ANYmal at the ARGOS Challenge – Tools and Experiences from the Autonomous Inspection of Oil & Gas Sites with a...

Presentation · October 2016
DOI: 10.13140/RG.2.2.31165.67043

17 authors, including:

Péter Fankhauser
ETH Zurich

Christian Gehring
ETH Zurich

Michael Bloesch
Imperial College London

Ralf Kaestner
ETH Zurich

ANYmal at the ARGOS Challenge

Tools and Experiences from the Autonomous Inspection of Oil & Gas Sites with a Legged Robot

Péter Fankhauser

Remo Diethelm, Samuel Bachmann, Christian Gehring, Martin Wermelinger, Dario Bellicoso, Vassilios Tsounis, Andreas Lauber, Michael Bloesch, Philipp Leemann, Gabriel Hottiger, Dominik Jud, Ralf Kaestner, Linus Isler, Mark Hoepflinger, Roland Siegwart, Marco Hutter


ANYmal for the
Oil & Gas Industry

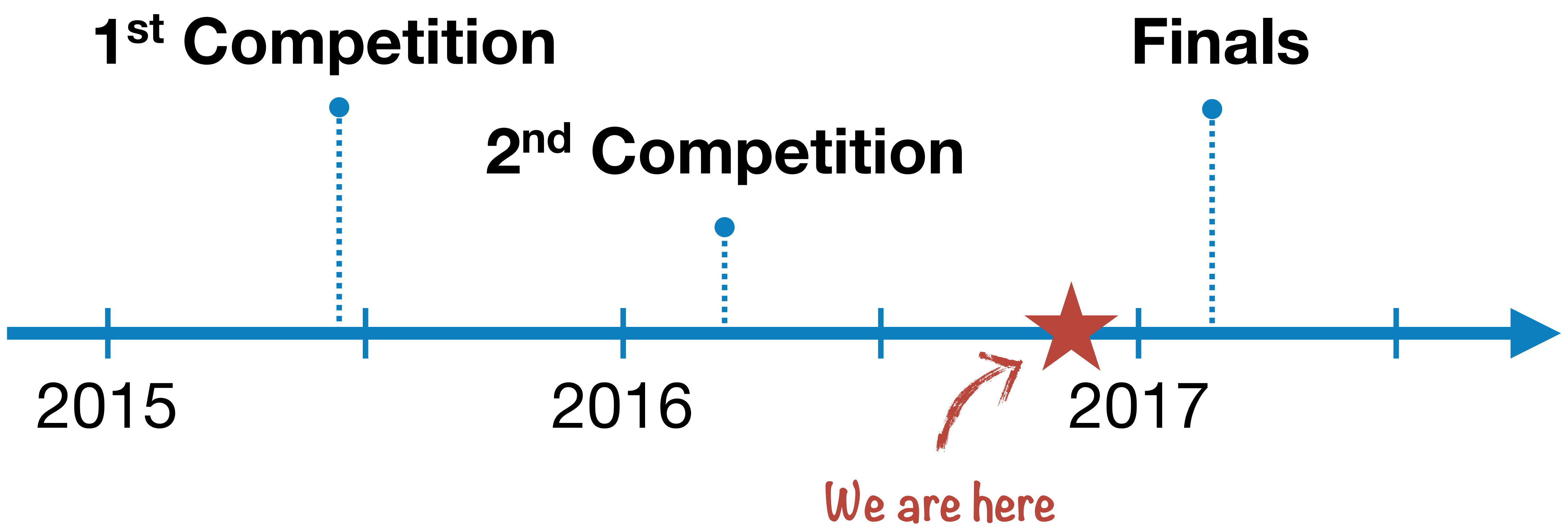
ARGOS Challenge

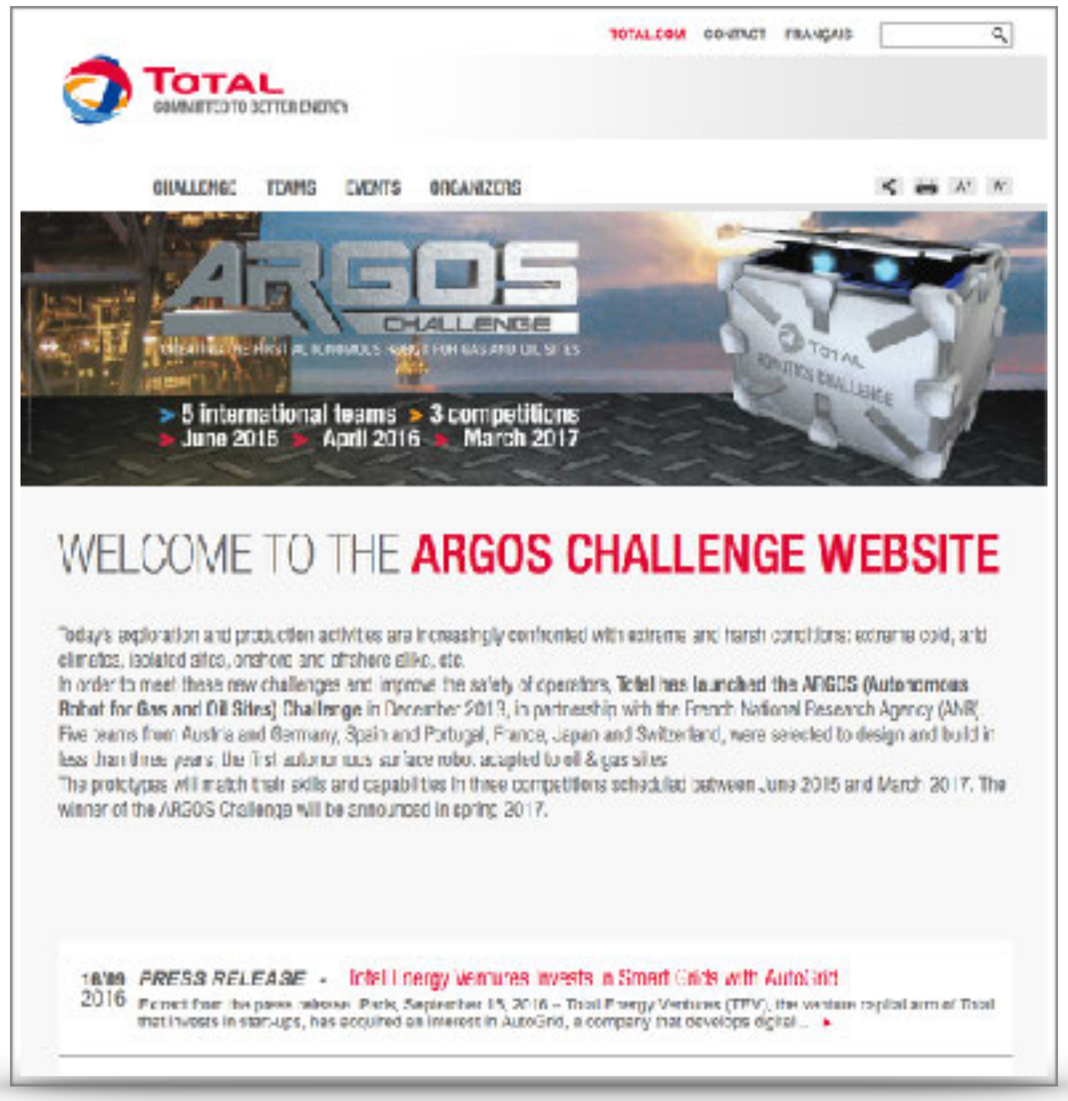
"Creating the first autonomous robot for gas and oil sites"

ARGOS CHALLENGE
LOOKING FOR THE FIRST AUTONOMOUS ROBOT FOR GAS AND OIL SITES

TOTAL COMMITTED TO BETTER ENERGY

Agence Nationale de la Recherche ANR

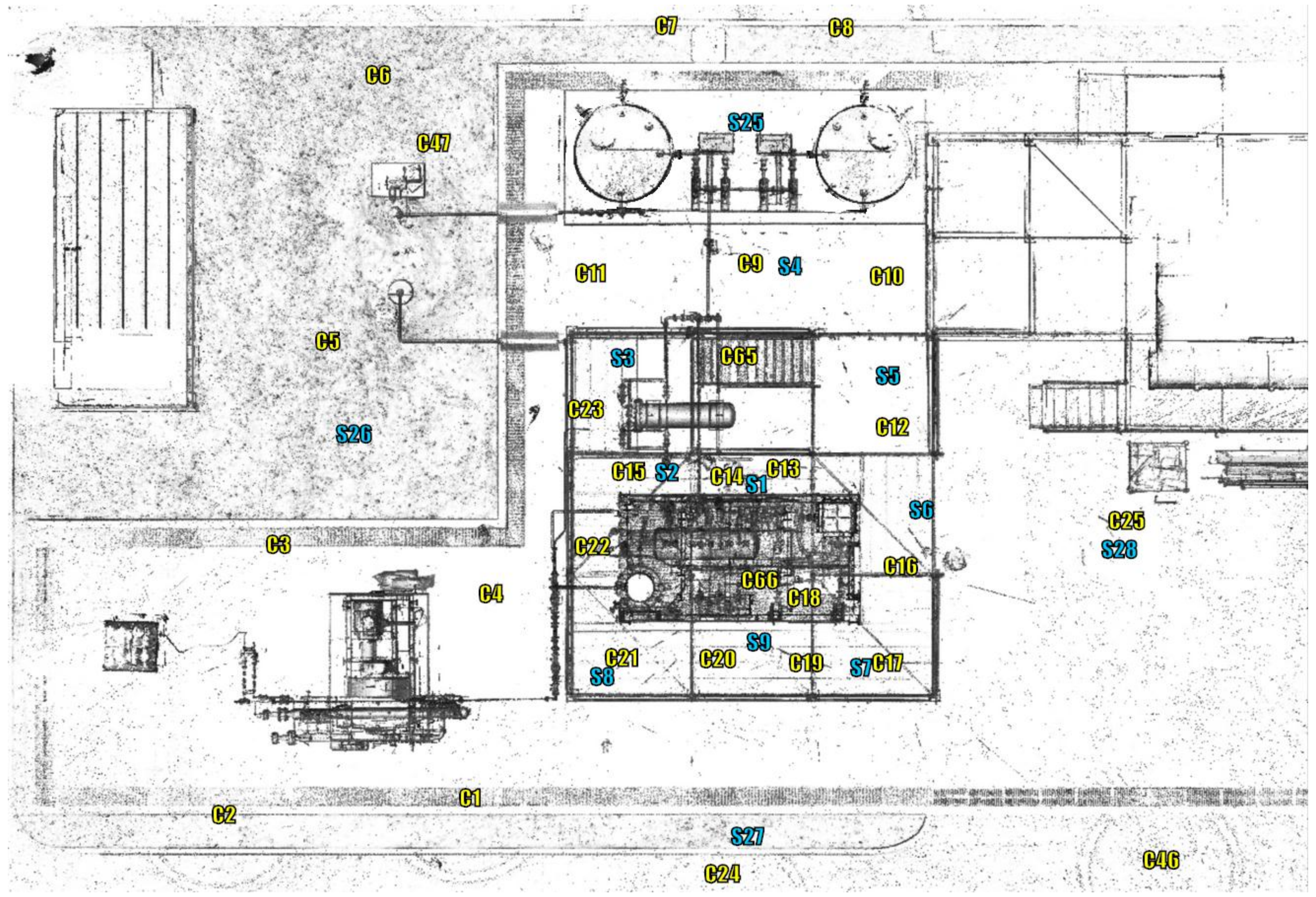


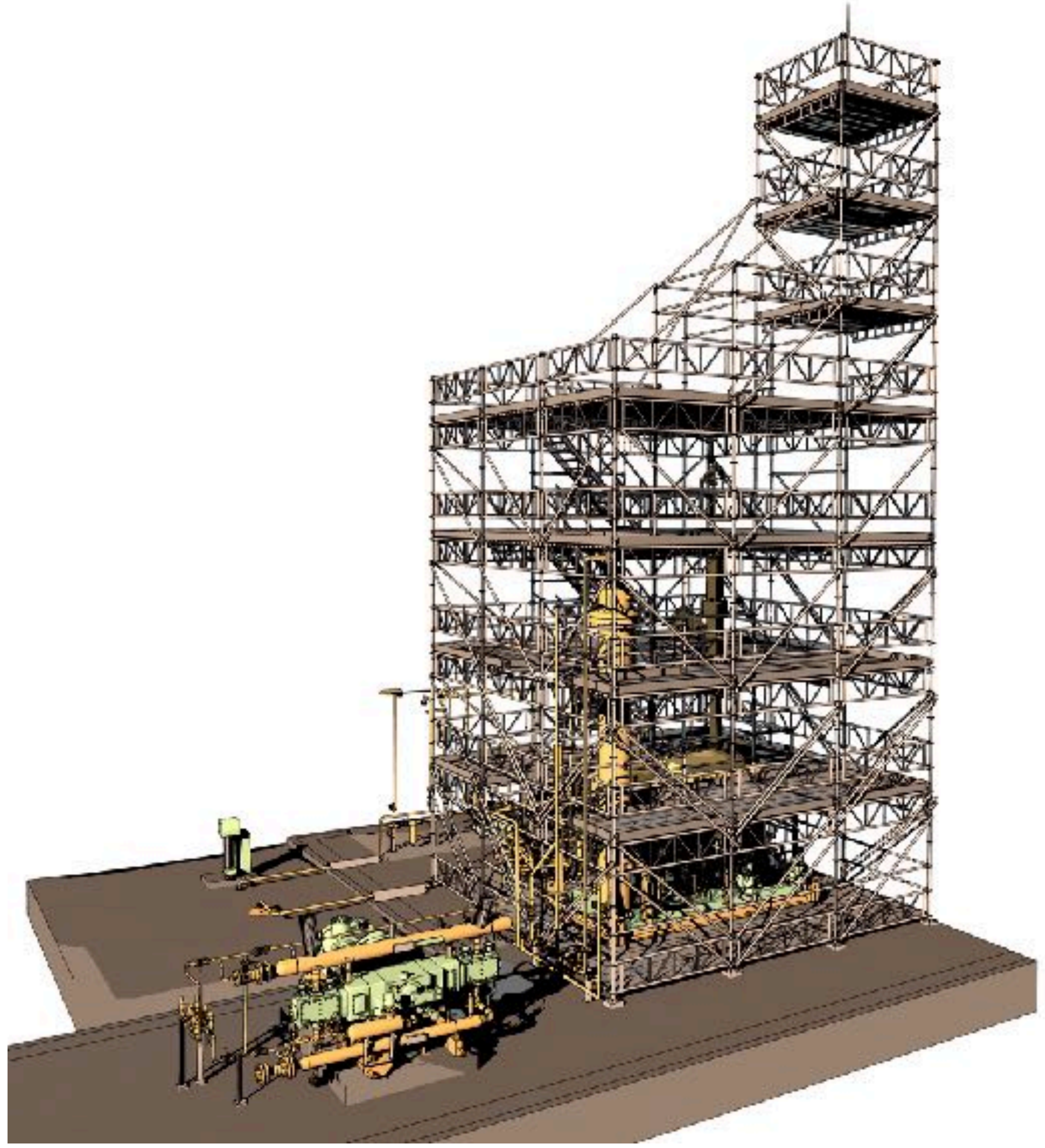


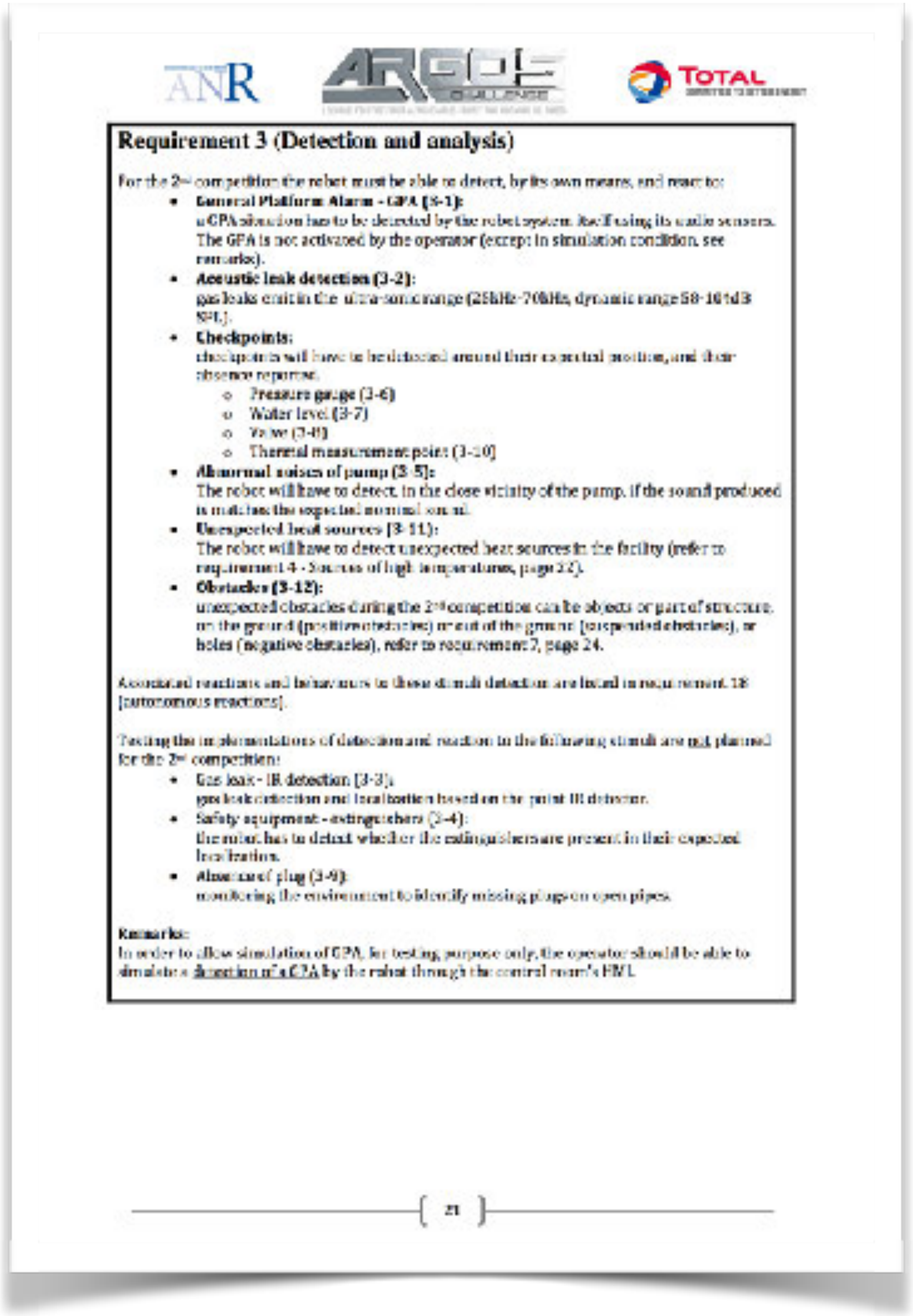
http://www.argos-challenge.com

ARGOS Challenge

Autonomous Inspection of Oil and Gas Sites







ANR ARGOS CHALLENGE TOTAL

Requirement 3 (Detection and analysis)

For the 2nd competition the robot must be able to detect, by its own means, and react to:

- **General Platform Alarm - GPA (3-1):**
  a GPA situation has to be detected by the robot system itself using its audio sensors. The GPA is not activated by the operator (except in simulation condition, see remarks).
- **Acoustic leak detection (3-2):**
  gas leaks emit in the ultra-sonic range (25kHz-70kHz, dynamic range 58-101dB SPL).
- **Checkpoints:**
  - Pressure gauge (3-6)
  - Water level (3-7)
  - Valve (3-8)
  - Thermal measurement point (3-10)
- **Abnormal noises of pump (3-5):**
  The robot will have to detect, in the close vicinity of the pump, if the sound produced
- **Unexpected heat sources (3-11):**
  The robot will have to detect unexpected heat sources in the facility (refer to
- **Obstacles (3-12):**
  unexpected obstacles during the 2nd competition can be objects or part of structure, holes (negative obstacles), refer to requirement 7, page 24.

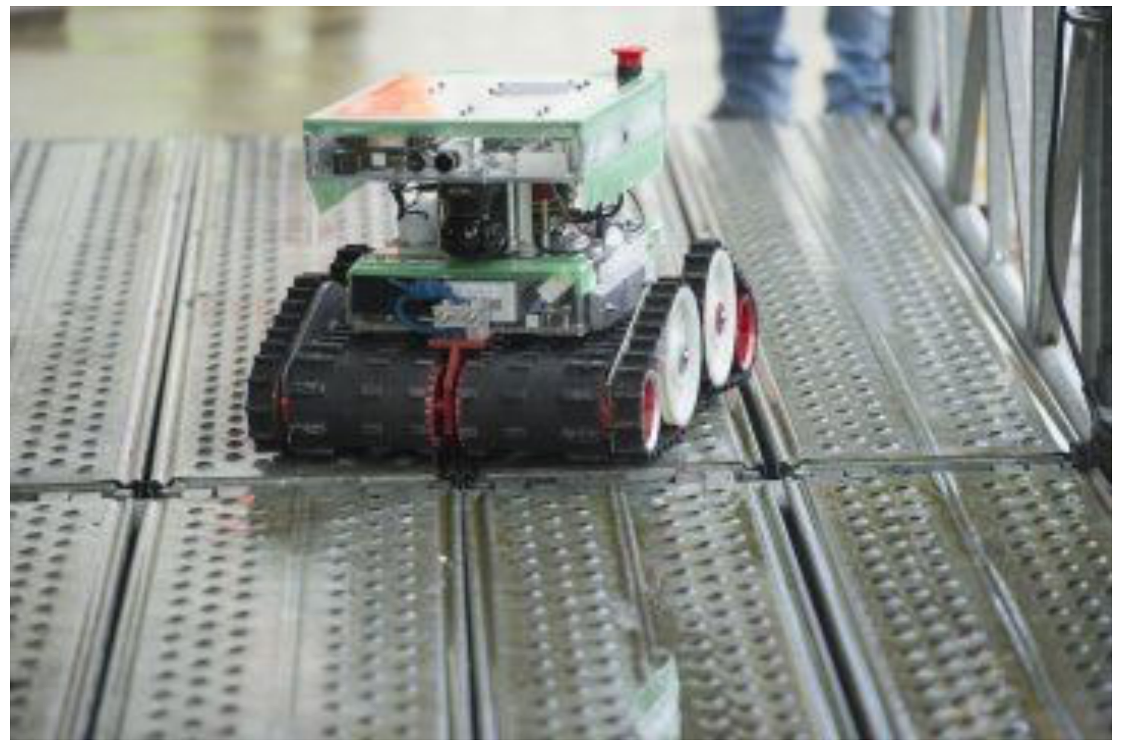
for the 2nd competition:

- Gas leak - IR detection (3-3):
- Safety equipment - extinguishers (3-4):
- Absence of plug (3-9):
  monitoring the environment to identify missing plugs on open pipes.

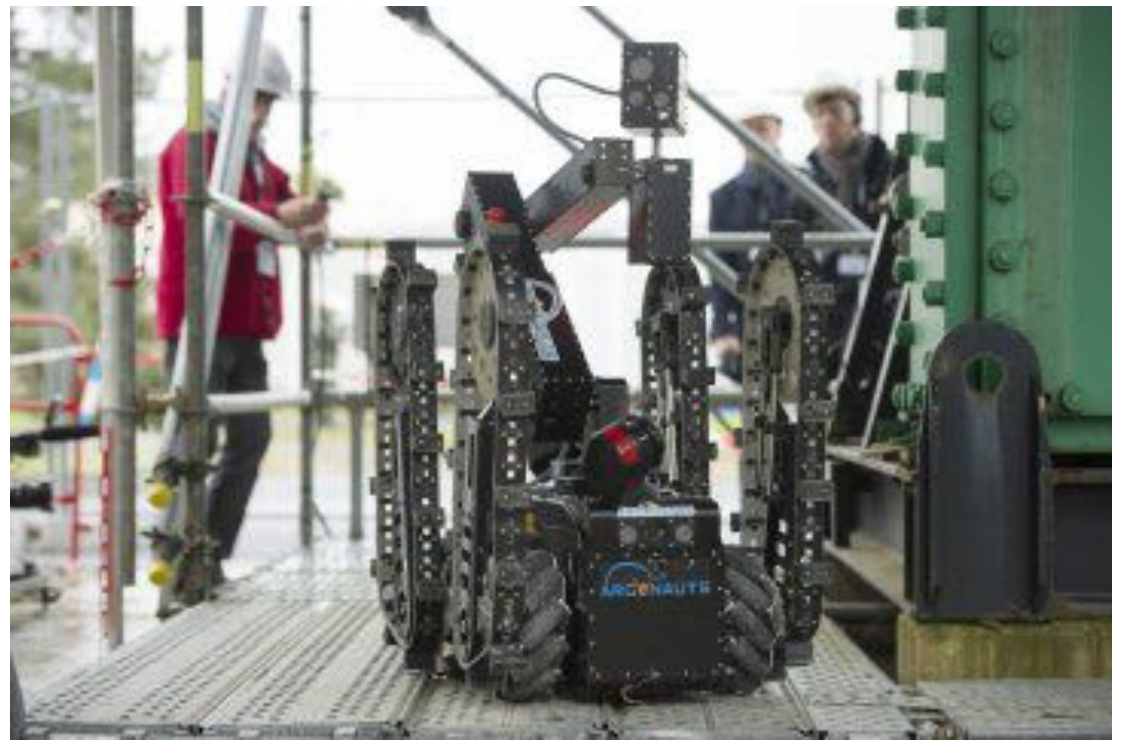
Remarks:

In order to allow simulation of GPA, for testing purpose only, the operator should be able to

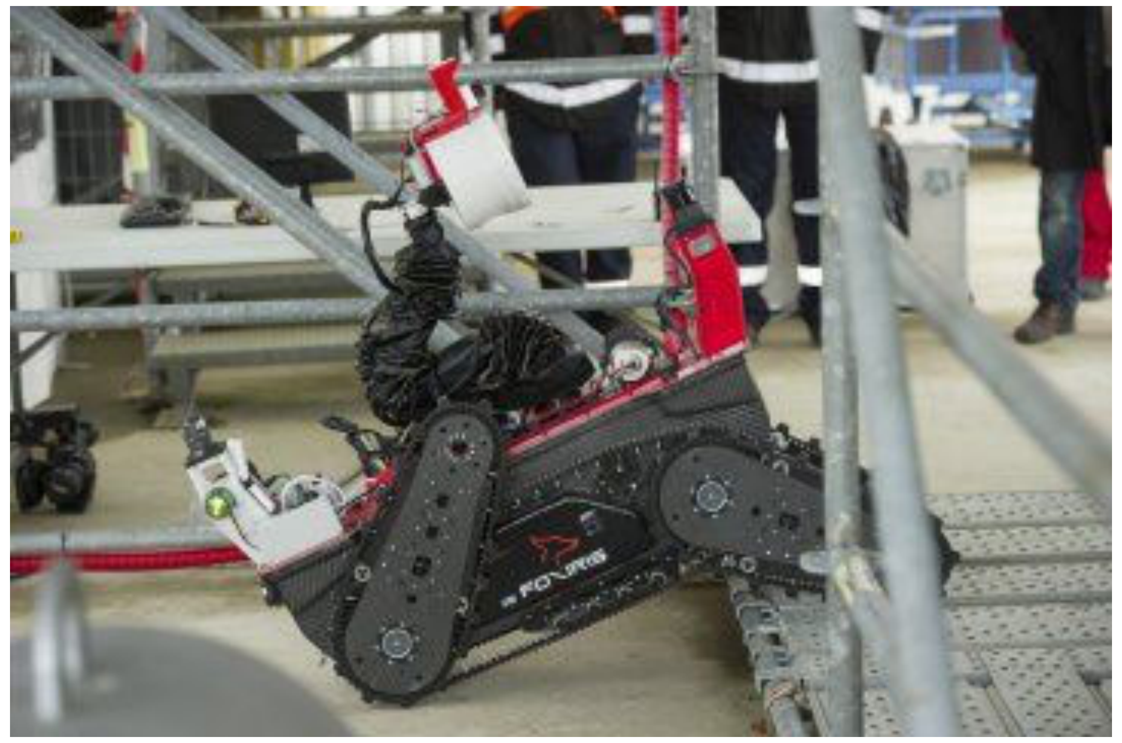
ARGOS Challenge
5 International Teams



AIR-K
Japan



ARGONAUTS
Austria & Germany



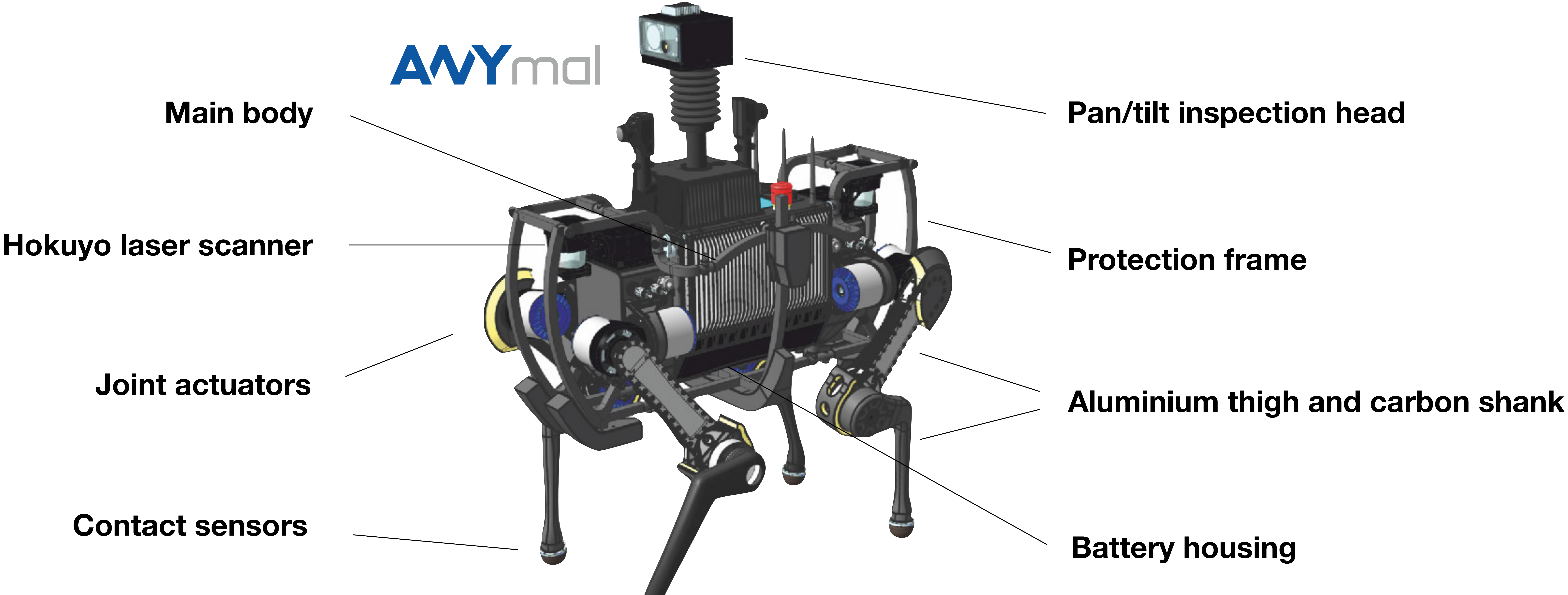
FOXIRIS
Spain & Portugal

VIKINGS
France

LIO
Switzerland

ARGOS Challenge
Team LIO

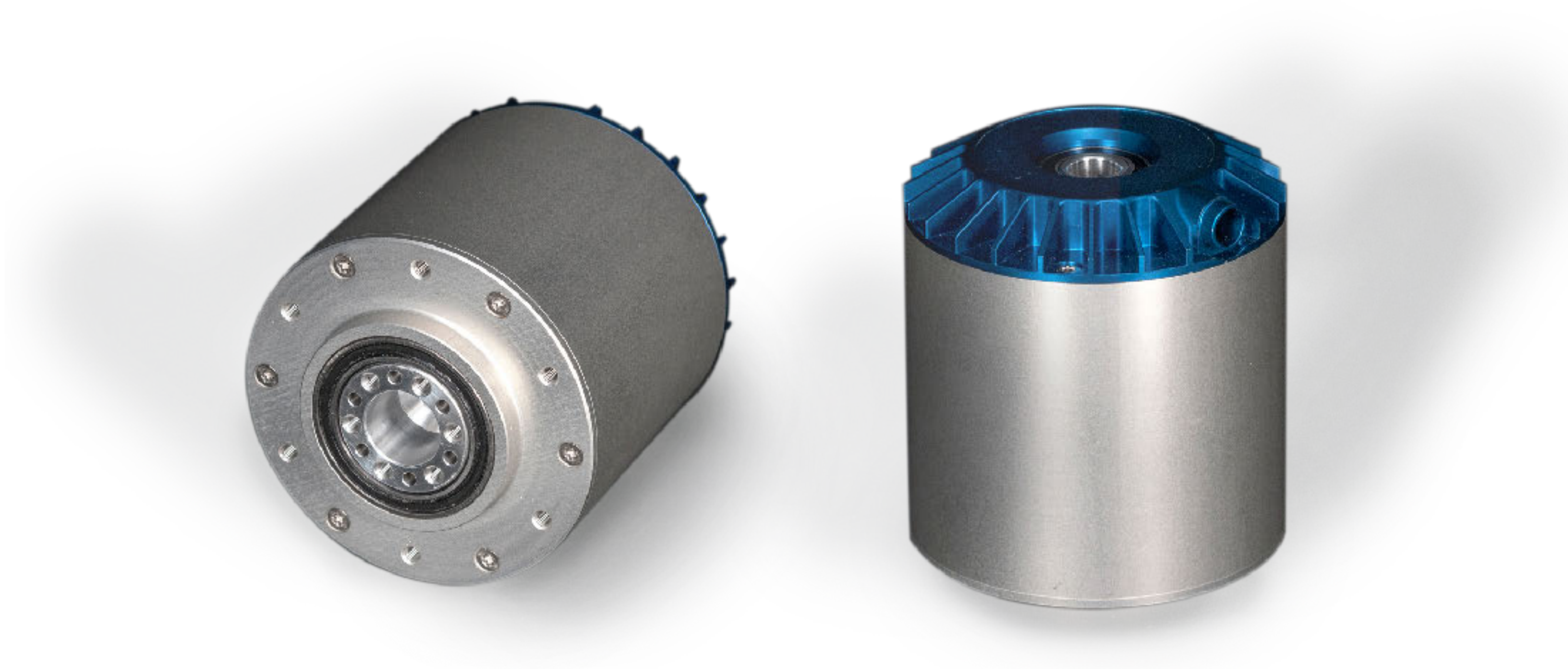
ANYmal – A High-Performance & Versatile Quadrupedal Robot



M. Hutter, C. Gehring, D. Jud, A. Lauber, C. D. Bellicoso, V. Tsounis, J. Hwangbo, P. Fankhauser, M. Bloesch, R. Diethelm, and S. Bachmann,
"ANYmal - A Highly Mobile and Dynamic Quadrupedal Robot," IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), 2016.

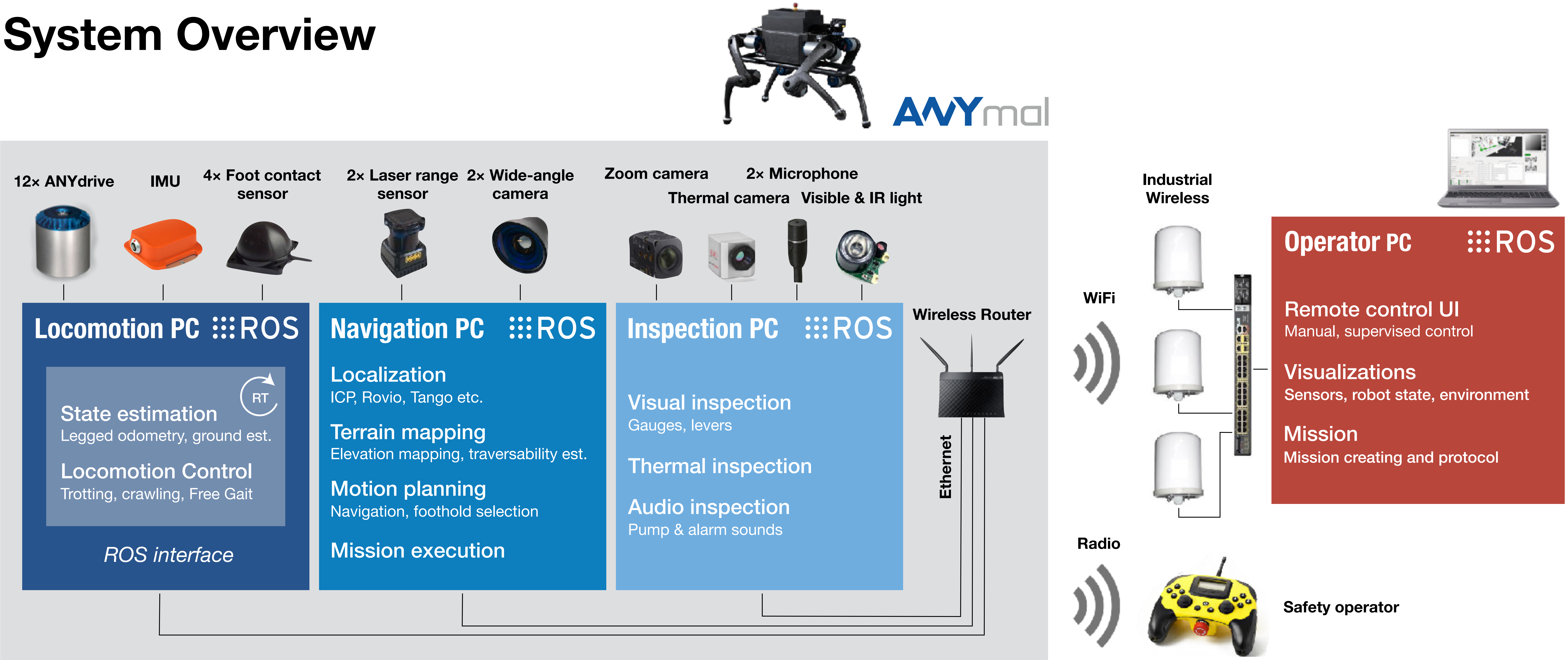
ANYdrive – A Integrated, Robust, Torque-Controllable Robot Joint

- Fully integrated
- Accurate position & torque control
- Absolute position sensing
- Programmable controller
- Impact robust
- Hollow-shaft
- Water-proof

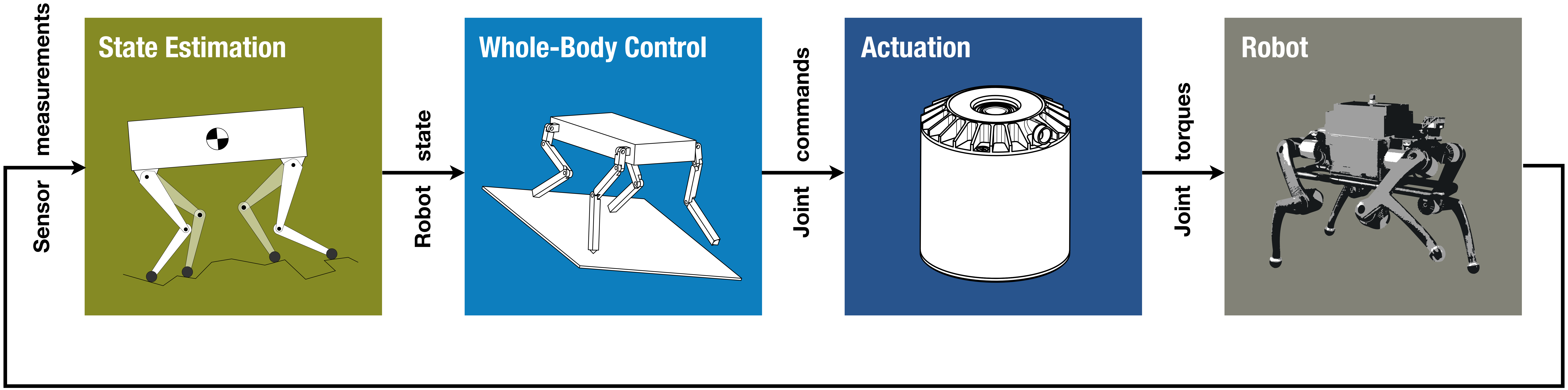


ANYdrive

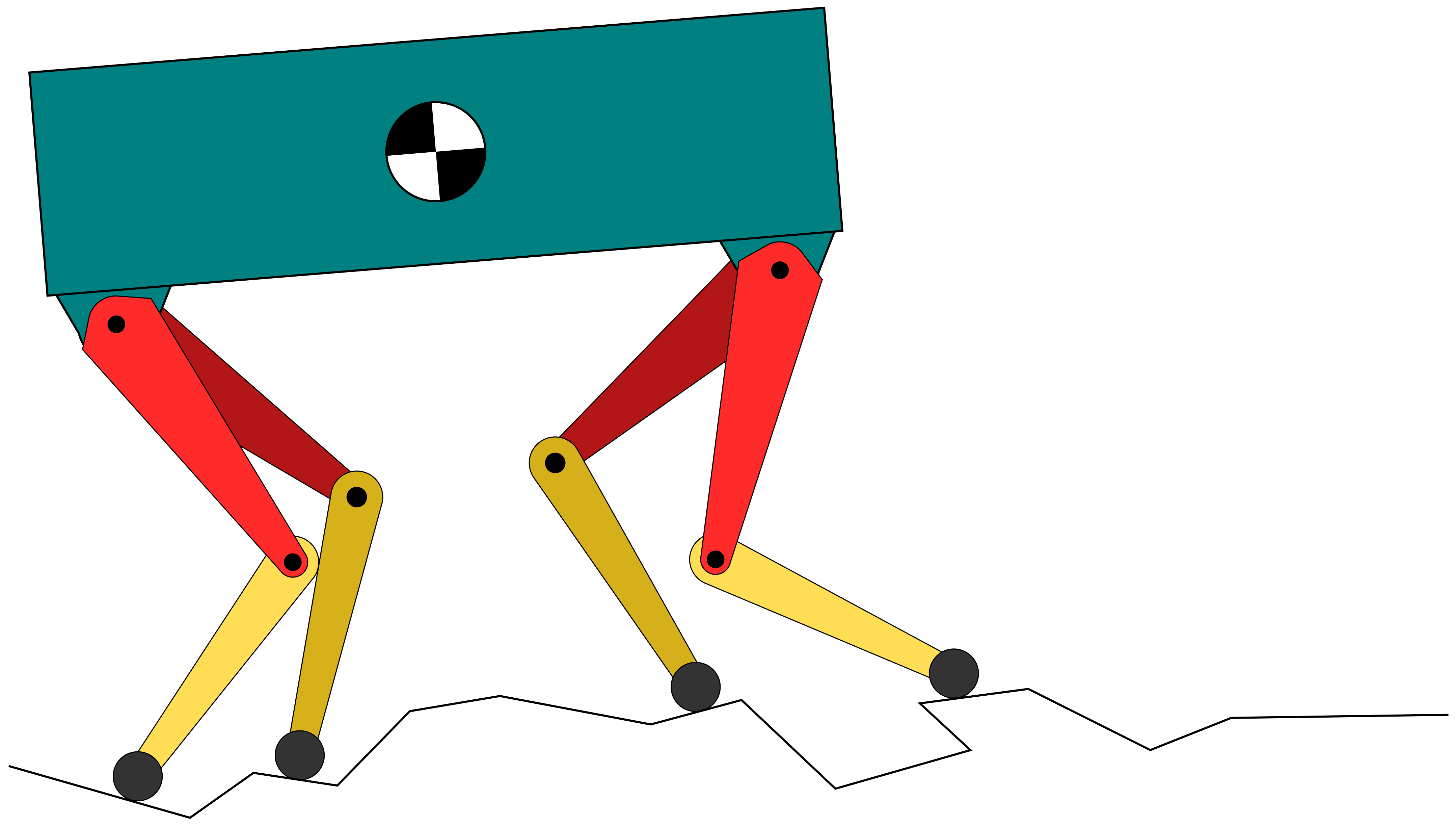
System Overview



Locomotion


Sensor measurements
State Estimation
Robot state
Whole-Body Control
Joint commands
Actuation
Joint torques
Robot

Locomotion
State Estimation

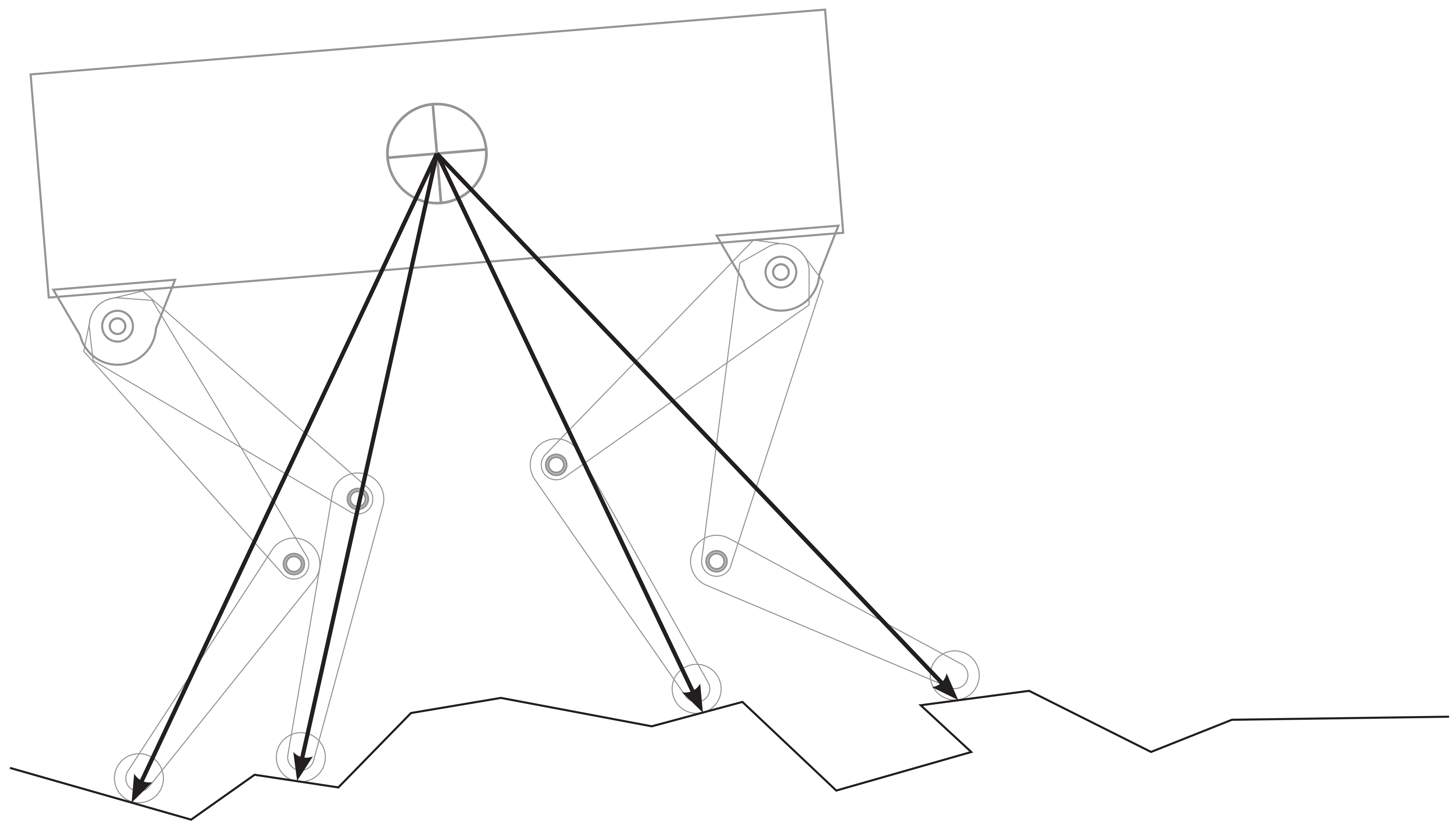


Extended Kalman Filter

- No assumption on terrain

M. Bloesch, C. Gehring, P. Fankhauser, M. Hutter, M. A. Hoepflinger and R. Siegwart, "**State Estimation for Legged Robots on Unstable and Slippery Terrain**", in International Conference on Intelligent Robots and Systems (IROS), 2013.

Locomotion
State Estimation

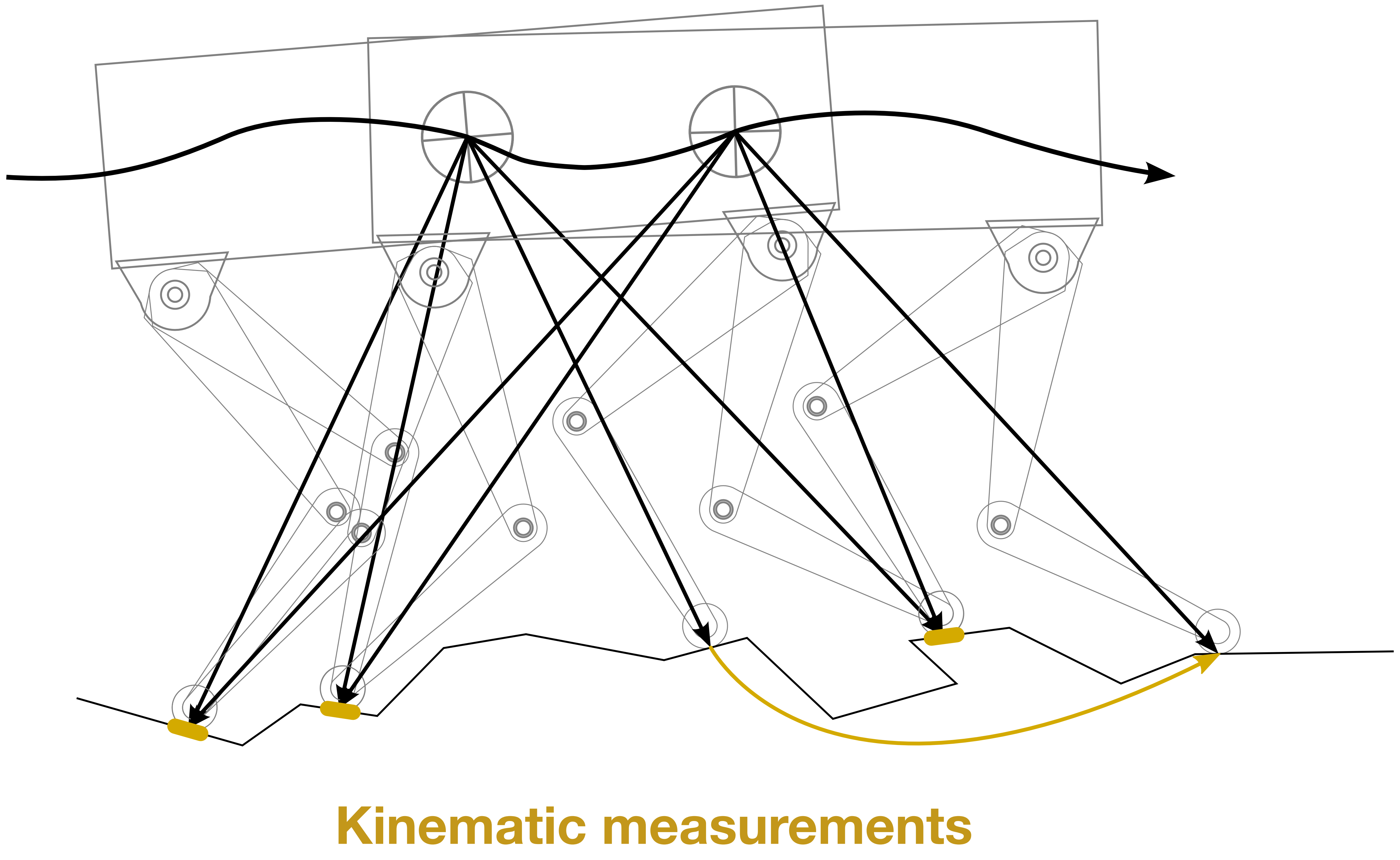


Extended Kalman Filter

- No assumption on terrain
- Kinematic measurements (encoders) for legs in contact

M. Bloesch, C. Gehring, P. Fankhauser, M. Hutter, M. A. Hoepflinger and R. Siegwart, "**State Estimation for Legged Robots on Unstable and Slippery Terrain**", in International Conference on Intelligent Robots and Systems (IROS), 2013.

Locomotion
State Estimation

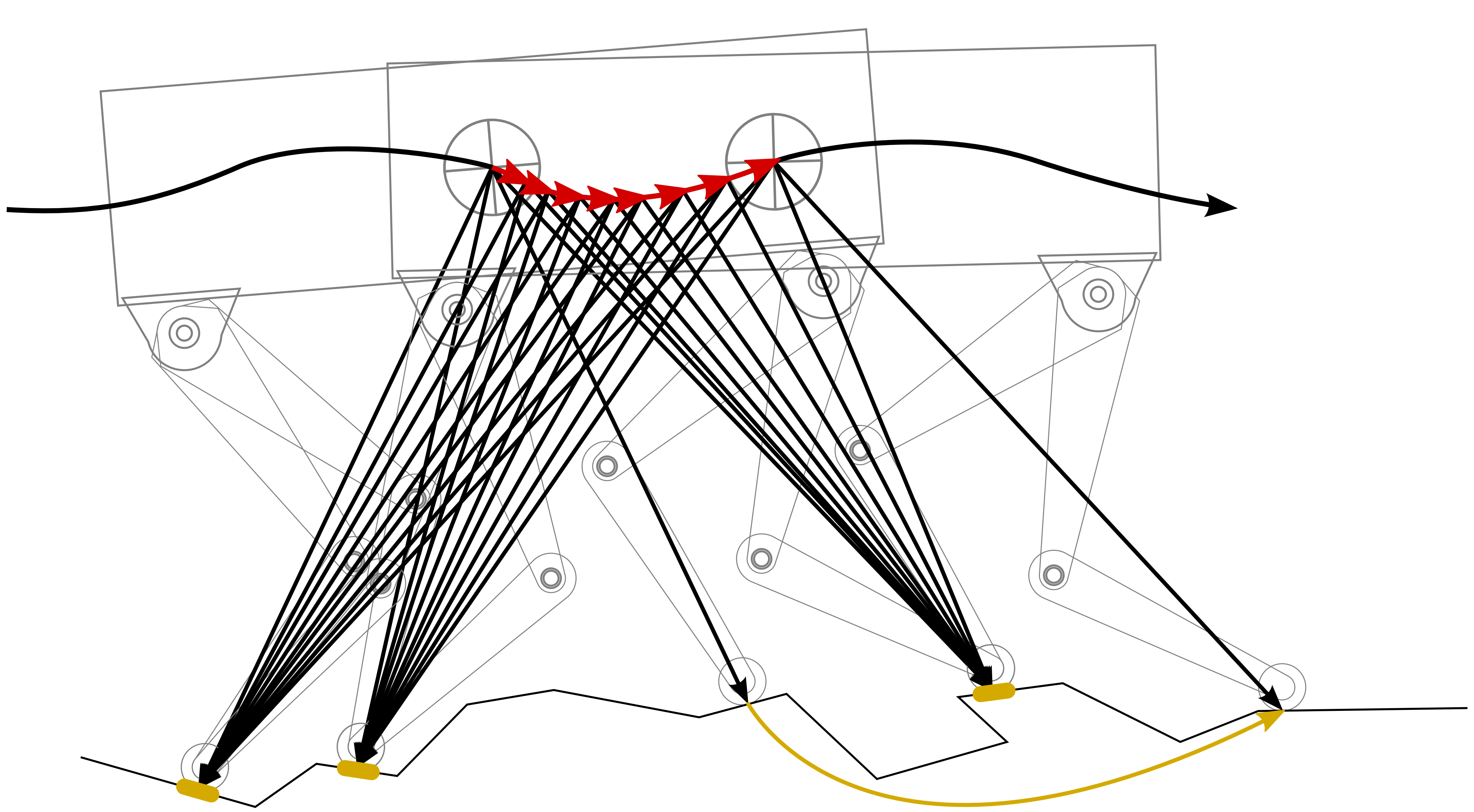


Extended Kalman Filter

- No assumption on terrain
- Kinematic measurements (encoders) for legs in contact

M. Bloesch, C. Gehring, P. Fankhauser, M. Hutter, M. A. Hoepflinger and R. Siegwart, “**State Estimation for Legged Robots on Unstable and Slippery Terrain**”, in International Conference on Intelligent Robots and Systems (IROS), 2013.

Locomotion
State Estimation

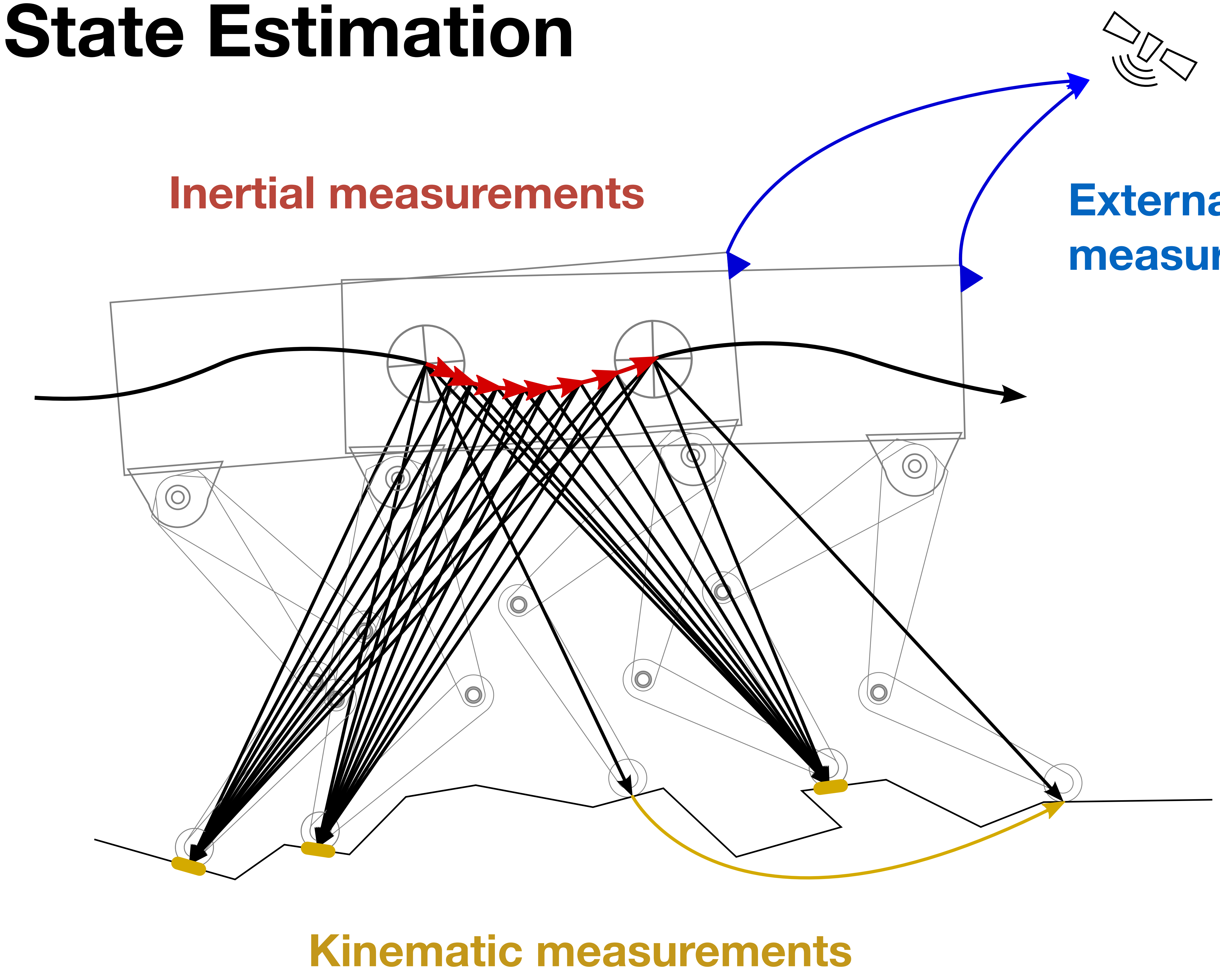


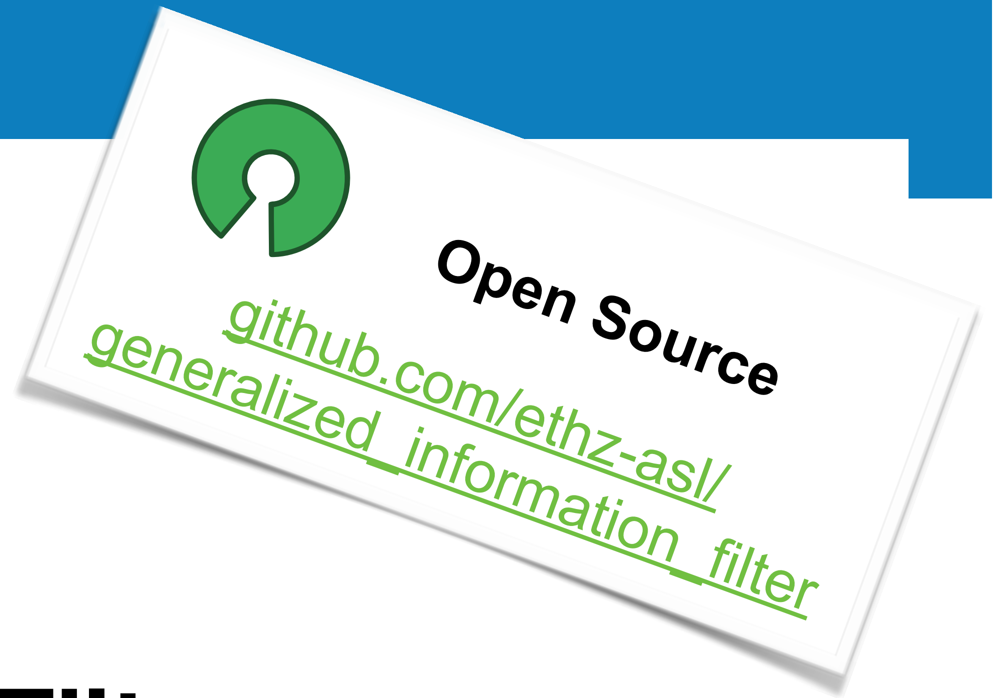
Extended Kalman Filter

- No assumption on terrain
- Kinematic measurements (encoders) for legs in contact
- Fused with inertial measurements (IMU)
- Error < 5% over distance

M. Bloesch, C. Gehring, P. Fankhauser, M. Hutter, M. A. Hoepflinger and R. Siegwart, "**State Estimation for Legged Robots on Unstable and Slippery Terrain**", in International Conference on Intelligent Robots and Systems (IROS), 2013.

Locomotion
State Estimation





Extended Kalman Filter

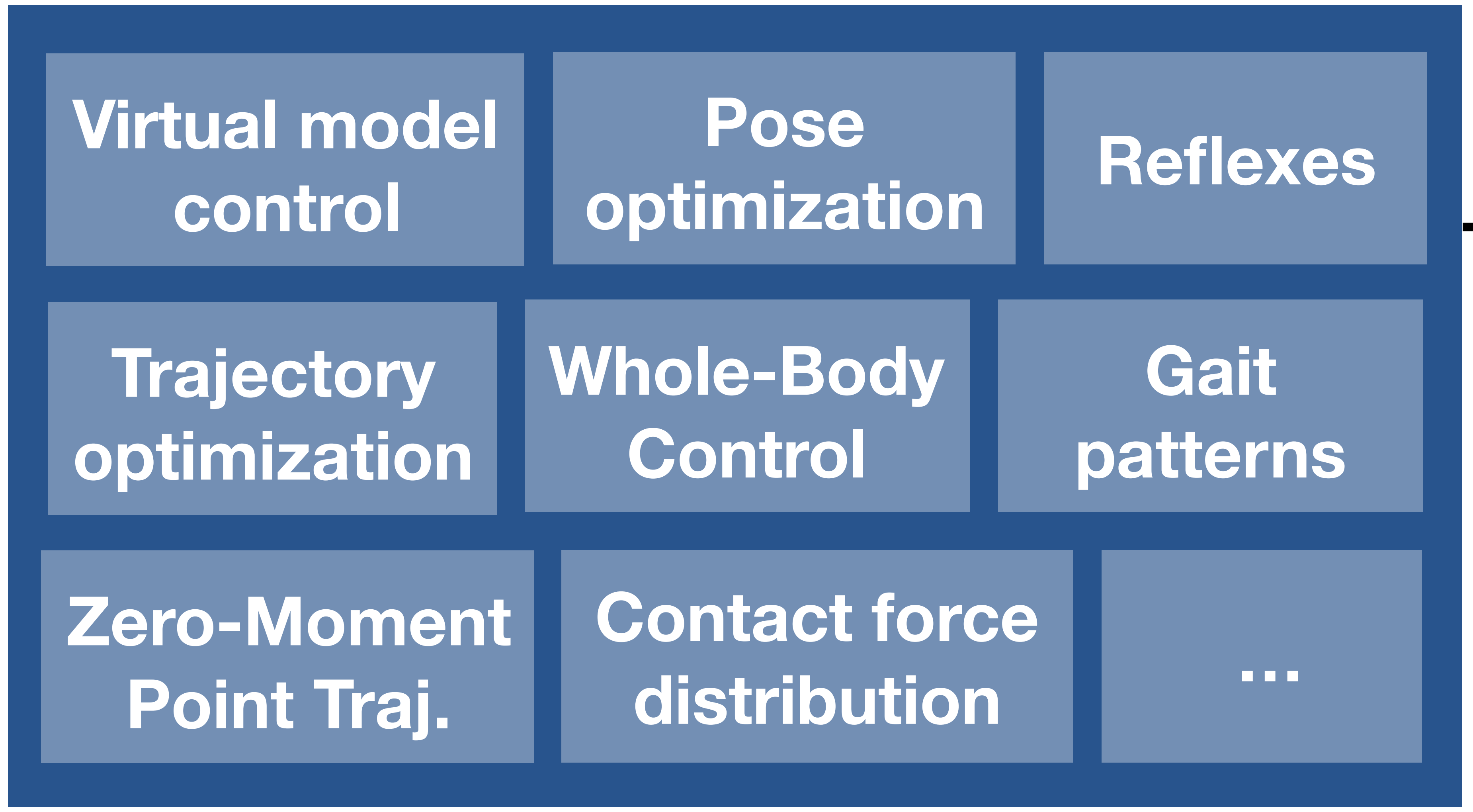
- No assumption on terrain
- Kinematic measurements (encoders) for legs in contact
- Fused with inertial measurements (IMU)
- Error < 5% over distance
- Optionally combined with external pose (GPS, laser, vision, etc.)

M. Bloesch, C. Gehring, P. Fankhauser, M. Hutter, M. A. Hoepflinger and R. Siegwart, "**State Estimation for Legged Robots on Unstable and Slippery Terrain**", in International Conference on Intelligent Robots and Systems (IROS), 2013.

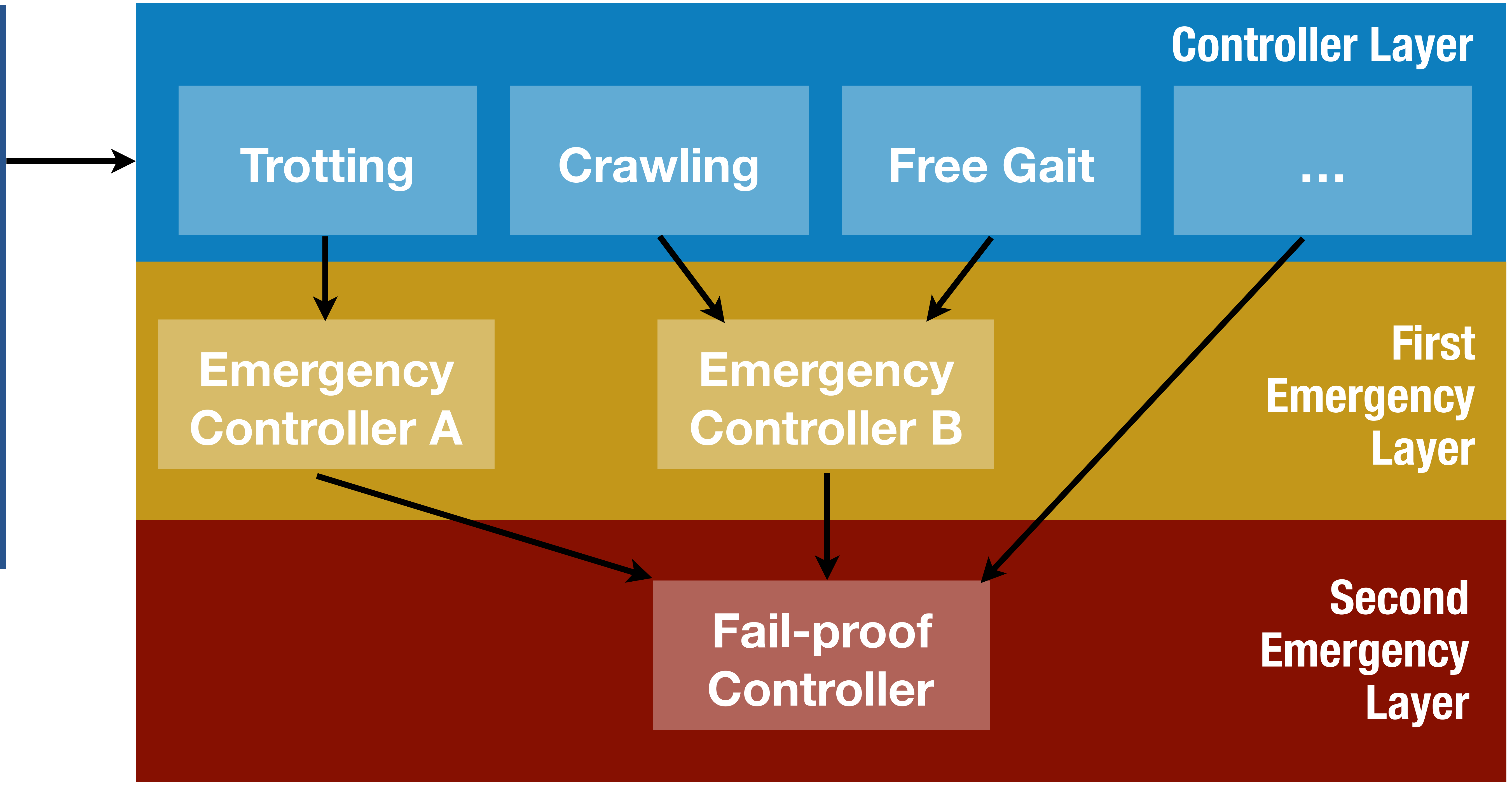
Locomotion
Whole-Body Control

Open Source (coming soon)

Locomotion Controller Modules (Loco)



Robot Controller Manager (Rocoma)

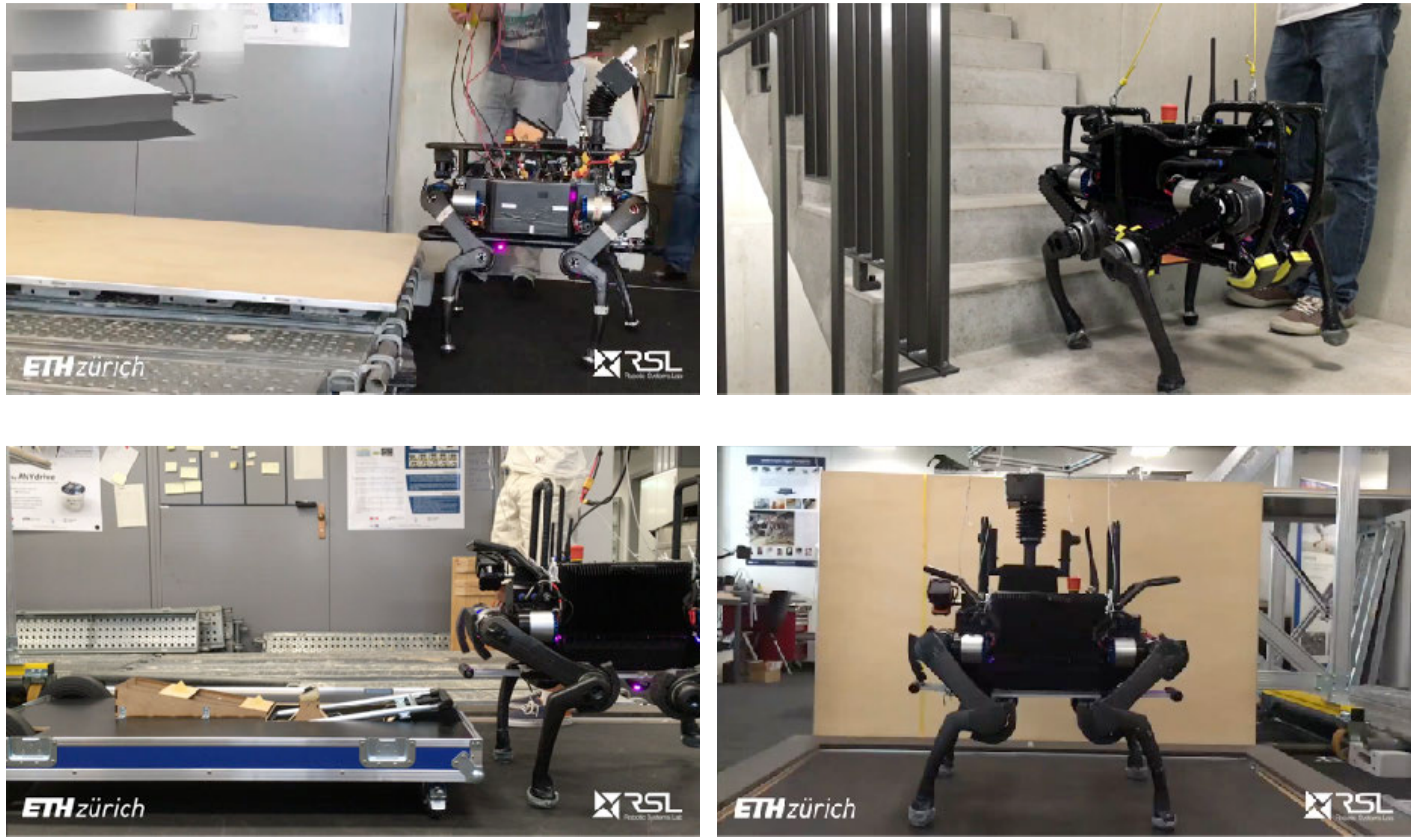


C. Gehring, S. Coros, M. Hutter, D. Bellicoso, H. Heijnen, R. Diethelm, M. Bloesch, P. Fankhauser, J. Hwangbo, M. A. Hoepflinger, and R. Siegwart, "**Practice Makes Perfect: An Optimization-Based Approach to Controlling Agile Motions for a Quadruped Robot.**", in IEEE Robotics & Automation Magazine, 2016.

C. Dario Bellicoso, C. Gehring, J. Hwangbo, P. Fankhauser, M. Hutter, "**Emerging Terrain Adaptation from Hierarchical Whole Body Control**," in IEEE Internal Conference on Humanoid Robots (Humanoids), 2016.

Open Source
github.com/leggedrobotics/free_gait (in progress)

Locomotion
Free Gait – An Architecture for the Versatile Control of Legged Robots

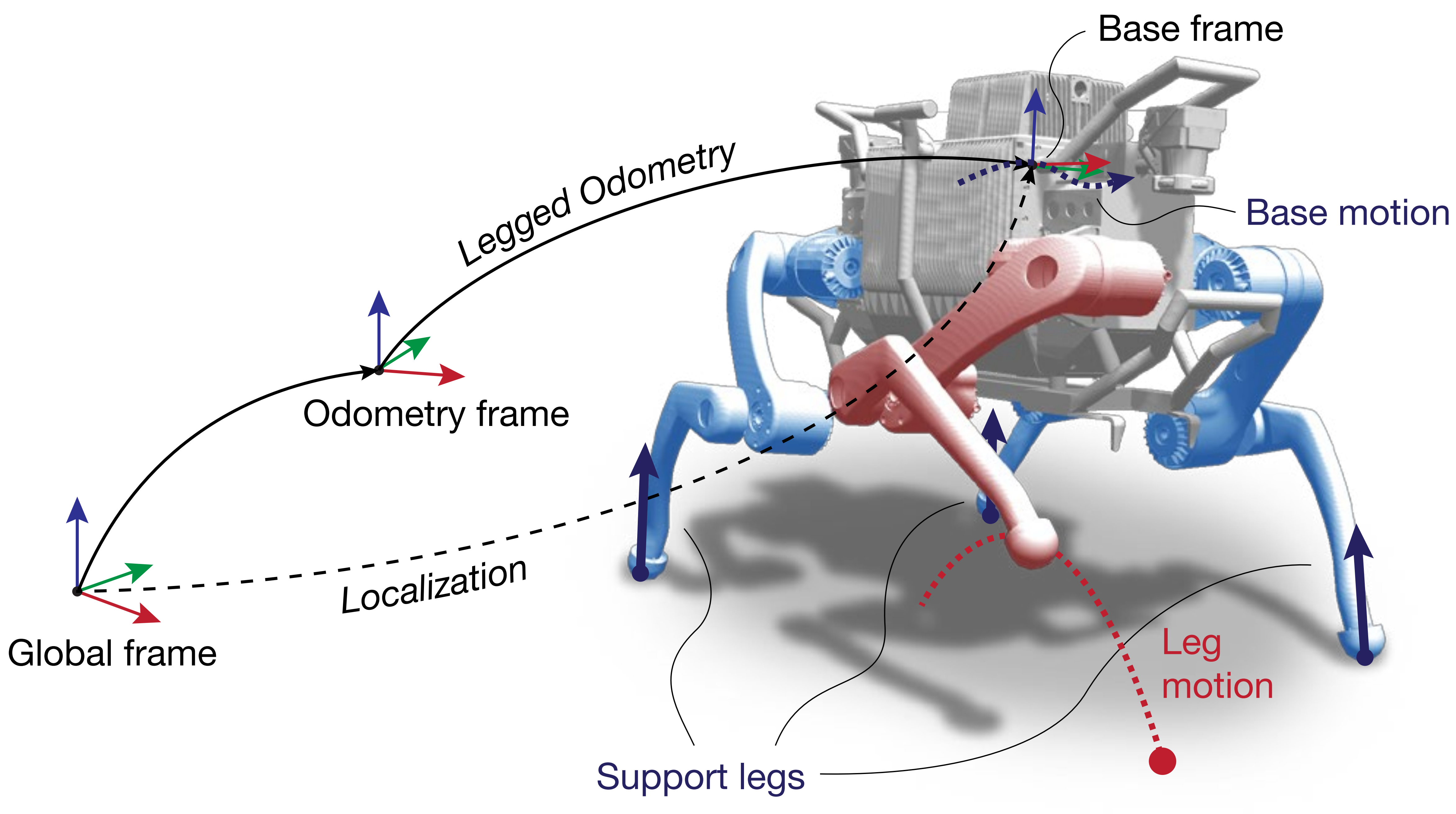


- Abstraction Layer for Whole-Body Motions (Free Gait API)

P. Fankhauser, D. Bellicoso, C. Gehring, R. Dubé, A. Gawel, and M. Hutter, **"Free Gait – An Architecture for the Versatile Control of Legged Robots,"** in IEEE-RAS International Conference on Humanoid Robots (Humanoids), 2016.

Open Source
github.com/leggedrobotics/free_gait (in progress)

Locomotion
Free Gait – An Architecture for the Versatile Control of Legged Robots



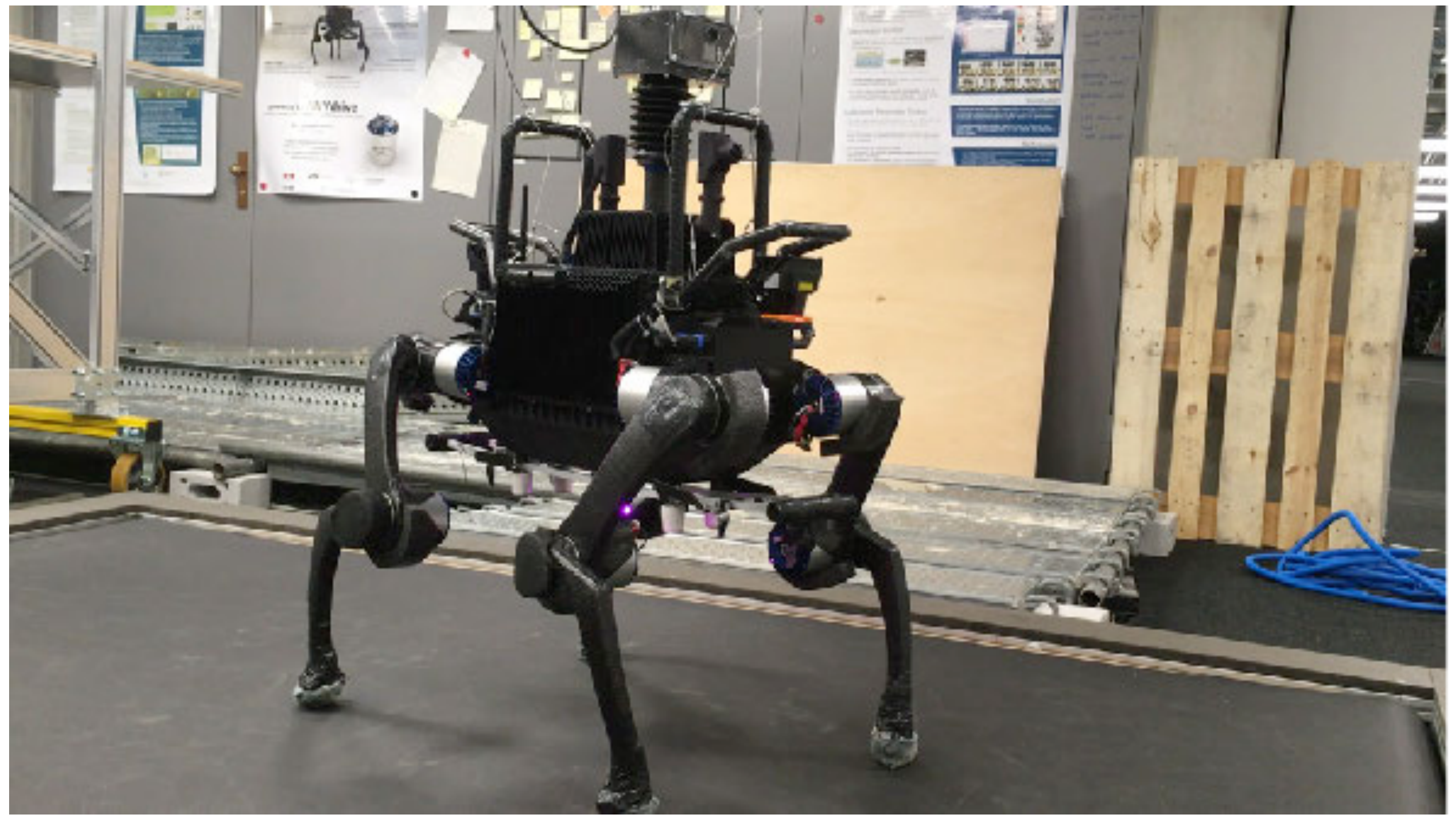
- Abstraction Layer for Whole-Body Motions (Free Gait API)
- Robust motion execution in task space

P. Fankhauser, D. Bellicoso, C. Gehring, R. Dubé, A. Gawel, and M. Hutter, **"Free Gait – An Architecture for the Versatile Control of Legged Robots,"** in IEEE-RAS International Conference on Humanoid Robots (Humanoids), 2016.

Open Source
github.com/leggedrobotics/free_gait (in progress)

Locomotion
Free Gait – An Architecture for the Versatile Control of Legged Robots

```
steps:
 - step:
   - base_auto:
 - step:
   - end_effector_target:
      name: RF_LEG
      ignore_contact: true
      target_position:
       frame: footprint
       position: [0.39, -0.24, 0.20]
 - step:
   - base_auto:
      height: 0.38
      ignore_timing_of_leg_motion: true
   - end_effector_target: &foot
      name: RF_LEG
      ignore_contact: true
      ignore_for_pose_adaptation: true
      target_position:
       frame: footprint
       position: [0.39, -0.24, 0.20]
 - step:
   - base_auto:
      height: 0.45
      ignore_timing_of_leg_motion: true
   - end_effector_target: *foot
 - step:
   - footstep:
      name: RF_LEG
      profile_type: straight
      target:
       frame: footprint
       position: [0.32, -0.24, 0.0]
 - step:
   - base_auto:
```



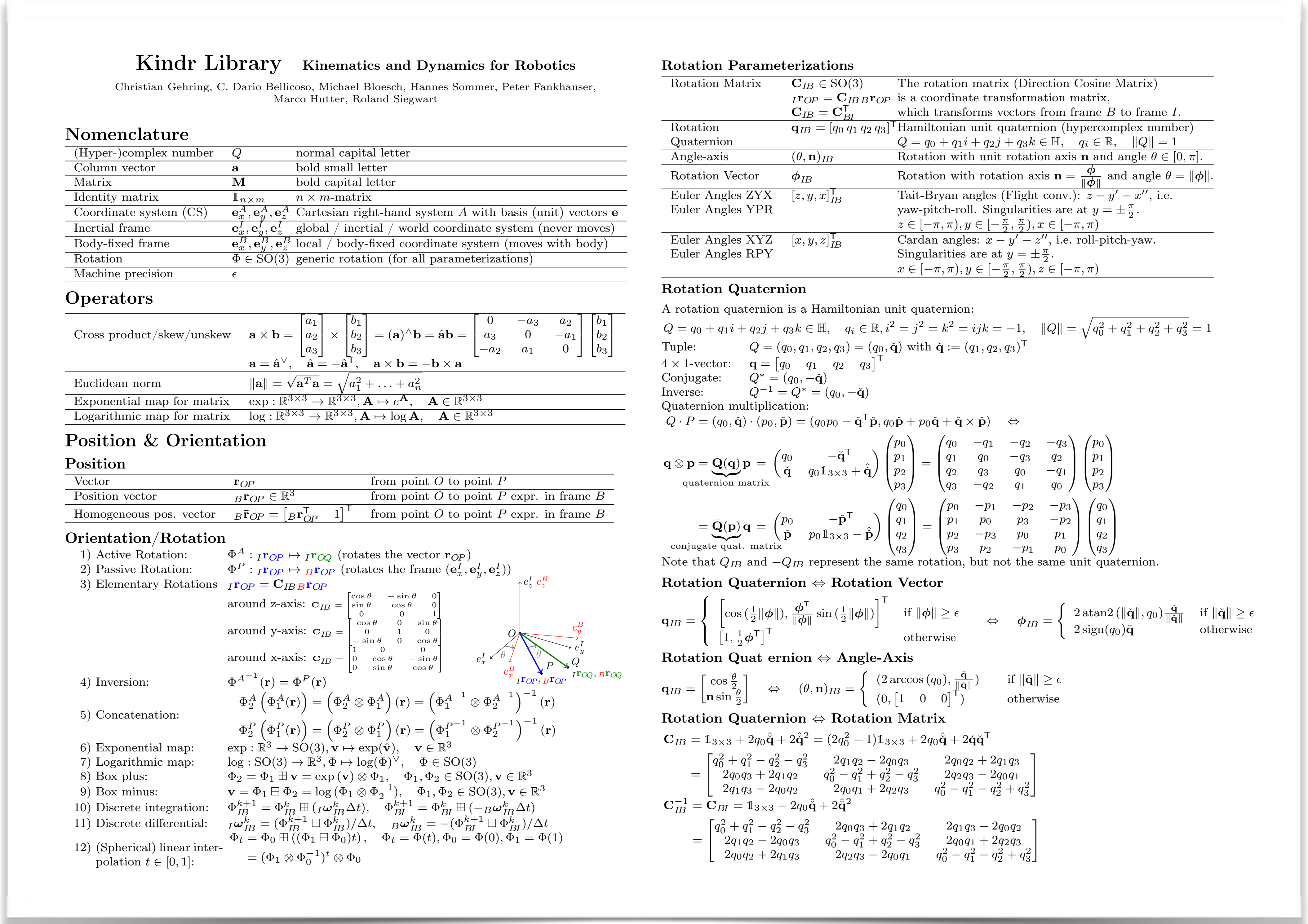
- Abstraction Layer for Whole-Body Motions (Free Gait API)
- Robust motion execution in task space
- Implemented as ROS Action (with frameworks for YAML, Python, C++)

P. Fankhauser, D. Bellicoso, C. Gehring, R. Dubé, A. Gawel, and M. Hutter, **"Free Gait – An Architecture for the Versatile Control of Legged Robots,"** in IEEE-RAS International Conference on Humanoid Robots (Humanoids), 2016.

Locomotion
Kindr – Kinematics and Dynamics for Robotics

Open Source
github.com/ethz-asl/kindr

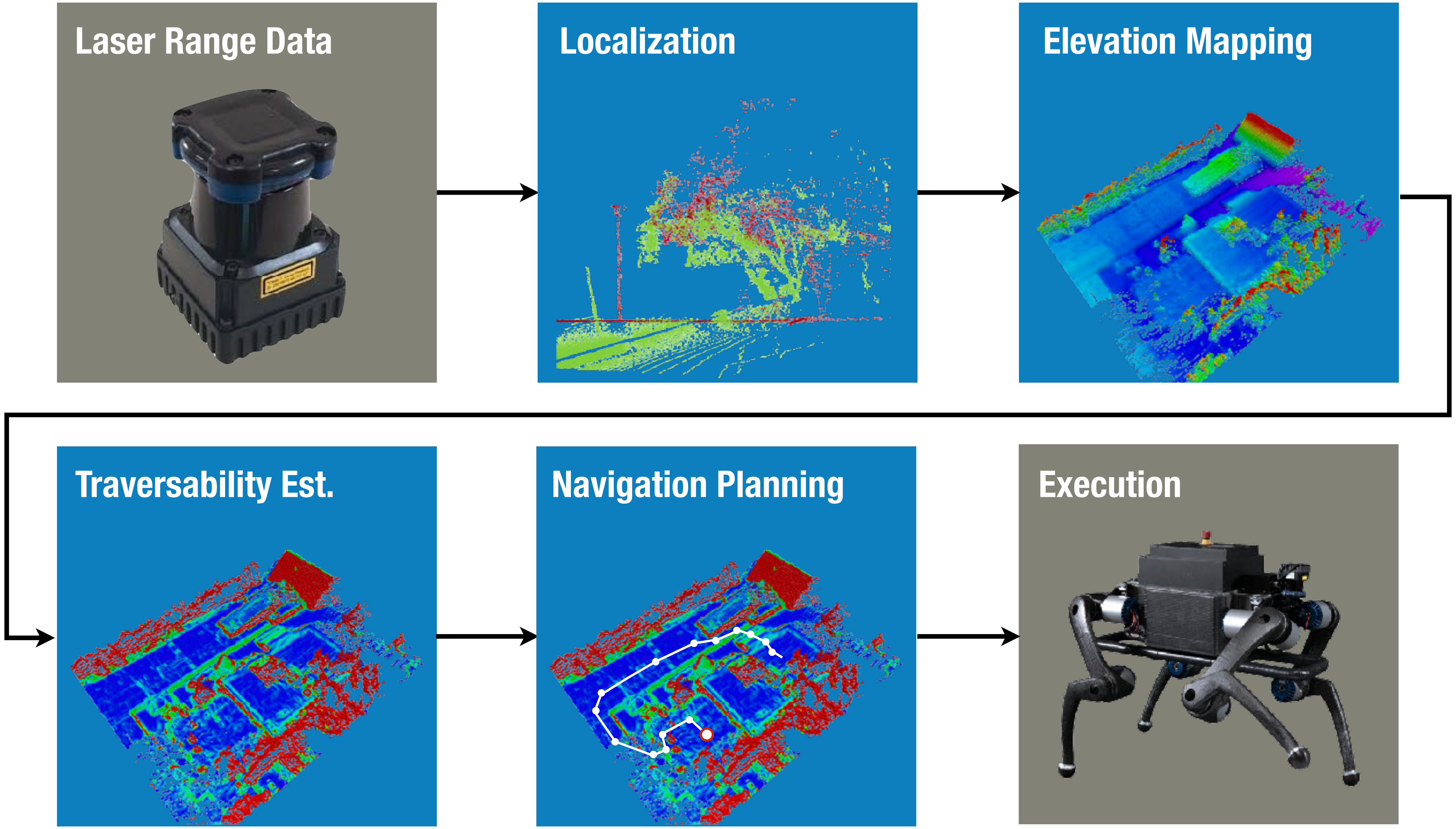
- C++ library for the consistent handling of 3d position and **rotations**
- Support for *rotation matrices*, *quaternions*, *angle-axis*, *rotation vectors*, *Euler angles*, etc.
- Support for all common operations and includes time-derivates
- ROS interface available
- Based on Eigen, 1000+ unit tests



Cheat-sheet (incl. derivations)

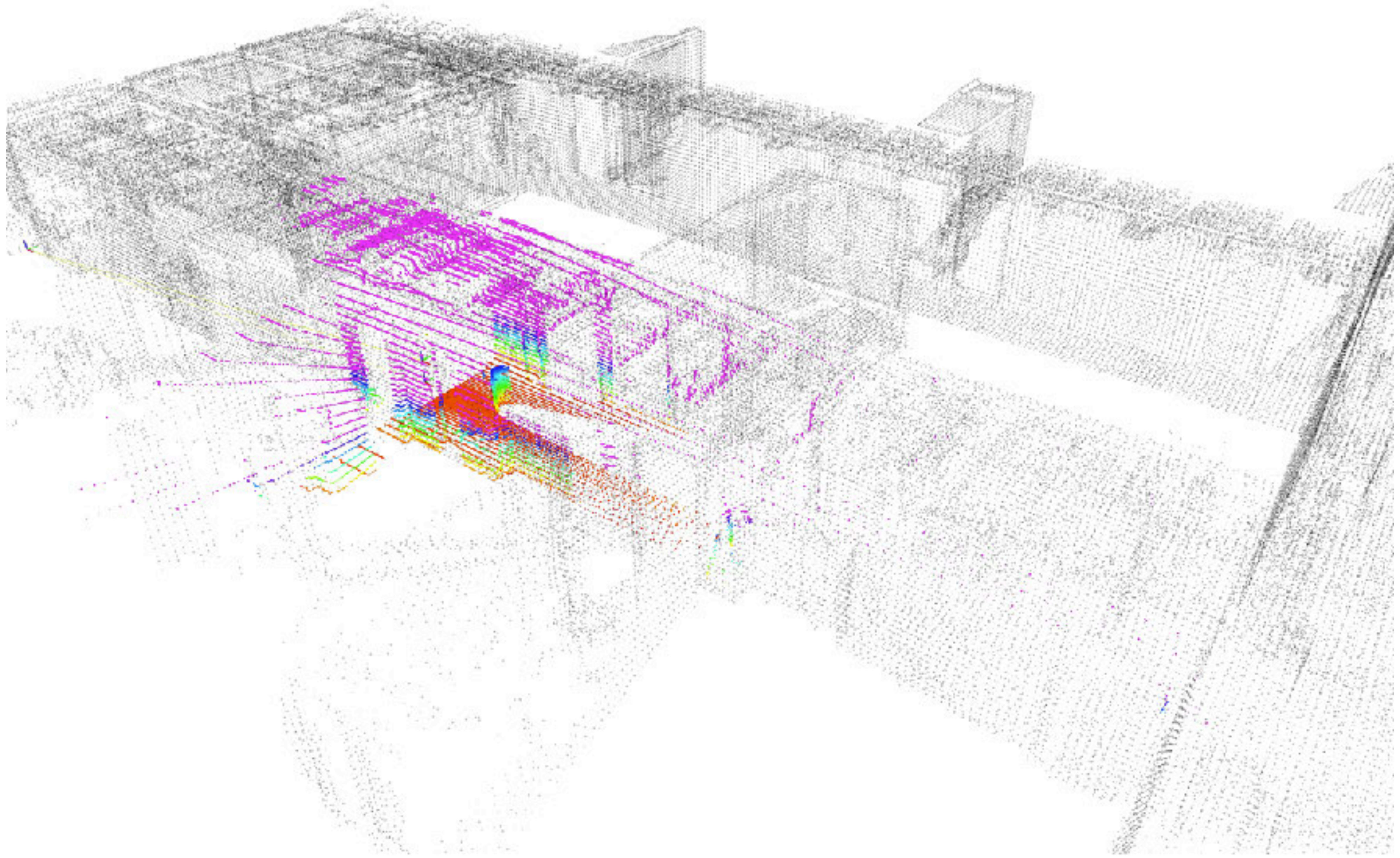
M. Bloesch, H. Sommer, T. Laidlow, M. Burri, G. Nuetzi, P. Fankhauser, D. Bellicoso, C. Gehring, S. Leutenegger, M. Hutter, R. Siegwart, **"A Primer on the Differential Calculus of 3D Orientations,"** in arXiv:1606.05285, 2016.

Navigation


Laser Range Data
Localization
Elevation Mapping
Traversability Est.
Navigation Planning
Execution

Navigation
Laser-Based Localization (Iterative Closest Point (ICP))

Open Source
github.com/ethz-asl/ethzasl_icp_mapping

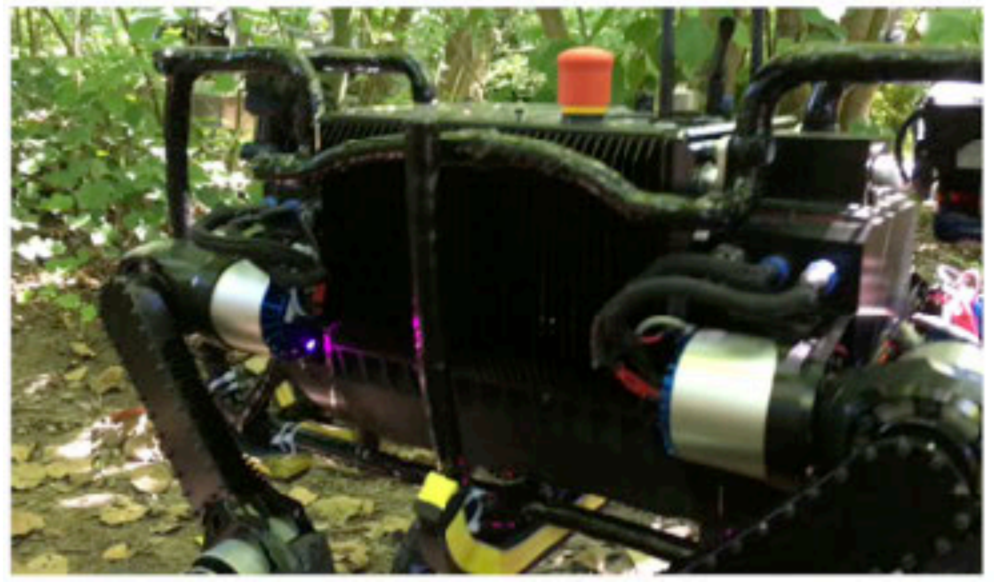


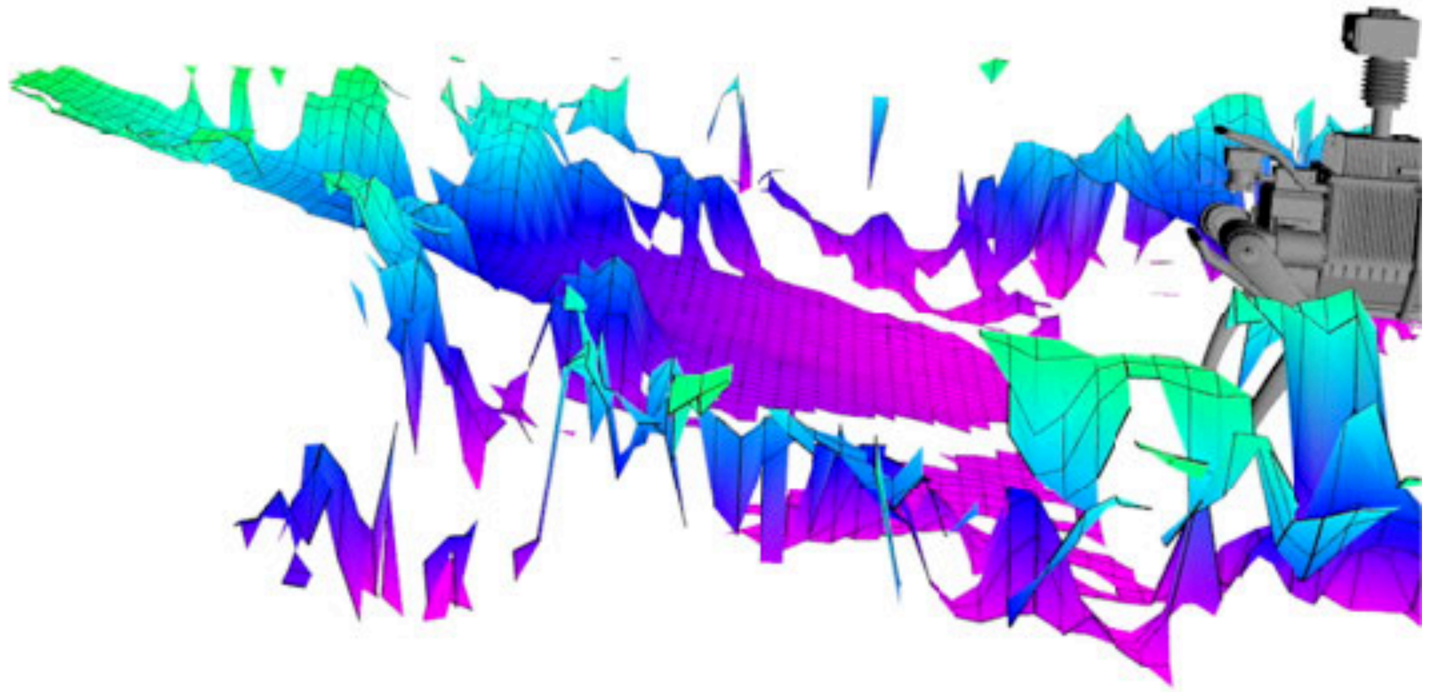
- Point cloud registration for localization in reference map
- Full rotation of LiDAR is aggregated for point cloud
- Use of existing maps or online mapping

Pomerleau, F., Colas, F., Siegwart, R., Magnenat, S., "**Comparing ICP variants on real-world data sets**", in Autonomous Robots, 2013.

Navigation
Elevation Mapping – Dense Terrain Mapping

Open Source
github.com/ethz-asl/elevation_mapping

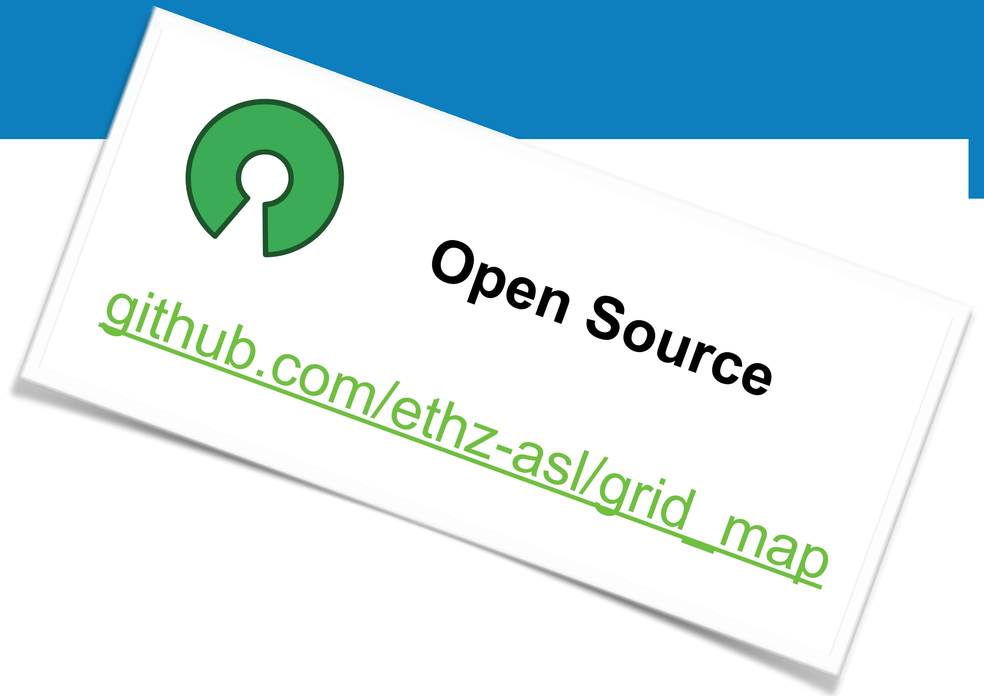


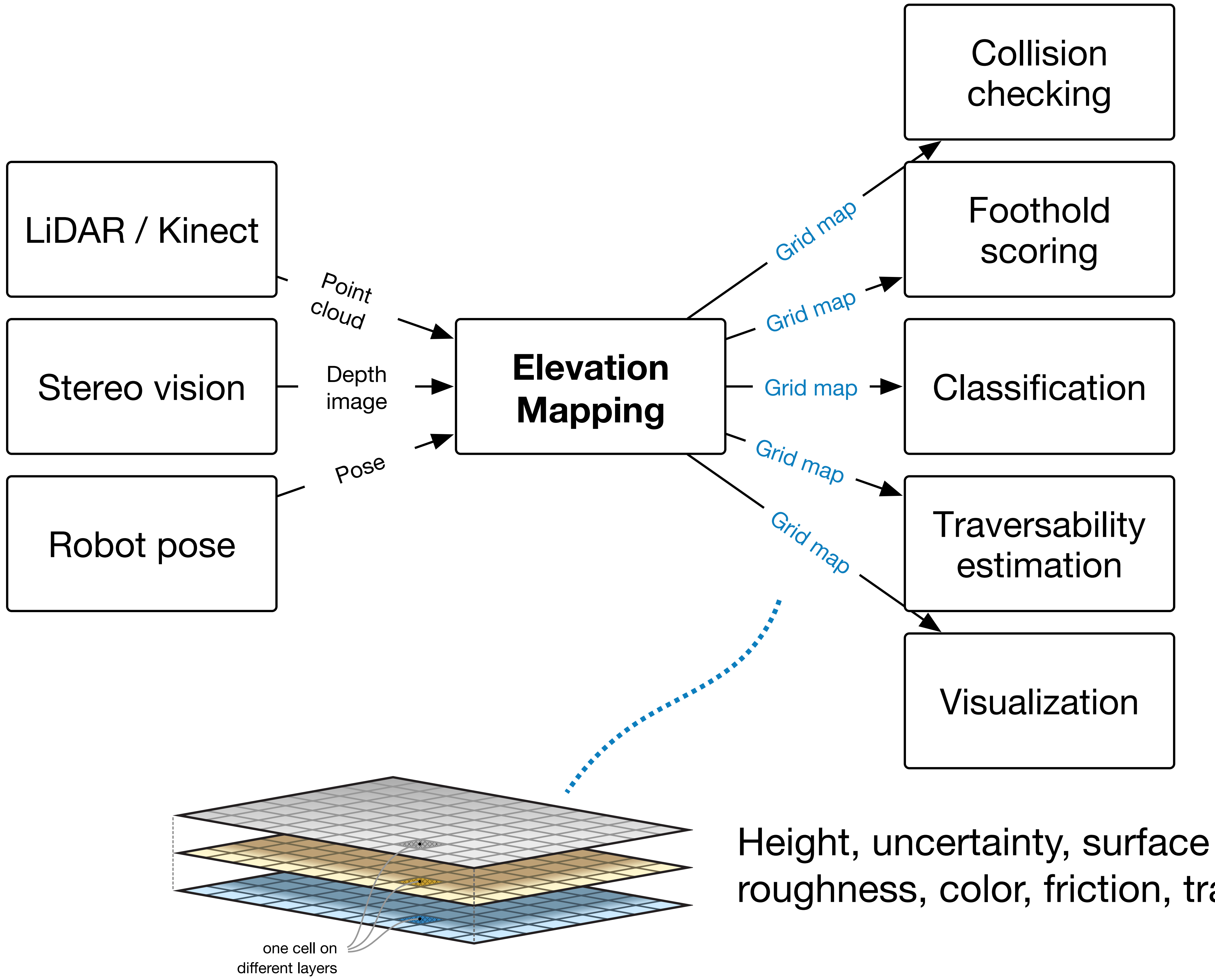


- Probabilistic fusion of range measurements and pose estimation
- Explicitly handles drift of state estimation (robot-centric)
- Input data from laser, Kinect, stereo cameras, Velodyne etc.

P. Fankhauser, M. Bloesch, C. Gehring, M. Hutter, R. Siegwart **"Robot-Centric Elevation Mapping with Uncertainty Estimates,"** in International Conference on Climbing and Walking Robots (CLAWAR), 2014.

Navigation
Grid Map – Universal Multi-Layer Grid Map Library



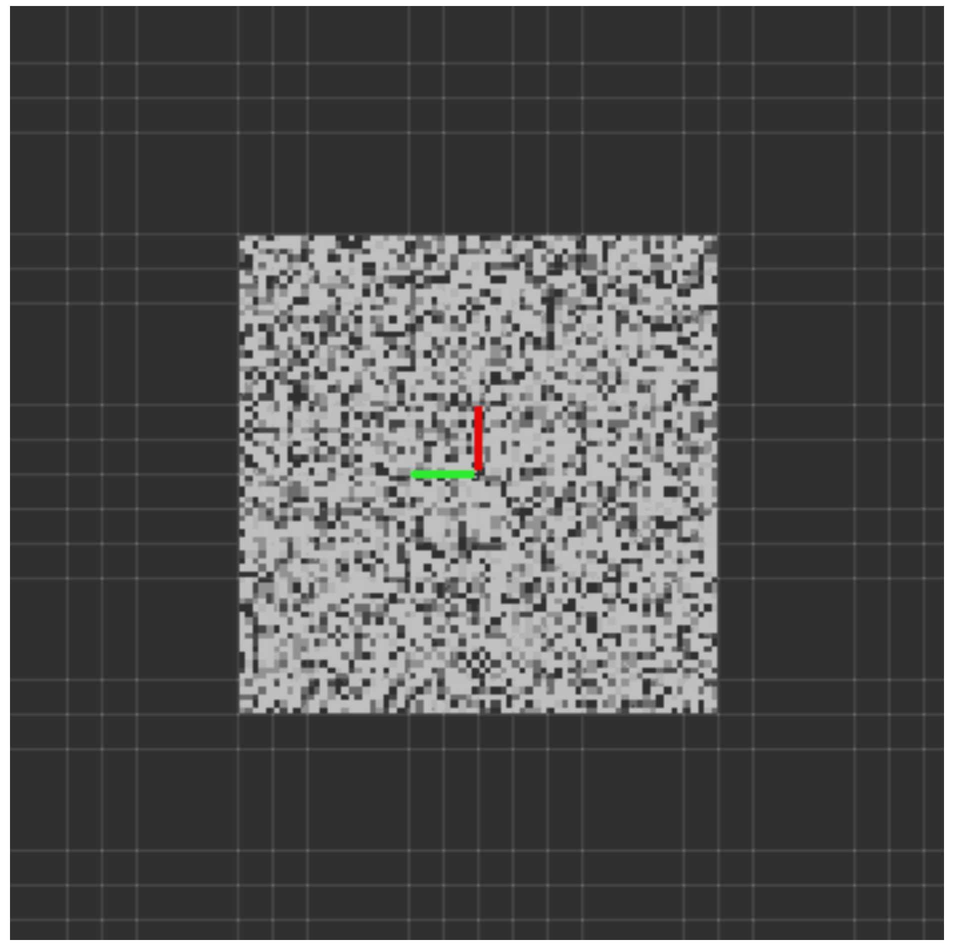


Height, uncertainty, surface normal, roughness, color, friction, traversability …

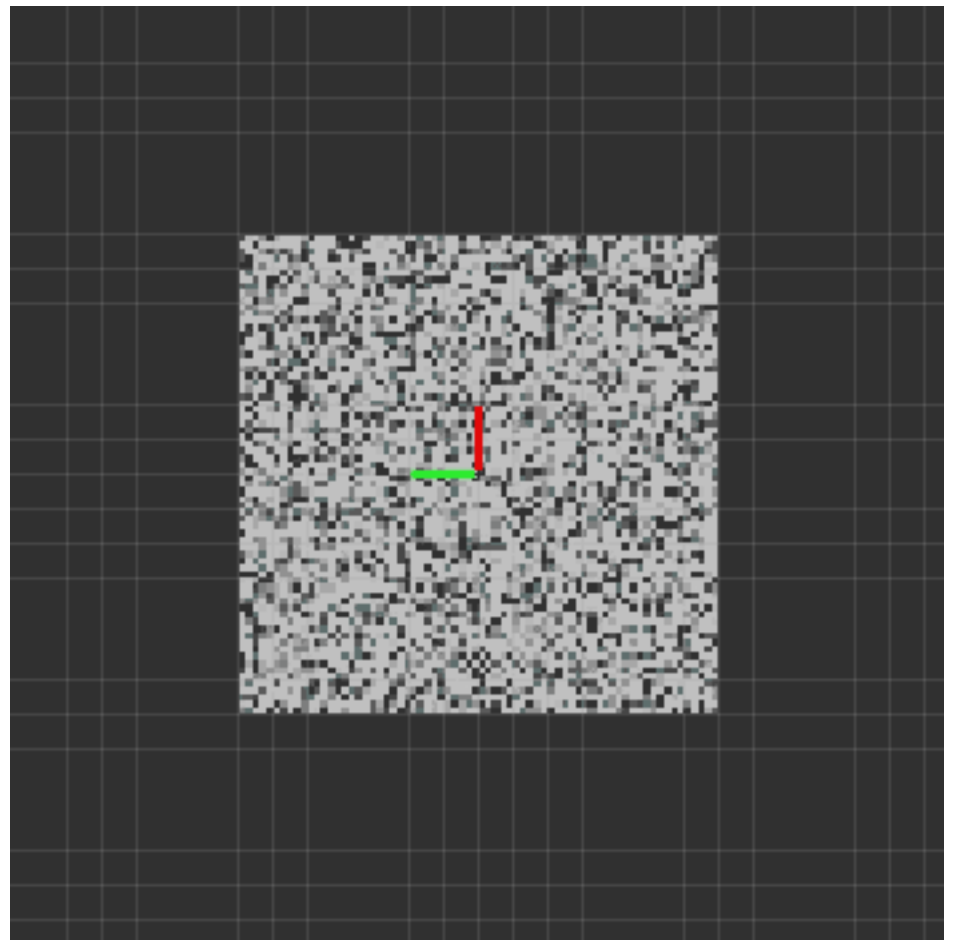
P. Fankhauser and M. Hutter, **"A Universal Grid Map Library: Implementation and Use Case for Rough Terrain Navigation,"** in Robot Operating System (ROS) - The Complete Reference, Springer, 2015.

Navigation
Grid Map – Universal Multi-Layer Grid Map Library

Open Source
github.com/ethz-asl/grid_map



`setPosition(…)`



`move(…)`

- 2D circular buffer data structure
  - ➡ Efficient map repositioning

P. Fankhauser and M. Hutter, **"A Universal Grid Map Library: Implementation and Use Case for Rough Terrain Navigation,"** in Robot Operating System (ROS) - The Complete Reference, Springer, 2015.

Navigation
Grid Map – Universal Multi-Layer Grid Map Library

Open Source
github.com/ethz-asl/grid_map

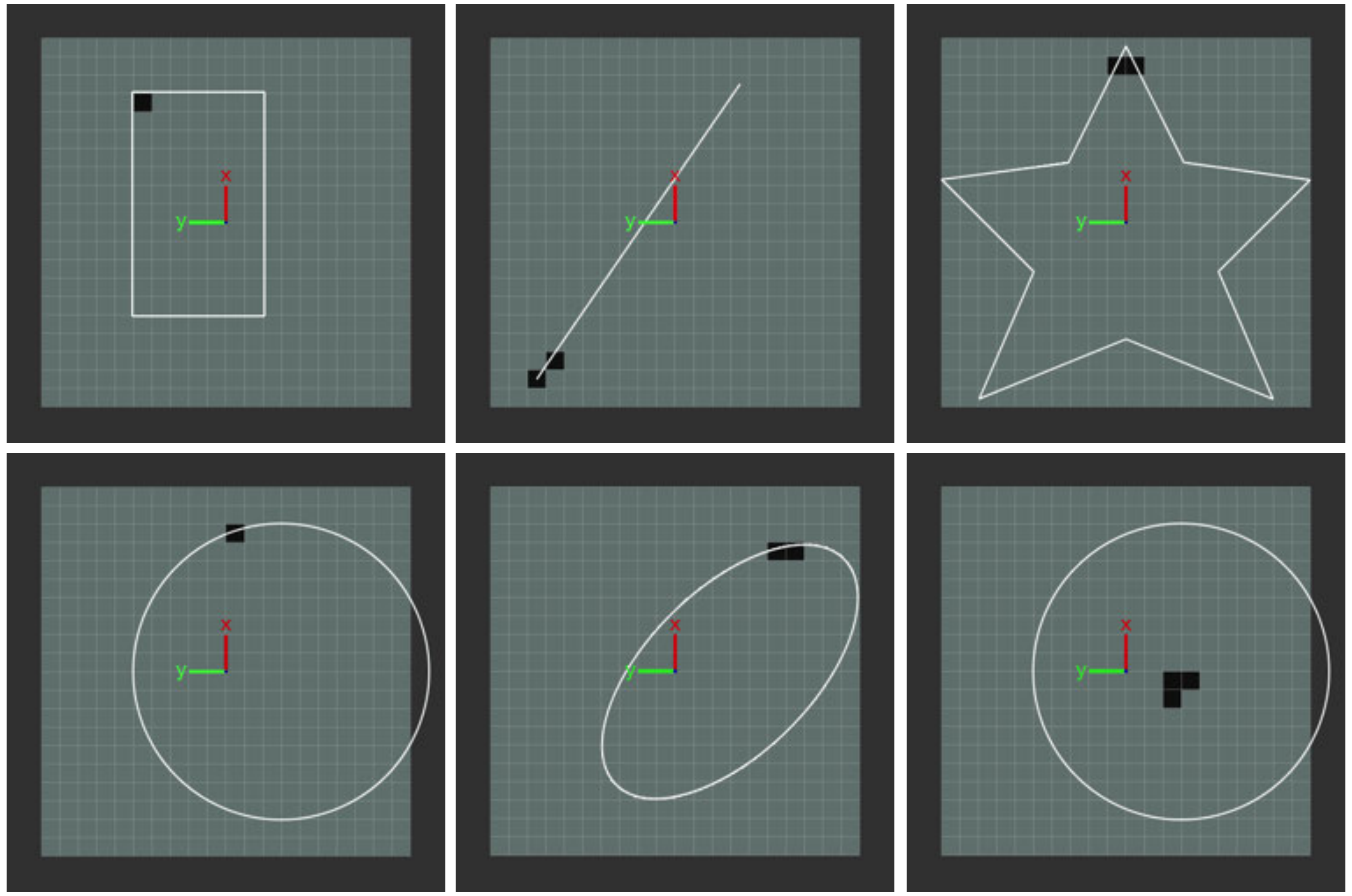
- 2D circular buffer data structure
  - ➡ Efficient map repositioning
- Based on Eigen (C++)
  - ➡ Versatile and efficient data manipulation

```
double rmse =
      sqrt(map["error"].array().pow(2).sum() / nCells);
```

P. Fankhauser and M. Hutter, **“A Universal Grid Map Library: Implementation and Use Case for Rough Terrain Navigation,”** in Robot Operating System (ROS) - The Complete Reference, Springer, 2015.

Navigation
Grid Map – Universal Multi-Layer Grid Map Library

Open Source
github.com/ethz-asl/grid_map

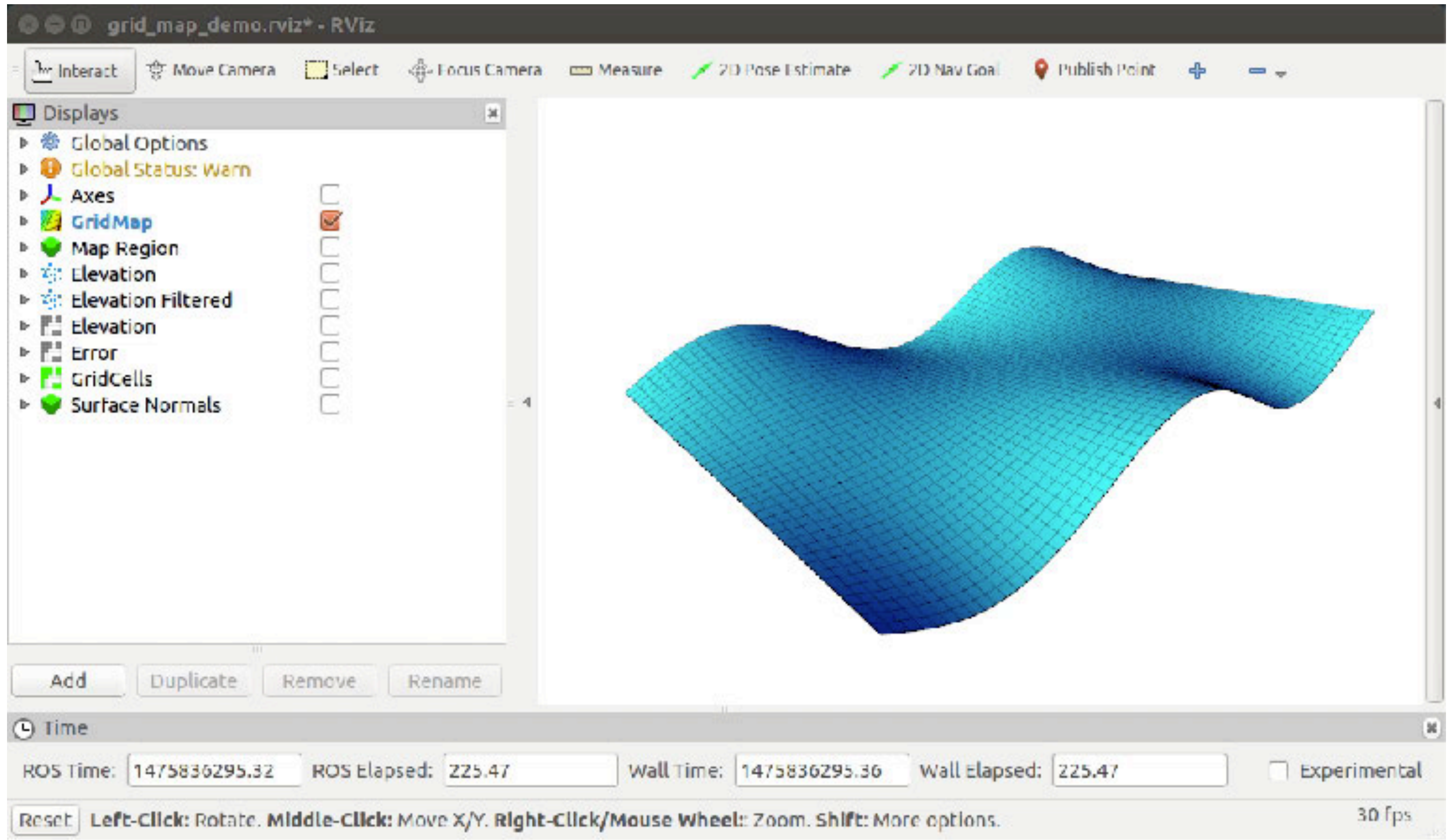


- 2D circular buffer data structure
  - ➡ Efficient map repositioning
- Based on Eigen (C++)
  - ➡ Versatile and efficient data manipulation
- Convenience functions
  - ➡ Iterators, math tools, etc.

P. Fankhauser and M. Hutter, **“A Universal Grid Map Library: Implementation and Use Case for Rough Terrain Navigation,”** in Robot Operating System (ROS) - The Complete Reference, Springer, 2015.

Navigation
Grid Map – Universal Multi-Layer Grid Map Library

Open Source
github.com/ethz-asl/grid_map

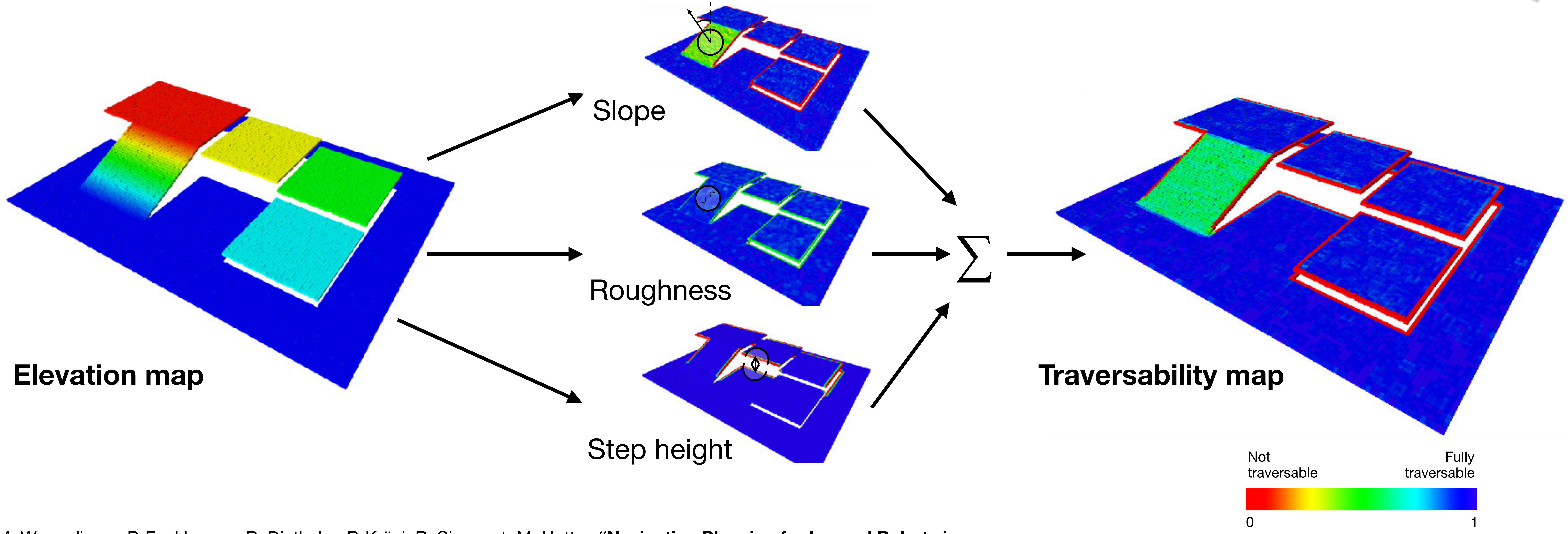


- 2D circular buffer data structure
  - ➡ Efficient map repositioning
- Based on Eigen (C++)
  - ➡ Versatile and efficient data manipulation
- Convenience functions
  - ➡ Iterators, math tools, etc.
- ROS & OpenCV interfaces
  - ➡ Conversion from/to images, point clouds, occupancy grids, grid cells

P. Fankhauser and M. Hutter, **"A Universal Grid Map Library: Implementation and Use Case for Rough Terrain Navigation,"** in Robot Operating System (ROS) - The Complete Reference, Springer, 2015.

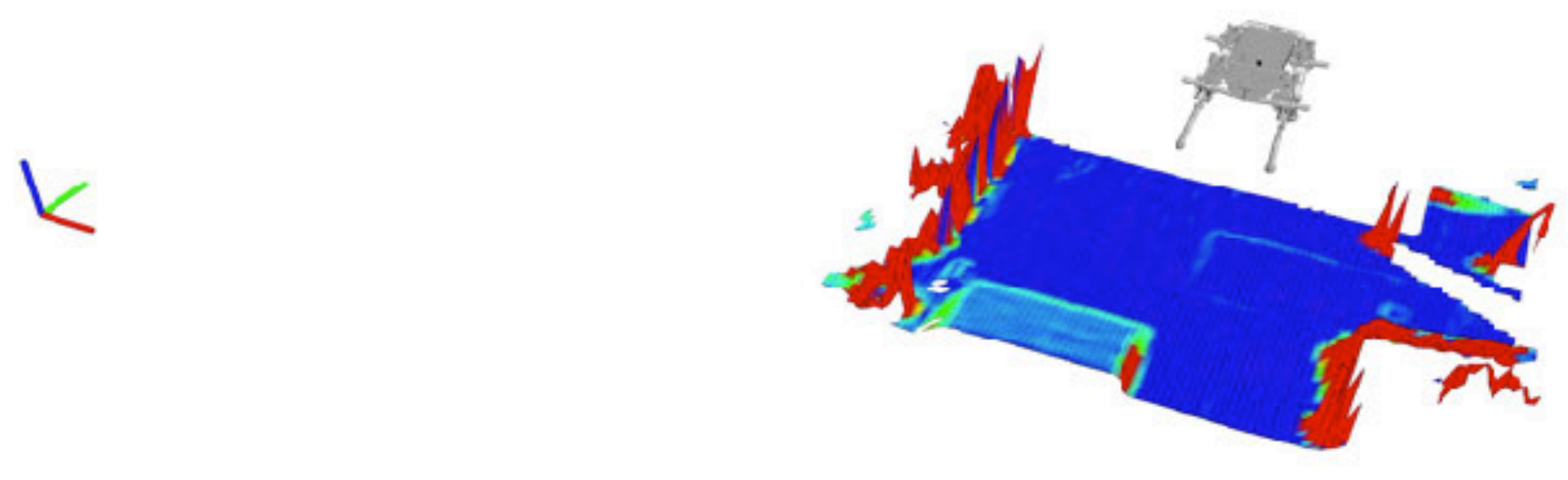
Navigation
Traversability Estimation

Open Source
github.com/ethz-asl/traversability_estimation



M. Wermelinger, P. Fankhauser, R. Diethelm, P. Krüsi, R. Siegwart, M. Hutter, **“Navigation Planning for Legged Robots in Challenging Terrain,”** in IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), 2016.

Navigation
Navigation Planning

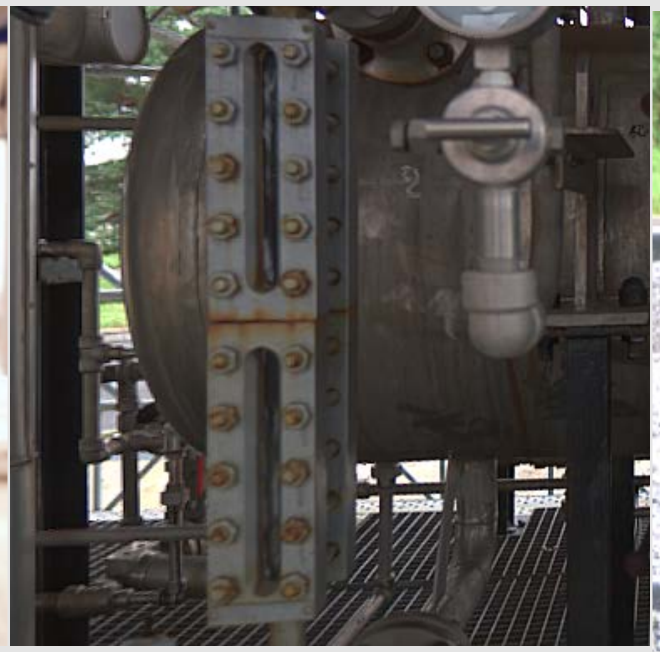


- Online navigation planning based on RRT* (OMPL)
- Works with and without initial map
- Continuous for changing environments

M. Wermelinger, P. Fankhauser, R. Diethelm, P. Krüsi, R. Siegwart, M. Hutter, **“Navigation Planning for Legged Robots in Challenging Terrain,”** in IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), 2016.

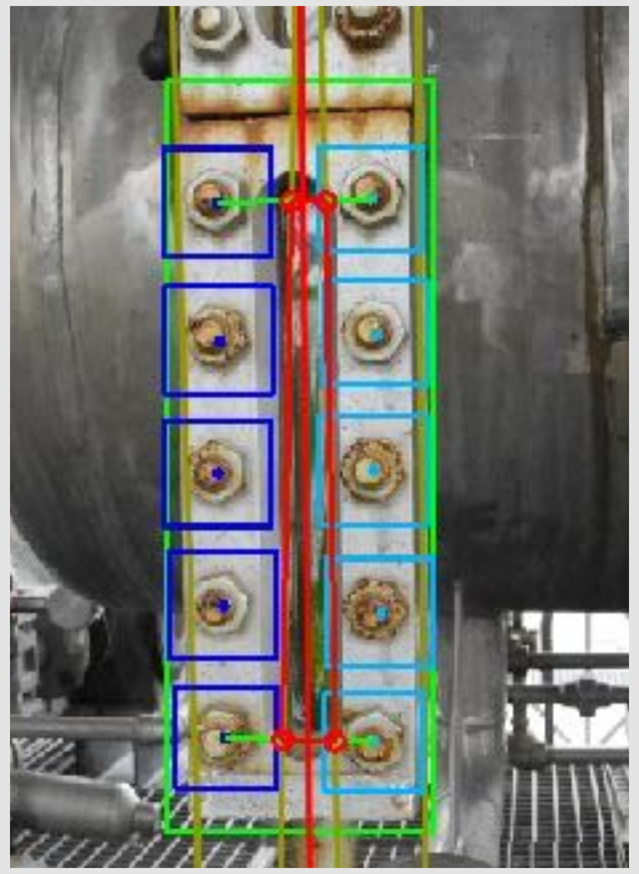
Inspection

Visual inspection

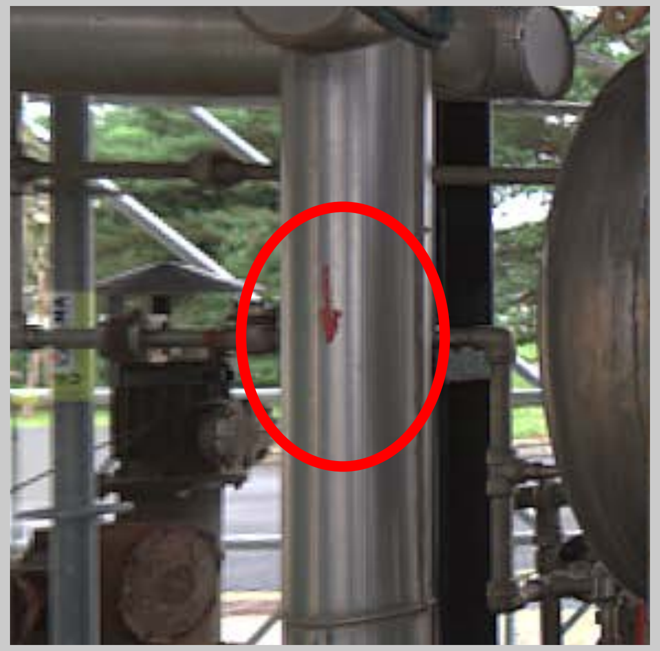



Pressure & Level gauges Valves


Zoom-camera

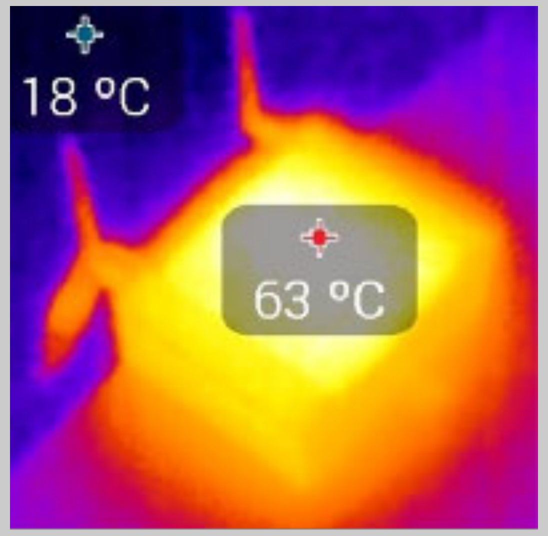


Thermal Inspection



Thermal points


Thermal-camera

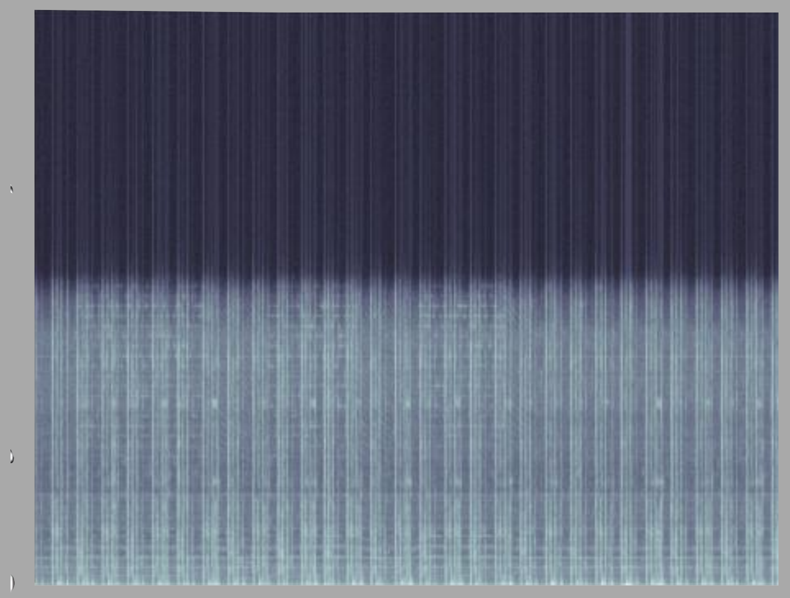


Auditive Inspection



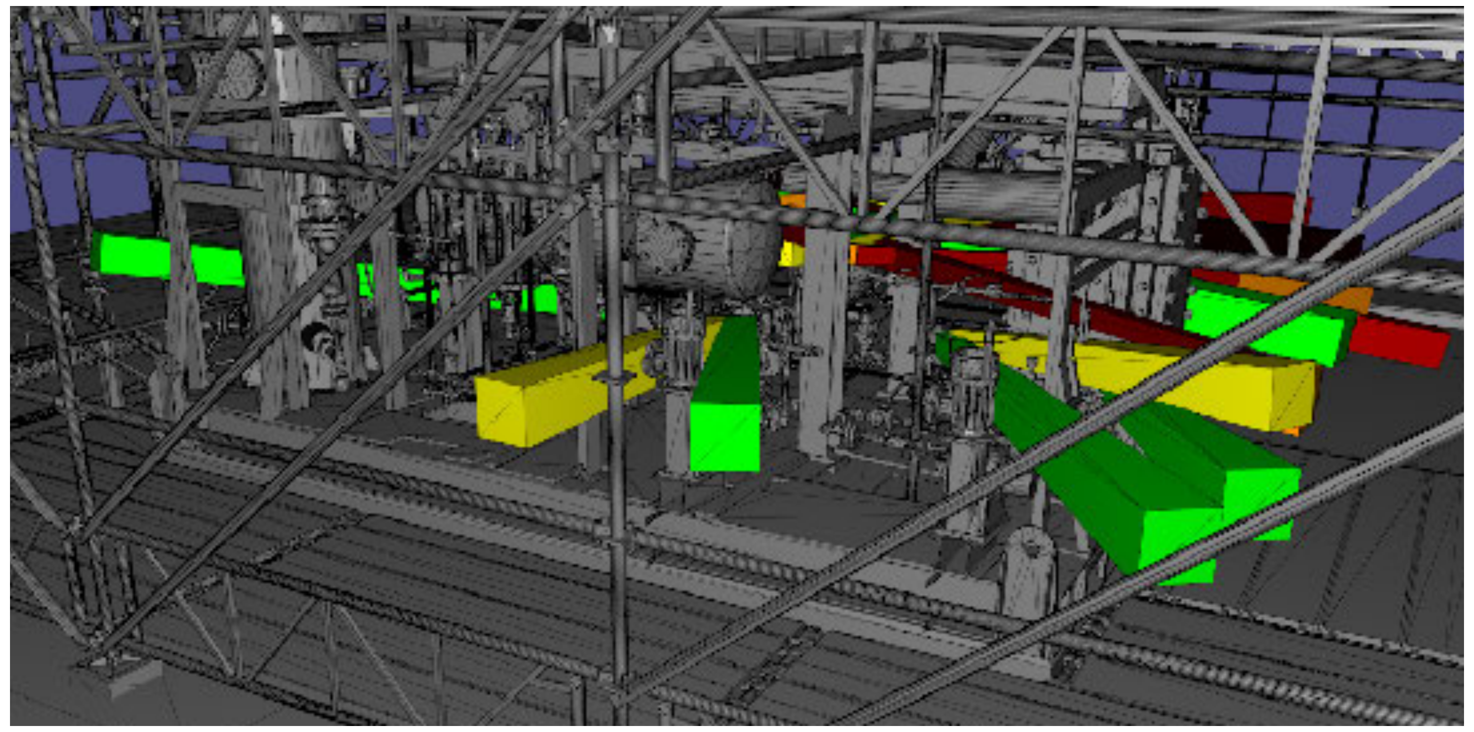
Pumps Gas leaks Platform alarm


Microphones (audible and ultra-sonic)

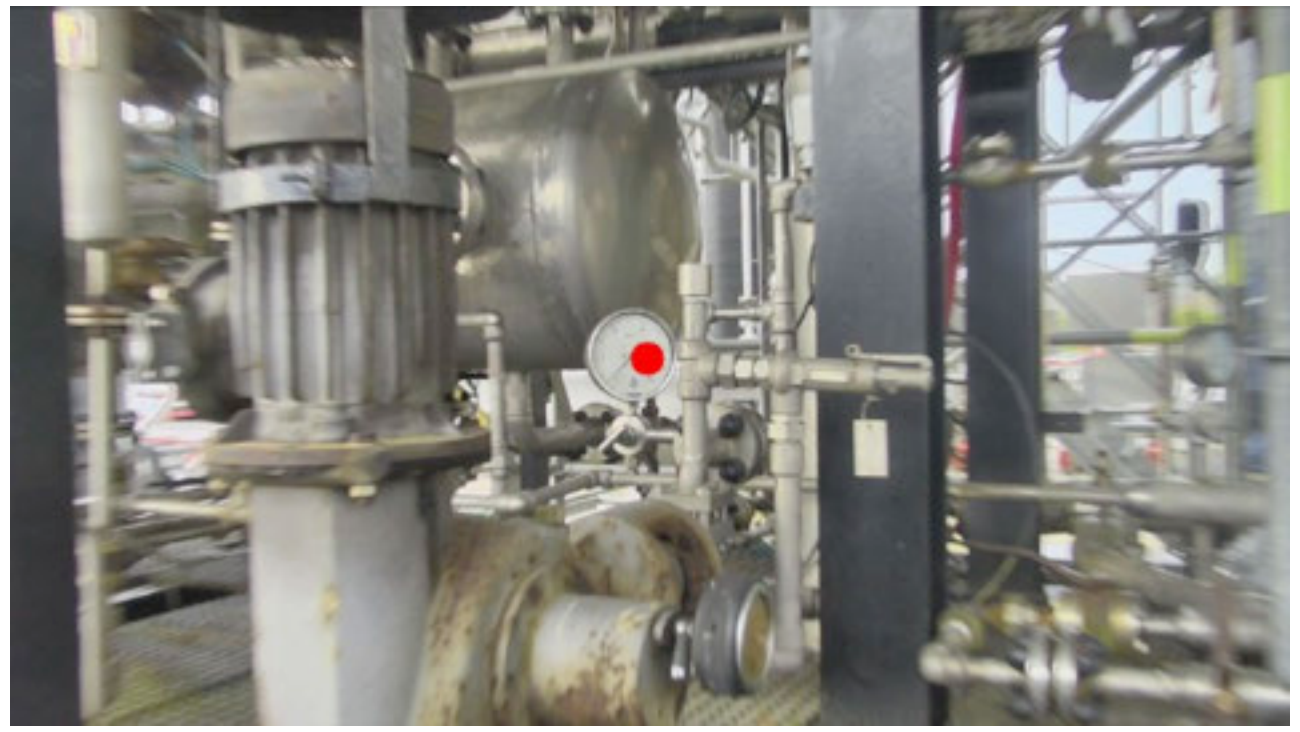


Inspection
Visual Inspection of Pressure Gauges

A Automatic view point generation

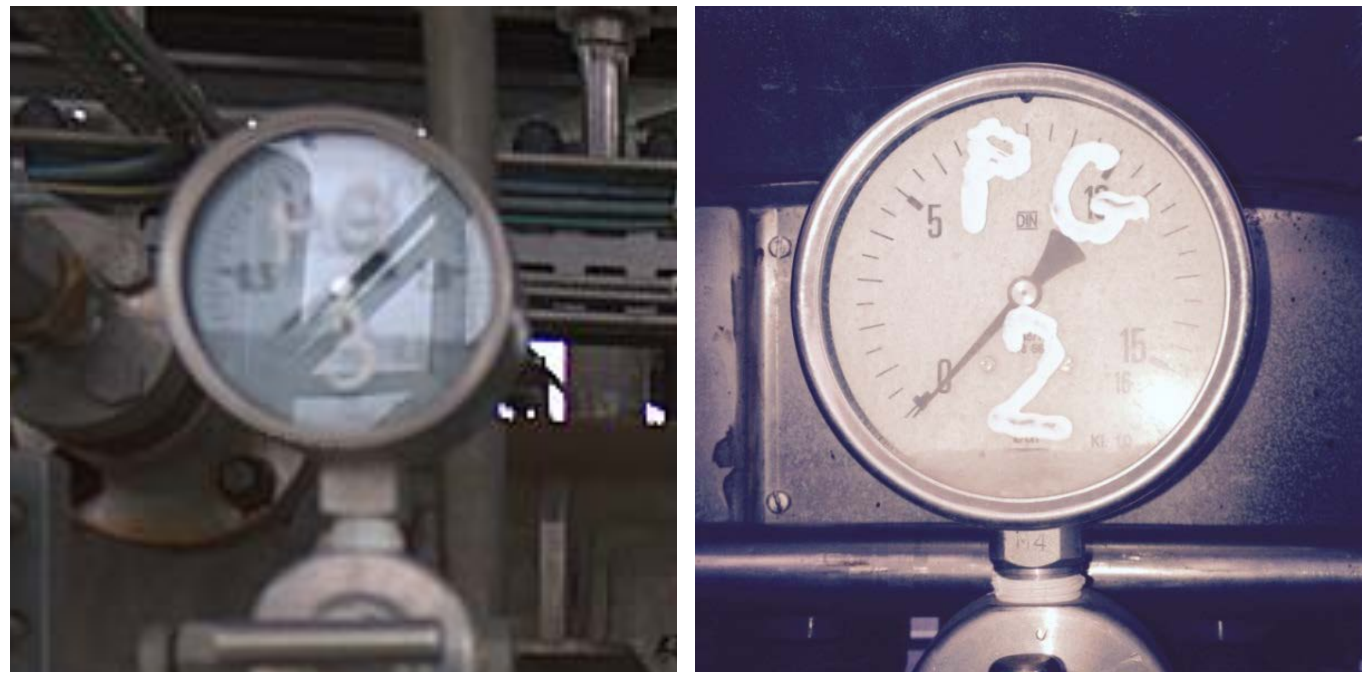


C Visual servoing



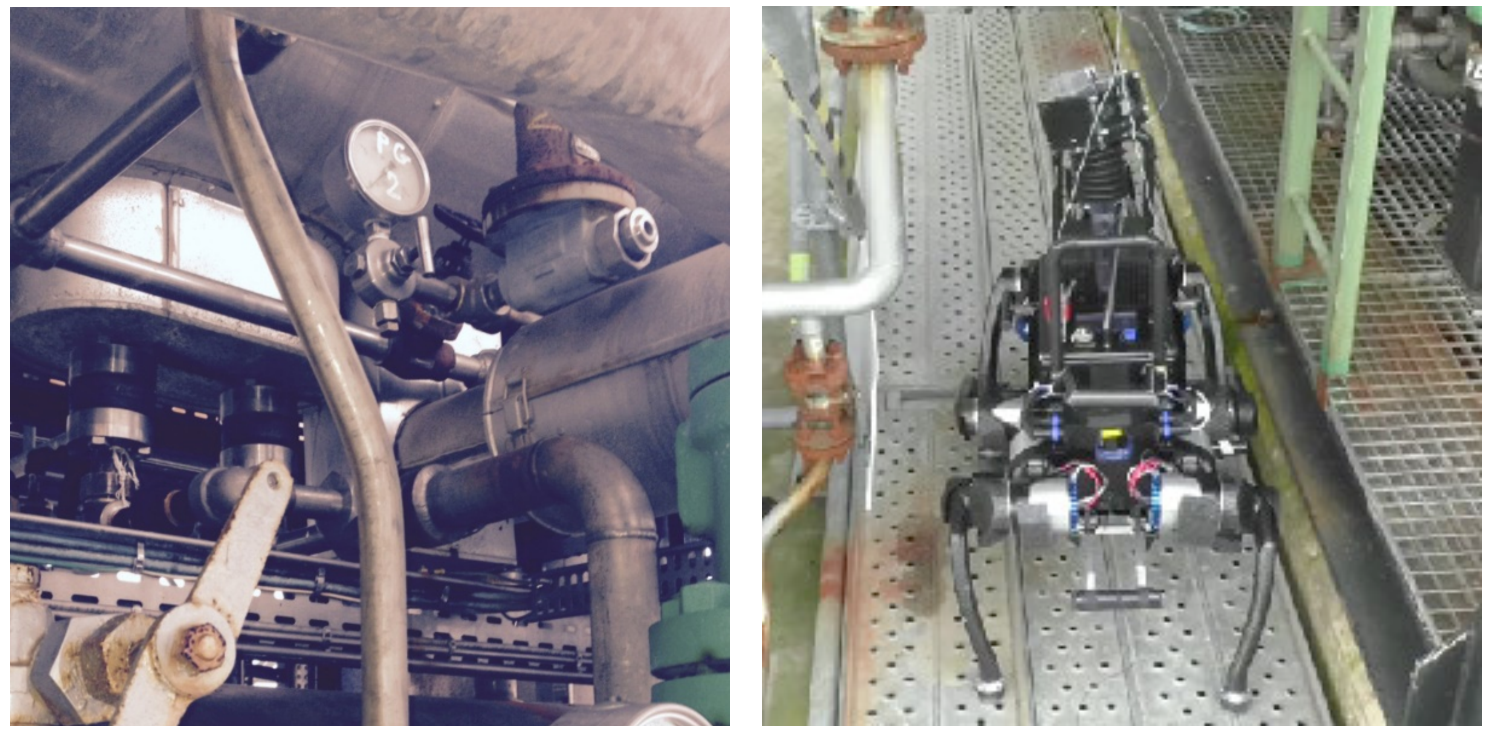
E Indicator reading

➔ Reading ok

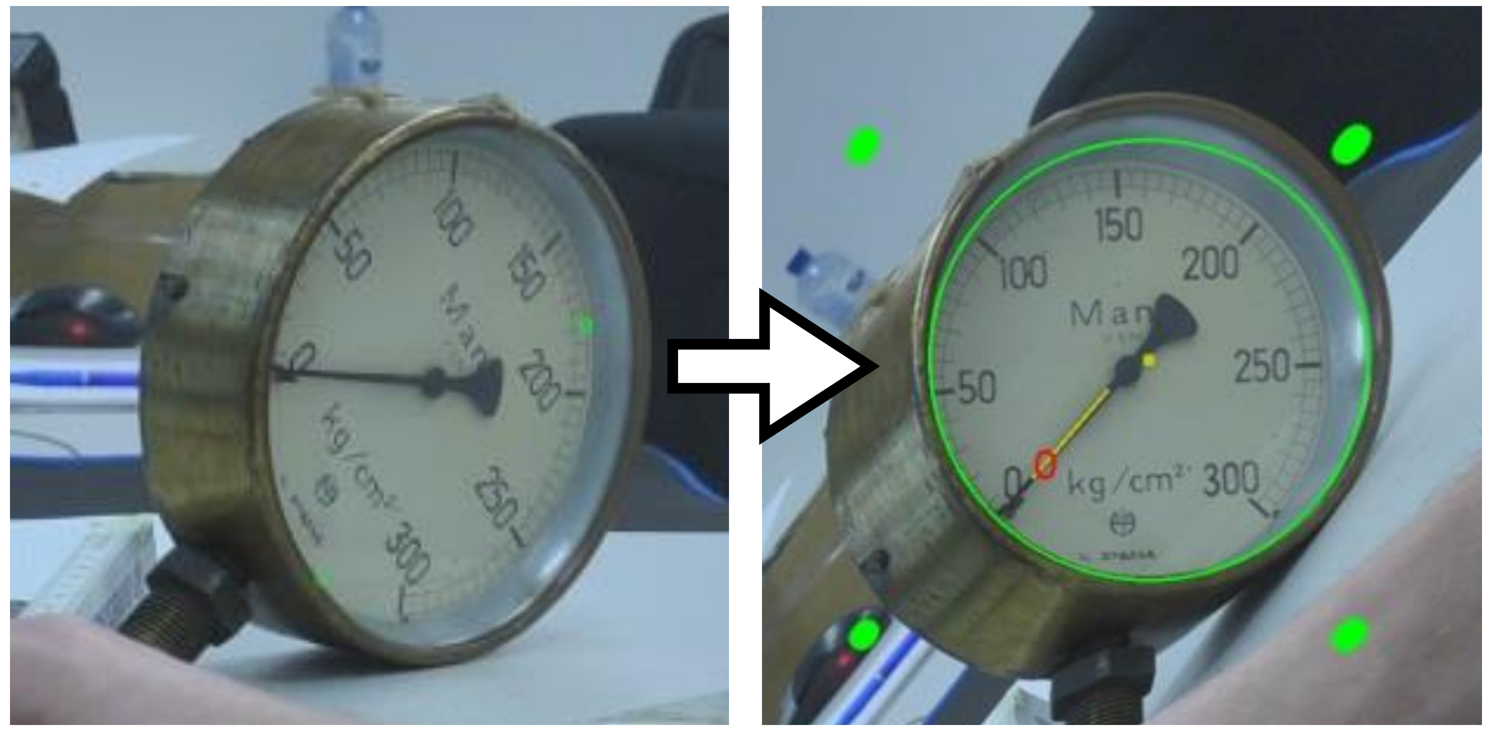


➔ Reading unsuccessful, try alternative position or report as unknown

B Whole-body camera positioning



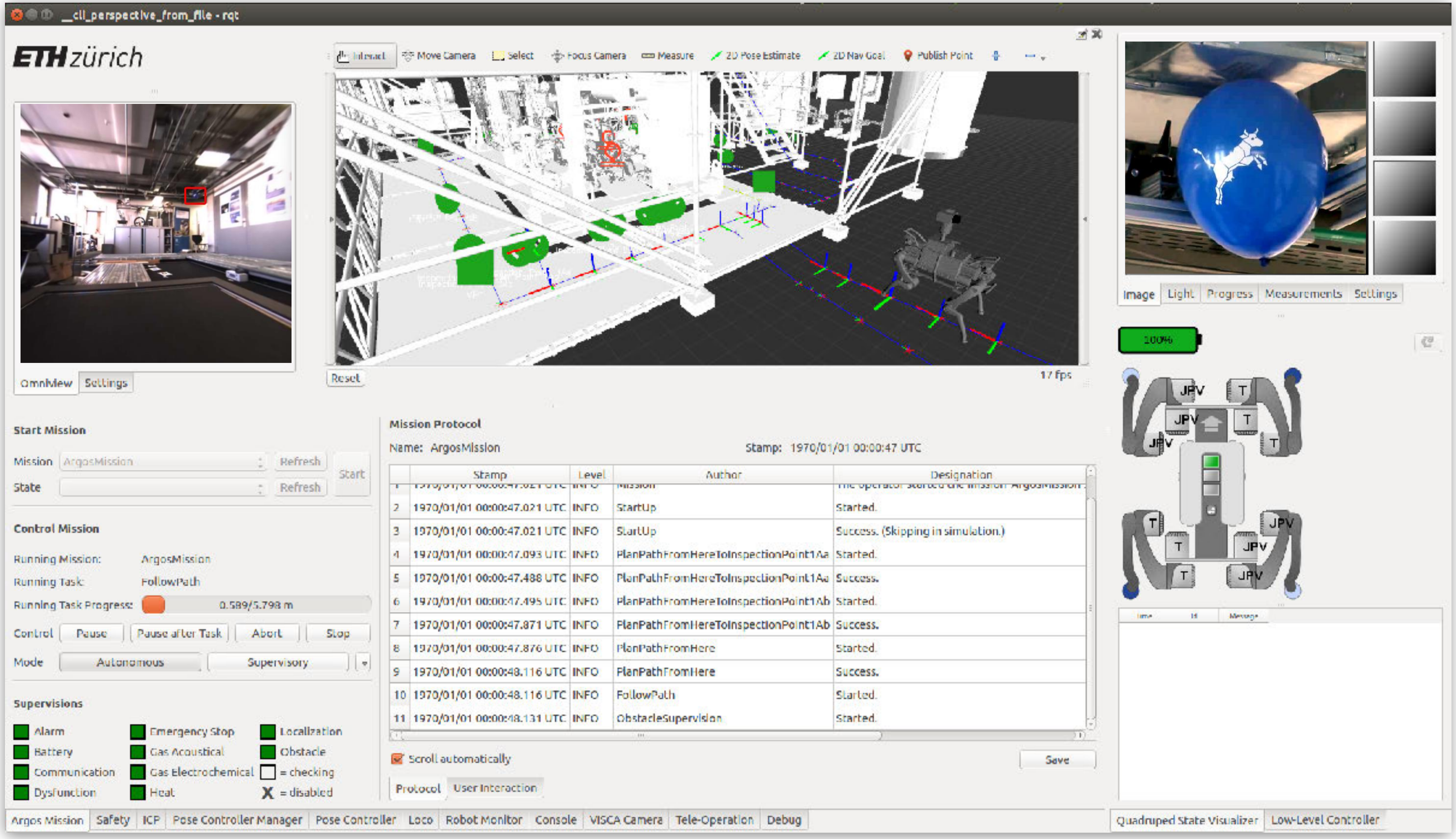
D Image de-warping



S. Bachmann, “**Visual Inspection of Manometers and Valve Levers**”, Master’s Thesis, ETH Zurich, 2015.

User Interface

Interface for remote control, semi-, and full autonomous operation.



User Interface

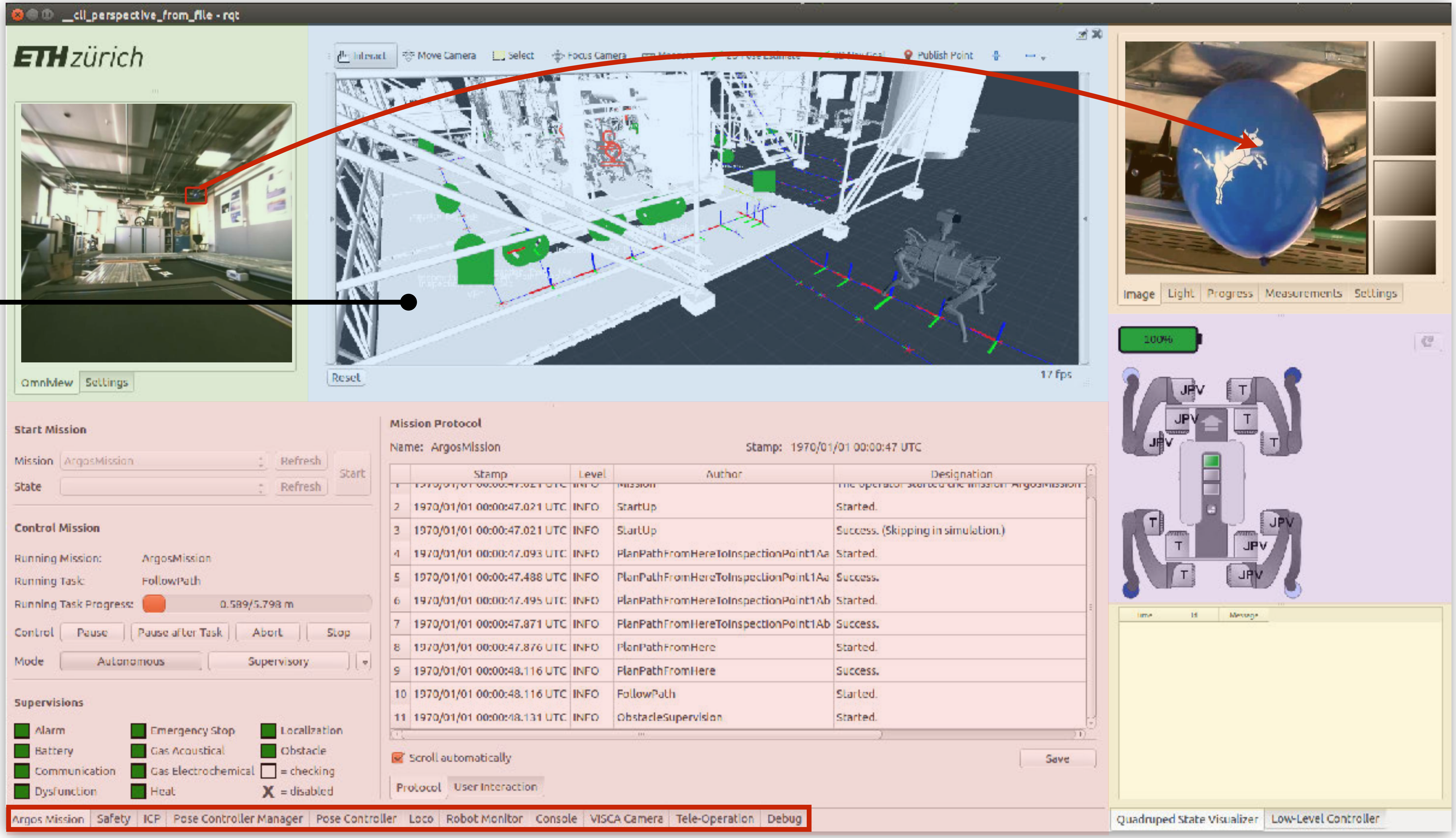
Interface for remote control, semi-, and full autonomous operation.

Situational camera

3D view (RViz)

Mission control & protocol

Other modules

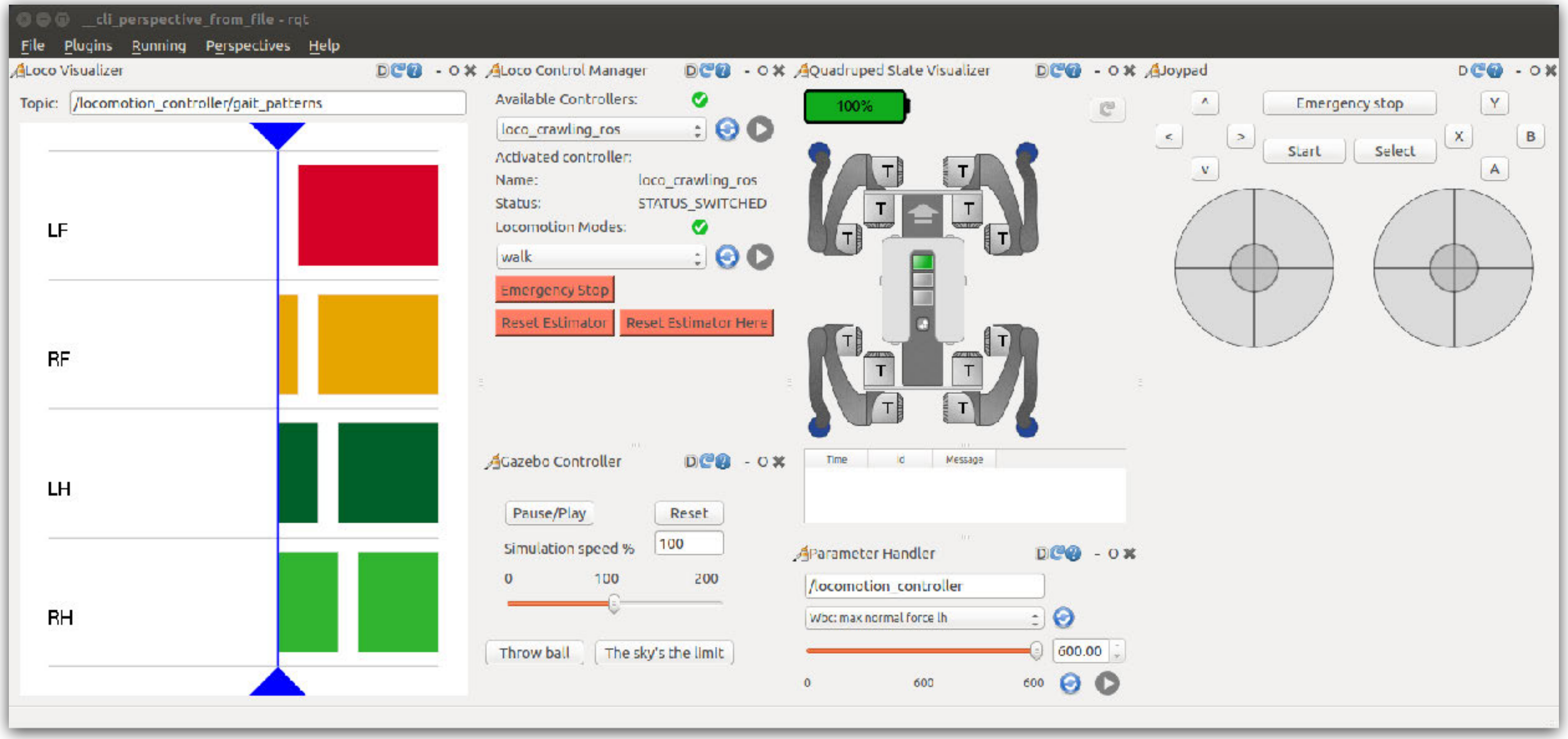


Inspection cameras

Robot actuators & sensors

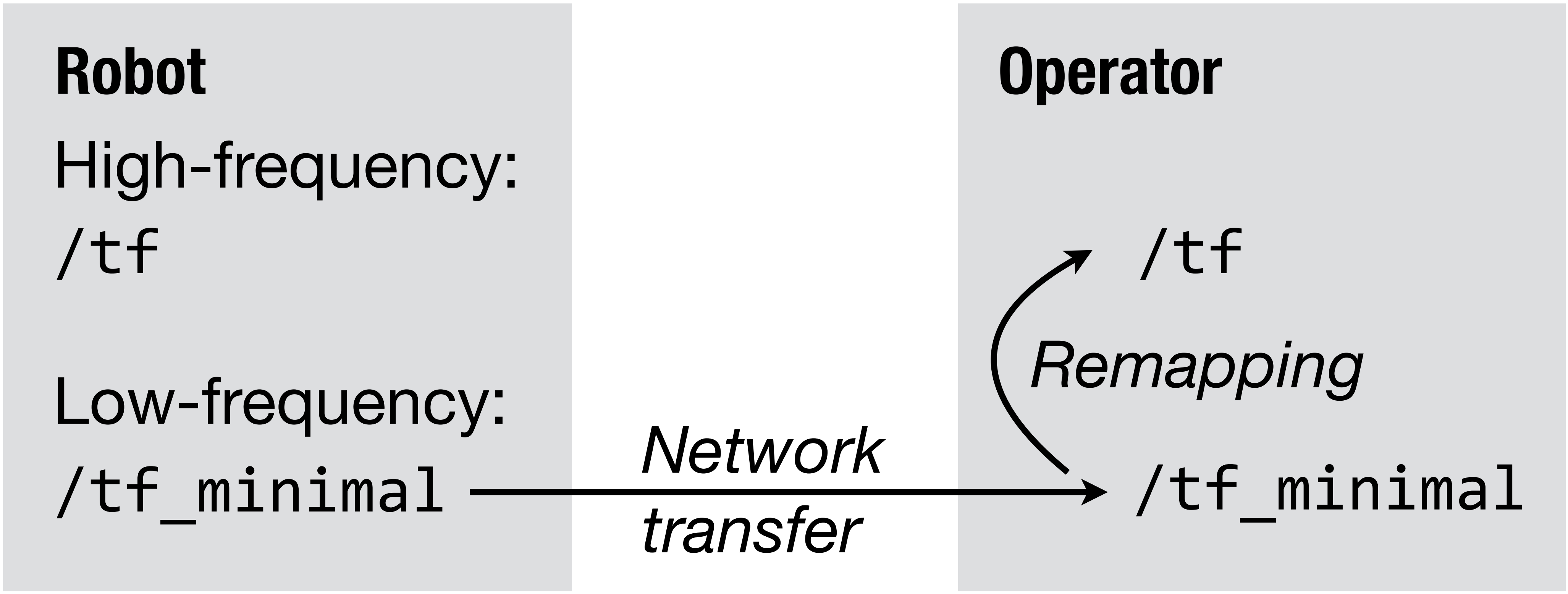
Error protocol

User Interface

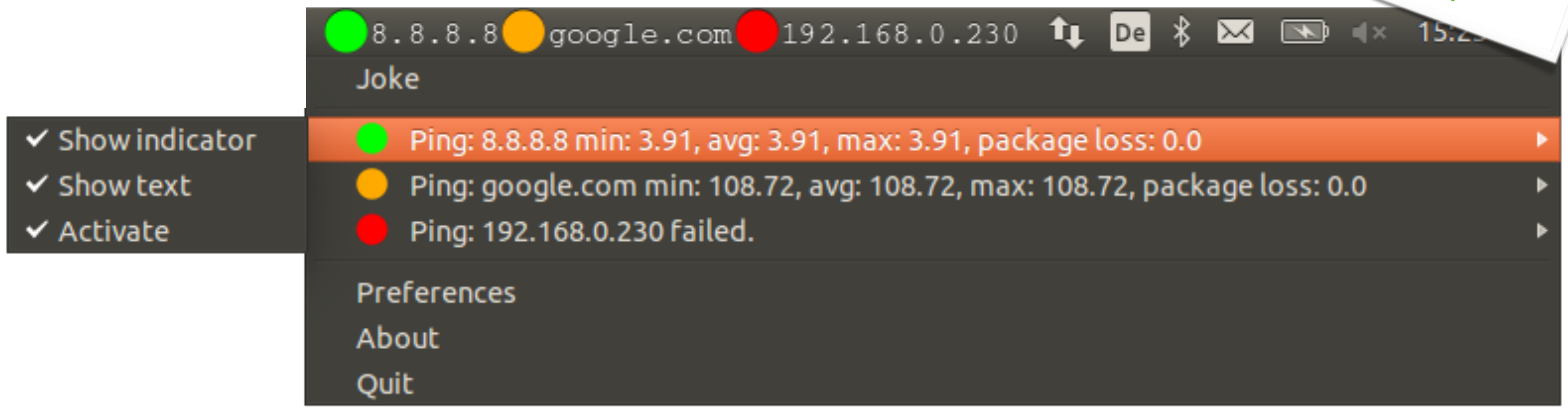


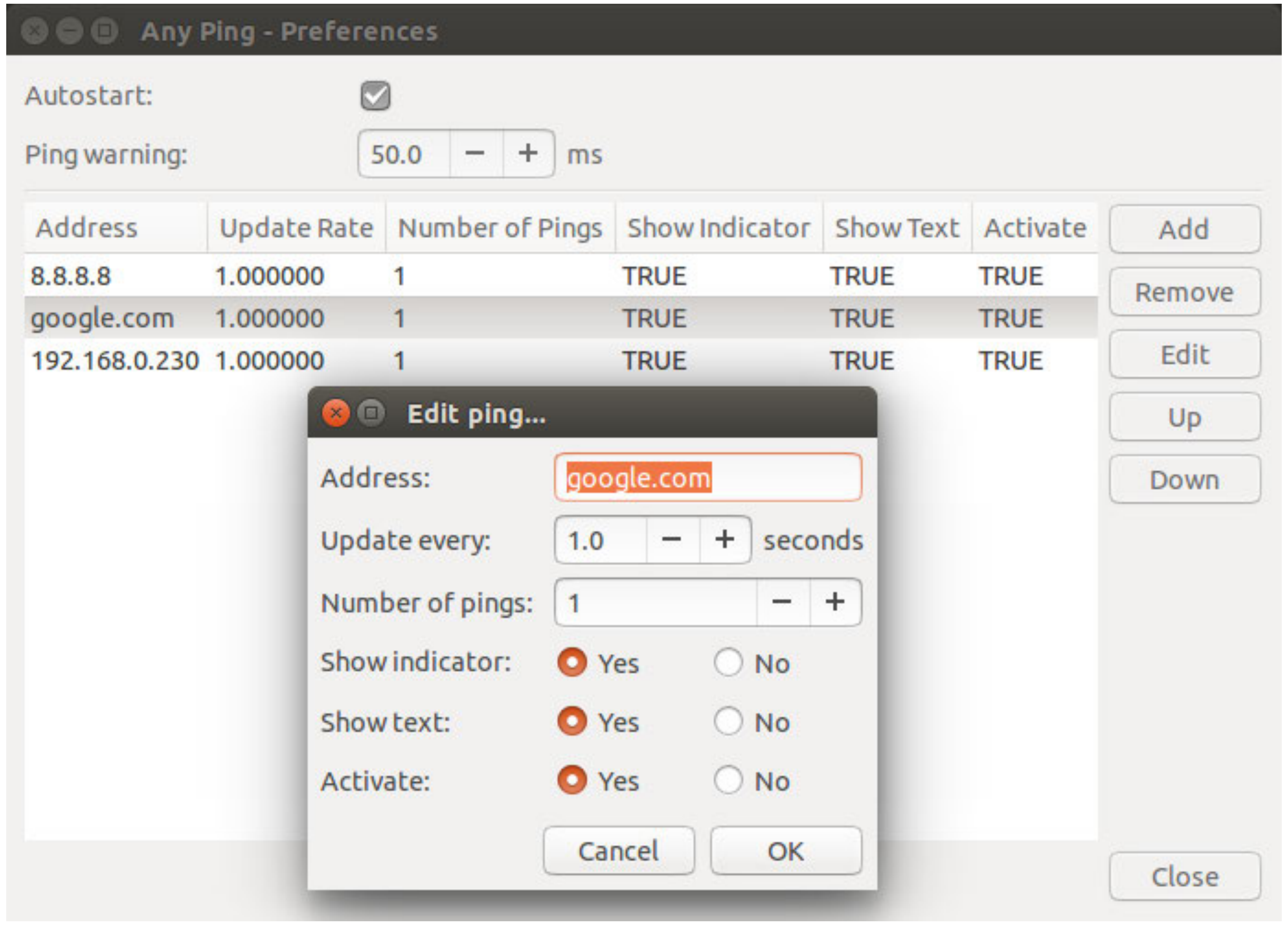
User Interface
Bandwidth Considerations

- Only critical data is transmitted by default (robot state and position)
- Other data is transmitted on demand (video, maps, etc.)
- Separation of onboard TF and operator TF
- Connection status node monitors WiFi status and triggers recovery behavior



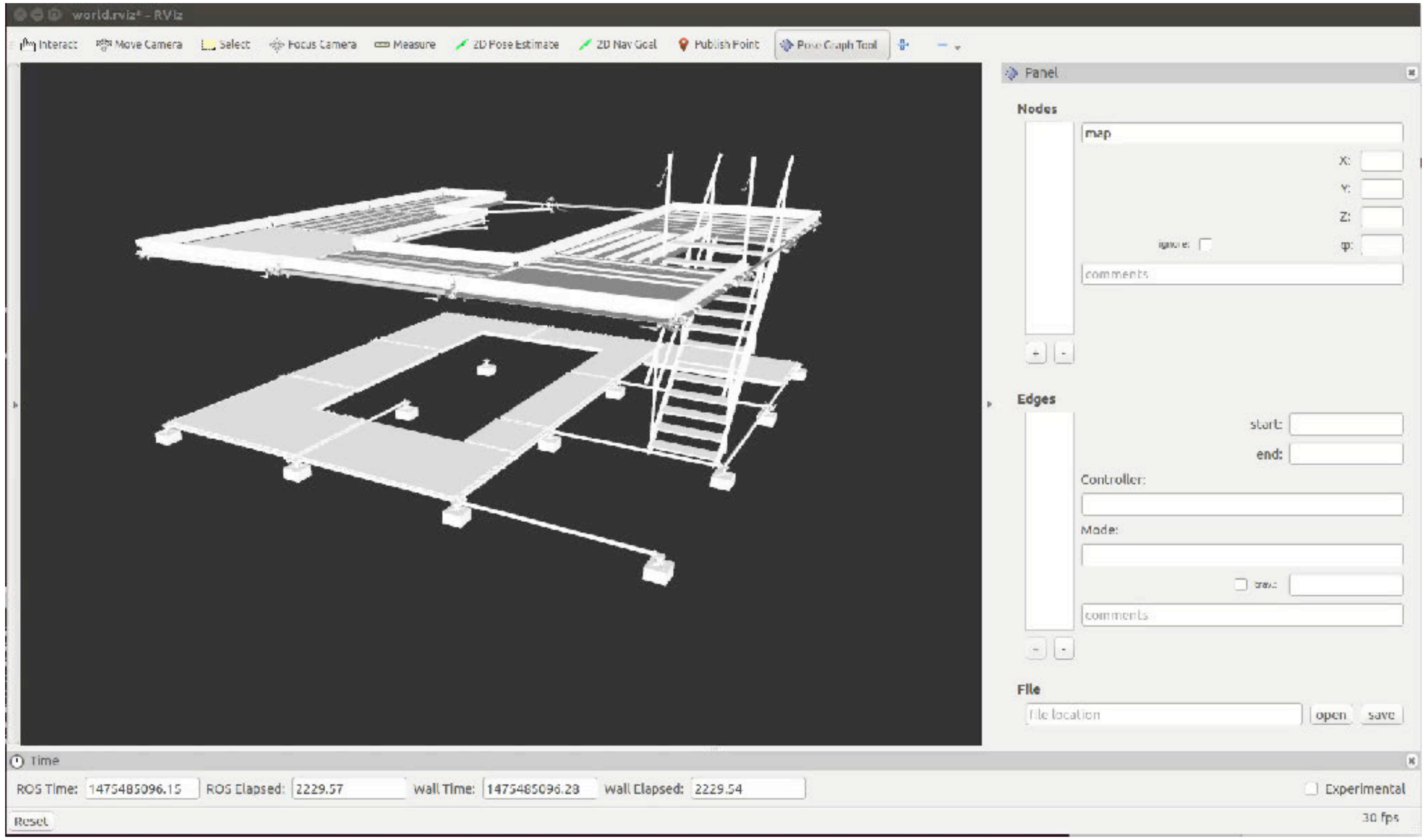
User Interface
ANYping Indicator





- Indicates PC network availability in Ubuntu menu bar

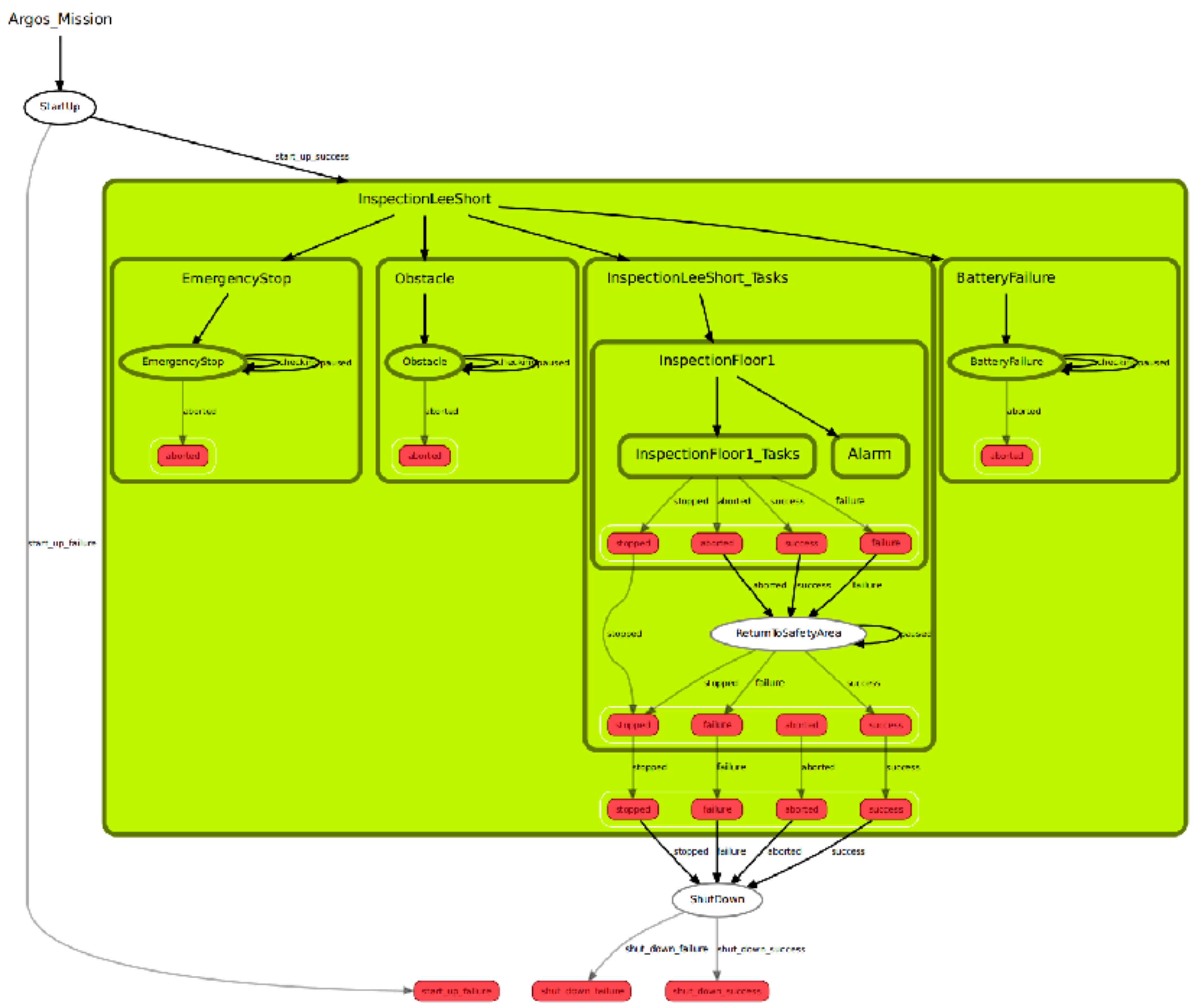
User Interface
Pose Graph



- Pose graph for inspection, special maneuvers (e.g. stairs), docking station etc.
- Visualization and interactive editing of pose graph
- Continuous updating and (re-)planning on pose graph during mission

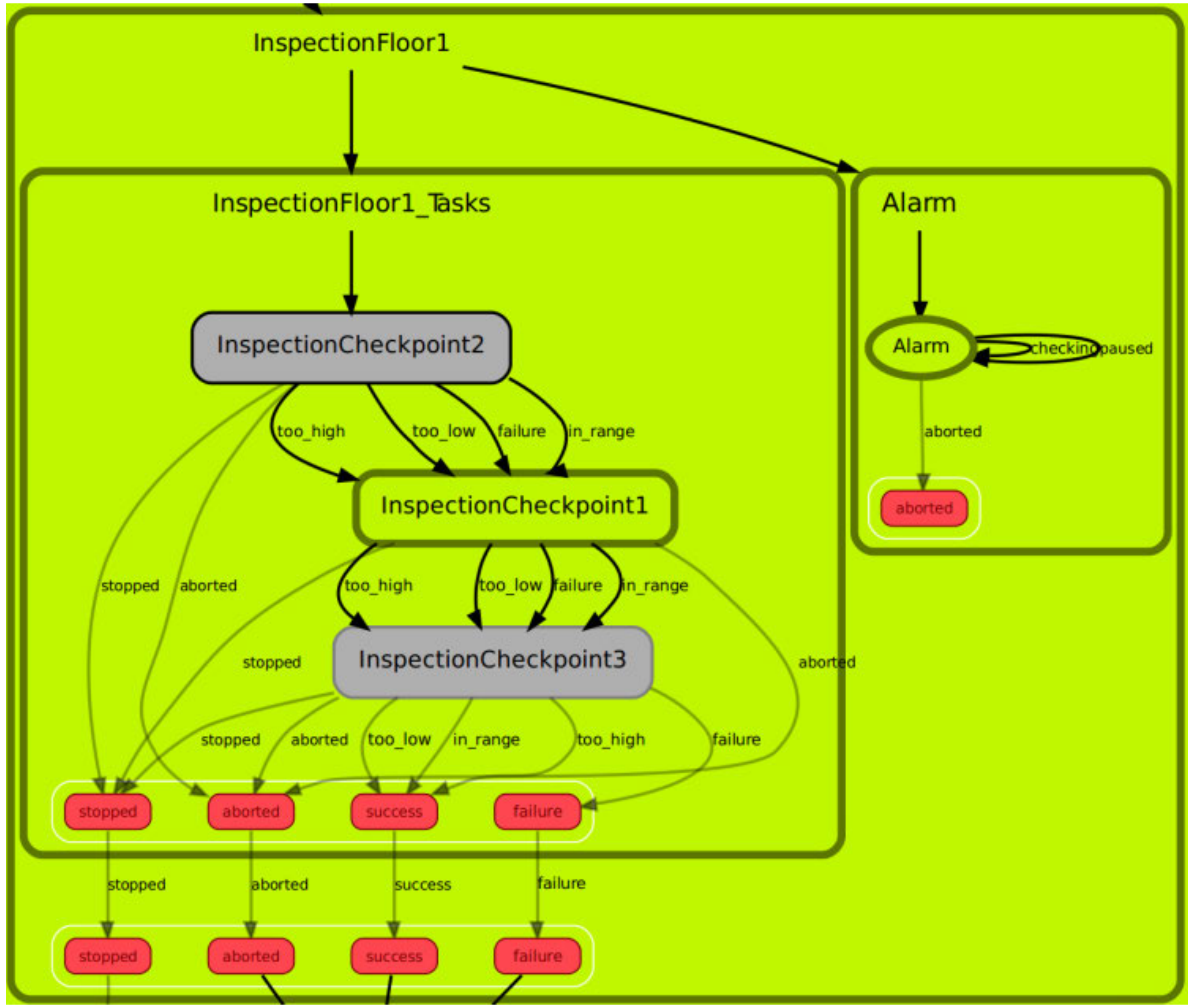
User Interface
Mission Creation

- Task-level state machine (C++ library, similar to SMACH)
- State machine defined in YAML format
- Common building blocks to facilitate construction



User Interface
Mission Creation

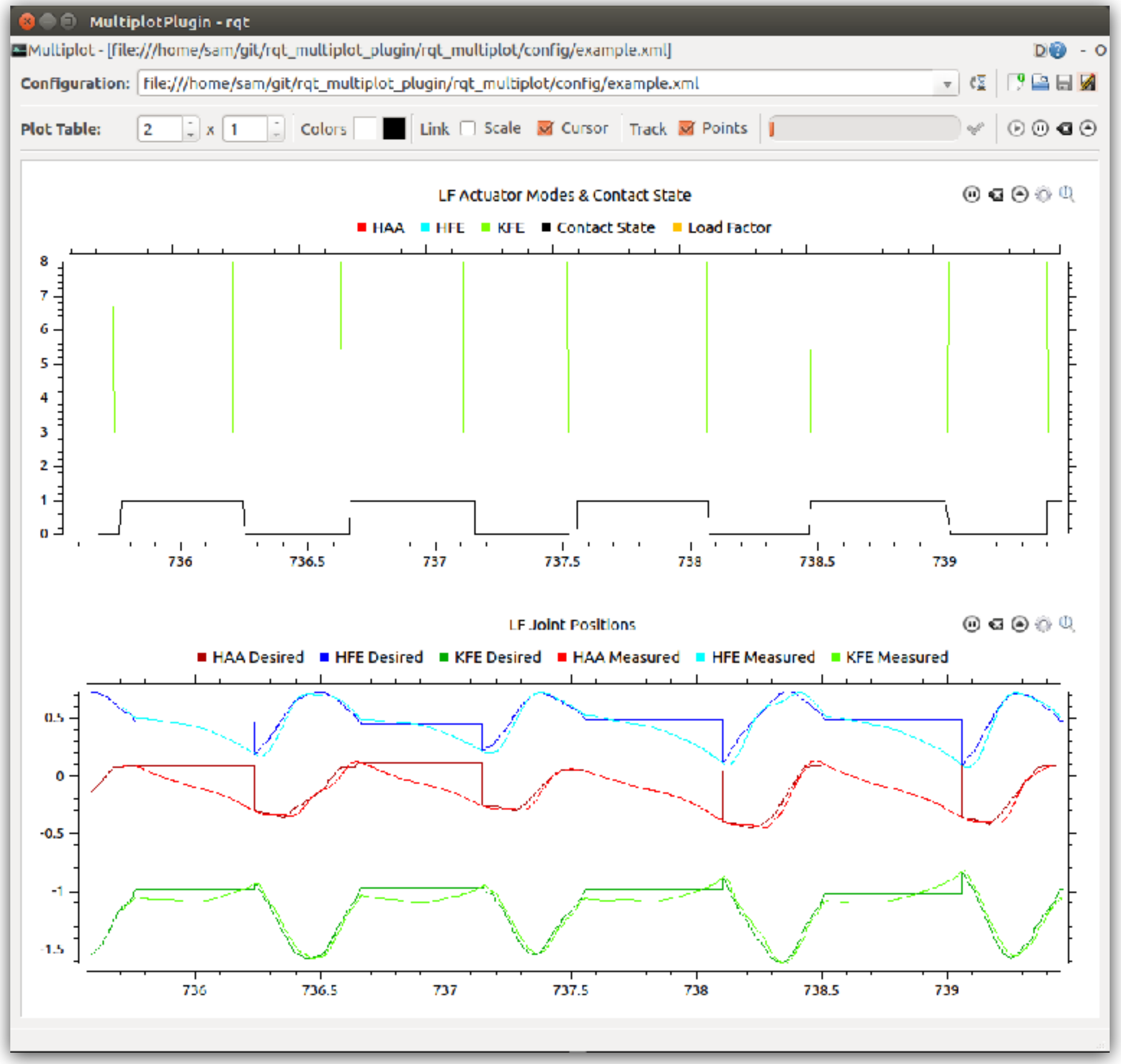
- Task-level state machine (C++ library, similar to SMACH)
- State machine defined in YAML format
- Common building blocks to facilitate construction
- Typical missions programmed in 5–20 minutes



Open Source
github.com/ethz-asl/rqt_multiplot_plugin

Open Source
github.com/ethz-asl/variant

RQT Multiplot Plugin & Variant Topic Tools

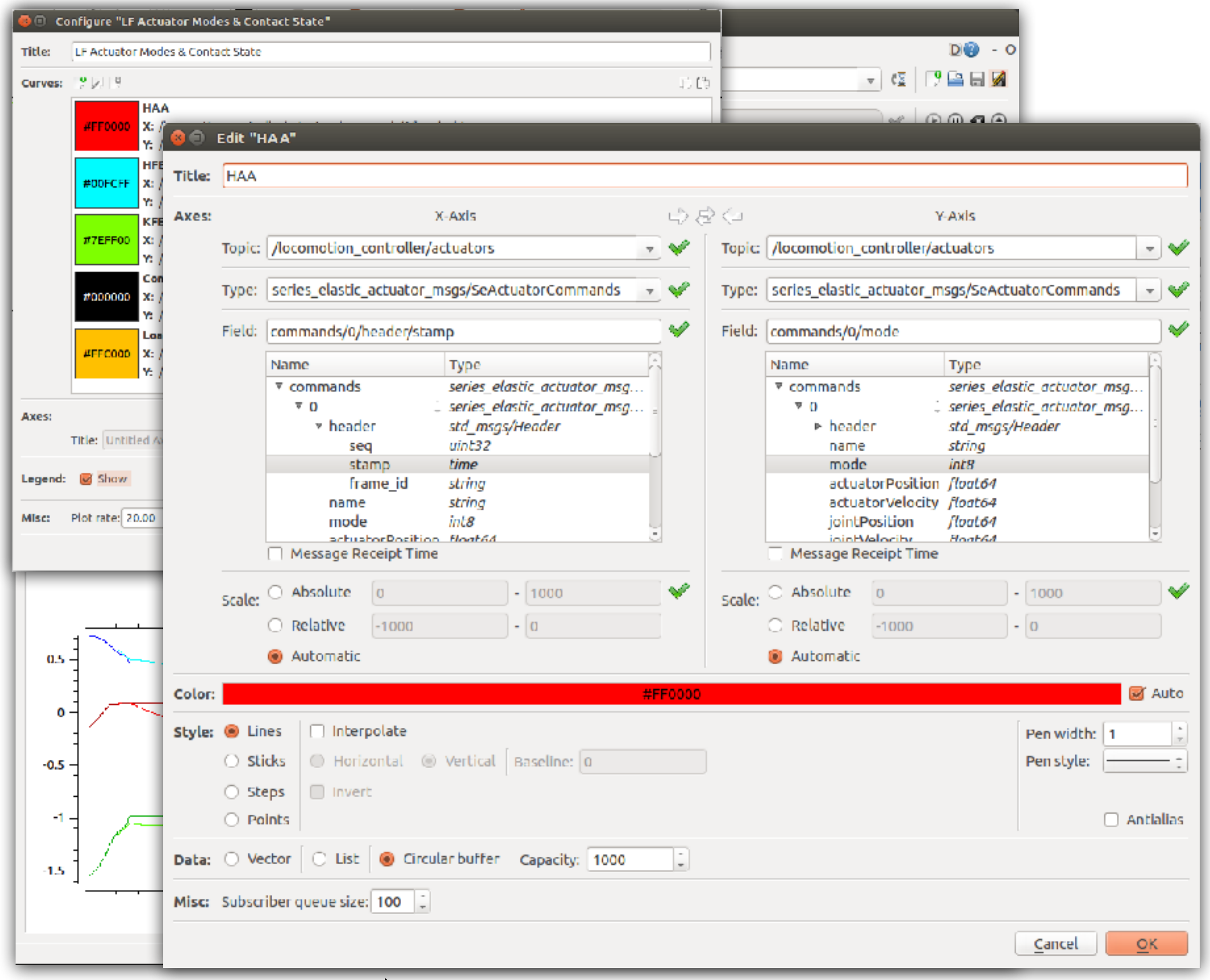


- C++ library (alternative to `rqt_plot`)
- Multiple plots in one window
- Edit, save, and load configurations
- Live plotting or load rosbags

Open Source
github.com/ethz-asl/rqt_multiplot_plugin

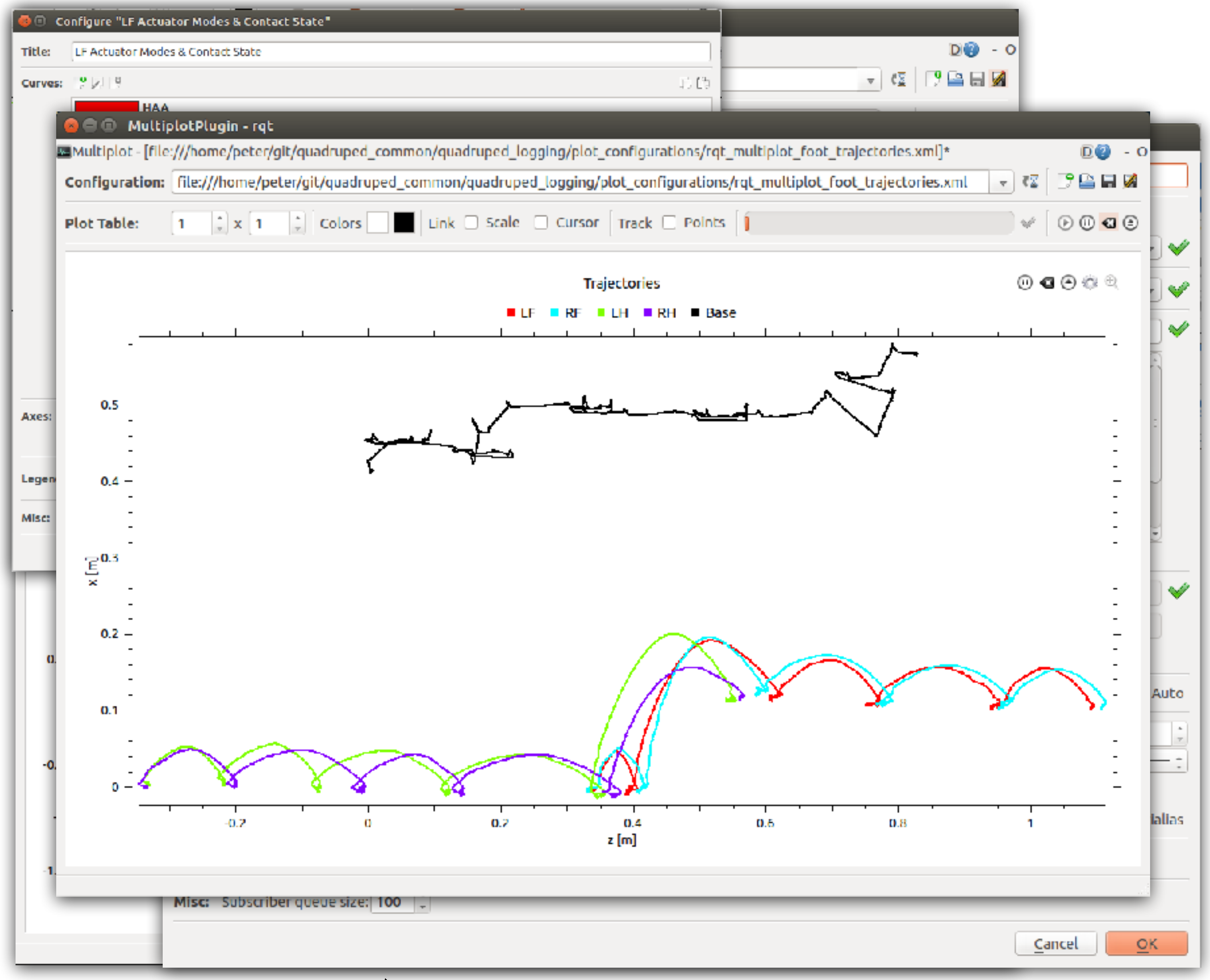
Open Source
github.com/ethz-asl/variant

RQT Multiplot Plugin & Variant Topic Tools

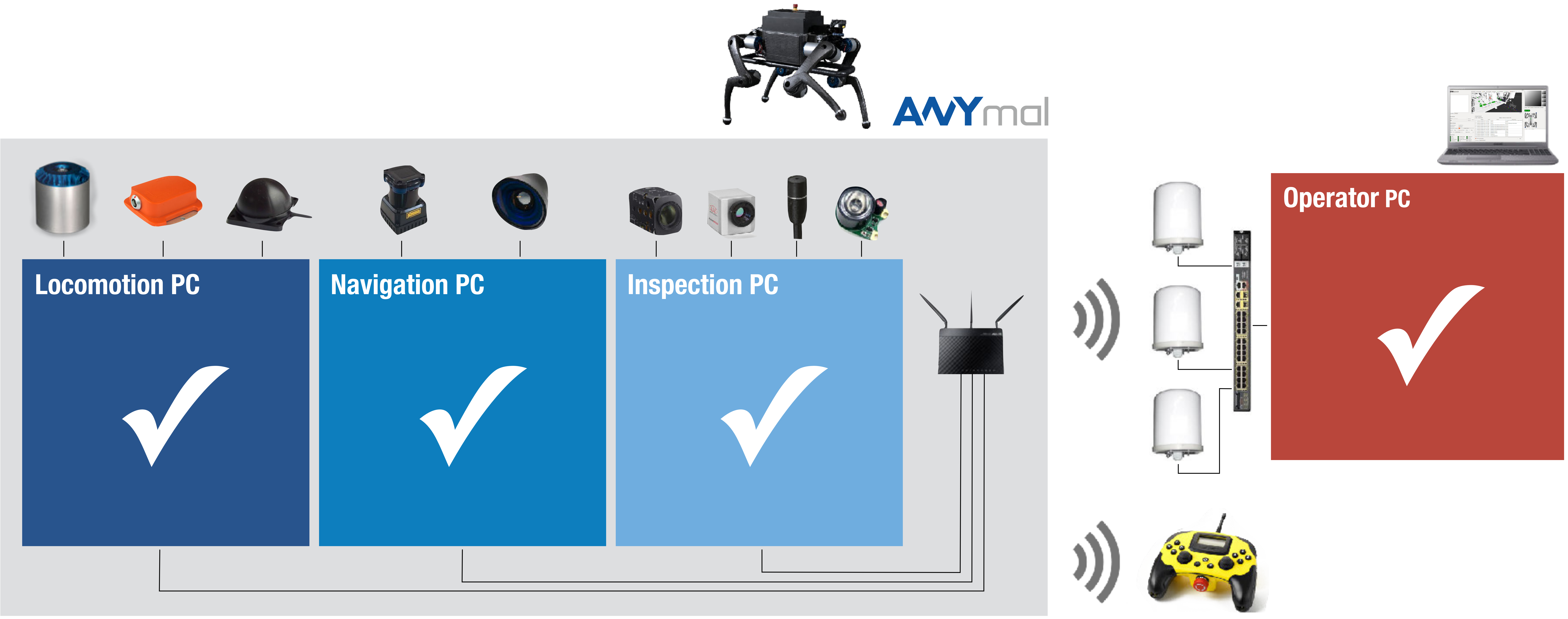


- C++ library (alternative to `rqt_plot`)
- Multiple plots in one window
- Edit, save, and load configurations
- Live plotting or load rosbags
- Easy to setup configurations

RQT Multiplot Plugin & Variant Topic Tools



- C++ library (alternative to `rqt_plot`)
- Multiple plots in one window
- Edit, save, and load configurations
- Live plotting or load rosbags
- Easy to setup configurations
- *x*- and *y*-axis freely configurable


ANYmal
Locomotion PC
Navigation PC
Inspection PC
Operator PC

Software Tools – How We (Try) To Keep Things Smooth

- All developers and robots same setup
  - ➡ Ubuntu 14.04 LTS, ROS Indigo

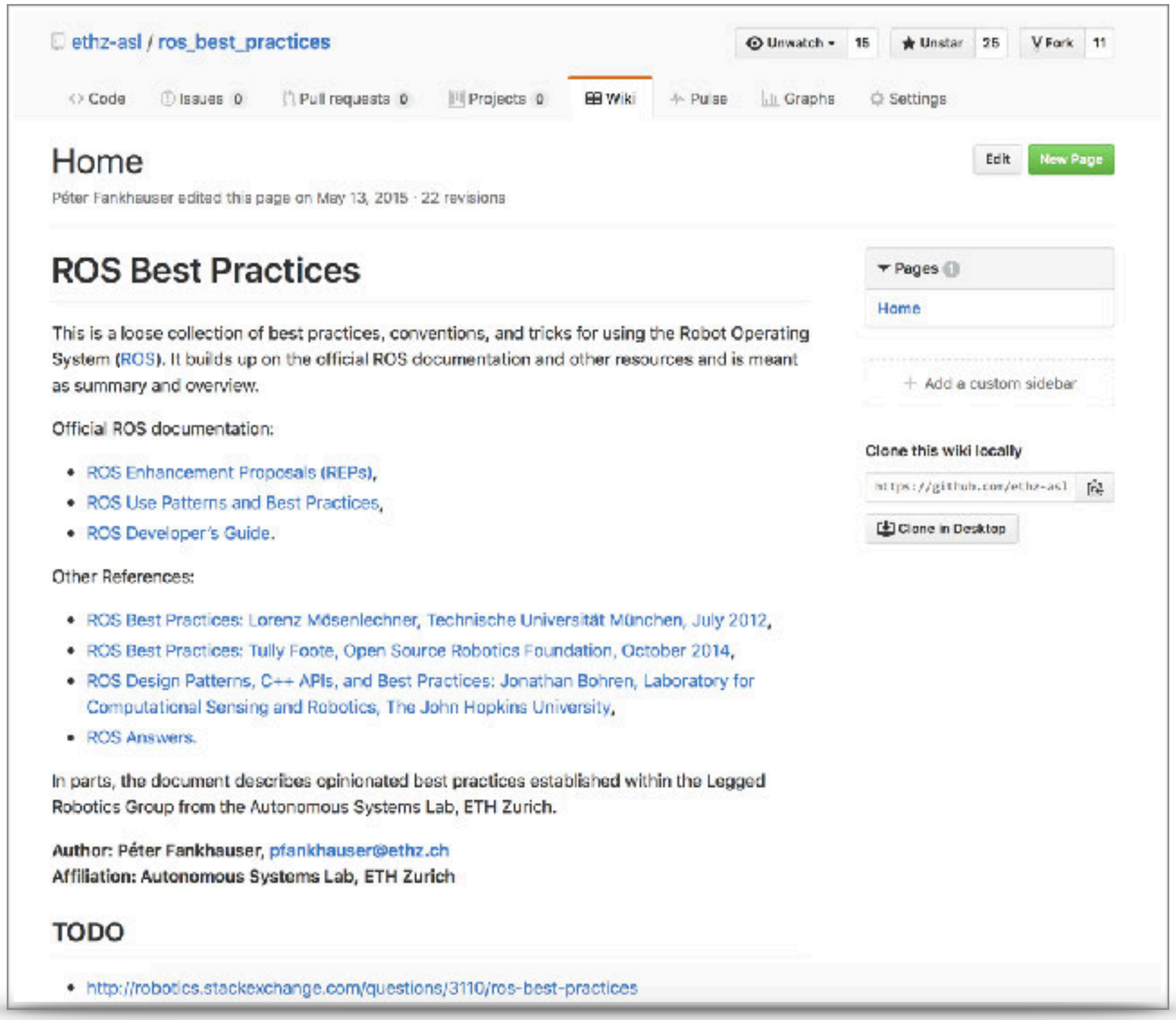
ubuntu
14.04 LTS

Software Tools – How We (Try) To Keep Things Smooth

- All developers and robots same setup
  - ➡ Ubuntu 14.04 LTS, ROS Indigo
- Software version control with Git
  - ➡ Bitbucket & GitHub

Software Tools – How We (Try) To Keep Things Smooth

- All developers and robots same setup
  - ➡ Ubuntu 14.04 LTS, ROS Indigo
- Software version control with Git
  - ➡ Bitbucket & GitHub
- Conventions for package structure, format, naming, and code style
  - ➡ github.com/ethz-asl/ros_best_practices/wiki


ethz-asl / ros_best_practices

Unwatch 15 | Unstar 25 | Fork 11

Code | Issues 0 | Pull requests 0 | Projects 0 | Wiki | Pulse | Graphs | Settings

Home

Edit | New Page

Péter Fankhauser edited this page on May 13, 2015 · 22 revisions

ROS Best Practices

This is a loose collection of best practices, conventions, and tricks for using the Robot Operating System (ROS). It builds up on the official ROS documentation and other resources and is meant as summary and overview.

Official ROS documentation:

- ROS Enhancement Proposals (REPs),
- ROS Use Patterns and Best Practices,
- ROS Developer's Guide.

Other References:

- ROS Best Practices: Lorenz Mösenlechner, Technische Universität München, July 2012,
- ROS Best Practices: Tully Foote, Open Source Robotics Foundation, October 2014,
- ROS Design Patterns, C++ APIs, and Best Practices: Jonathan Bohren, Laboratory for Computational Sensing and Robotics, The John Hopkins University,
- ROS Answers.

In parts, the document describes opinionated best practices established within the Legged Robotics Group from the Autonomous Systems Lab, ETH Zurich.

Author: Péter Fankhauser, pfankhauser@ethz.ch
Affiliation: Autonomous Systems Lab, ETH Zurich

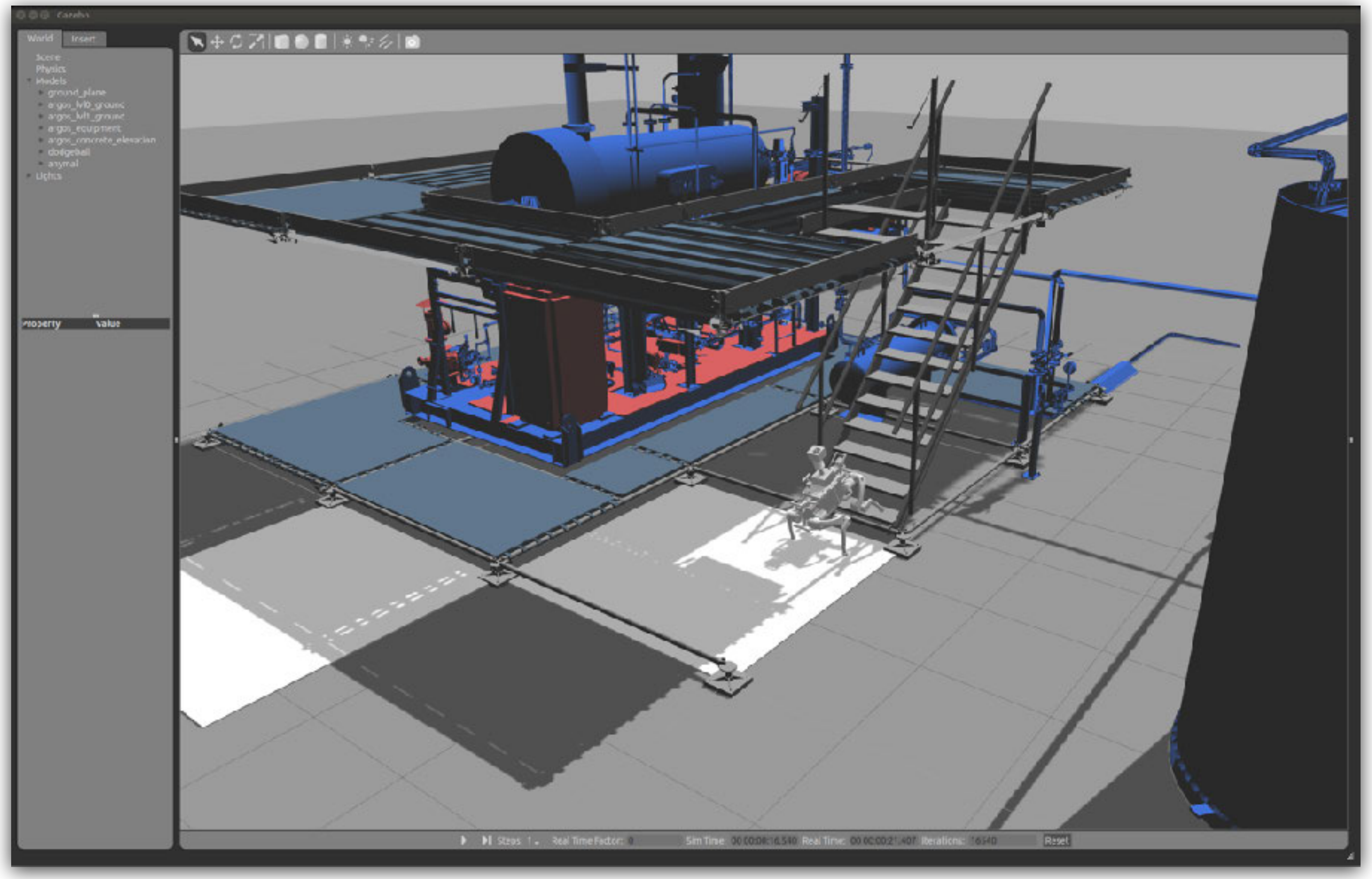
TODO

- http://robotics.stackexchange.com/questions/3110/ros-best-practices

Pages
Home
Add a custom sidebar
Clone this wiki locally
Clone in Desktop

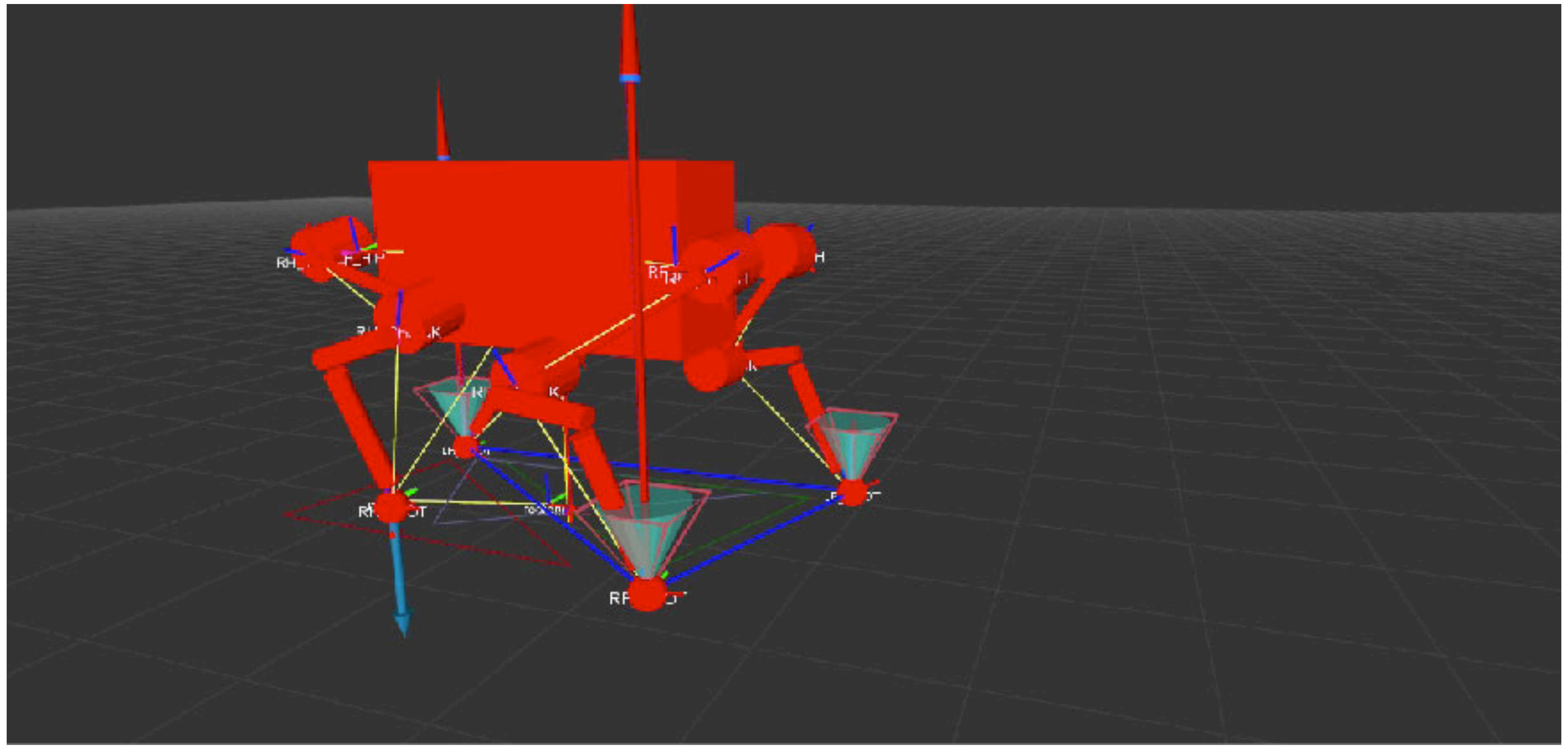
Software Tools – How We (Try) To Keep Things Smooth

- All developers and robots same setup
  - ➡ Ubuntu 14.04 LTS, ROS Indigo
- Software version control with Git
  - ➡ Bitbucket & GitHub
- Conventions for package structure, format, naming, and code style
  - ➡ github.com/ethz-asl/ros_best_practices/wiki
- Extensive use of simulation
  - ➡ Gazebo



Software Tools – How We (Try) To Keep Things Smooth

- All developers and robots same setup
  - ➡ Ubuntu 14.04 LTS, ROS Indigo
- Software version control with Git
  - ➡ Bitbucket & GitHub
- Conventions for package structure, format, naming, and code style
  - ➡ github.com/ethz-asl/ros_best_practices/wiki
- Extensive use of simulation
  - ➡ Gazebo
- Visualizing as much as possible



Software Tools – How We (Try) To Keep Things Smooth

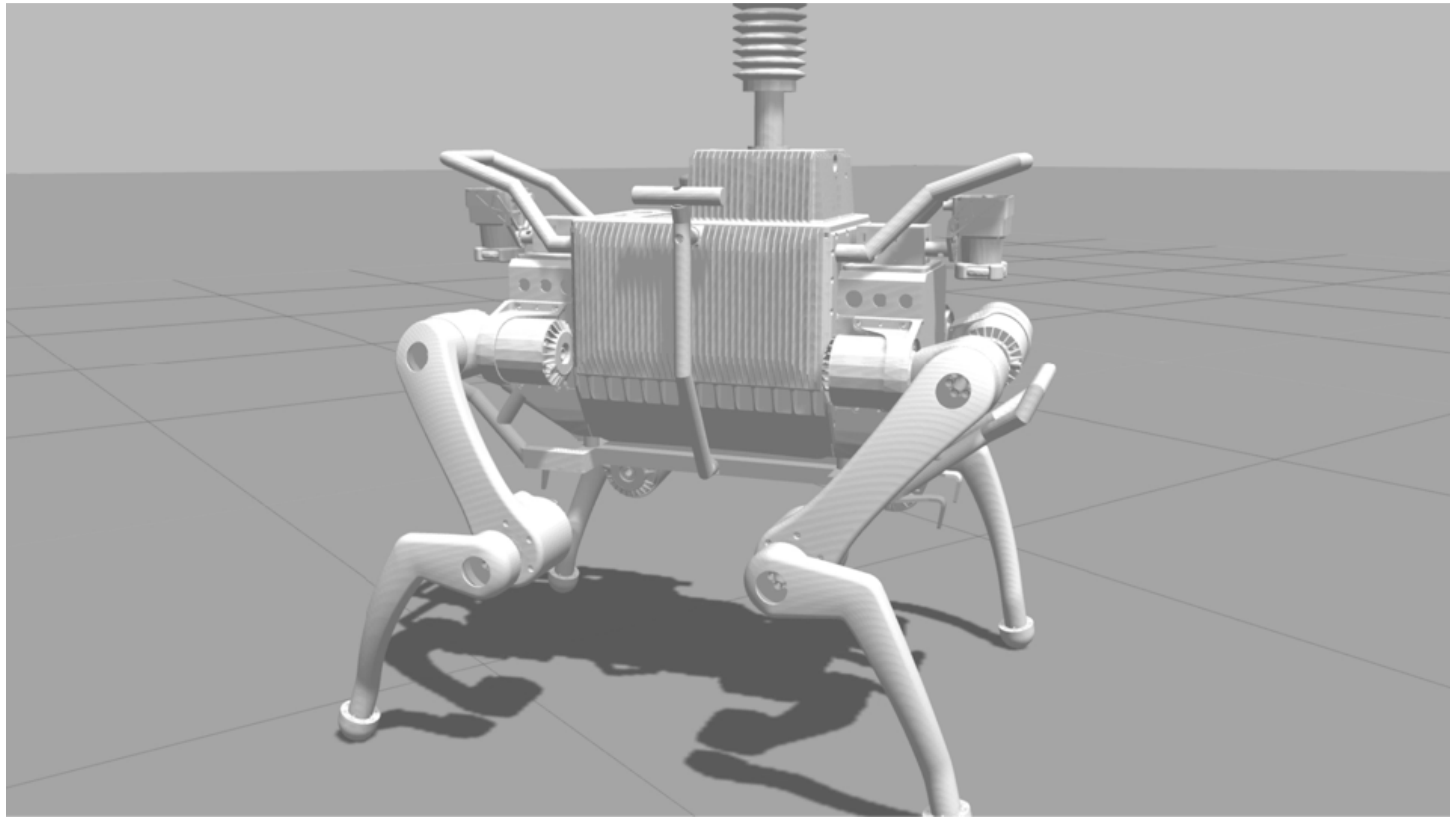
- Lots of tests on hardware
  - ➡ Weekly “shakeouts” for defined tasks
  - ➡ Lots of demos

Software Tools – How We (Try) To Keep Things Smooth

- Lots of tests on hardware
  - ➡ Weekly “shakeouts” for defined tasks
  - ➡ Lots of demos
- Continuous Integration
  - ➡ Jenkins
  - ➡ Unit tests (after each change)
  - ➡ ROS integration tests (at night)

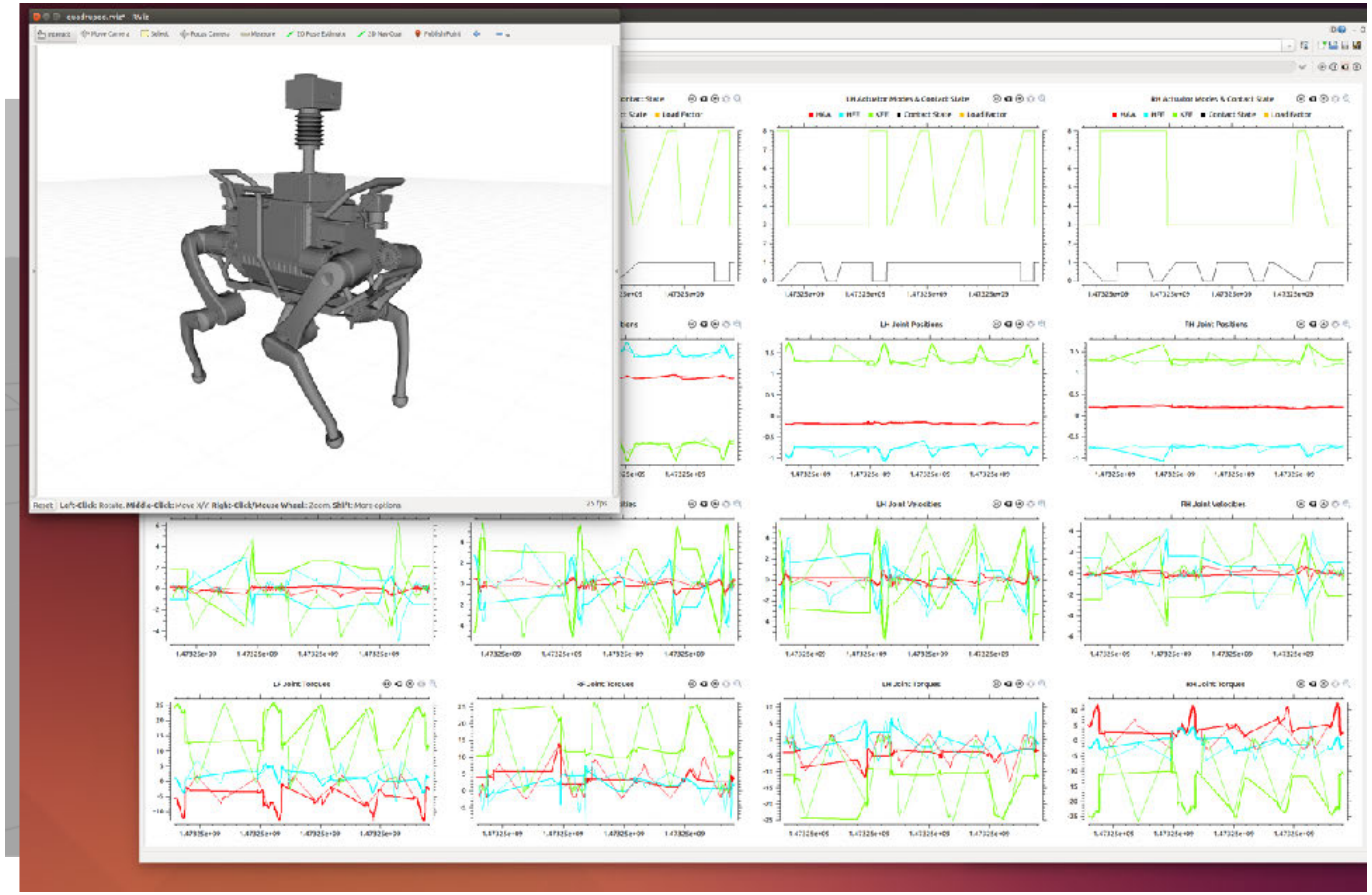
Software Tools – How We (Try) To Keep Things Smooth

- Lots of tests on hardware
  - ➡ Weekly "shakeouts" for defined tasks
  - ➡ Lots of demos
- Continuous Integration
  - ➡ Jenkins
  - ➡ Unit tests (after each change)
  - ➡ ROS integration tests (at night)



Software Tools – How We (Try) To Keep Things Smooth

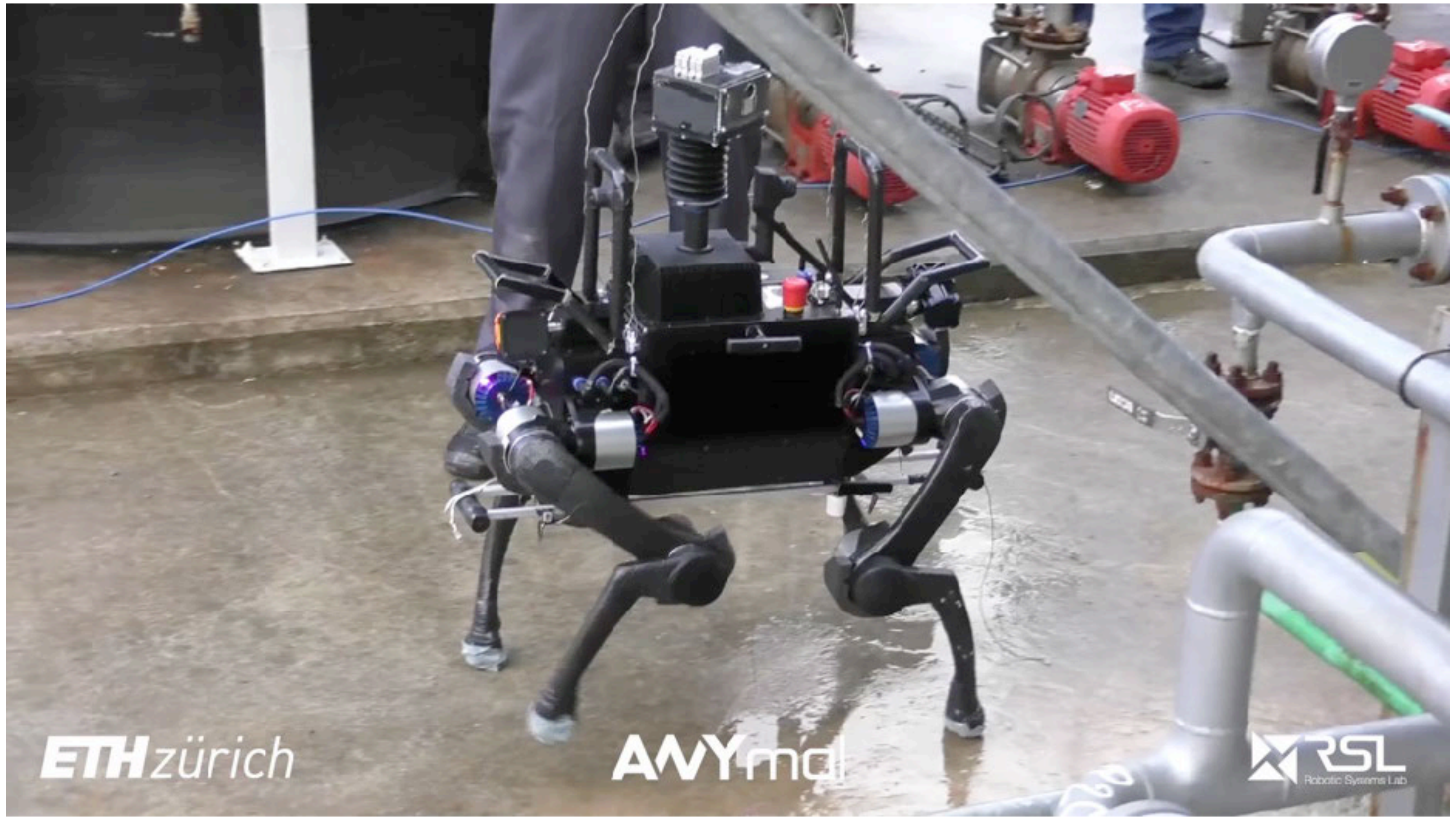
- Lots of tests on hardware
  - ➡ Weekly "shakeouts" for defined tasks
  - ➡ Lots of demos
- Continuous Integration
  - ➡ Jenkins
  - ➡ Unit tests (after each change)
  - ➡ ROS integration tests (at night)
- Logging (rosbag)
  - ➡ All important information is always logged
  - ➡ Review logs with RViz and RQT Multiplot



Conclusion

- Introduced 10 open-source packages, 250+ internal packages
- Coordination of a big team is hard
- Good naming is important
- ROS as “glue”
- WiFi is often problematic
- Reliability is crucial

Thank You



Open-Source Software

github.com/ethz-asl

github.com/leggedrobotics

Péter Fankhauser
pfankhauser@ethz.ch

Samuel Bachmann
Dario Bellicoso
Michael Bloesch
Remo Diethelm
Christian Gehring
Mark Hoepflinger
Gabriel Hottiger
Marco Hutter
Jemin Hwangbo
Linus Isler
Dominic Jud
Ralf Kaestner
Bruno Kaufmann
Philipp Krüsi
Andreas Lauber
Philipp Leemann
Konrad Meyer
Roland Siegwart
Vassilios Tsounis
Martin Wermelinger

Robotic Systems Lab

www.rsl.ethz.ch

Autonomous Systems Lab

www.asl.ethz.ch