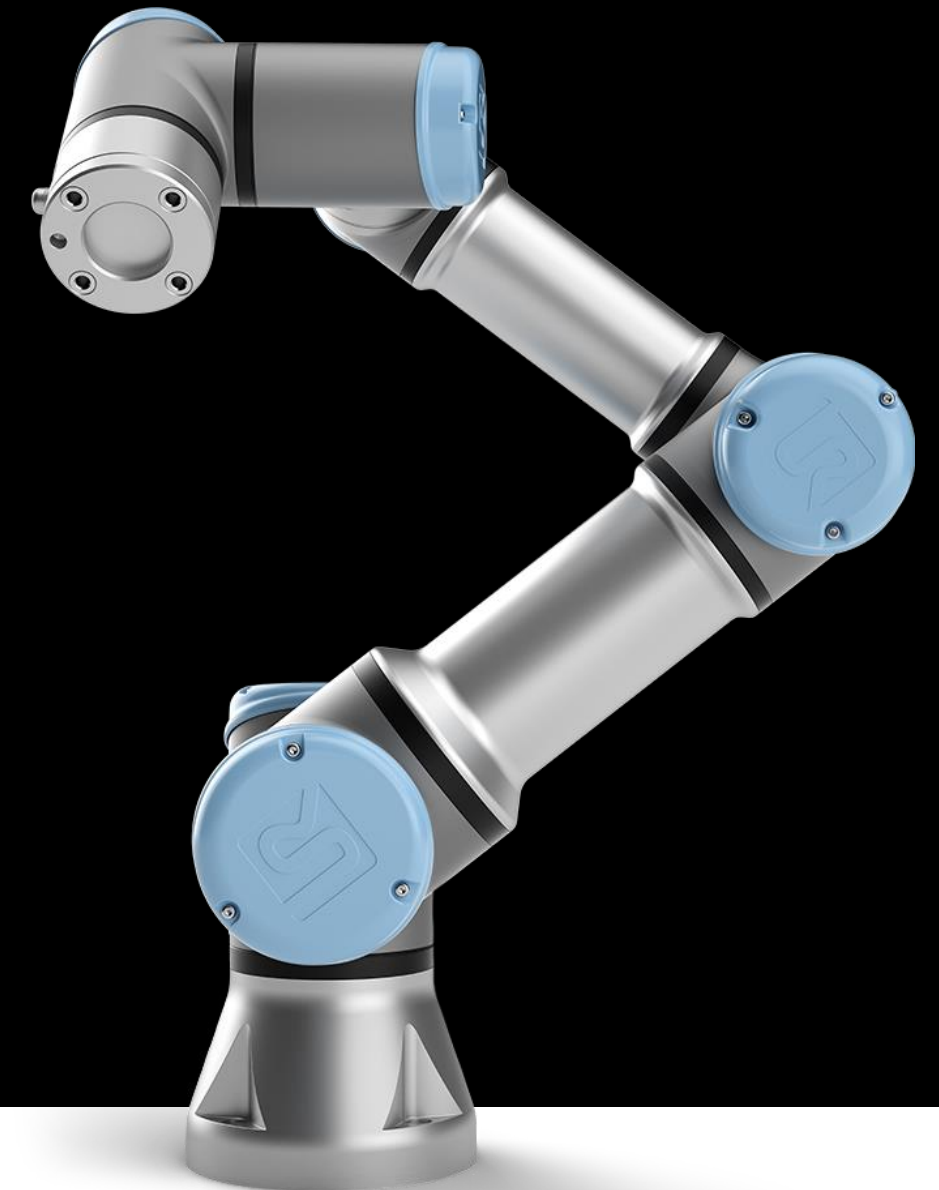


# TOWARDS INDUSTRIAL-GRADE ROBOT CONTROL IN ROS

Anders Billesø Beck  
Head of Innovation























# THE NEED FOR A NEW ROS DRIVER

## Clouded landscape

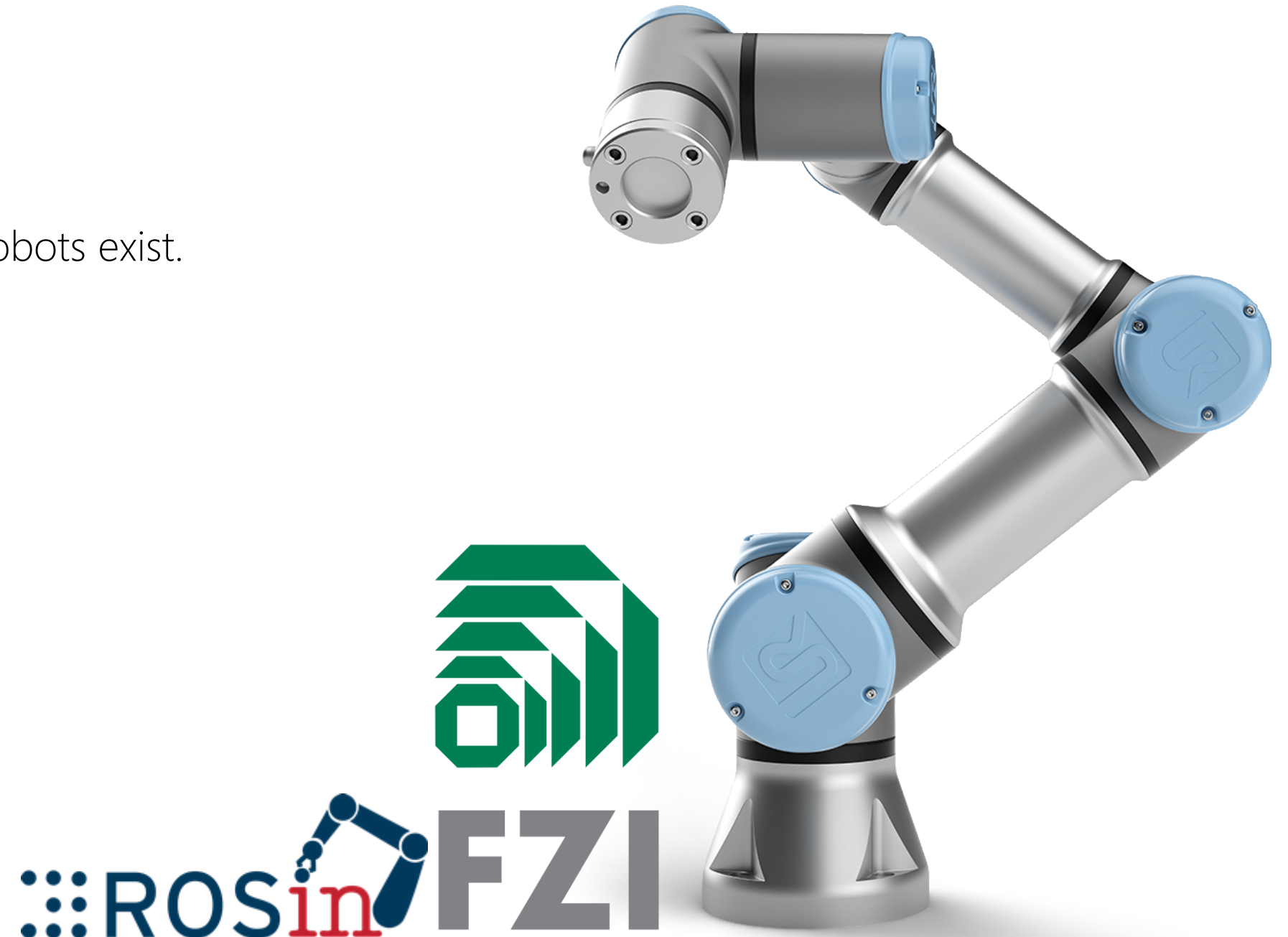
- More than 200 variants of a ROS driver for UR robots exist.
- Instability towards API changes

## A lot could be improved

- Many, especially new features are lacking
- Sub-optimal performance
- Only preliminary e-Series support

## The new driver

- ROSIN FTP in collaboration with FZI





# GOALS

## Ease of use

- Easy and plug-and-play to use a UR robot with ROS

## Performance

- Full utilization of all features of the robots
- As industrial grade an interface as current ROS practice allows

## Stability

- The driver will build on stable and versioned APIs
- The driver will be integrated into the software testing regime of Universal Robots

## Community

- The driver will remain open source and relying on future community contributions



# OFFICIAL UNIVERSAL ROBOTS ROS DRIVER

Beta Program Launch - June 1st 2019

73 beta partners from 12 countries

- 13 from Large enterprises
- 16 from SMEs and startups
- 12 from Research organizations
- 14 from Universities



# OFFICIAL UNIVERSAL ROBOTS ROS DRIVER

Public Launch – October 9th 2019

Hosted at Universal Robots GitHub  
- 25 daily unique clones!

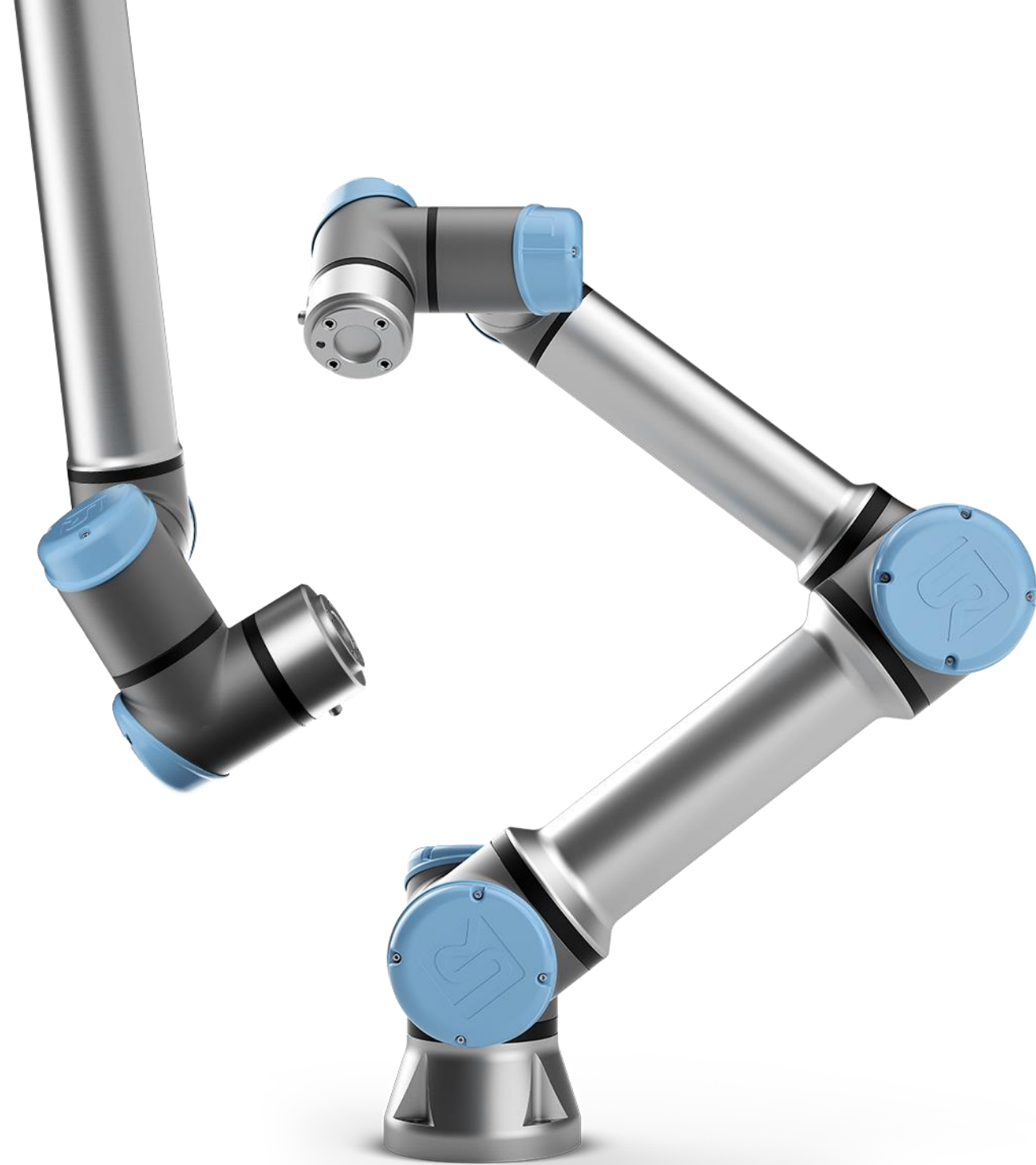






# HIGHLIGHTS

- Stable control interfaces
- Teach-pendant integration
  - Use ROS as part of a program
  - Pause, stop, restart and scale speed
- Factory Calibration in ROS
- Safety compliance speed scaling
- eSeries tool communication forwarding
- Full safety system compliance





# PROGRAMMING FLOW INTEGRATION

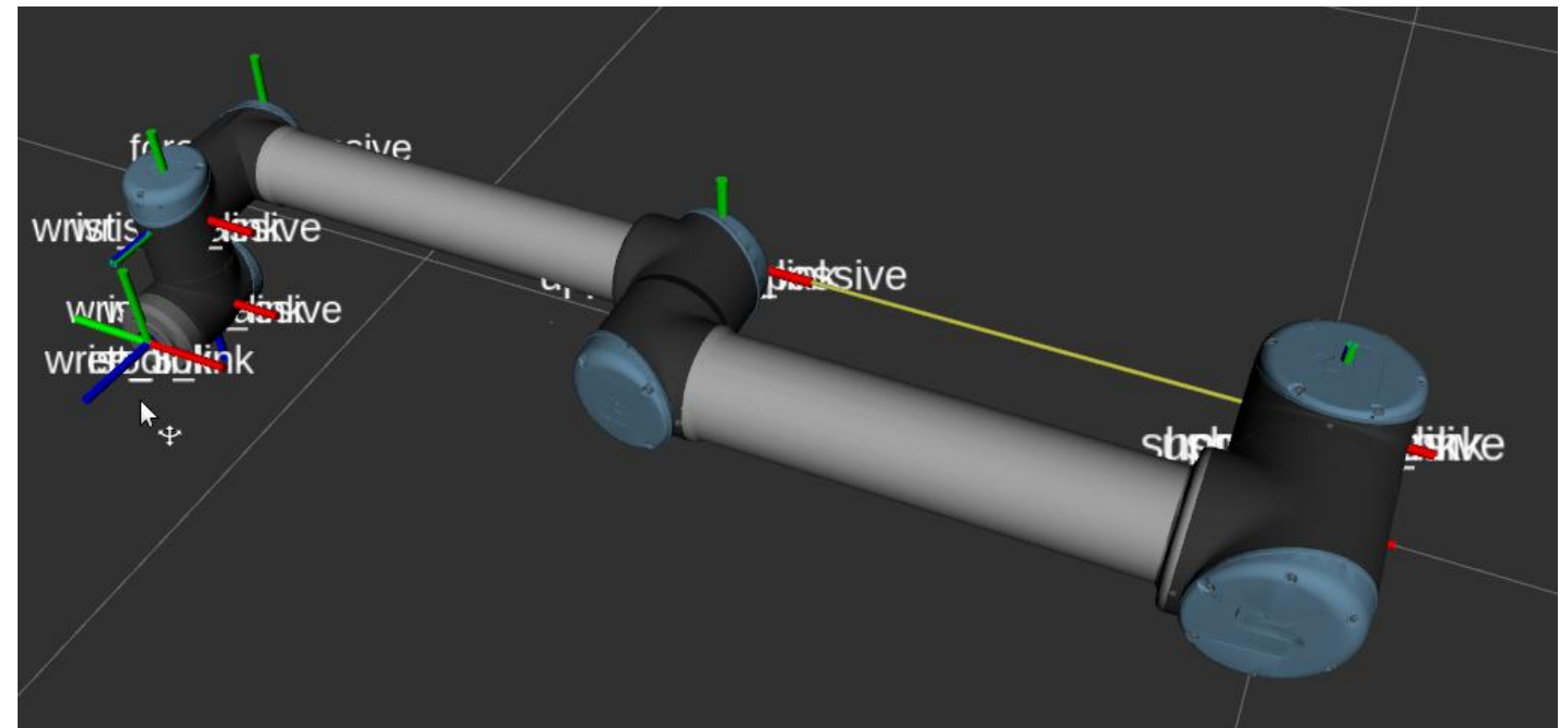
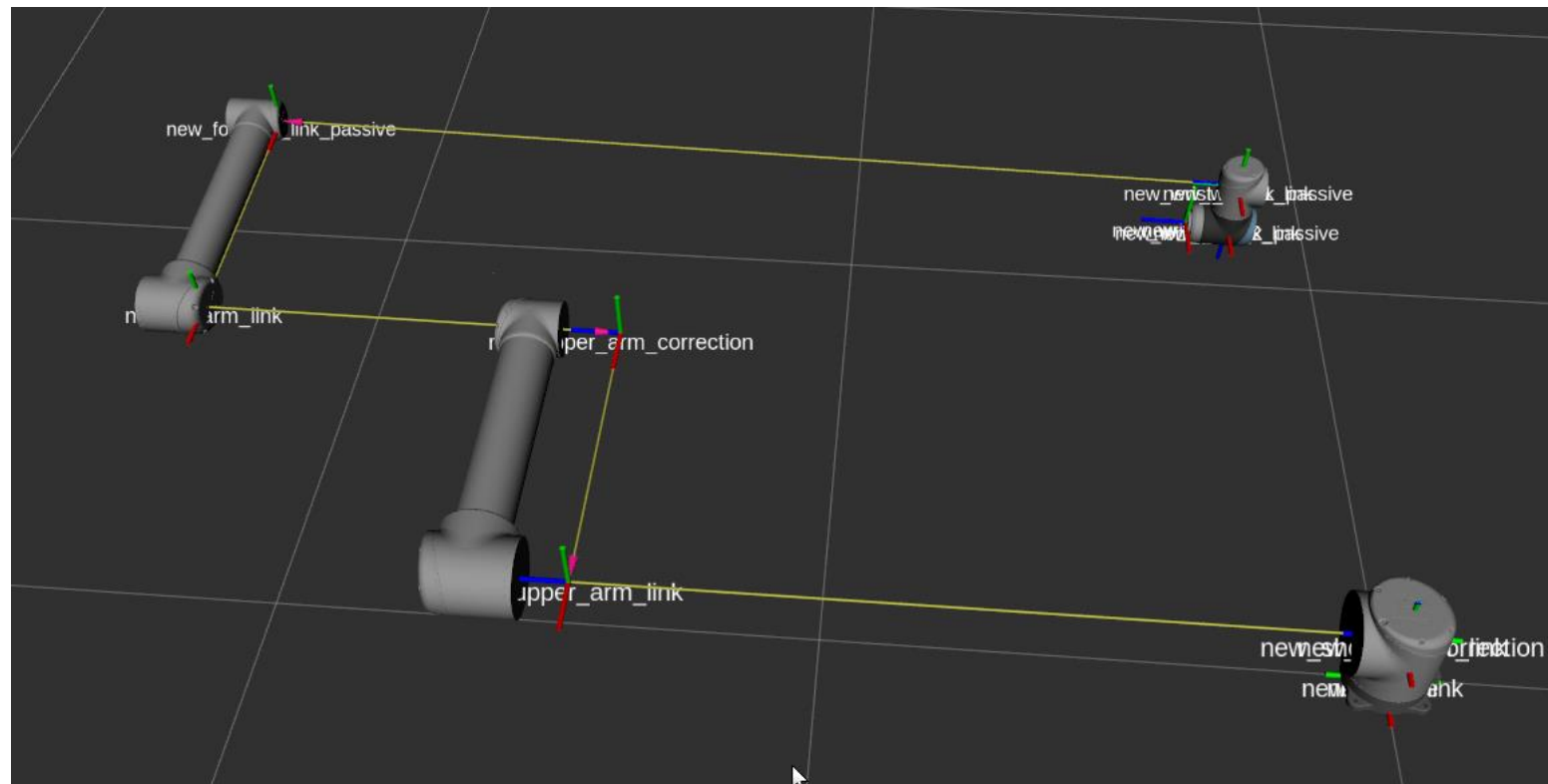
Use ROS where it is needed and be in full control of  
the robot during ROS control





# CALIBRATION FOR ABSOLUTE ACCURACY

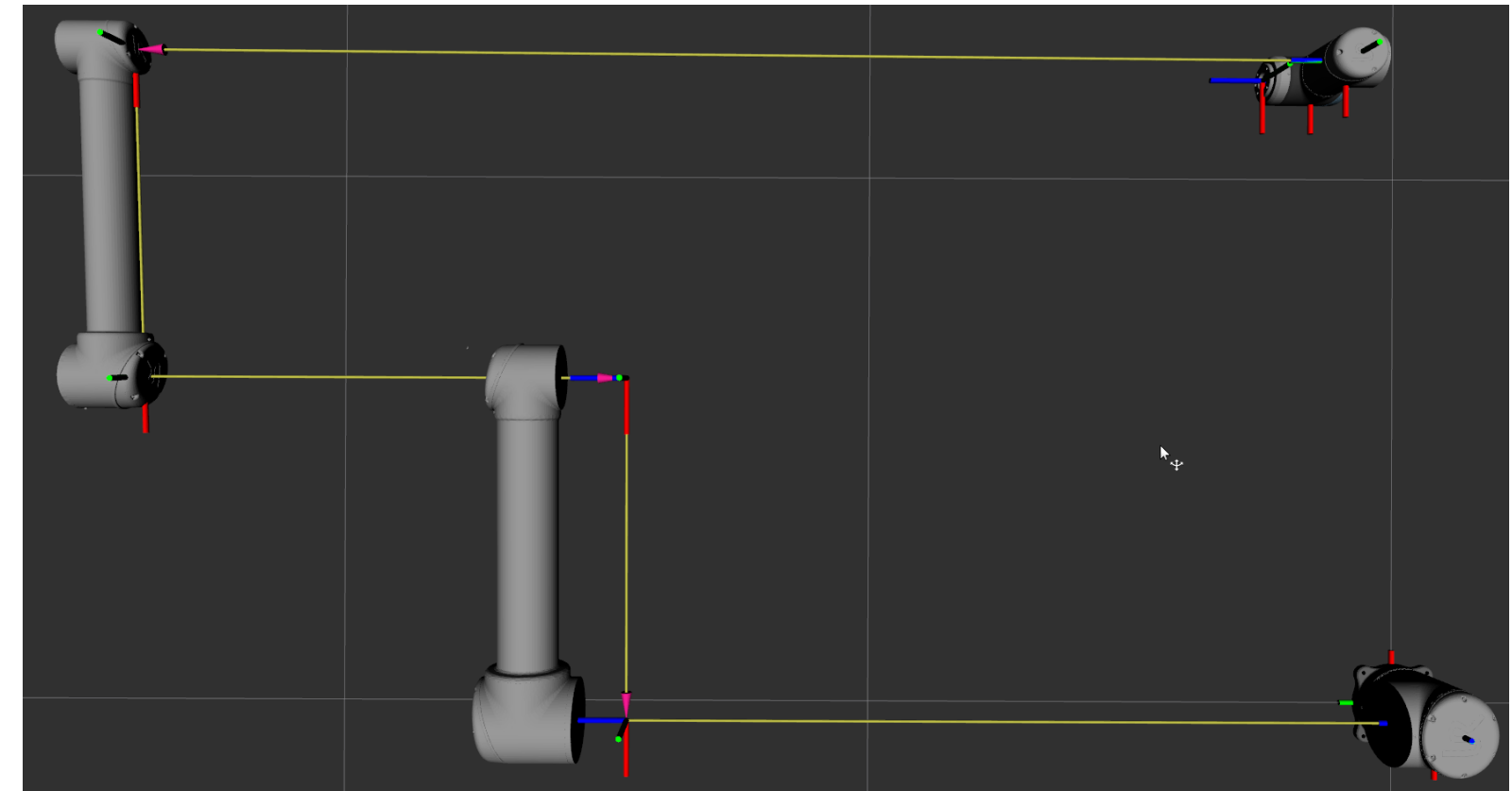
- All UR robots are individually factory calibrated for absolute accuracy
- Factory calibration not directly applicable for ROS
- Change structure to match URDF





# CALIBRATION CORRECTION

- Transform DH parameters to 6D representation
- Move in arm segments **on their rotation axis**
- Save resulting 6D parameters of each joint as .yaml parameter set
- Automatically happening at start
- Parameter sets are saved in **\$ROS\_HOME**
- Launch files provide a *serial\_number* argument to load correct calibration





# COLLABORATIVE & SAFE

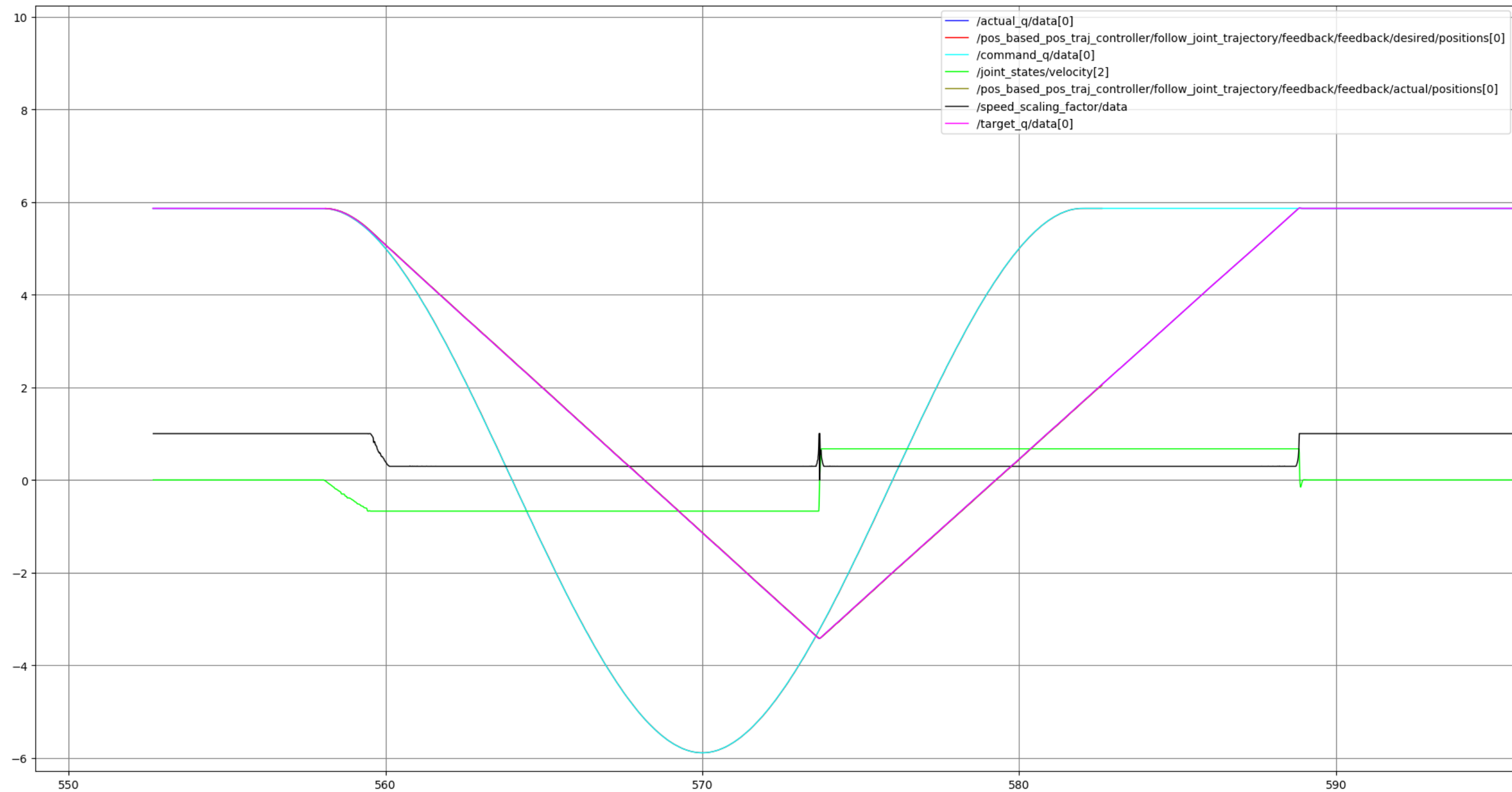
Safety compliance in ROS



UNIVERSAL ROBOTS

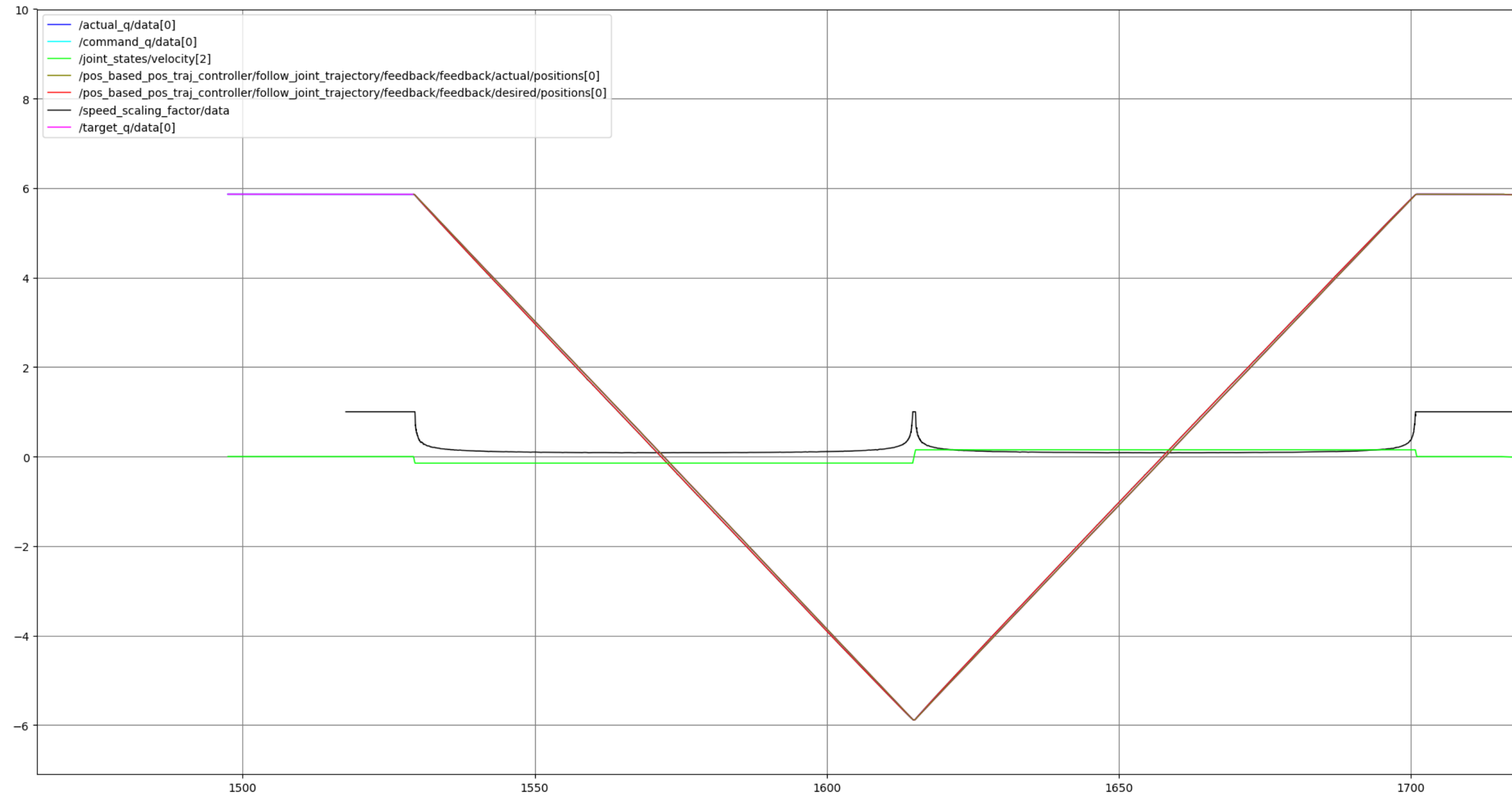


# THE PROBLEM





# WITH SAFETY COMPLIANCE CONTROLLER





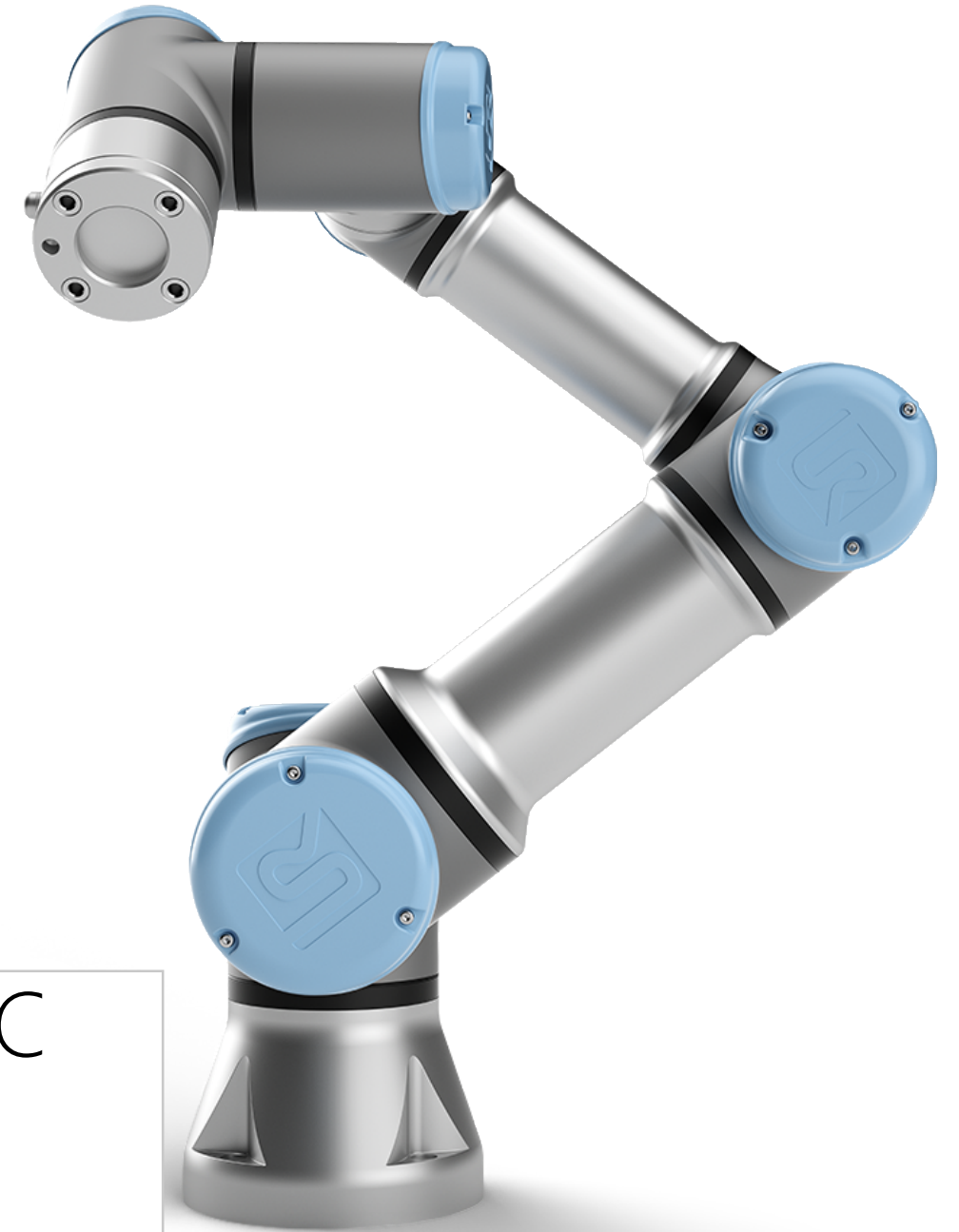
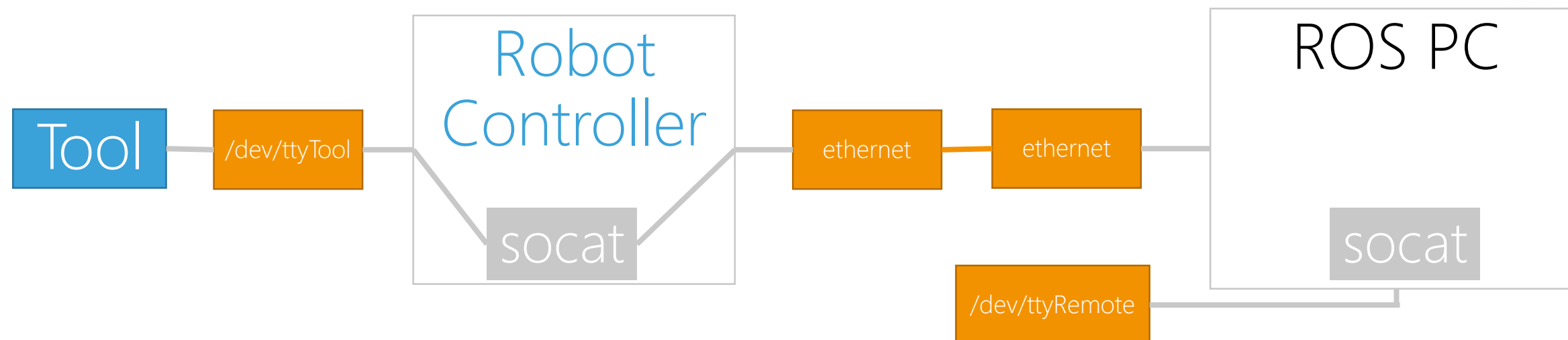
# e-Series TOOL COMMUNICATION IN ROS

## Tool communication connector

- RS485 Software UART, up to 5 Mbps

## UART in ROS

- Forwarded as virtual serial port to ROS machine
- Re-use of existing drivers for serial devices in ROS





WHAT'S NEXT?



# UNIVERSAL ROBOTS ROS DRIVER 2.0

Industrial-grade performance, features and  
reliability





# TOWARDS INDUSTRIAL-GRADE

Industrial-grade performance and stability require an interface that respect real-time constraints and leverage the full performance of all components.

- Easy access to all control interfaces in ROS
- Exploit the **full motion quality and responsiveness** of the robot
- Industrial **stability** and robustness
- Integrated and **seamless cooperation** between ROS and Robot
- Making the way for using ROS for **actual production and technology commercialization**





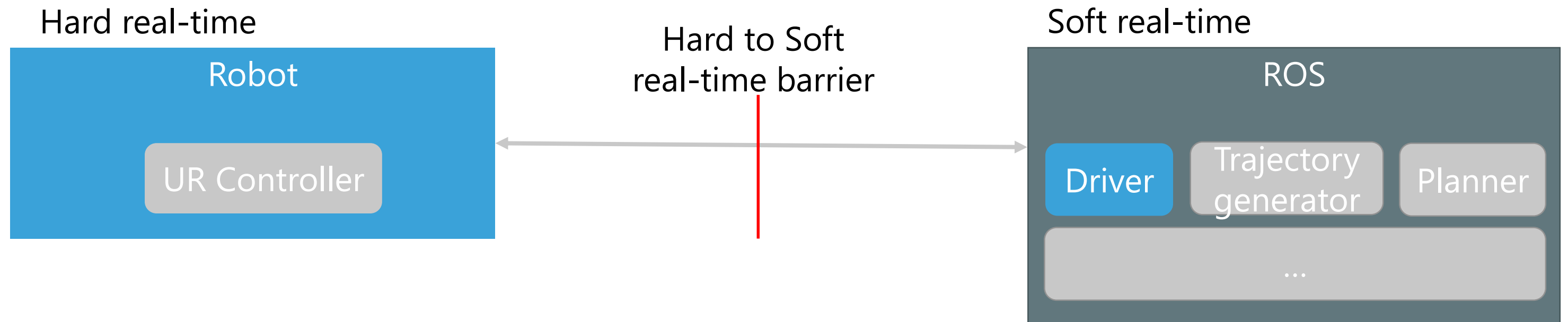
# GOALS

- **Easy-to-use cartesian motion interface in ROS**
  - Definition of Cartesian input for robot control (Try to set a standard)
  - New hardware interfaces for Cartesian commands.
- **Seamless switching of execution flow between ROS and robot**
  - On the robot execution of movements and functions in real-time
  - A Further enhanced coupling of ROS Control and the UR Controller
  - Real-time use of new robot features
- **ROS as a Service**
  - Two-way communication channel between ROS and Robot controller
  - User-defined ROS actions and services can be called from a UR program tree lowering the barriers of utilizing ROS-I inside an industrial automation process



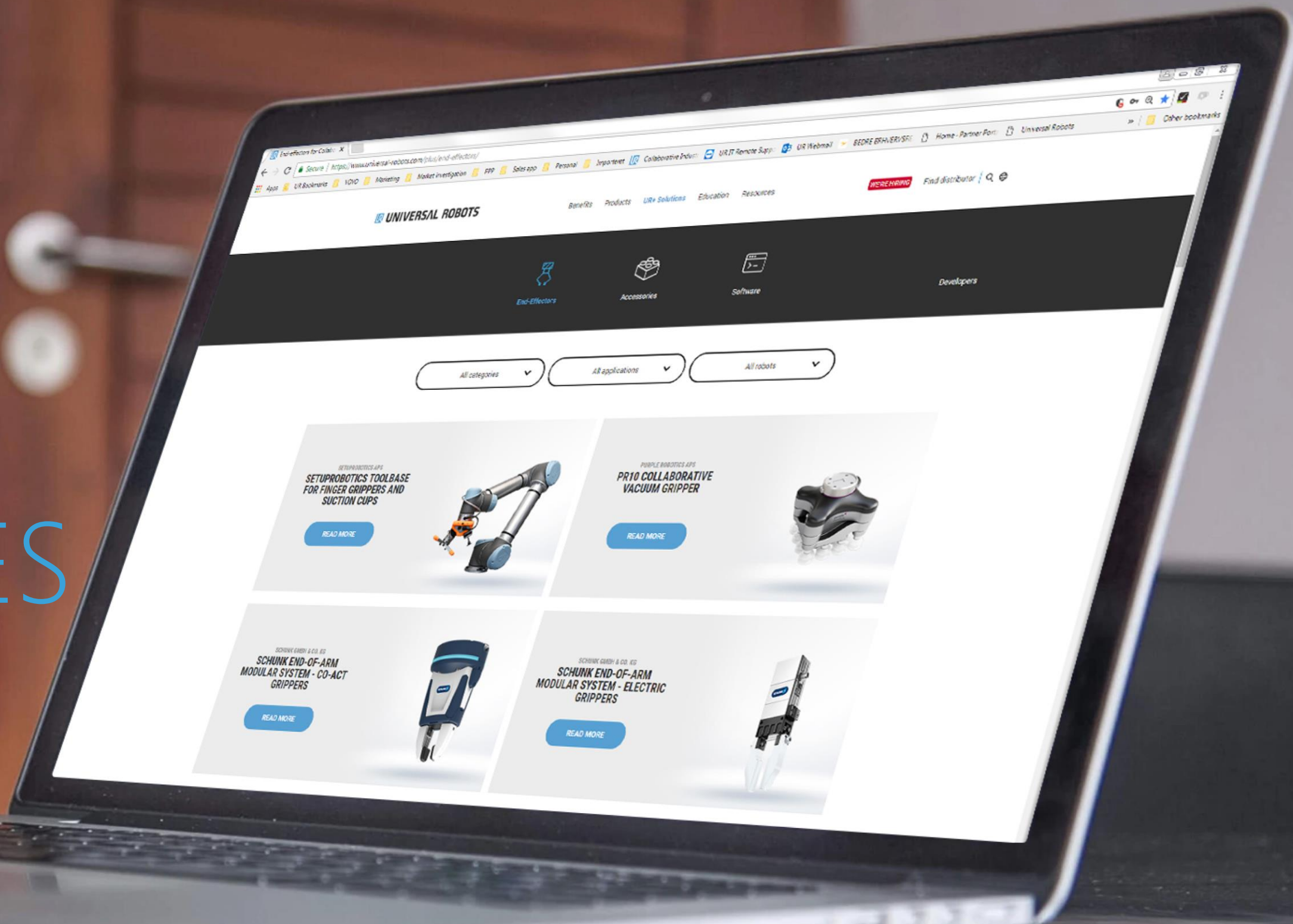
# PERFORMANCE ACROSS REAL-TIME BARRIERS

- Ensuring hard real-time decisions in the robot controller
- Enabling seamless transition to soft real-time ROS control when needed for advanced tasks





# SUPPORTING NEW FEATURES





# REACTIVE MOTION PROGRAMMING (NEW 5.3-5.4)

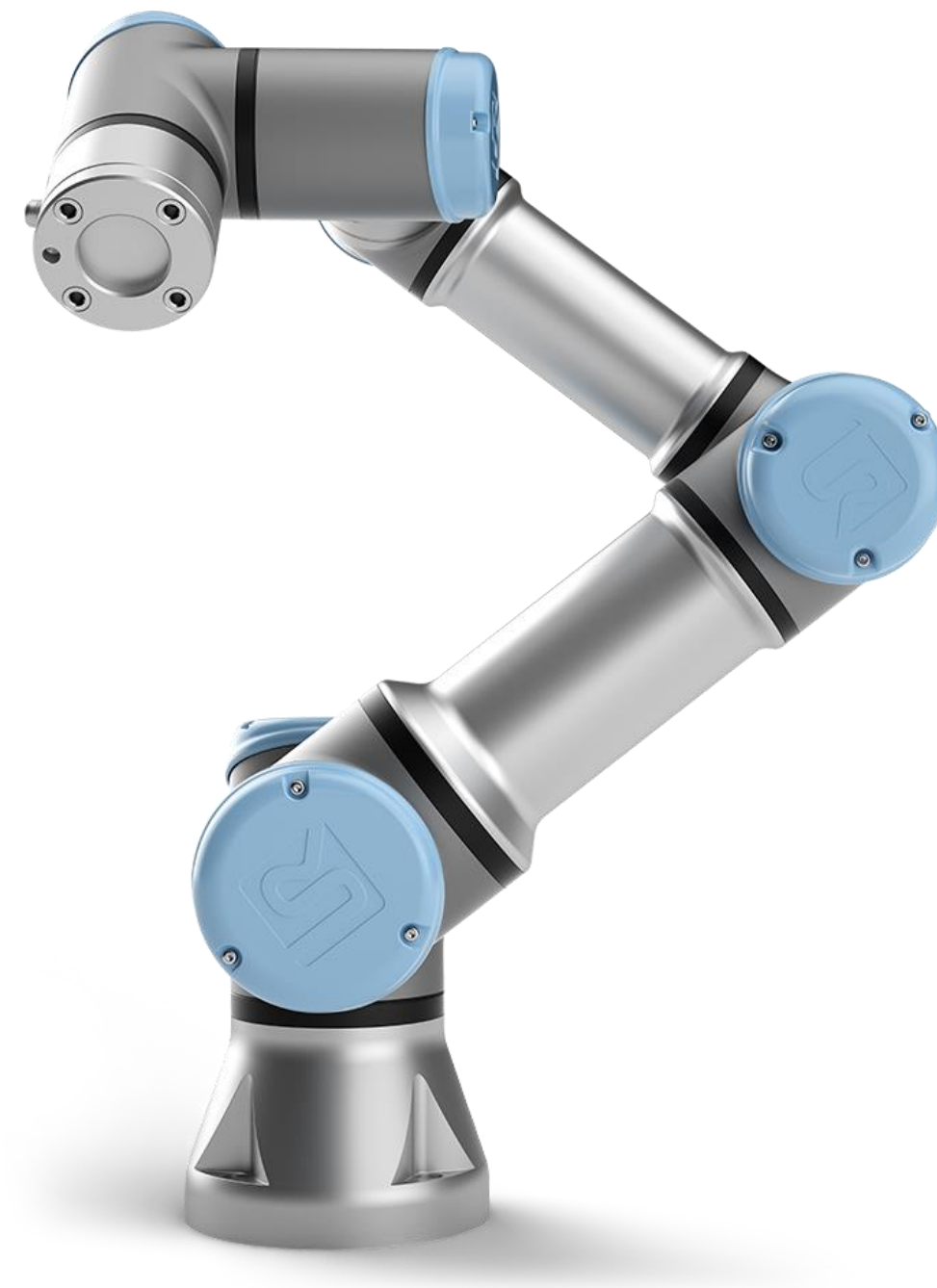
- Fully reactive and event driven motions
- Movement and actions **until** events
- **Contact detection**
  - Disturbance tolerant
  - Detects hard and relatively soft contacts
  - Retaction to contact surfaces
- Utilize 2 ms hard real-time responses





# DYNAMIC PATH-OFFSET (NEW 5.6 – 5 DAYS AGO)

- Dynamic path-offset allows real-time adjustment to robot motions in 6DOF
- Motions can be compensated relative to
  - Tool of the robot
    - e.g. weaving for welding, admittance control
  - Base of the robot
    - E.g. Use with mobile robots or linear axis



# CALL TO ACTION

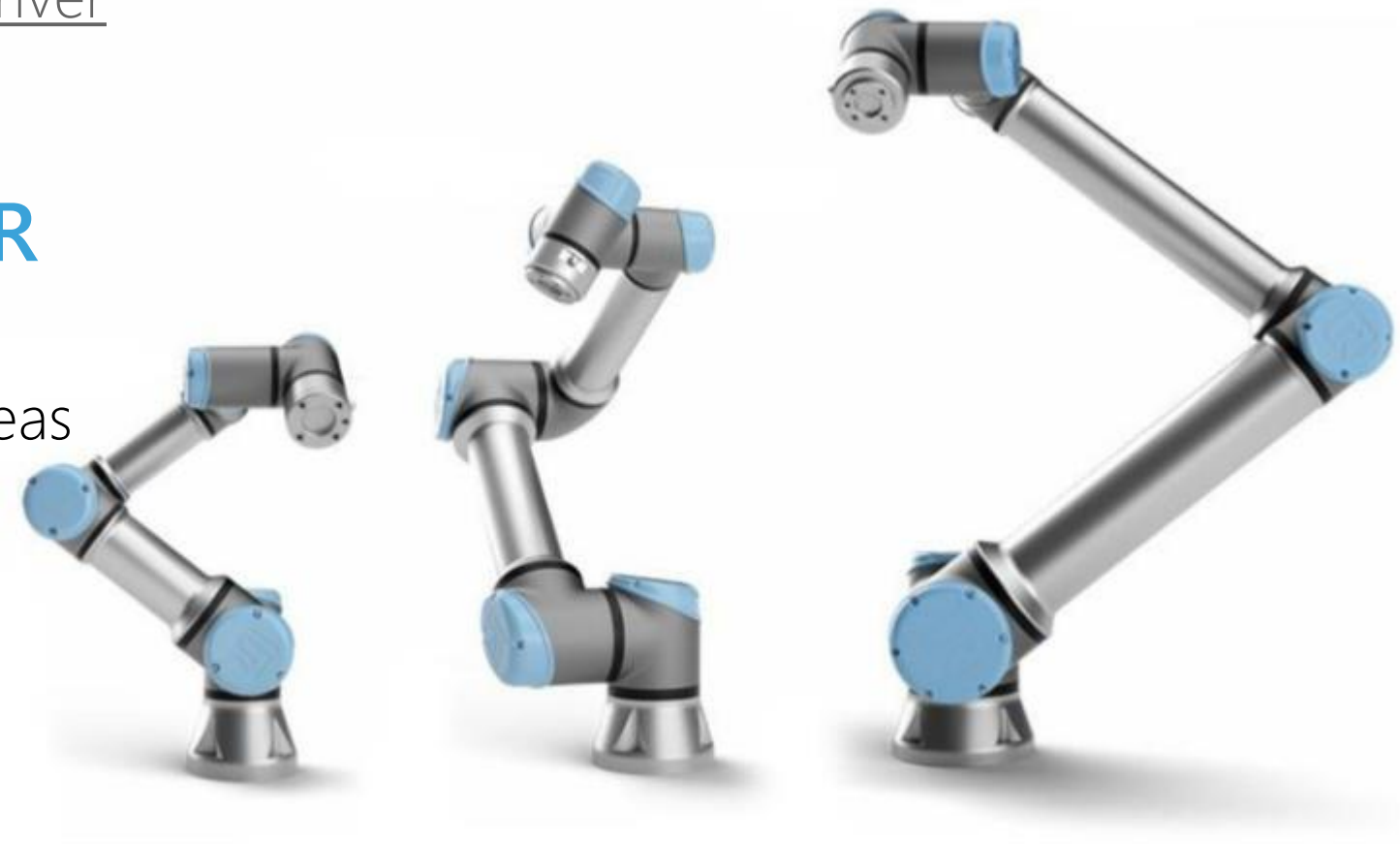
## UNIVERSAL ROBOTS ROS DRIVER

Leverage your UR robots by using the new driver, and give us your feedback

[https://github.com/UniversalRobots/Universal\\_Robots\\_ROS\\_Driver](https://github.com/UniversalRobots/Universal_Robots_ROS_Driver)

## PROVIDE YOUR INPUT FOR THE 2.0 DRIVER

Contact us on [ros@universal-robots.com](mailto:ros@universal-robots.com) for feedback and ideas





Two Universal Robots robotic arms are shown against a black background. The arms are positioned as if they are clapping, with their hands facing each other. The arms have a grey metallic finish with light blue accents on the joints and end effectors. The lighting highlights the texture of the metal and the smooth curves of the joints.

THANKS!

DON'T FORGET TO TRY THE NEW DRIVER