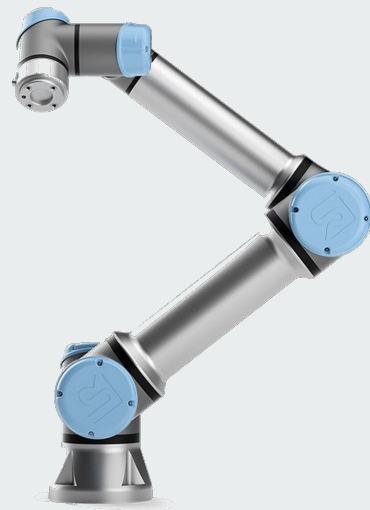




A ROS2 Driver for Universal Robots

13 April 2021

Andy Zelenak
PickNik Robotics
zelenak@picknik.ai





Outline

- Rundown of the current state in ROS2 manipulators
- Video time!
 - Teleop, trajectory execution, compliance
- Capabilities
- Limitations
 - Jerk-limited trajectories
 - Velocity control
- Request for Beta testers
- Challenges
 - Launch file args
 - Evolving `ros2_control` API/ABI
- Contributors

Manipulators having ROS2 Drivers



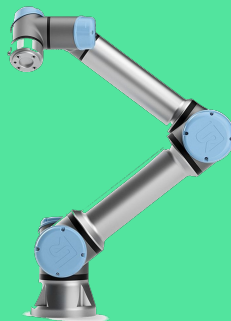
Acutronic, 2019
(Now closed)



Hello Robot
Stretch R1 mobile manipulator
May demo for ROS2 Groovy



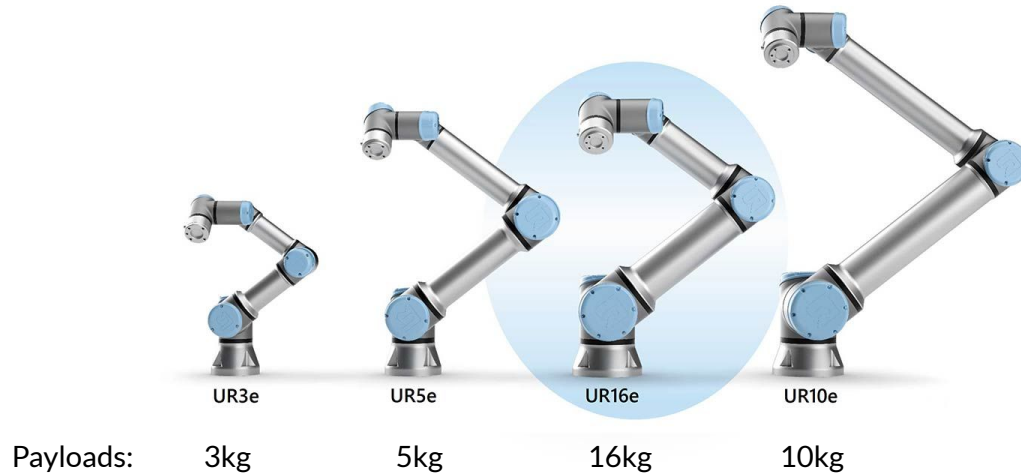
Doosan Robotics
Announced today



Universal Robots
7 supported robot models
Beta release today!



Supported UR Models - all of them!



As well as the UR3, UR5, UR10.

What can UR robots do in ROS2?

- For the most part, it can do the same things as the ROS1 package
- Teleoperation
- Compliance → Streaming commands
- Trajectory execution → Series of waypoints
- End-effector GPIO
- Force/torque sensors
- Motion speed scaling via teach pendant
 - Thanks FZI



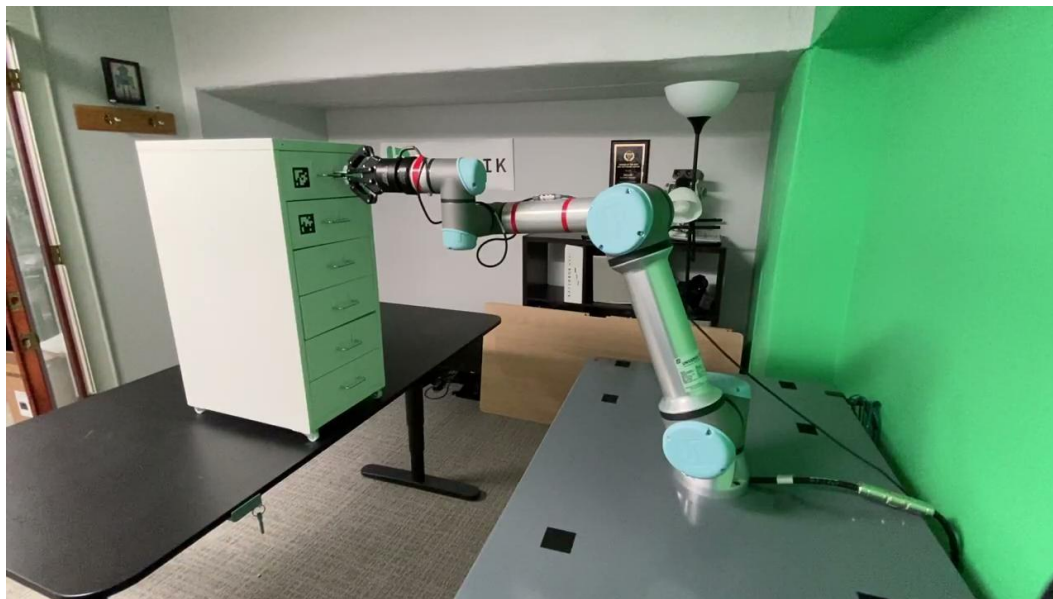
Video time!

"A video is worth a thousand words"



Trajectory Execution

This video uses ROS2 exclusively
Supervised autonomy with *Movel Studio*
Notice gripper integration



Streaming Commands - the Jeff Bezos video

This video uses ROS1 -- but it should be possible in ROS2 now
Shadow Robot and HaptX Dexterous Hand at Amazon re:MARS



Alan Boyle, GeekWire

Streaming Commands - Compliance

This video uses ROS1 -- but it should be possible in ROS2 now
FZI Karlsruhe



Benefits of flexible control modalities

Especially for startups and researchers, it's great to have a robot that --

- Is ROS-compatible
- Offers many control modalities
 - Streaming vs. trajectories
 - Position vs. velocity control
- Is somewhat hackable at a low level, when desired
 - Example:
 - We have a project where fast cycle time is critical
 - For a demonstration, we increased the robot acceleration limit by **4X**
 - Hardware longevity will suffer but we don't care

For people who “usually try to do unusual things”

Performance Benefits of ROS2

- ✓ “Nodes” → shared library, launched as a component in a single process
 - Often called “node components”
 - Less internal data transfer
 - Decreased latency
- ✓ Improved security
 - Tunable DDS middleware
- ✓ Improved communication between nodes
 - Tunable DDS middleware
- Deterministic launching
 - Node A needs to wait on Node B

Request for Beta testers

[https://github.com/PickNikRobotics/Universal Robots ROS2 Driver](https://github.com/PickNikRobotics/Universal_Robots_ROS2_Driver)

Limitations of the Beta release

- Trajectories cannot be executed in velocity mode yet
 - Easy to add, coming soon
- Yes acceleration-limited trajectories, no jerk-limited trajectories
 - Usually required by large industrial robots
 - Two promising new open-source packages have been released
 - [TopiCo](#)
 - [Ruckig](#)
 - Should be integrated with MoveIt in ~6 mos.
- Cannot run the kinematics calibration routine

Development challenges

- Continuously evolving ros2_control ABI/API
- Found it difficult to add arguments to launch files
 - Example from ROS1:

```
<arg name="robot_ip" value="192.168.1.14"/>
```

ROS2 launch file arguments

- **Declare arguments**

```
declared_arguments = []
declared_arguments.append(
    DeclareLaunchArgument(
        "robot_ip",
        description="IP address by which the robot can be reached.",
    )
)
```

- **Initialize arguments**

```
robot_ip = LaunchConfiguration("robot_ip")
```

- **Use the arguments**

```
robot_launch = IncludeLaunchDescription(
    PythonLaunchDescriptionSource([ThisLaunchFileDir(),
"/ur_control.launch.py"]),
    launch_arguments={
        "robot_ip": robot_ip,
        ...
    }
)
```

Contributors

- PickNik Robotics
 - Denis Stogl, Lovro Ivanov, Abi Sivaraman, Andy Zelenak, Nathan Brooks
 - FZI Forschungszentrum Informatik
 - Marvin Besselmann
 - Delft University of Technology
 - Gijs van der Hoorn
 - Universal Robots
 - Rune Sørensen
-