

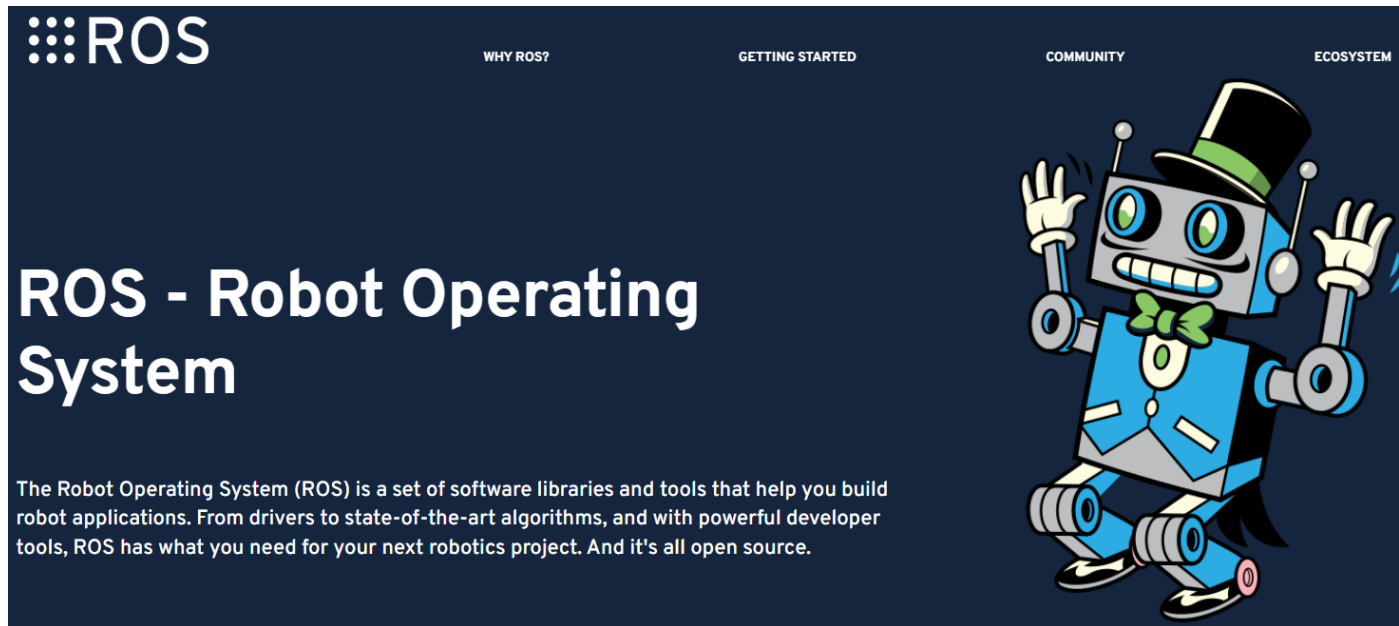
ROS2 – Fundamentals plus some practical insights



© maxon motor ag, 2026
Author: CCMC mmagwj
Sachseln, July 2026

maxon

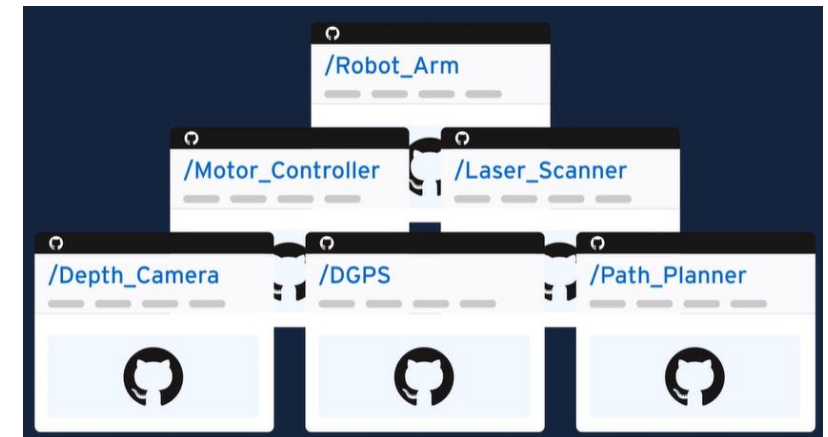
ROS – The official explanation



Check out
the official website:
[ROS: Home](https://www.ros.org)
(<https://www.ros.org>)

Watch the official «What is ROS» video:

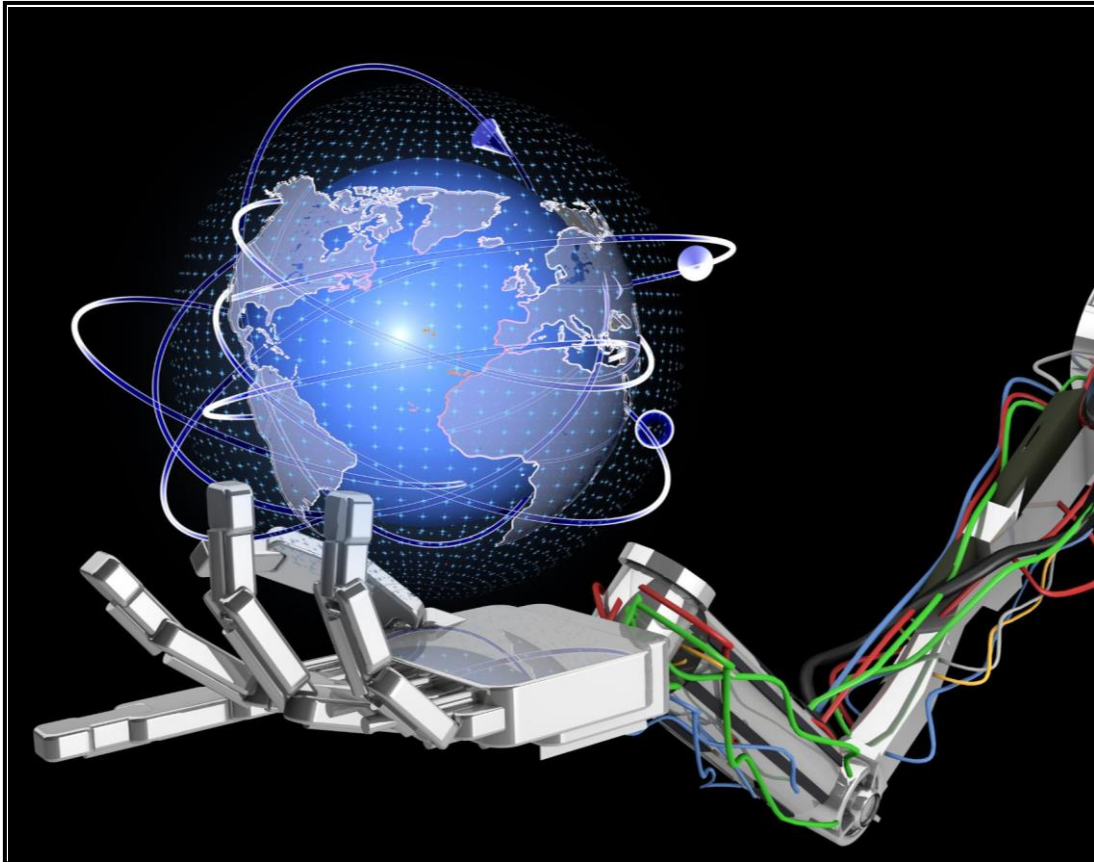
-> [ROS2 Introduction \(Vimeo, captioned\)](https://vimeo.com/639236696)
(<https://vimeo.com/639236696>)



ROS2

Design requirements

(or short comings of ROS1)



Security

ROS2 needs to be safe with proper encryption where needed.

Flexible system design

ROS2 needs to be able to run on embedded systems as well as to run and communicate across vast networks from LAN to multi-satellite hops.

Real-time computing

ROS2 needs to be able to perform computation in real-time reliably.

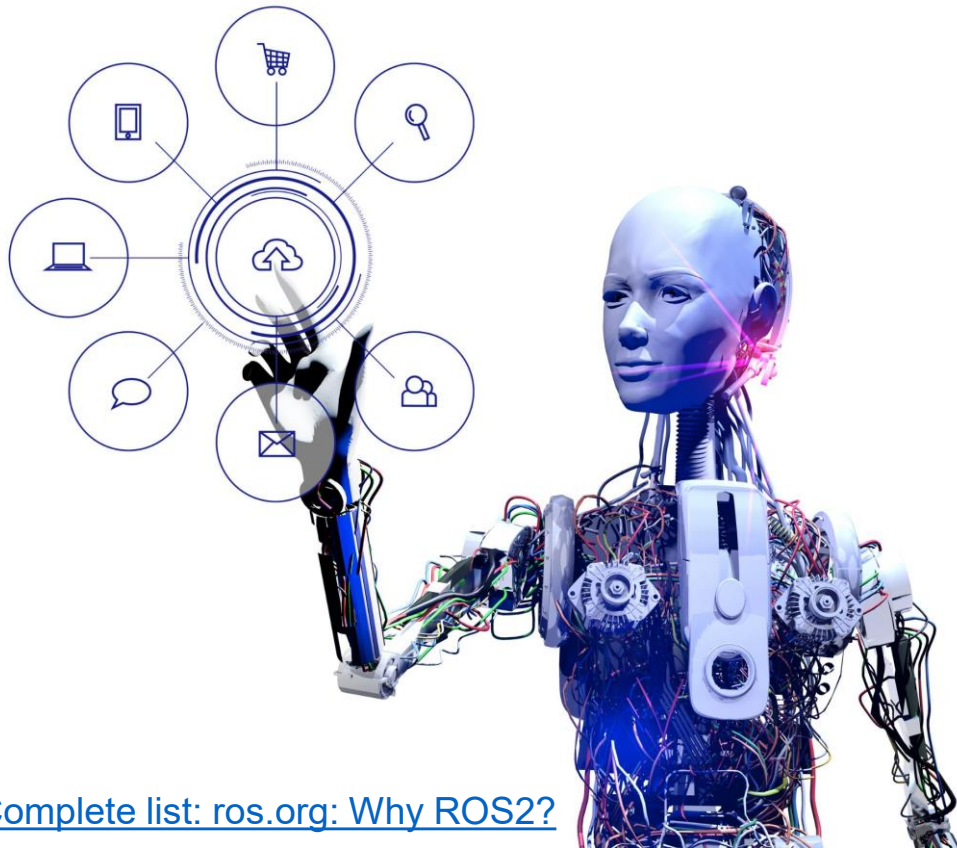
Product readiness

ROS2 needs to conform to relevant safety and industrial standards.

ROS2

Four main Objectives

(to overcome short comings of ROS1)



[Complete list: ros.org: Why ROS2?](https://www.ros.org/Why-ROS2/)

ROS2 - Modular architecture

ROS2 consists of defined building blocks that form a flexible and extensible system with clear interfaces, which can also be executed on multiple devices.

ROS2 - Platform Independence

ROS2 runs on various operating systems, such as Linux, Windows, and real-time OS.

ROS2 - Real-Time Capability

ROS2 supports real-time capability for multi-axis and synchronized operation of system components.

ROS2 – Designed for Industry

ROS2 offers high reliability and a safety critical design based on open standards which are widely deployed in industry as well as aerospace and military applications.

ROS2 Objectives versus Practice



ROS2 – Real Time Capability

- Real-time capability demands for an operating system which is real-time capable.
- There are quite often standard Linux versions in use, which are not real-time capable.

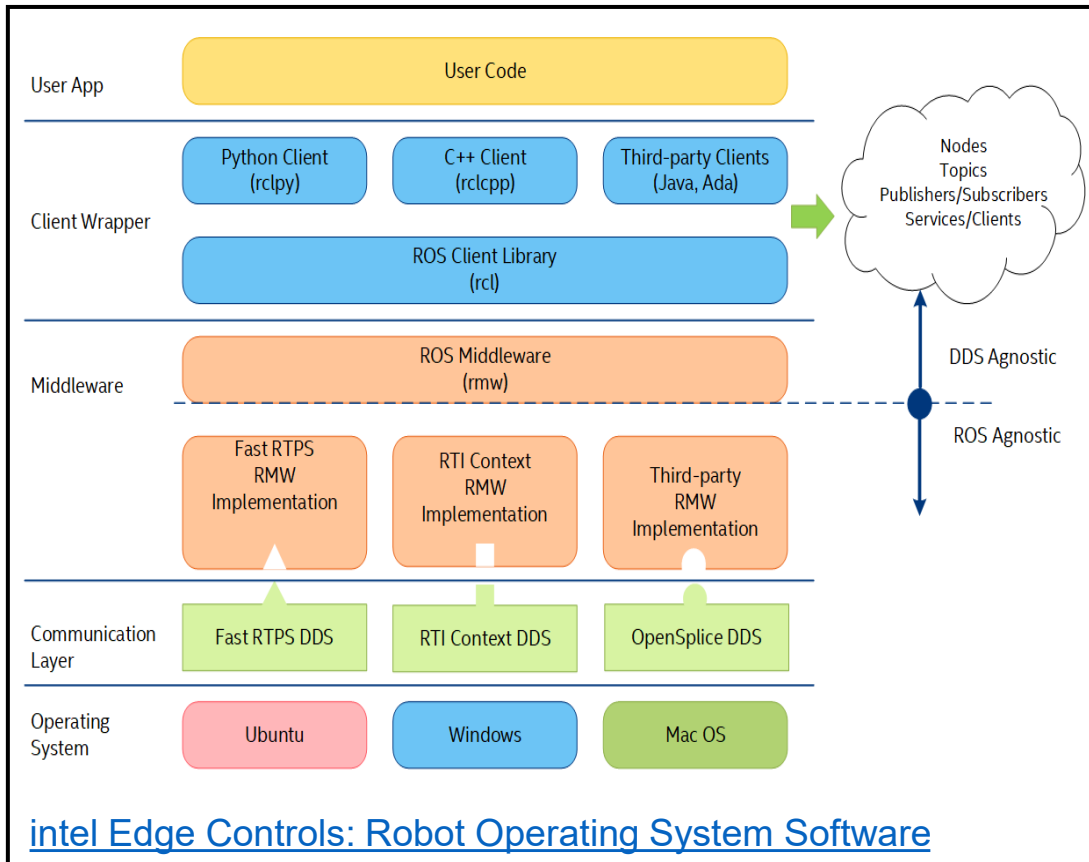
ROS2 – Platform Independence

In practice, platform independence is often severely limited out of mainly two restrictions:

- In case of a ROS2 real-time system, there is the need to select a real-time operating system resp. a Linux real-time version or add-on.
- Most of the ROS2 libraries and tools are based on specific Linux versions, i.e. do not interact properly with libraries or OS with mismatching versions.

ROS2

Fundamentals & System design



ROS - Common misconception

ROS is not(!) an operating system, even though the acronym stands for “Robotic Operating System”.

ROS2

is a Framework and Toolkit

for modular and structured robotic systems;
in other words, it is a framework and collection of open-source libraries and tools that are primarily developed by robotic engineers and students.

ROS2

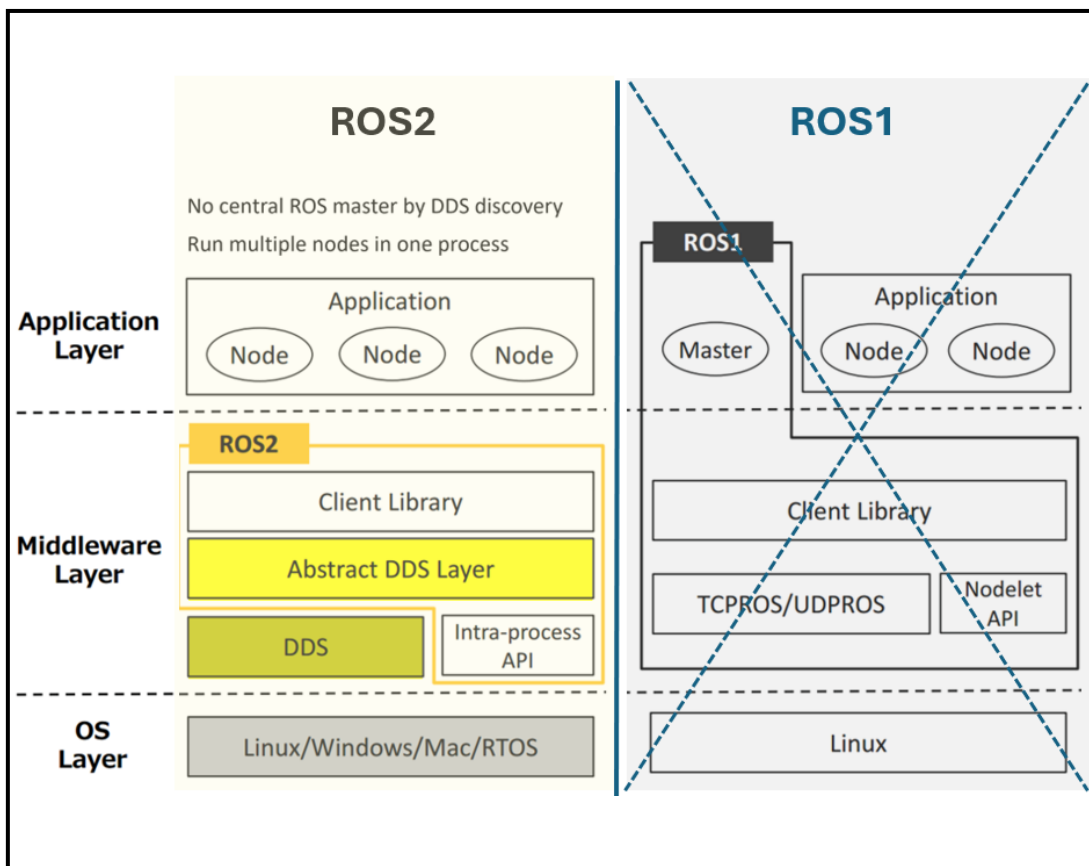
is a Middleware Software layer

that connects sensors, actuators, and controllers in industrial systems (such as robots, AGVs).

[MATLAB Help Center: Introduction to ROS2 basic concepts](#)

maxon

ROS2 standard: DDS Data Distribution Service



ROS2 & DDS

- DDS is the network protocol for all communication that happens internally in ROS2.
- ROS2 uses DDS in the background and abstracts technical details to keep operation simple.

DDS - Communication standard

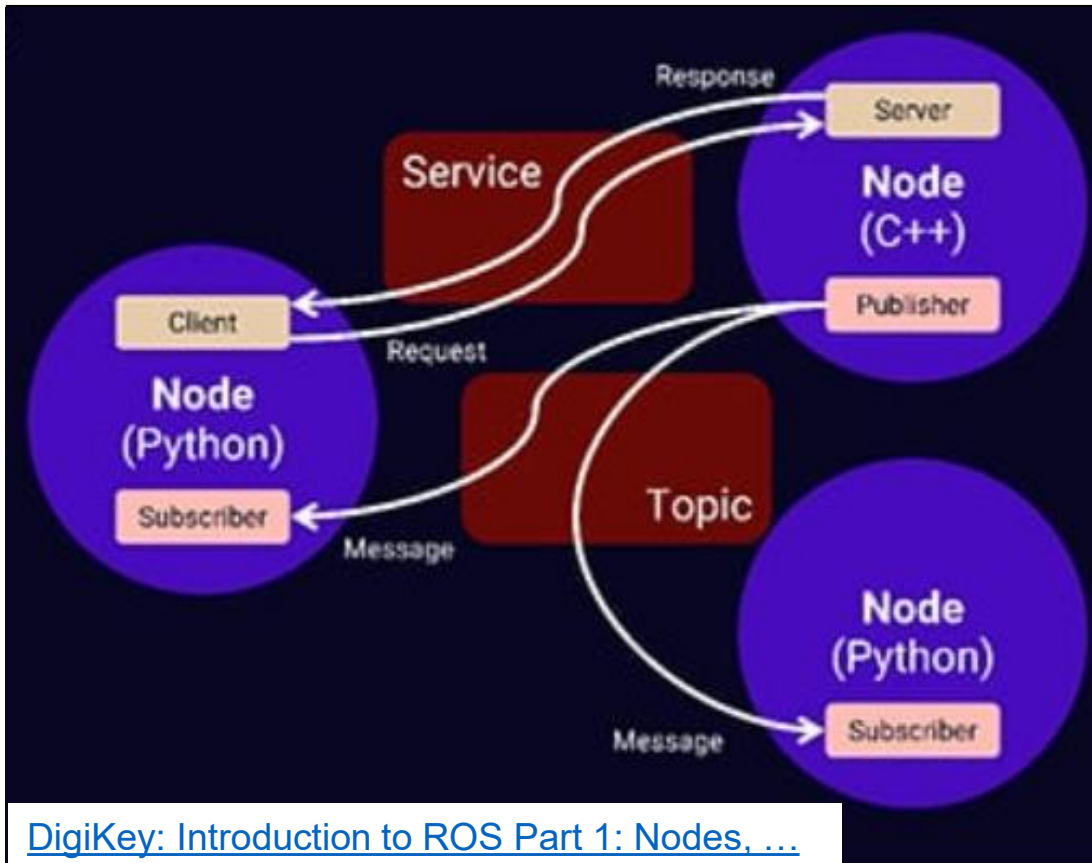
DDS is an established standard for data exchange in distributed systems and is used in various industrial sectors including safety critical systems in use by military, aerospace and space applications.

DDS - Automatic partner discovery

DDS provides a publish-subscribe data transport. Communication partners on the network are automatically discovered without a central “master”. This makes systems more robust.

ROS2

Basics & Terms of the system architecture



[DigiKey: Introduction to ROS Part 1: Nodes, ...](#)

Node: Process -> Action

- Self-contained processes that perform specific functions, e.g. reading a sensor, processing camera data, controlling a motor, ...

Topic: Communication channel

- Named communication channels for publishing and subscribing of data streams, e.g. streaming sensor data or broadcasting commands.

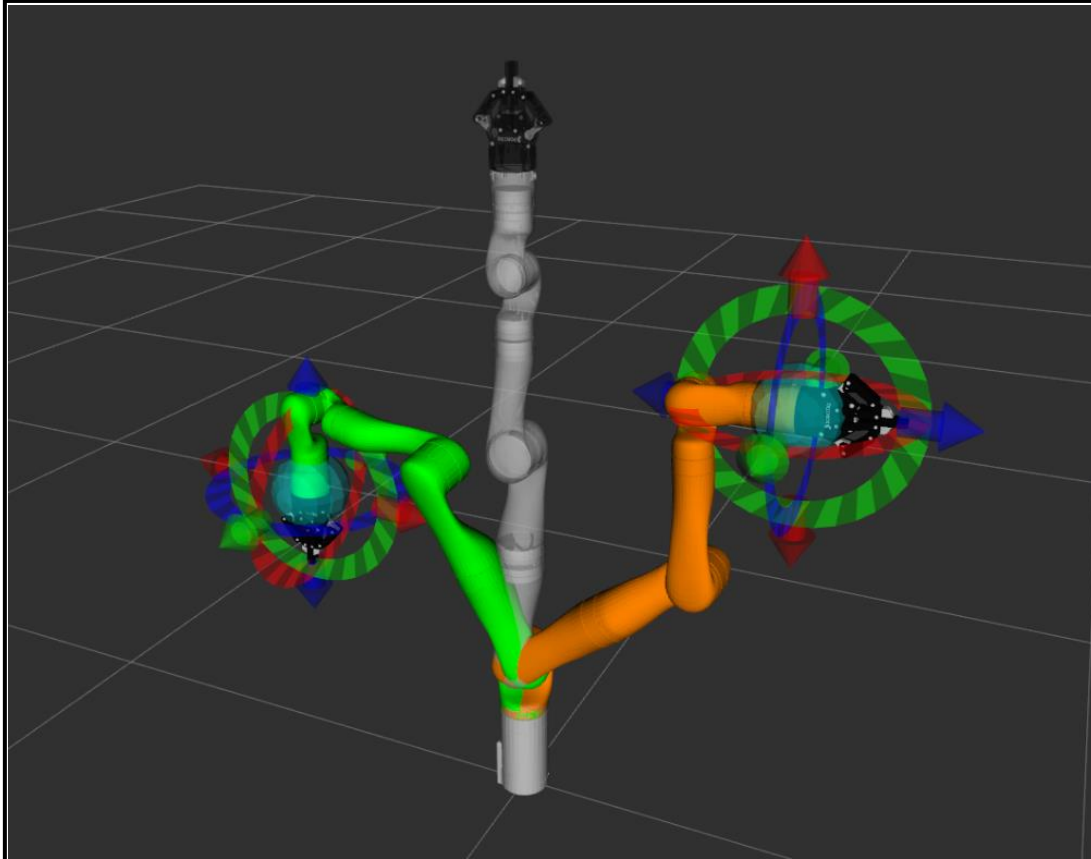
Service: Synchronous data interfaces

- Synchronous request / response interfaces for direct communication, e.g. querying a battery level.

Message: Data

- Structured data type with one or more strongly-typed fields.

ROS2 Tools



Visualization with [Rviz](#)

Graphical display of sensor data, maps, and robot states for better system analysis.

Simulation with [Gazebo](#)

Virtual testing of robot systems to verify behavior before deployment on real hardware.

Motion Planning and Control with [MoveIt](#)

Library for motion planning, control, inverse kinematics, and collision checking.

Diagnostics and Monitoring with [RQt](#)

Plugins for diagnostics and configuration to monitor the status of the robot system.

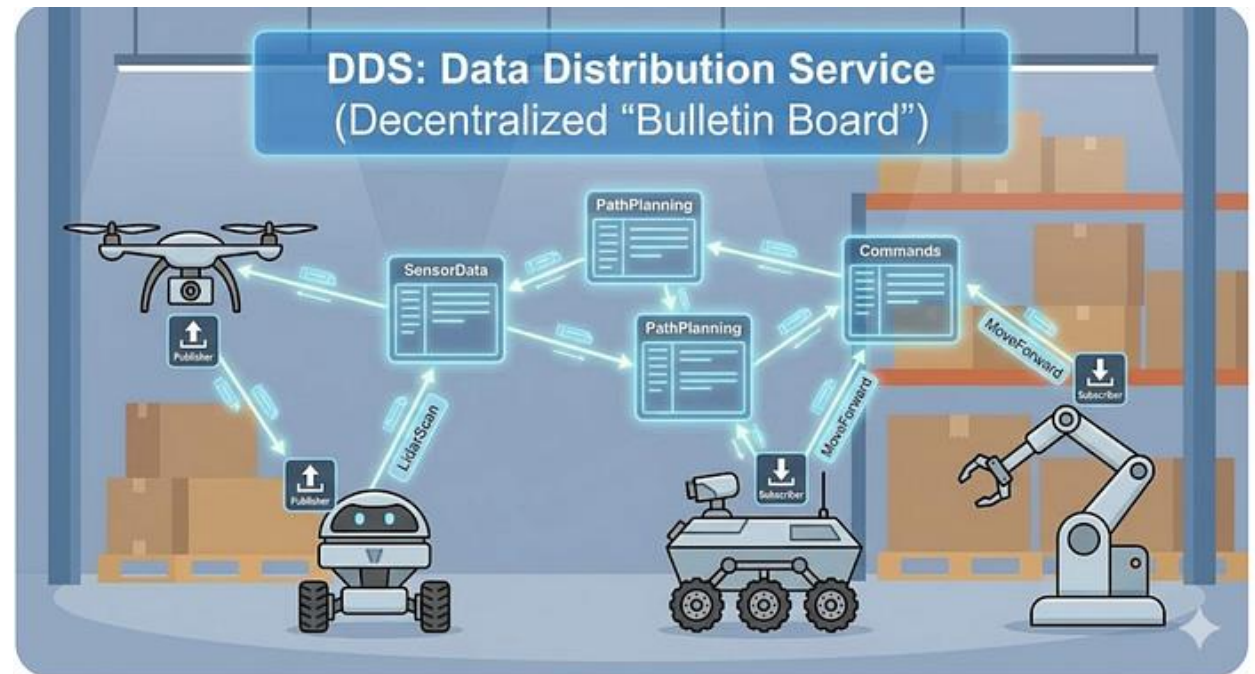
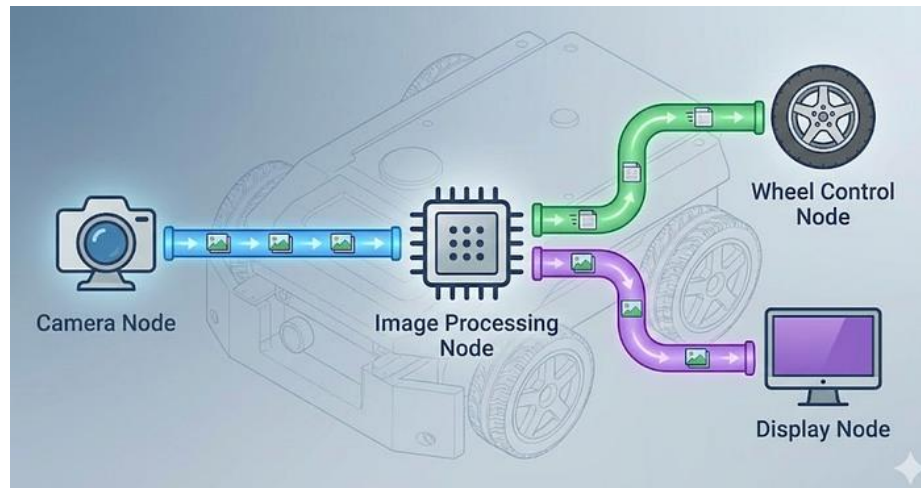
ROS2 Command Line ([CLI](#))

Control nodes, monitor topics, query system info.

ROS2 – Try to understand the idea

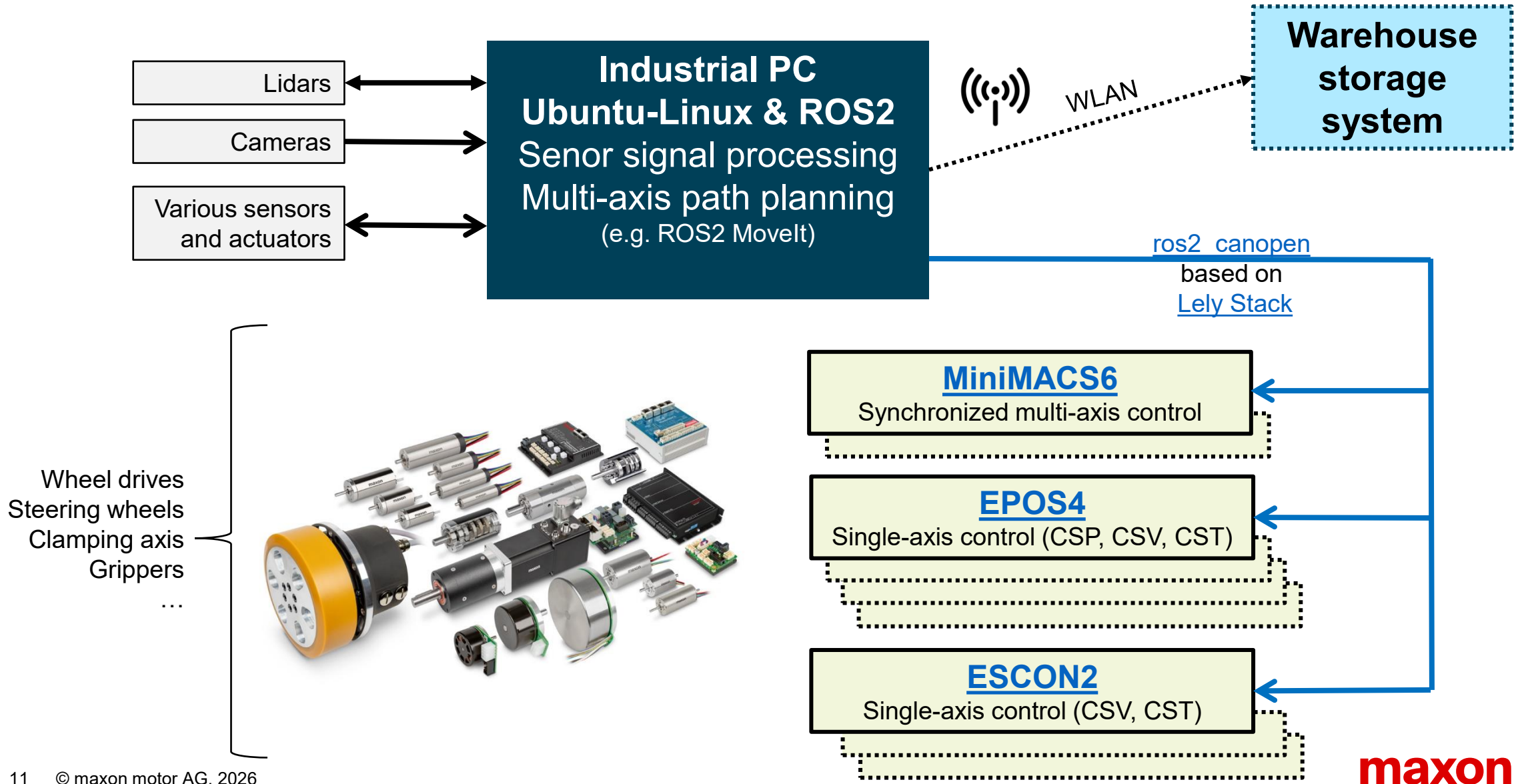
Think of a ROS2 system like a busy hospital. You have different specialists (doctors, nurses, lab techs) who need to share information in different ways depending on the situation.

- ROS2 acts as the “glue” in between the different “specialists” (= components = “Nodes”).
- ROS2 provides “packages” (= pre-written code = “Topics”) for common tasks.
- ROS abstracts the hardware, so that high-level code stays mostly the same even if design or components change.



[ROS2: Robot Operating System. 1. What is ROS2? | by Dilip Kumar | Medium](#)

A typical robot's system overview



maxon / zub: ROS2 AGV (prototype for testing)

Top-level controller

(reComputer J3011)

- NVIDIA Jetson Orin Nano 8GB
- Ubuntu 22.04 / ROS2 Humble

MiniMACS6-IF1 Ethernet

(Motion Controller)

Castor Wheels

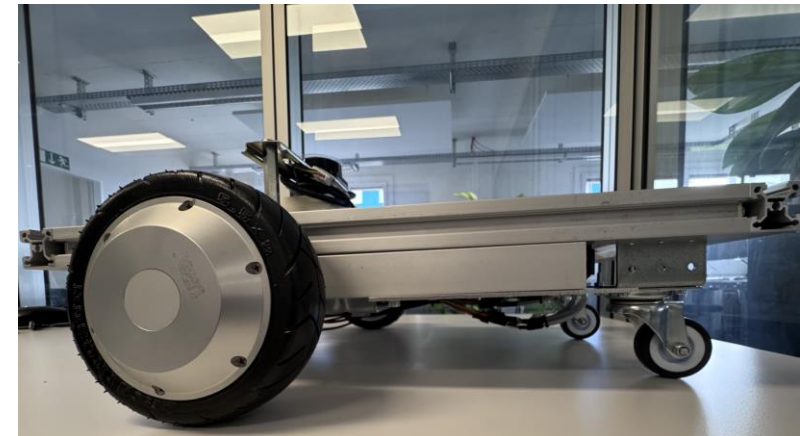
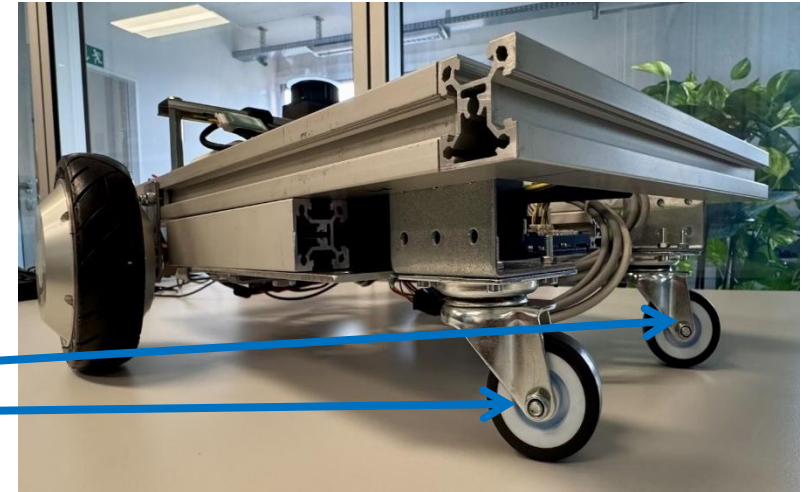
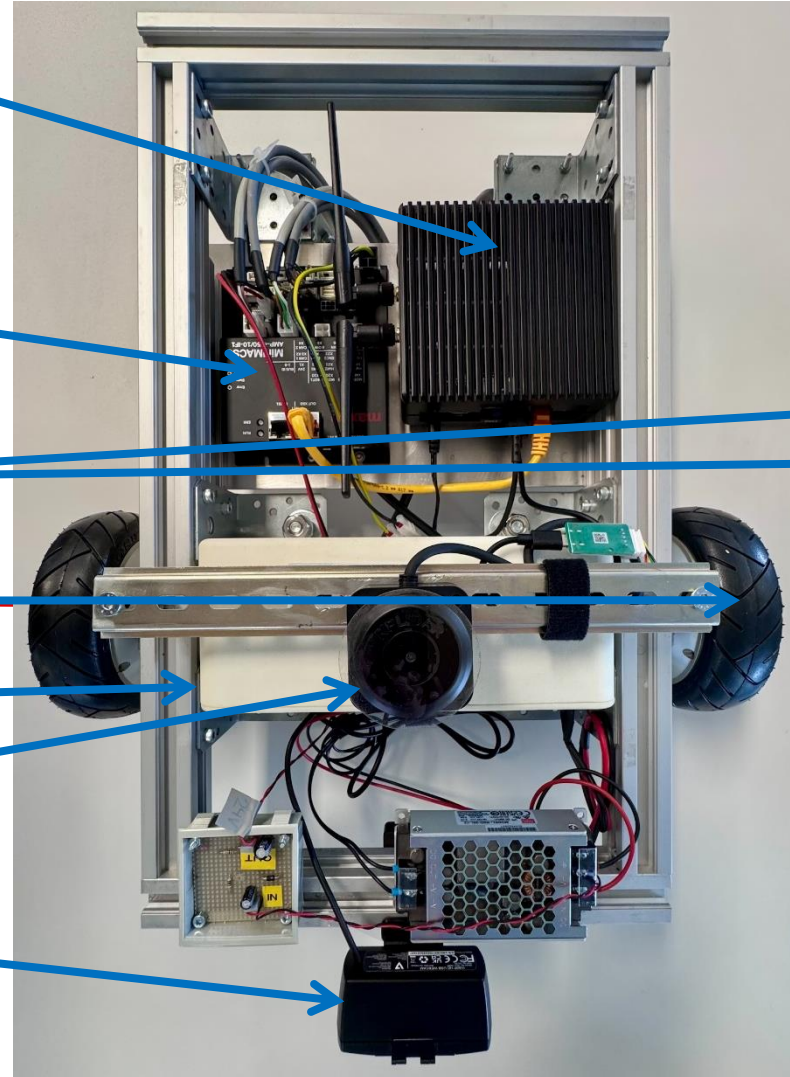
InWheel Drives

(EC90flat + MILE)

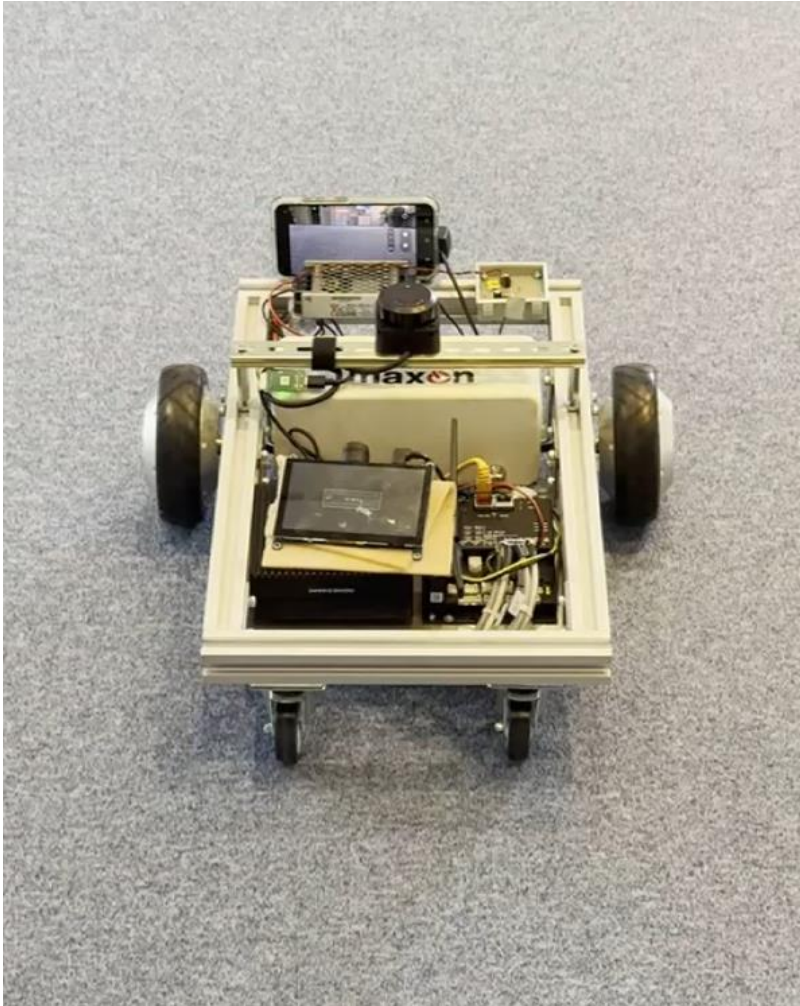
Battery

Lidar Sensor

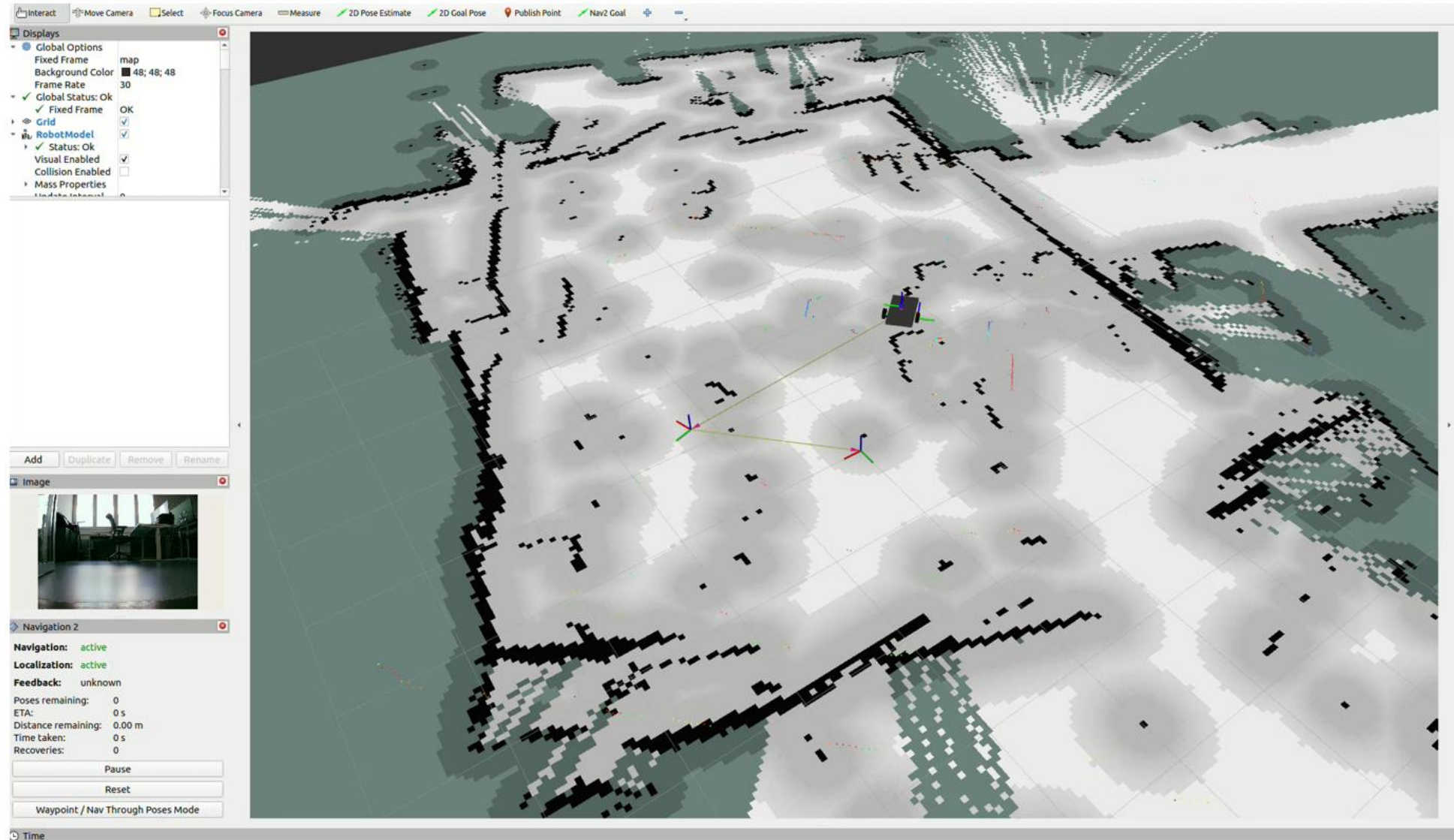
Camera



maxon / zub: See it with the eyes of ROS powered AGV



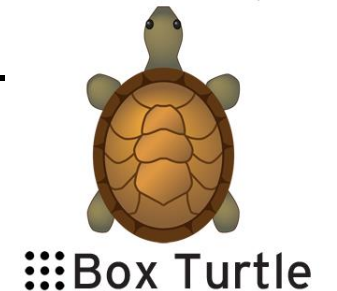
maxon / zub: Visualization of generated map (in Rviz)



ROS & ROS2: History

ROS development started on Nov. 2007

- Joint development of Stanford Artificial Intelligence Lab and “Willow Garage”. (“Willow Garage” is influential robotics research lab founded by a Google engineer in 2006)
- **First ROS 1.0** (= “ROS box Turtle”) was officially **released March 2010**.



ROS2 proposal in 2014 and start of development in 2015 / 2016

- ROS2 was primarily driven and developed by Open Robotics (OSRF) and the “ROS2 Technical Steering Committee”, which includes companies such as Amazon, Microsoft, Intel, and Apex.AI.
- ROS2 addressed the shortcomings of ROS by providing real-time, IT security, and cross-platform capabilities for fully industrial solutions.
- **First ROS2** version “Ardent Apalone” was officially **released Dec. 2017**. ROS2 follows a yearly release cadence every May 23rd plus 4 year support.



ROS2: Up-to-date & supported versions

<https://docs.ros.org/>



LATEST ROS 2 LTS RELEASE

Lyrical Luth

Our **latest long term support (LTS)** ROS 2 distro, and the one we recommended for all ROS users.

Released May 2026

[Documentation](#)

[Platform Support](#)



Jazzy Jalisco

A ROS 2 distro with support until May, 2029.

[Documentation](#)



Humble Hawksbill

A ROS 2 distro with support until May, 2027.

[Documentation](#)



Rolling Ridley

Our **rolling release** and the bleeding edge!

[Documentation](#)



Kilted Kaiju

A non LTS ROS 2 distro with support until November, 2026.

[Documentation](#)

ROS2 – Why is ROS2 still not widely in use in industry?

- **Dominance of Established Proprietary Solutions:**

Industrial robot manufacturers (such as FANUC, KUKA, and ABB) have relied for decades on their own, highly optimized control systems. These systems are considered fail-safe, robust, and offer direct, professional support. Switching to a new platform like ROS2 entails high costs and risks.

- **Perception and Certification (Safety):**

ROS2 is often still associated with its origins in academic research rather than directly with industry-ready, certified software. Obtaining safety certifications (e.g., SIL 2/3) requires complex safety verifications, which are time-consuming for open-source software.

- **Complexity and development effort:**

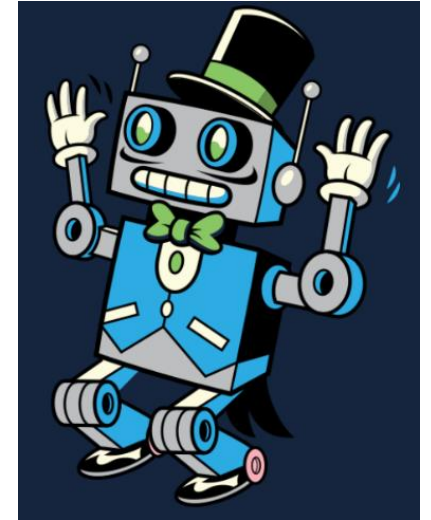
ROS2 is a powerful but complex framework. It requires highly qualified engineers who possess both ROS expertise plus deep understanding of motion control and real-time OS. Even companies which use ROS2 and gathered required knowledge did not want to share details of their expertise and system integration. They rate their knowledge of “How to use and integrate / customize ROS2 components” as their “IP – Intellectual Property”.

ROS2: Video and Tutorial / Document links

www.ros.org

Video links

- [ROS Introduction: What is ROS?](#)
- [Introduction to ROS Part 1: What is the Robot Operating System](#)
- [What is ROS2? – Framework Overview](#)
- [Getting started with ROS2](#)
- [ROS2 Crashkurs - ROS1 oder ROS2 - Einsteiger-Tutorial](#)



Documents:

- <https://docs.ros.org>
- [Introduction to Robot Operating System 2 \(ROS 2\) - MATLAB & Simulink](#)
- [ROS2: Robot Operating System. 1. What is ROS2? | by Dilip Kumar | Medium](#)
- [Robot Operating System 2: Design, architecture, and uses in the wild | Science Robotics](#)
- [Autonome Systeme – Roboter-Betriebssysteme: ROS2 bügelt Schwächen aus](#)
- [ROS2 vs. ROS1— key differences and which one is better? | by Omar Elmofty | Medium](#)
- [OSS Robotik-Framework ROS / ROS 2 – RimA | Roboter im Alltag](#)

ROS2: Cross references, Libraries and add-on links

Technical details:

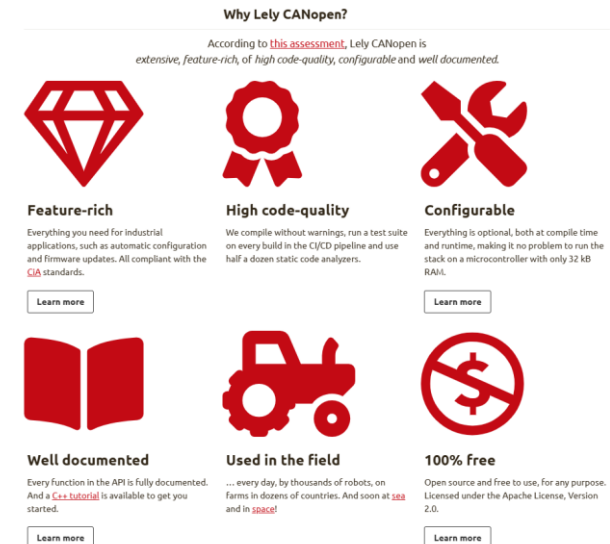
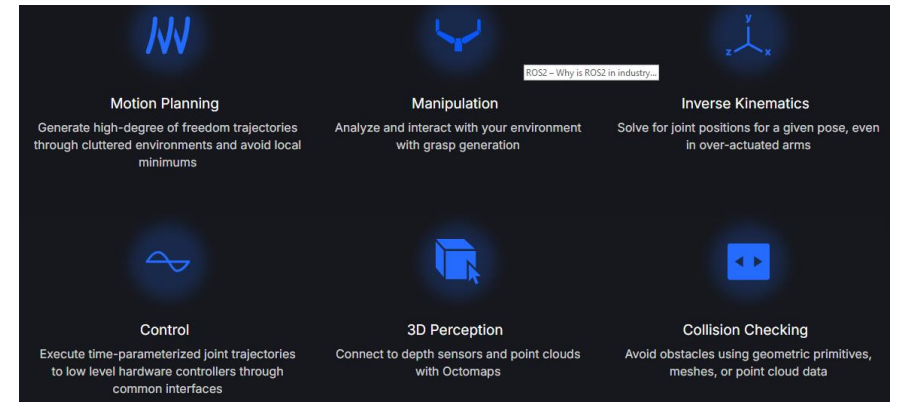
- [Understanding nodes — ROS2 Documentation](#)
- [Understanding topics — ROS2 Documentation](#)
- [ROS on DDS](#)

Most popular ROS2 libraries:

- [MoveIt: Motion Planning Framework](#)
- [MoveIt · GitHub](#) / [MoveIt – Tutorials](#)
- [Lely CANopen](#) / [Lely CANopen FAQs](#)
- [ros2_canopen: CANopen driver framework for ROS2 · GitHub](#)
- [maxon / zub's EPOS4 & ROS2 package](#)

ROS2 feedback based on practical experience:

- [How popular is ROS2 actually in the robotics industry? : r/ROS](#)
- [What companies actually use ROS2 in production? : r/ROS](#)



Thanks for your attention



maxon