

Agenda & Training Details

DAY 1 - ROS2 Fundamentals

Monday, October 25, 2021

9:00 am CEST	Opening
9:15 - 10:30 am CEST	ROS2 Basics Session 1
10:30 am CEST	Coffee Break
10:45 am - 12:15 pm CEST	ROS2 Basics Session 2
12:15 - 1:00 pm CEST	Lunch Break
1:00 - 1:15 pm CEST	Real Robot Project Introduction
1:15 - 4:15 pm CEST	Work on real robot project + 1hr practice on real robots

- **ROS2 BASIC CONCEPTS:**

- Learn how to work with ROS2 packages
- Create your first Hello World python ROS2 program
- Create your first Hello World ROS2 launch file
- Understand the setup.py file for source installation for ROS2 python based packages
- Understand how ROS2 nodes work.
- Use environmental ROS2 variables

- **ROS2 TOPICS:**

- Learn about all the ROS2 Topic basic commands, like listing running topics and message types.
- Publish a ROS2 topic message through the command line
- Create a ROS2 topic publisher and ROS2 topic subscriber in python.
- Learn how to create a custom ROS2 topic and use it in a python script.
- Learn how to remap topics in ROS2 in a launch file.

- **MANAGING NODES:**

- Learn about the problem of using multiple callbacks in the same python script.
- Learn how Callback groups and multithreading can solve those issues.
- Create multiple publishers, subscriber python ROS2 script, that uses multithreading and callback groups to manage all the callbacks and threads to make the robot have updated values of the laser and odometry to decide how to avoid an obstacle.

- **DEBUGGING ROS2 PROGRAMS**

- How to use and manage different levels of Debugging messages in ROS2 python scripts.
- Learn how to visualize Camera Images, Pointclouds, TFs, robot models, and other complex data in RVIZ2.
- Learn how to generate a PDF with the TF tree structure of the current ROS2 robot-description system.
- Learn about ROS2Doctor debugging tool command, and how it can help you detect unused topics, old packages, and other system data like the RMW MIDDLEWARE used Topics list, and network setup.

DAY 2: ADVANCED ROS2 BASICS

Tuesday, October 26, 2021

9:00 - 10:30 am CEST	ROS2 Basics Session 3
----------------------	------------------------------

10:30 am CEST	Coffee Break
10:45 am - 12:15 pm CEST	ROS2 Basics Session 4
12:15 - 1:00 pm CEST	Lunch Break
1:00 - 4:00 pm CEST	Work on real robot project + 1hr practice on real robots

• UNDERSTANDING DDS IN ROS2

- Learn about the reasons behind using DDS in ROS2 instead of the ROS1 TCP.
- Learn the basic morphology and features of the default RMW MIDDLEWARE DDS used CycloneDDS.
- Learn about the three supported DDS implementations in Galactic.
- Learn how to change the DDS used to then execute scripts using that DDS implementation.
- Learn about what is the ROS2 Daemon and what's its relation with DDS.
- Learn about the need for Quality of Services (QoS) systems in ROS2.
- Learn how to start publishers and subscribers with different QoS and their compatibility.
- How to change QoS in ROSbags.
- How to change QoS setup in RVIZ2

• ROS2 SERVICES

- Learn about all the ROS2 Service basic commands, like listing running services and message types.
- Learn about the ROS2 services that can be synchronous or asynchronous.
- Create your first ROS2 service server and service clients.
- Create a custom Service message and use it in a python script to control the movement of the simulated Neobotix Mp-400.

• ROS2 ACTIONS

- Learn about the new update in ROS2 having actions their own space.
- Learn about all the ROS2 Action basic commands, like listing running actions and message types.
- Create your first ROS2 action server and action clients.
- Create a custom Action message and use it in a python script to control the movement of the simulated Neobotix Mp-400

DAY 3: ROS2 NAVIGATION2

Wednesday, October 27, 2021

9:00 - 10:30 am CEST	ROS2 NAV2 Session 1
10:30 am CEST	Coffee Break
10:45 am - 12:15 pm CEST	ROS2 NAV2 Session 2
12:15 - 1:00 pm CEST	Lunch Break
1:00 - 4:00 pm CEST	Work on real robot project + 1hr practice on real robots

- How to configure and launch a full navigation stack in ROS2
- How to create a map of the environment using SLAM-Toolbox
- How to provide the map to other applications using a map server
- How to localize the robot using the AMCL particle filter
- How to plan paths from one location to another using the planner for differential robots
- How to make the robot follow a trajectory while avoiding obstacles using the controller
- How to configure the costmaps (global and local) for effective obstacle avoidance
- How to use Behavior Trees to control the navigation behavior of the robot
- How to manage recovery behaviors
- How to manage and synchronize the navigation nodes using the BT-Navgation-Manager

- How to programmatically send requests and goals to the navigation system from external nodes using Python3

DAY 4: PERCEPTION & START MANIPULATION

Thursday, October 28, 2021

9:00 - 10:30 am CEST	ROS2 Perception
10:30 am CEST	Coffee Break
10:45 am - 12:15 pm CEST	ROS2 Manipulation Session 1
12:15 - 1:00 pm CEST	Lunch Break
1:00 - 4:00 pm CEST	Work on real robot project + 1hr practice on real robots

- Identify the position of the object to pick using a depth camera
- Provide the object location to the grasping algorithm
- Create a MoveIt2 package config specific to your robot
- Port a MoveIt1 package config to MoveIt2

DAY5: GRASPING USING MOVEIT2

Friday, October 29, 2021

9:00 - 10:30 am CEST	ROS2 Manipulation Session 2
10:30 am CEST	Coffee Break
10:45 am - 12:15 pm CEST	ROS2 Manipulation Session 3
12:15 - 1:00 pm CEST	Lunch Break
1:00 - 4:00 pm CEST	Work on real robot project + 1hr practice on real robots
4:00 - 4:15 pm CEST	Closing Ceremony

- Arm Trajectory Planning using MoveIt2
- Open Motion Planning Library (OMPL) basic configuration
- Test different planning algorithms such as RRT* or PRM*
- Create programs to control a robot arm using the MoveIt2 C++ API
- Gripper control using the Gripper Command action server
- Bridge of action servers from ROS1 to ROS2 using the action_bridge node
- Full Pick & Place pipeline based on:
 - Plan to the position of the object
 - Approach the object
 - Pick the object
 - Lift the object
 - Plan to the release position
 - Release the object