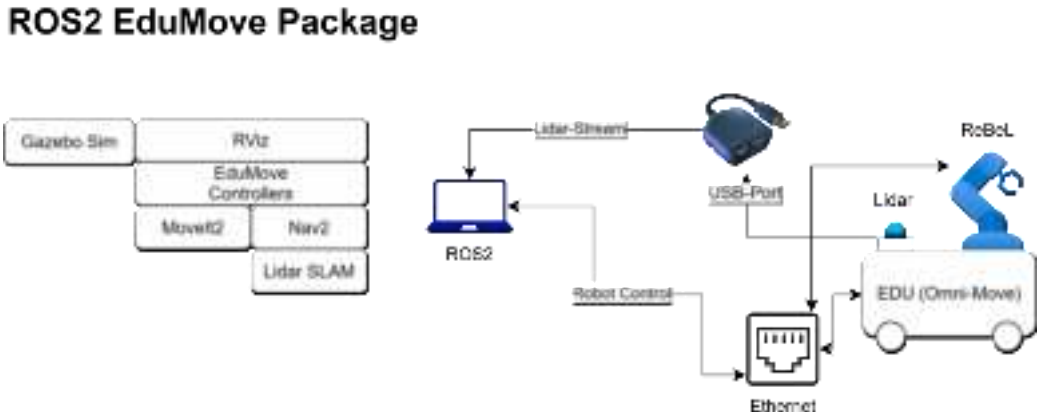
Dokumentation

ReBeL Edumove ROS Package

Abbildung 1 ReBeL EduMove

General Architecture

The control architecture of the EduMove system is modular and based on a combination of embedded computing, network communication, and the ROS2 framework:



- Die Verbindung zwischen der Steuerung und EduMove erfolgt nach jetzigem Stand über Ethernet.
- Die Informationen aus dem Lidarsensor werden über USB-Port an die ROS-Umgebung übermittelt

The following key components are summarized:

Visualisierung:

- **RViz2** is used to visualize robot state, sensor data, and navigation results.

Simulation:

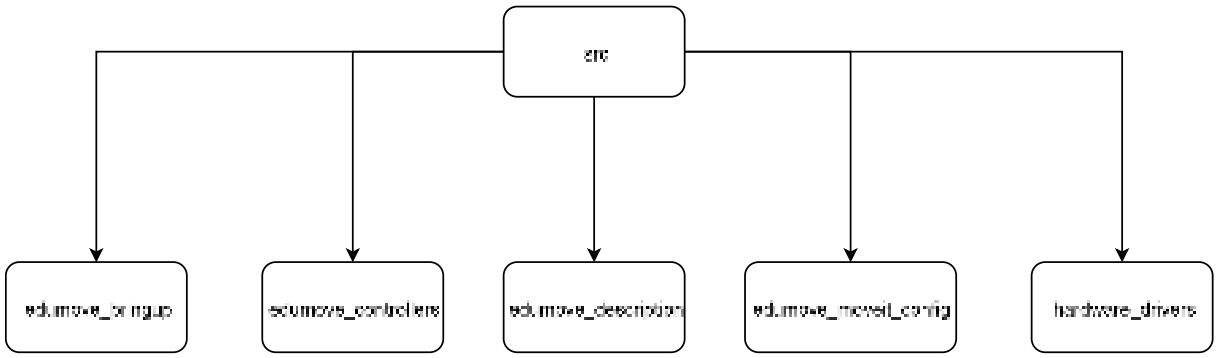
- **Gazebo** is used as the simulation environment to model the platform and the robotic arm and to perform tests.

Navigation:

- Zur For localization and mapping, the RPLidar A2M12 is used in combination with a **SLAM-Algorithmus**.
- The **Nav2-Stack** is implemented for autonomous navigation, handling path planning and collision avoidance.

Workspace Architecture ROS

The packages in the ROS workspace are located in the **/src** folder. There are five packages:



edumove_bringup:

- All configuration files for the workspace are stored under **config**, divided into **software** and **hardware**.
- All launch files are included in this package, also divided into **software** and **hardware**.
- Pre-generated maps are stored under **maps**.
- An automated script for map saving is located under **scripts**.

edumove_controllers:

- Hybrid interface for controlling the edumove platform and ReBeL-arm.

edumove_description:

- Contains meshes and URDF files for the ReBeL arm and the mobile platform.

edumove_moveit_config:

- Configuration package for using MoveIt, customized for the EduMove system.

hardware_drivers:

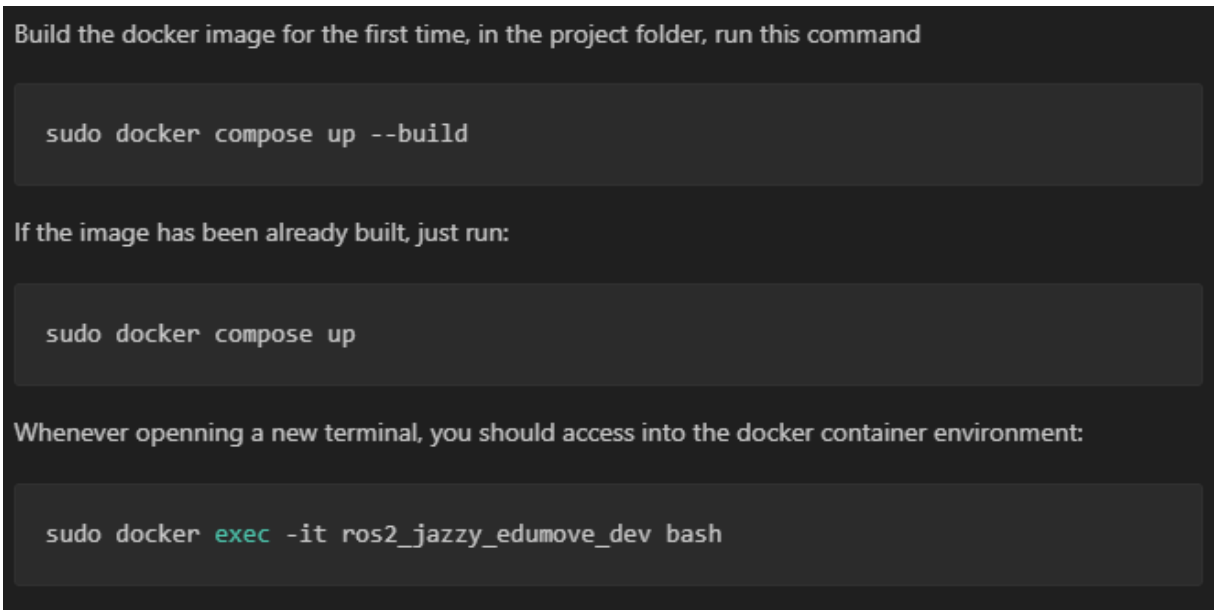
- Repo RPLidar A2M12

How to start for the first time

The software solution can be run on a computer with ROS Jazzy installed, or in a Docker environment. Both options are also described in a README file within the ROS workspace (located in the project folder).
Hint:

- For more information for using Nav2 see Nav2 doc.
- For more information for using MoveIt2 see MoveIt2 doc.

With Docker, execute the following commands in a terminal started from the project folder:



Without Docker, execute the following commands in a terminal started from the project folder.

Attention: Make sure that all required packages and libraries are installed on your device (this depends on how you installed ROS). The packages can be found in the package.xml files of the ROS packages within the workspace.
For installing with rosdep (in case you not already installed rosdep):

install rosdep:

```
apt-get install python3-rosdep
```

install dependencies:

```
sudo rosdep init
rosdep update
rosdep install --from-paths src -y --ignore-src
```

With the dependencies installed, you can build your workspace:

Build the code directly in the project folder:

```
colcon build
```

How to start mapping

Do not forget to source the ROS workspace before launching.

- Launch the main program:

```
ros2 launch edumove_bringup bringup.launch.py use_gui:='true' mode:='mapping'
```

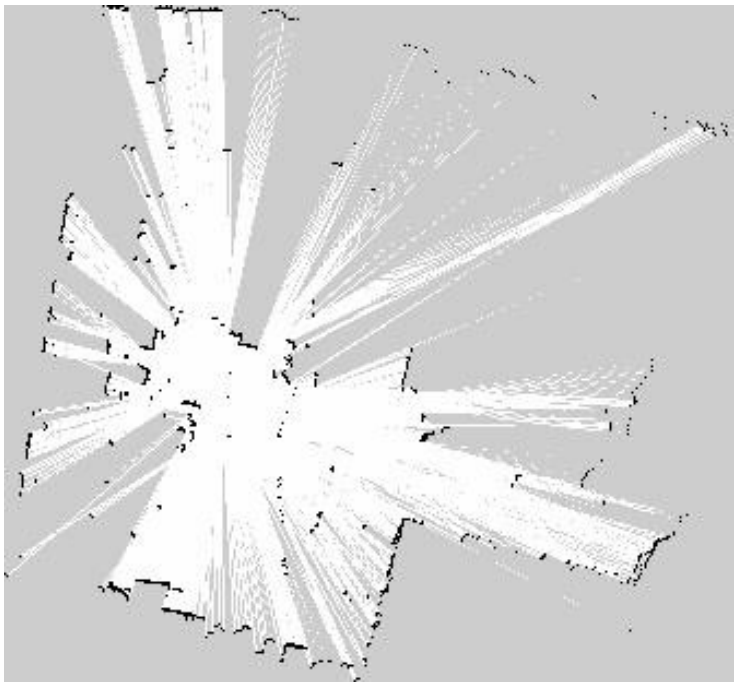
- Drive the platform using teleop_twist_keyboard

```
ros2 run teleop_twist_keyboard teleop_twist_keyboard --ros-args --remap
cmd_vel:=cmd_vel/joy
```

- Save map:

```
ros2 run edumove_bringup custom_map_saver.py --ros-args -p
map_name:=edumove_map
```

A map is created as a **pgm** file, and a **yaml** file references it. The map looks as follows:



How to start navigation program

Do not forget to source the ROS workspace before launching.

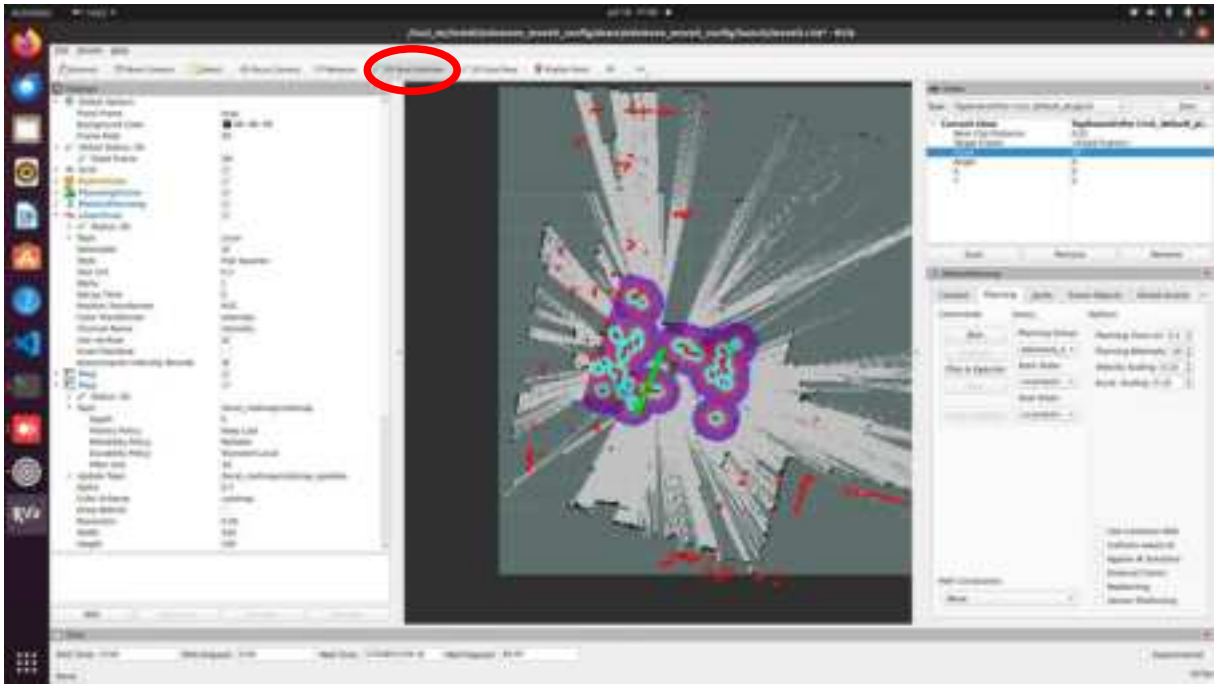
- Launch the main program:

```
ros2 launch edumove_bringup bringup.launch.py use_gui:='true'
mode:='localization' map_name:='edumove_map'
```

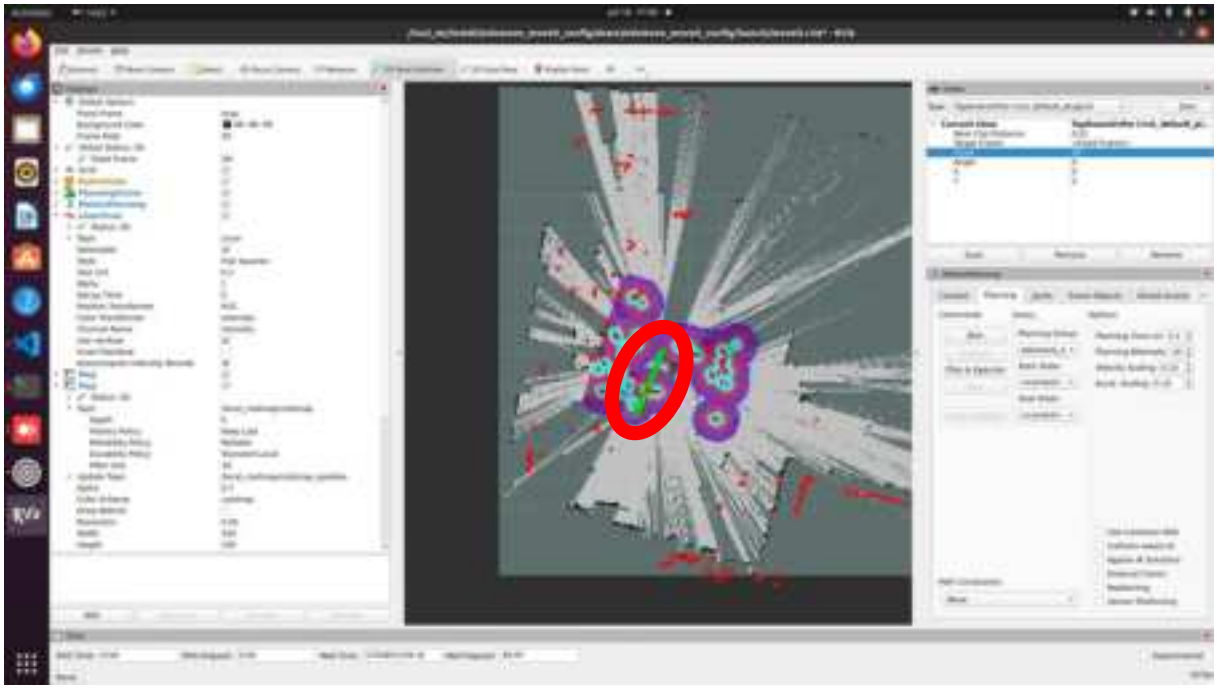
You should replace edumove_map by your own map

- Set an initial pose on rviz
- Set goal on rviz

The initial position must be set via the RViz interface after startup. To do this, click the following button in the top toolbar:



After that, click (and hold) on the map at the estimated position where EduMove is located, then drag in the direction EduMove is oriented, represented by the green arrow:



Setting a target position works in the same way. First, select **2D Goal Pose**, then click on a point on the map and drag to set the desired target orientation. After this, the planning process within Nav2 begins: