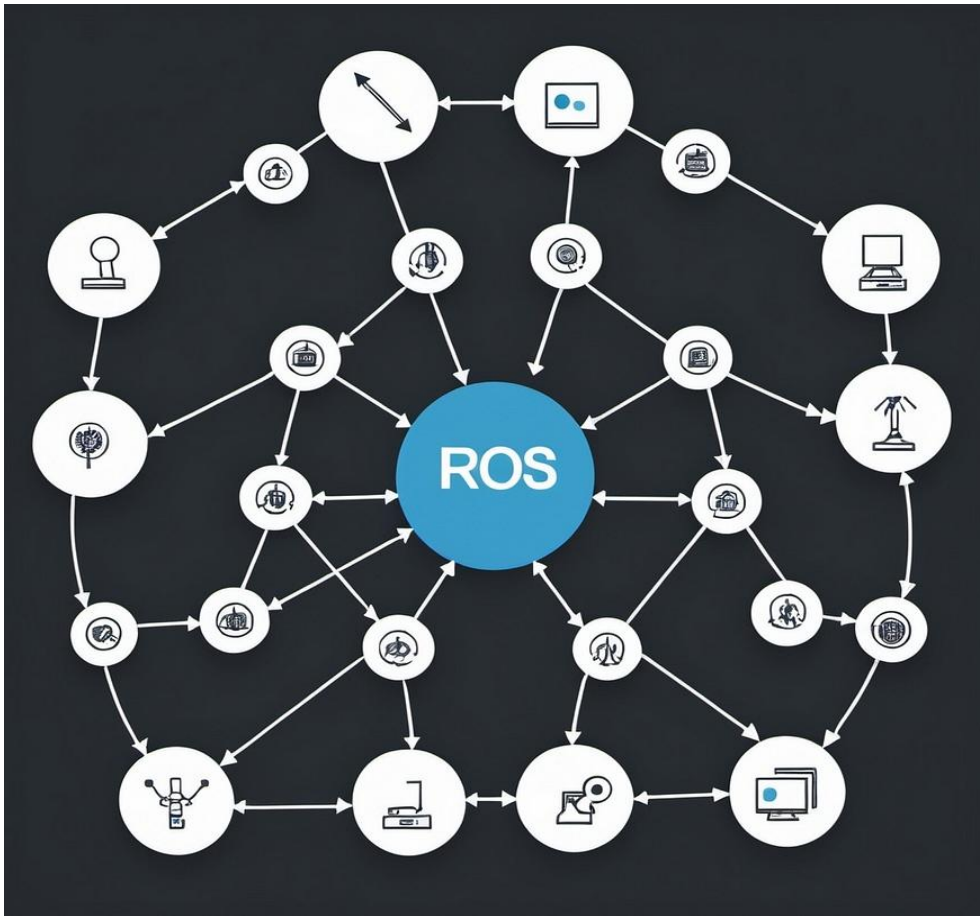


ROS2 Tutorial

Introduction to Intelligent Systems

Robot Learning Lab

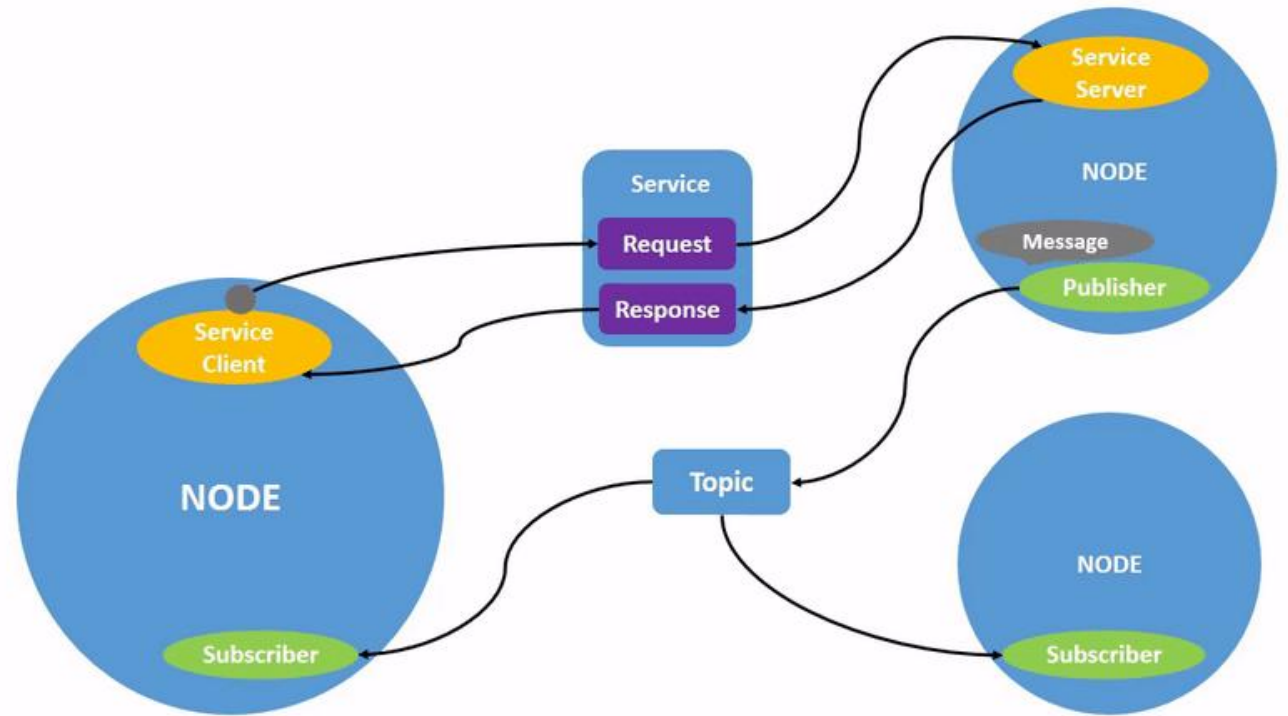
What is ROS2?



- Framework for building modular robot applications
- Functions are divided into 'nodes' that communicate with each other
- Uses DDS (Data Distribution Service) for reliable data exchange
- Ideal for complex systems like autonomous vehicles or industrial robots

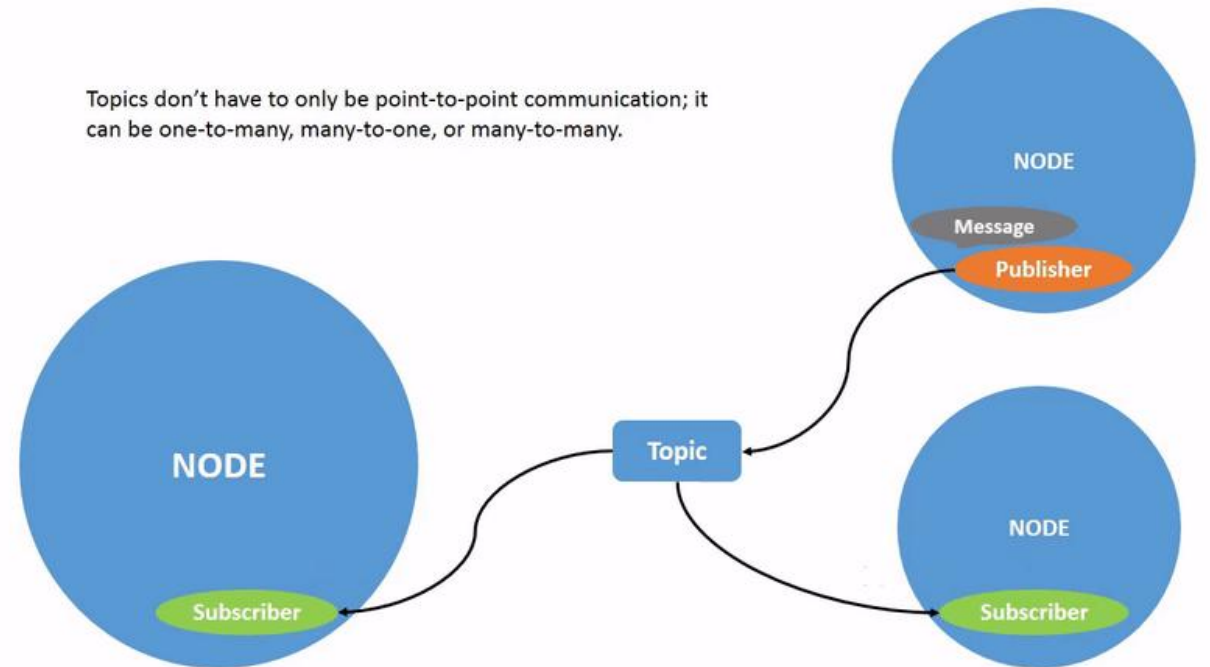
ROS2 Architecture

- Node
 - Smallest executable program
- Communication types
 - Topics, services, actions

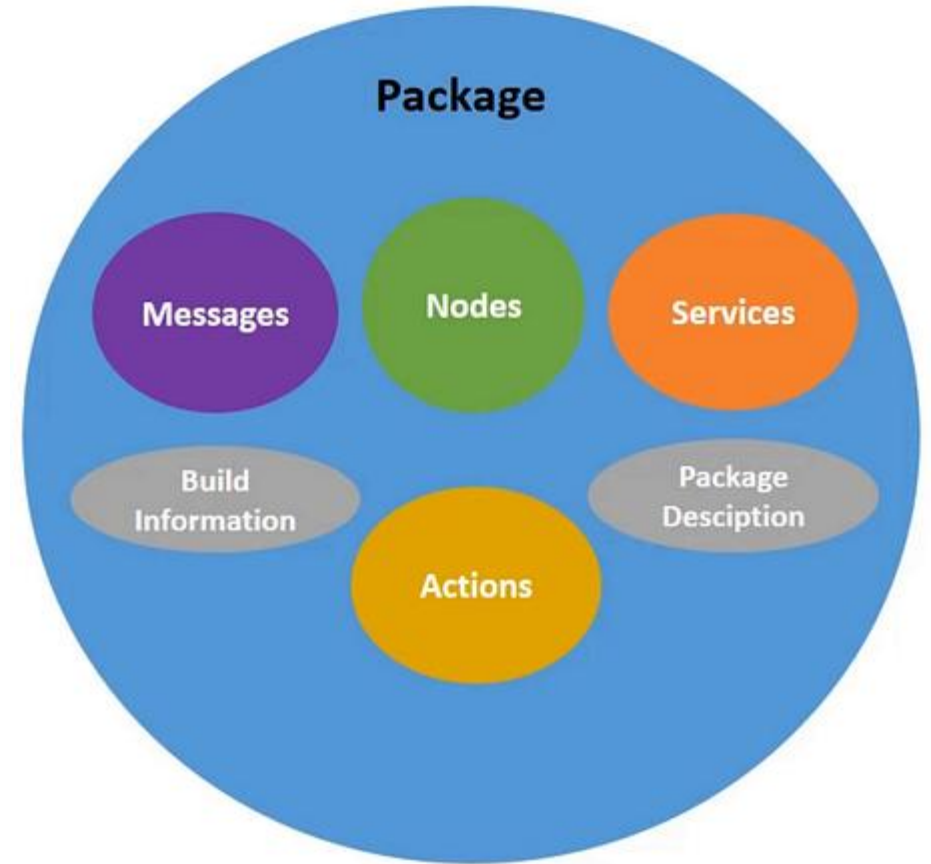
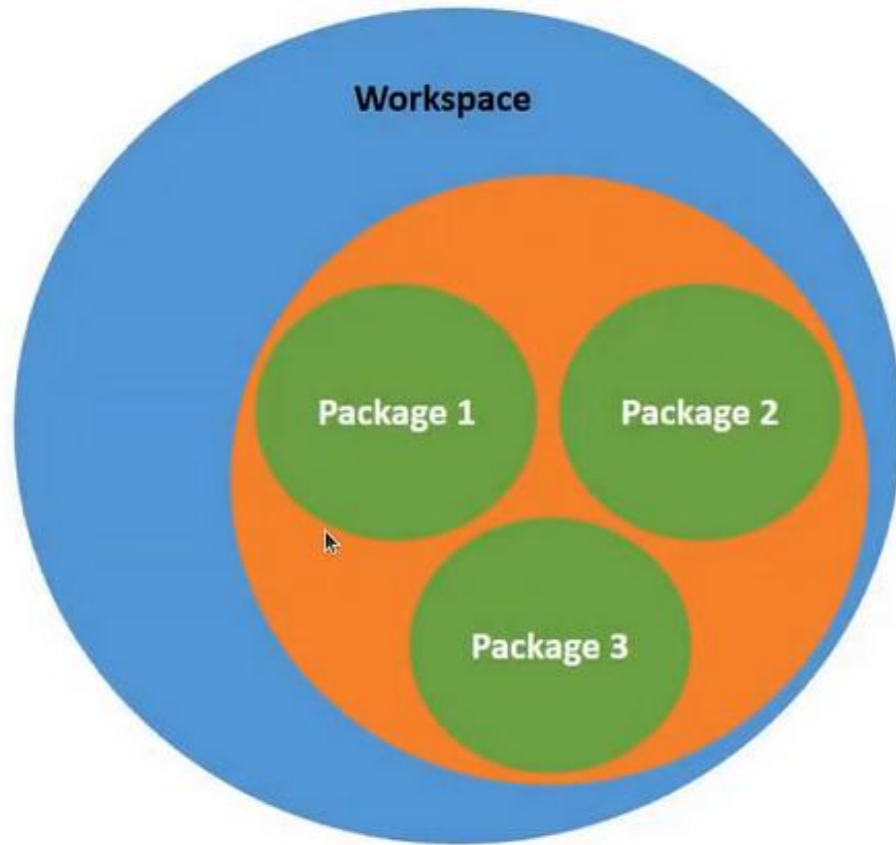


Topic & Message

- Characteristics of topic
 - Asynchronous, one-way message transmission
 - Communication between publisher and subscriber
 - 1:N, N:1, N:N communication
 - Continuous message transmission until communication is terminated



Workspace & Package



Ready to Use ROS2

1. Install 'colcon'

- **sudo apt install python3-colcon-common-extensions**
- For specific python virtual environment, **pip install colcon-common-extensions**

2. Source 'setup.bash'

- **source /opt/ros/foxy/setup.bash**
- Source 'setup.bash' in every terminal before using ROS2 commands!!
- For convenience, add **source /opt/ros/foxy/setup.bash** in .bashrc

Creating Workspace & Package

1. Create workspace

- **mkdir -p ~/example_ws/src**

2. Create package

- Move to /example_ws/src
- **ros2 pkg create --build-type ament_python <package_name>**
- **ros2 pkg create --build-type ament_python <package_name> --node-name <node_name> --dependencies <dependency>**

3. Resolve dependencies

- **rosdep update --rosdistro foxy**
- **rosdep install -i --from-path src --rosdistro foxy -y**

Creating Workspace & Package

4. Colcon build

- 'colcon build' every time after node is modified
- **colcon build --symlink-install**

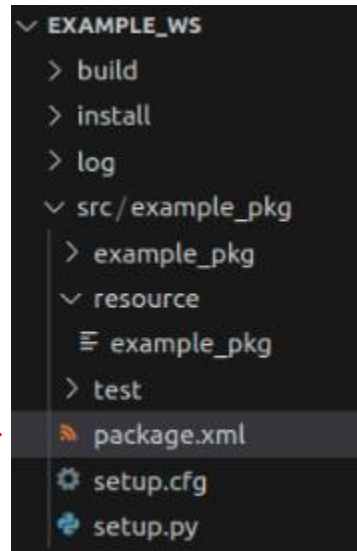
5. Source

- **source install/setup.bash**
- For convenience, add `~/<workspace_name>/install/setup.bash` in `.bashrc`

6. Run example_node

- **ros2 run <package_name> <executable_name>**

Package.xml



```
<?xml version="1.0"?>
<?xml-model href="http://download.ros.org/schema/package_format3.xsd" schematypens="http://www.w3.org/2001/XMLSchema"?>
<package format="3">
  <name>example_pkg</name>
  <version>0.0.0</version>
  <description>TODO: Package description</description>
  <maintainer email="kdjrvic7502@gmail.com">subin</maintainer>
  <license>TODO: License declaration</license>

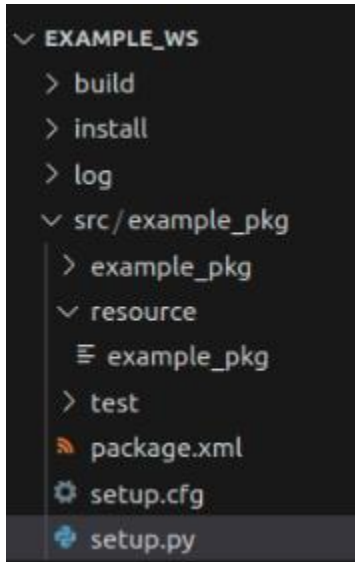
  <depend>rclpy</depend>
  <test_depend>ament_copyright</test_depend>
  <test_depend>ament_flake8</test_depend>
  <test_depend>ament_pep257</test_depend>
  <test_depend>python3-pytest</test_depend>

  <export>
    <build_type>ament_python</build_type>
  </export>
</package>
```

← `<depend>another_pkg</depend>`

`<exec_depend>another_pkg</exec_depend>`

Setup.py



```
from setuptools import setup

package_name = 'example_pkg'

setup(
    name=package_name,
    version='0.0.0',
    packages=[package_name],
    data_files=[
        ('share/ament_index/resource_index/packages',
         ['resource/' + package_name]),
        ('share/' + package_name, ['package.xml']),
    ],
    install_requires=['setuptools'],
    zip_safe=True,
    maintainer='subin',
    maintainer_email='kdjrvic7502@gmail.com',
    description='TODO: Package description',
    license='TODO: License declaration',
    tests_require=['pytest'],
    entry_points={
        'console_scripts': [
            'example_node = example_pkg.example_node:main'
        ],
    },
)
```

```
entry_points={
    'console_scripts': [
        'node_name = pkg_name.file_name:main'
    ],
}
```

Create Node - Publisher

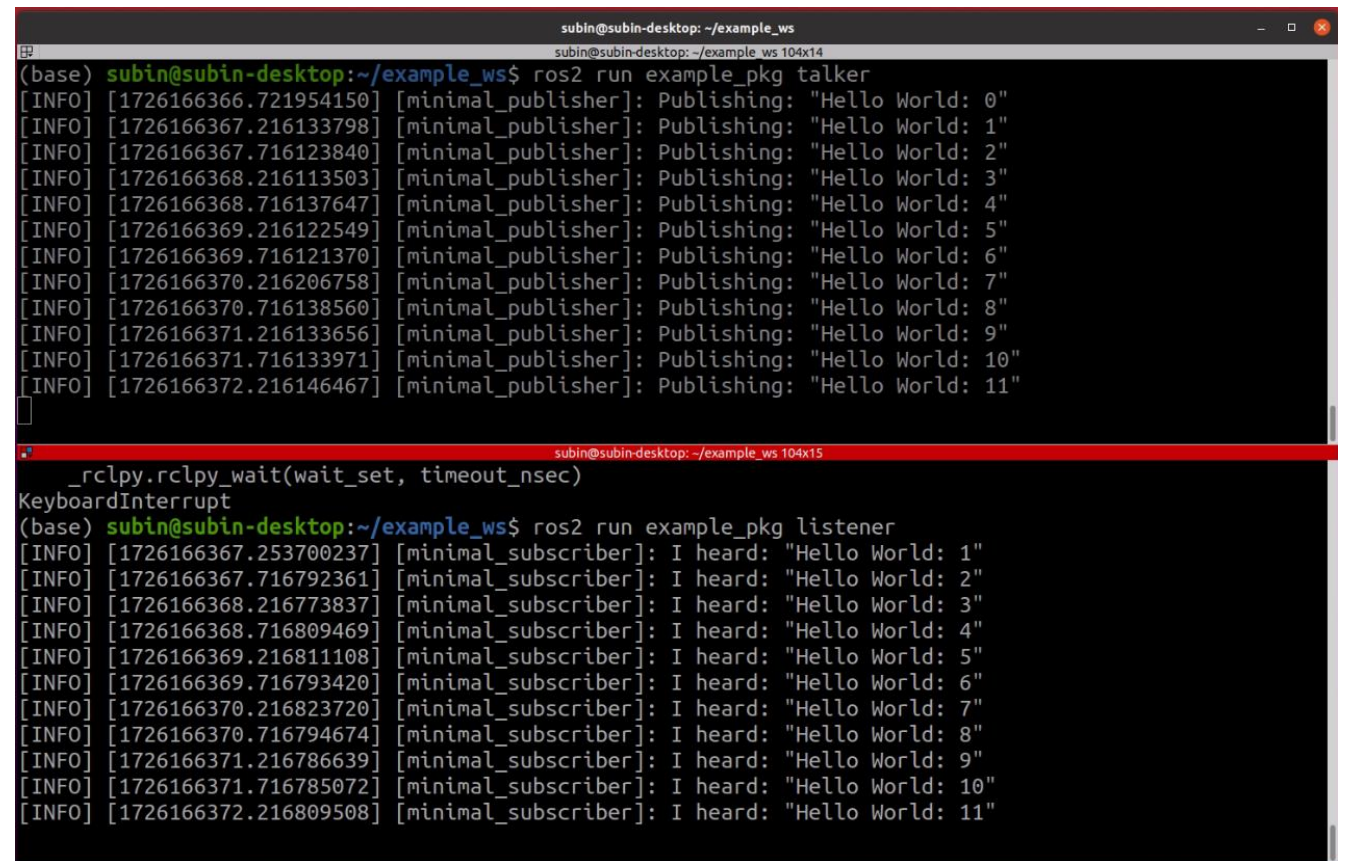
- Navigate into src/example_pkg/example_pkg
- Download example talker code
 - wget
https://raw.githubusercontent.com/ros2/examples/foxy/rclpy/topics/minimal_publisher/examples_rclpy_minimal_publisher/publisher_member_function.py
- Add depending package
 - `<depend>std_msgs</depend>`
- Add entry point
 - `'talker = example_pkg.publisher_member_function:main'`

Create Node - Subscriber

- Navigate into `src/example_pkg/example_pkg`
- Download example talker code
 - `wget`
`https://raw.githubusercontent.com/ros2/examples/foxy/rcldpy/topics/minimal\_subscriber/examples/rcldpy\_minimal\_subscriber/subscriber\_member\_function.py`
- Add entry point
 - `'listener = example_pkg.subscriber_member_function:main'`

Build & Run

- colcon build
- **source install/setup.bash**
- Run the talker node & listener node
 - **ros2 run example_pkg talker**
 - **ros2 run example_pkg listener**
- Package list, topic list, and node list
 - **ros2 pkg list**
 - **ros2 topic list**
 - **ros2 node list**



The image shows two terminal windows side-by-side. The top window is titled 'subin@subin-desktop: ~/example_ws' and shows the command 'ros2 run example_pkg talker' being executed. It displays a series of 12 log messages from the 'minimal_publisher' node, each publishing 'Hello World' with a counter from 0 to 11. The bottom window is titled 'subin@subin-desktop: ~/example_ws 104x15' and shows the command 'ros2 run example_pkg listener' being executed. It displays a series of 12 log messages from the 'minimal_subscriber' node, each receiving 'Hello World' with a counter from 1 to 11. A red status bar at the bottom of the terminal indicates the current command being executed: '_rclpy.rclpy_wait(wait_set, timeout_nsec)'.

```
(base) subin@subin-desktop:~/example_ws$ ros2 run example_pkg talker
[INFO] [1726166366.721954150] [minimal_publisher]: Publishing: "Hello World: 0"
[INFO] [1726166367.216133798] [minimal_publisher]: Publishing: "Hello World: 1"
[INFO] [1726166367.716123840] [minimal_publisher]: Publishing: "Hello World: 2"
[INFO] [1726166368.216113503] [minimal_publisher]: Publishing: "Hello World: 3"
[INFO] [1726166368.716137647] [minimal_publisher]: Publishing: "Hello World: 4"
[INFO] [1726166369.216122549] [minimal_publisher]: Publishing: "Hello World: 5"
[INFO] [1726166369.716121370] [minimal_publisher]: Publishing: "Hello World: 6"
[INFO] [1726166370.216206758] [minimal_publisher]: Publishing: "Hello World: 7"
[INFO] [1726166370.716138560] [minimal_publisher]: Publishing: "Hello World: 8"
[INFO] [1726166371.216133656] [minimal_publisher]: Publishing: "Hello World: 9"
[INFO] [1726166371.716133971] [minimal_publisher]: Publishing: "Hello World: 10"
[INFO] [1726166372.216146467] [minimal_publisher]: Publishing: "Hello World: 11"

subin@subin-desktop:~/example_ws 104x15
KeyboardInterrupt
(base) subin@subin-desktop:~/example_ws$ ros2 run example_pkg listener
[INFO] [1726166367.253700237] [minimal_subscriber]: I heard: "Hello World: 1"
[INFO] [1726166367.716792361] [minimal_subscriber]: I heard: "Hello World: 2"
[INFO] [1726166368.216773837] [minimal_subscriber]: I heard: "Hello World: 3"
[INFO] [1726166368.716809469] [minimal_subscriber]: I heard: "Hello World: 4"
[INFO] [1726166369.216811108] [minimal_subscriber]: I heard: "Hello World: 5"
[INFO] [1726166369.716793420] [minimal_subscriber]: I heard: "Hello World: 6"
[INFO] [1726166370.216823720] [minimal_subscriber]: I heard: "Hello World: 7"
[INFO] [1726166370.716794674] [minimal_subscriber]: I heard: "Hello World: 8"
[INFO] [1726166371.216786639] [minimal_subscriber]: I heard: "Hello World: 9"
[INFO] [1726166371.716785072] [minimal_subscriber]: I heard: "Hello World: 10"
[INFO] [1726166372.216809508] [minimal_subscriber]: I heard: "Hello World: 11"
```

Build & Run

- rqt_graph

