



Autonomous Air Vehicle Racing

Robot Operating System

July 9th, 2025



Introduction

Drones are complex devices made up of sensors, controllers, and actuators

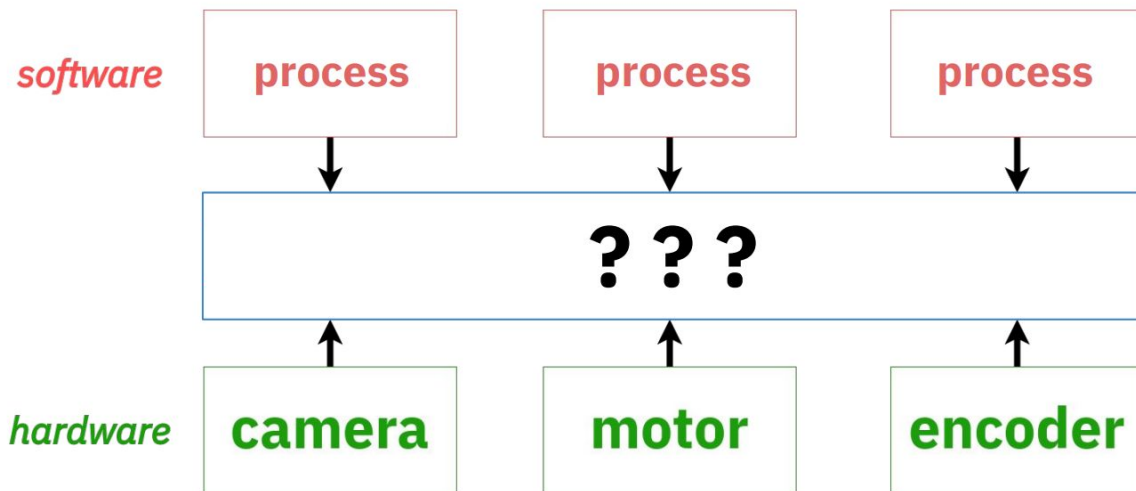
On the software level, how can we make these components work together?



Introduction

Drones are complex devices made up of sensors, controllers, and actuators

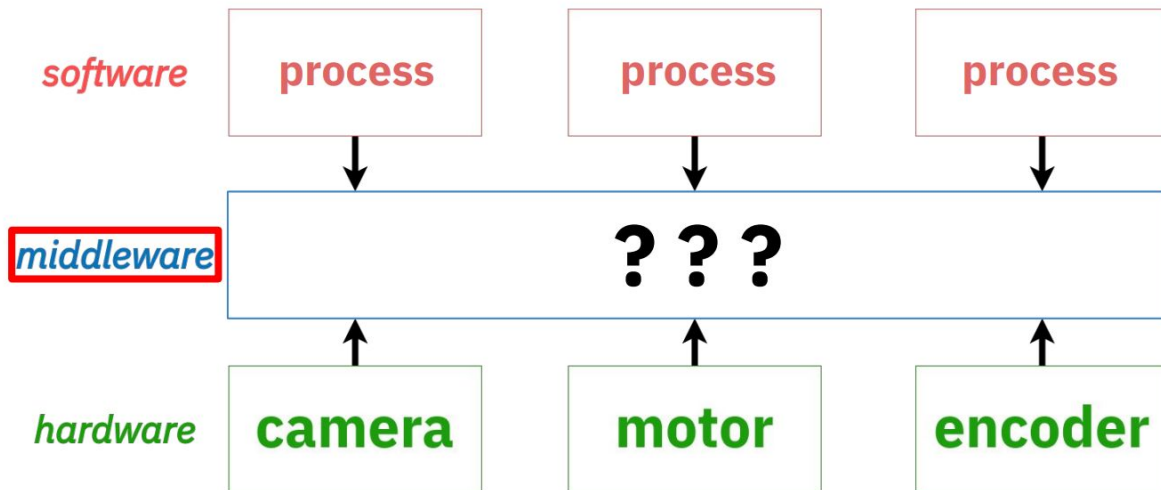
On the software level, how can we make these components work together?





The Solution: Middleware

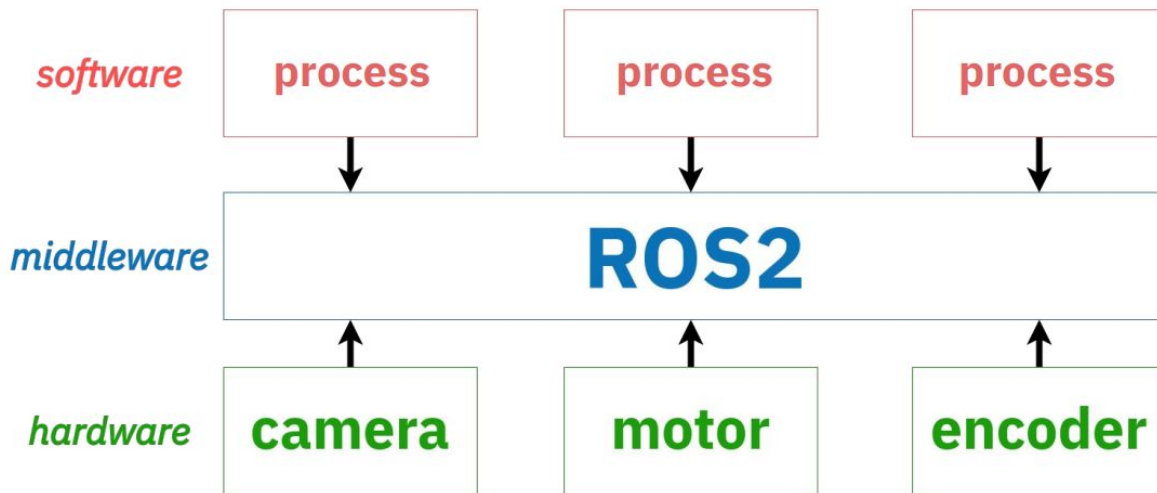
Middleware exists between the higher-level control/perception programs and the drone flight hardware





ROS2 Is Middleware

ROS2 (Robot Operating System 2) is a collection of software tools that provides the **industry standard** in robotic middleware

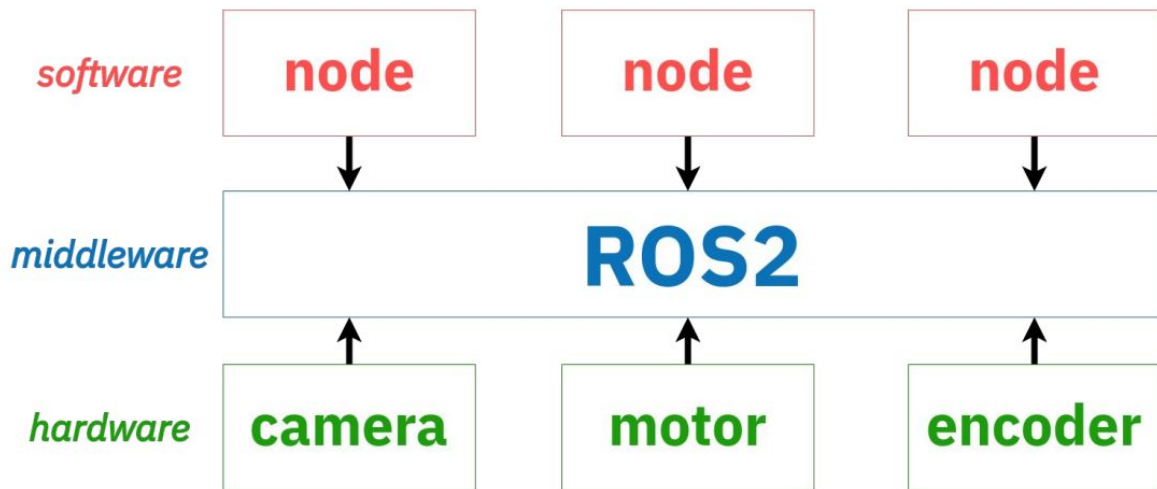




Aside

Typically, software processes in the ROS world are referred to as **ROS nodes**

For consistency, we'll use this terminology going forward





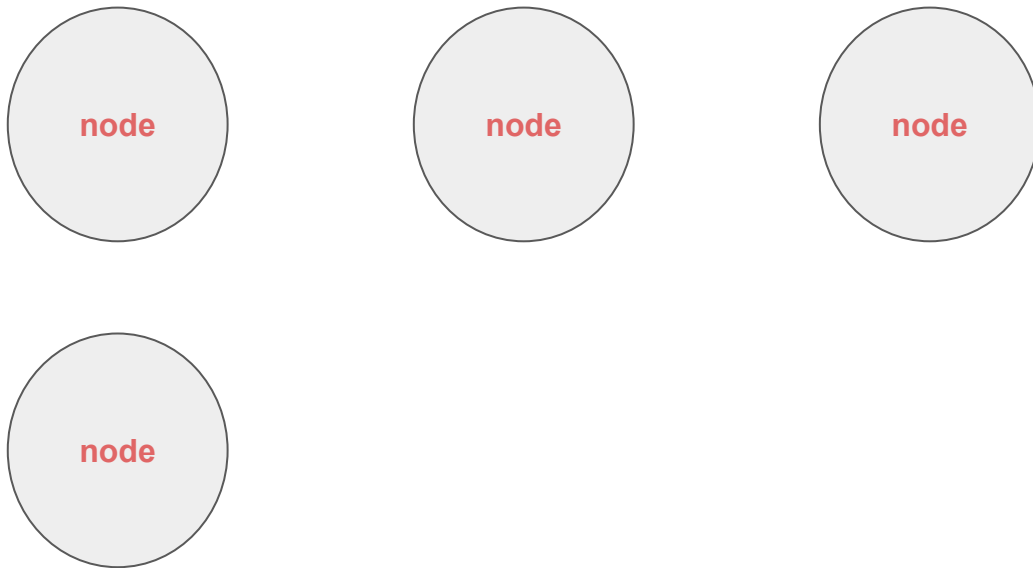
How does ROS actually work?



The ROS Graph

All ROS nodes exist in an abstract structure called the **ROS graph**

The ROS graph is a collection of ROS nodes running as concurrent software processes





ROS Nodes

Each ROS node serves some specific **purpose**

Node names should be **unique**, **descriptive**, and **concise**



/telem

/camera

/object_detection

/flight_controller

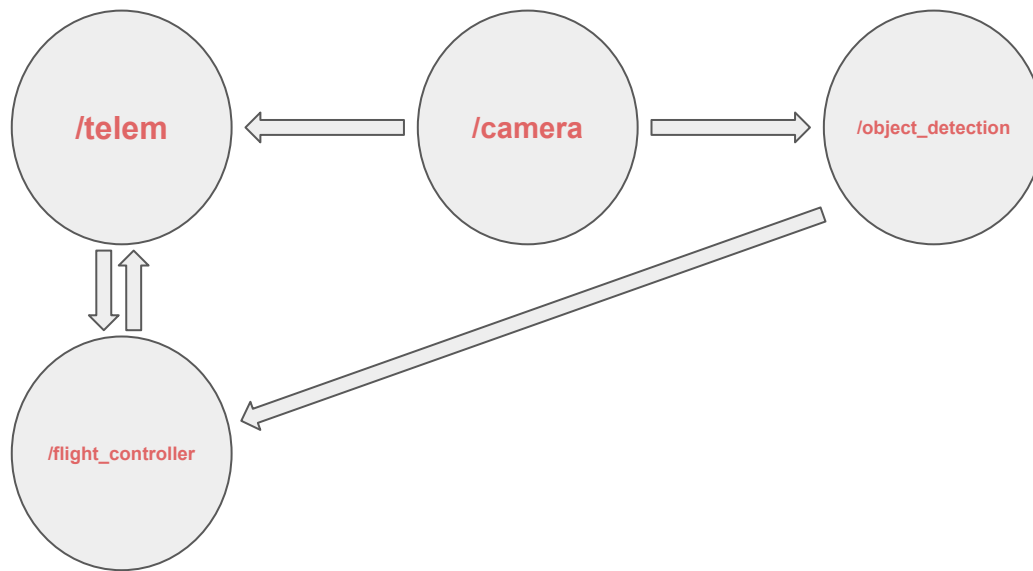


ROS Topics

Nodes are connected by **ROS topics**

When nodes **publish** to topics, they **transmit** information to the topic

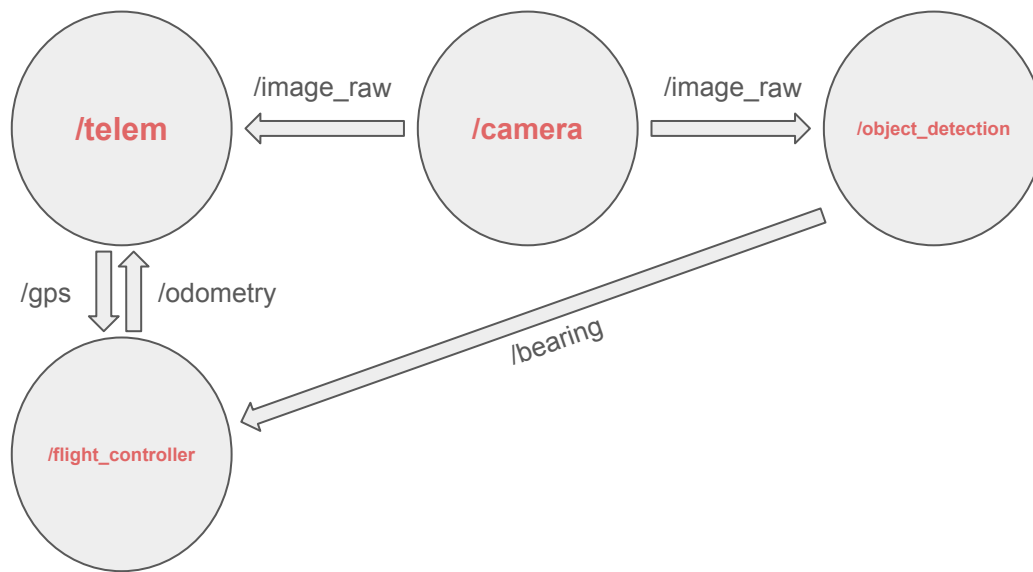
When nodes **subscribe** to topics, they are **notified** of new information





ROS Topics cont.

Similar to nodes, topics should be named appropriately





ROS Messages

ROS nodes pass **messages** through topics

This is an example ROS message containing two **fields**

```
DriveCmd {  
    float32 l_speed,  
    float32 r_speed  
}
```

Each field consists of
a **type** and a **name**



Lab Activity: FizzBuzz