
PiDrone Code Architecture

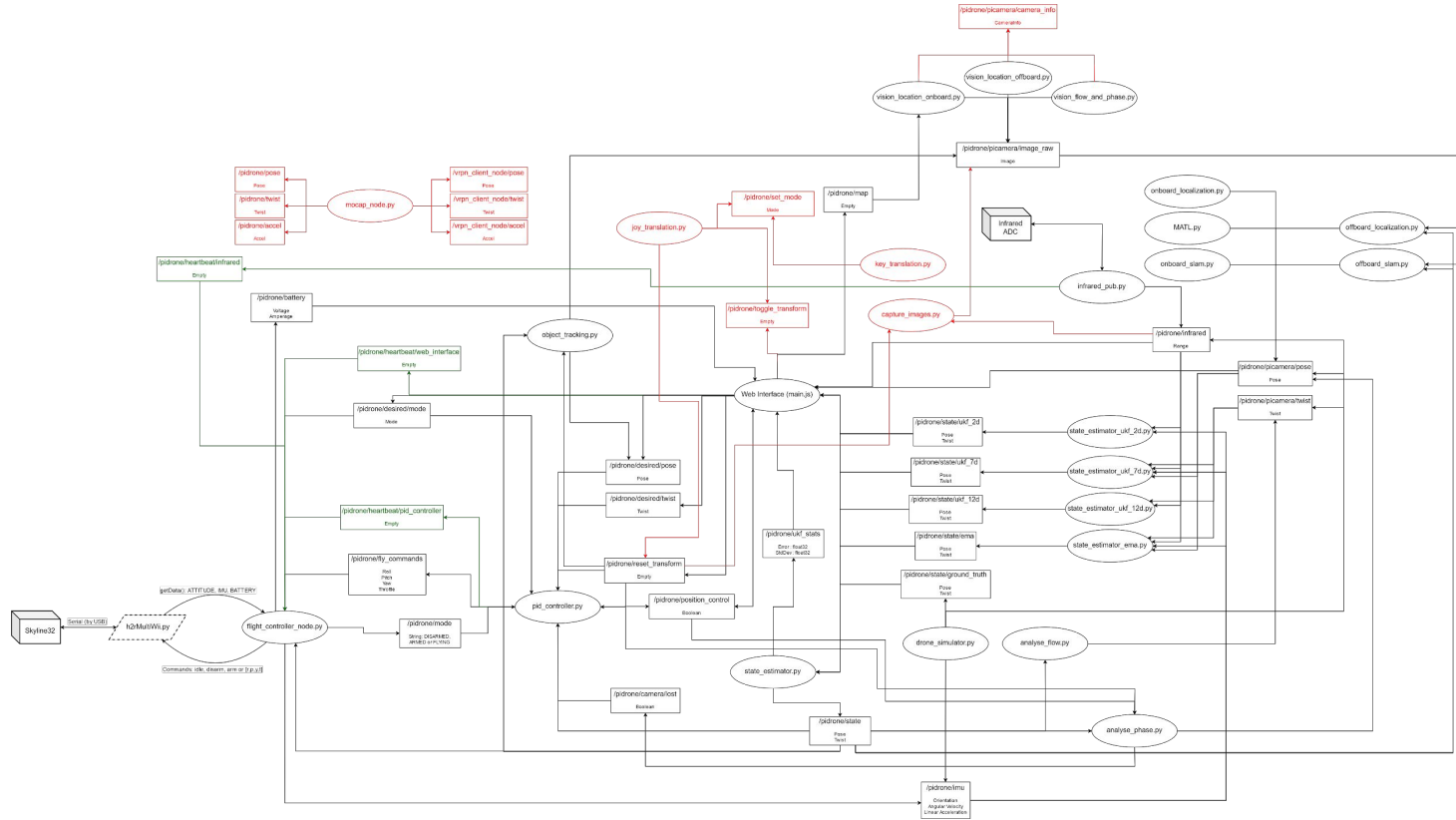
Source

- Source codes of PiDrone on Github: https://github.com/h2r/pidrone_pkg
- Architecture analysis of commit 57126cb, of August 16, 2019 (newest stable version on this presentation build)

Notation

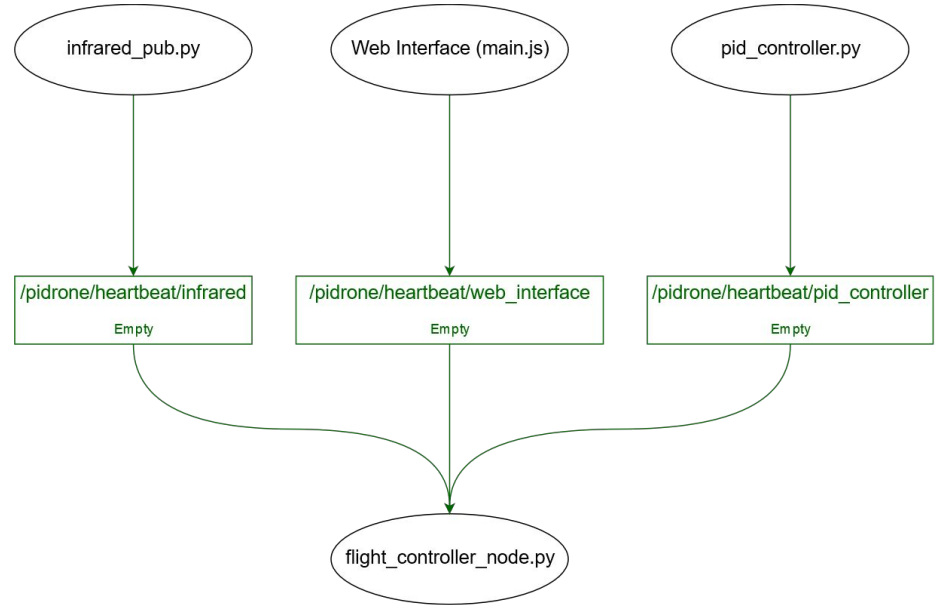
- Once PiDrone is a ROS project, the notation has as base the rqt_graph notation
 - Ellipses for ROS nodes
 - Squares for ROS topics
- Each topic has also the type of message that carries
- Non ROS codes are represented by a parallelogram
- Physical devices are represented by a cube
- Red nodes and topics are residual code
- Green topics are related to heartbeats (a security measure)

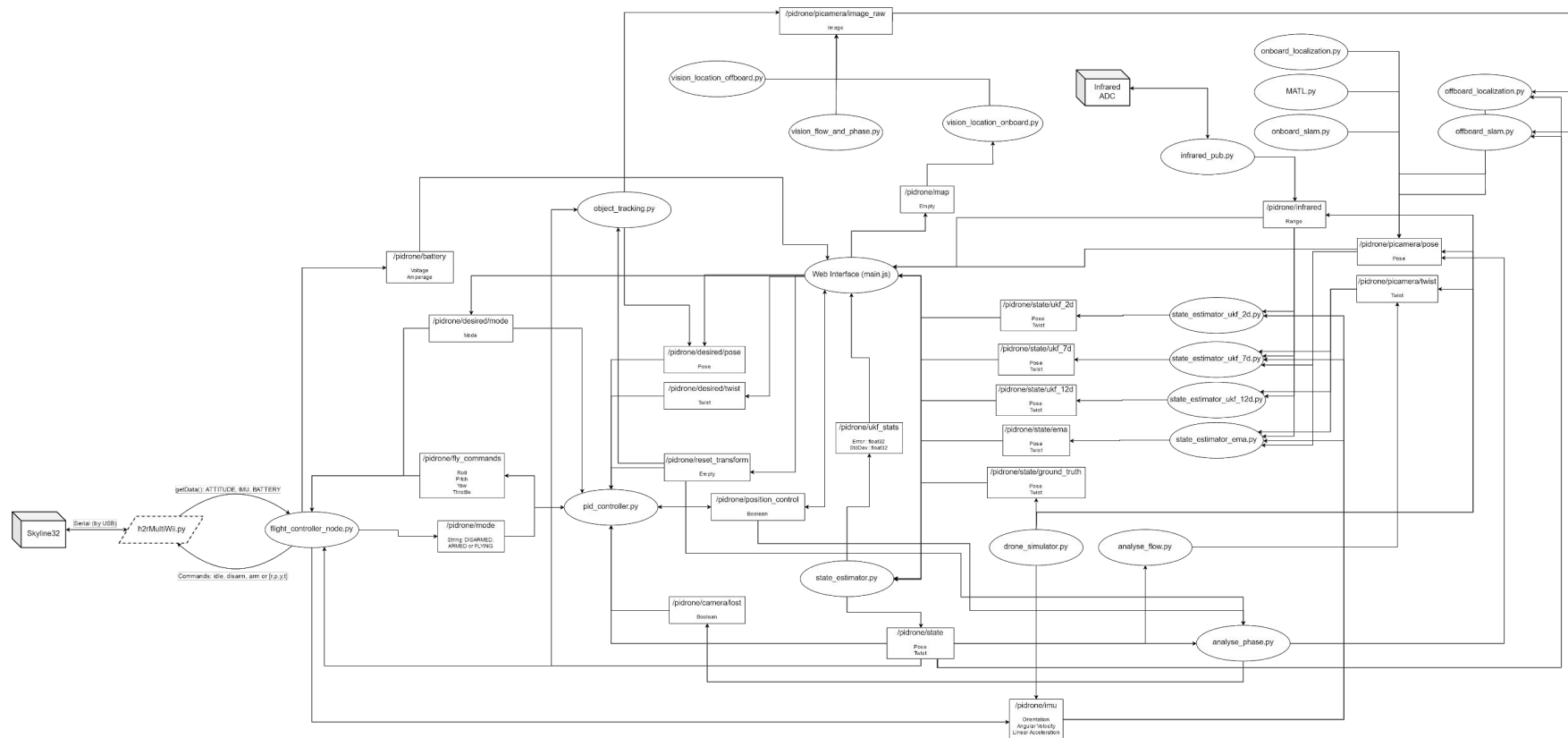
The Complete Architecture



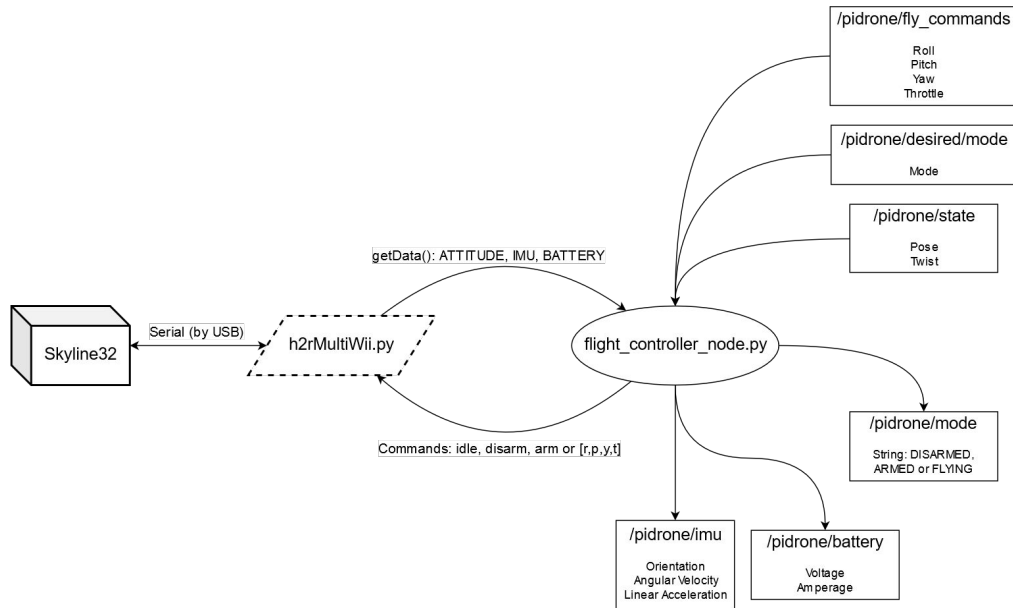
Heartbeats

- 60 times per second, the Infrared node, Web Interface and PID controller talk that are active.
- If one of the heartbeats is missing for more than 1 second, the Flight Controller disarm the drone.



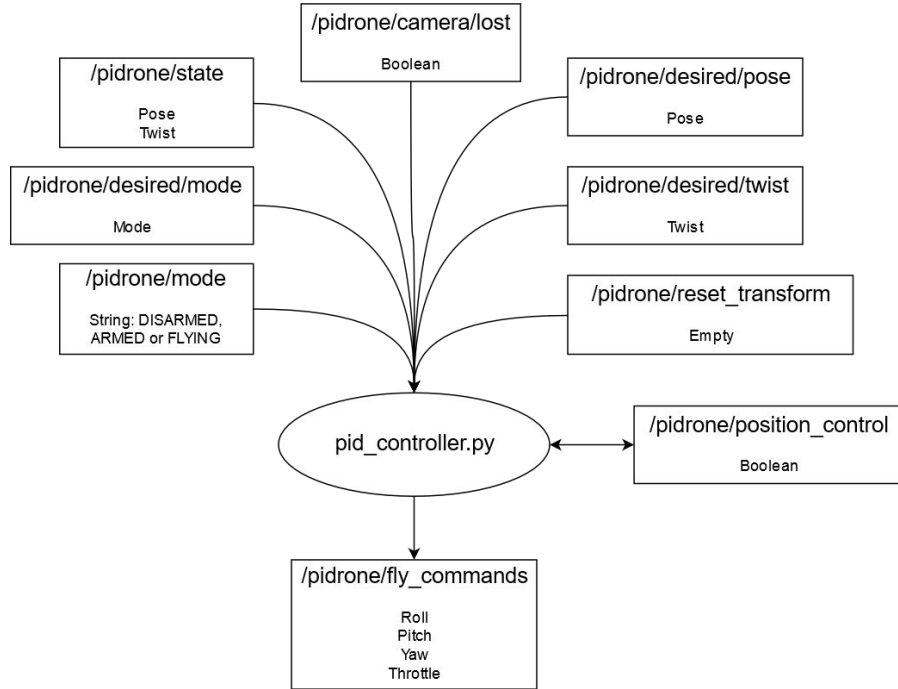


Flight Controller



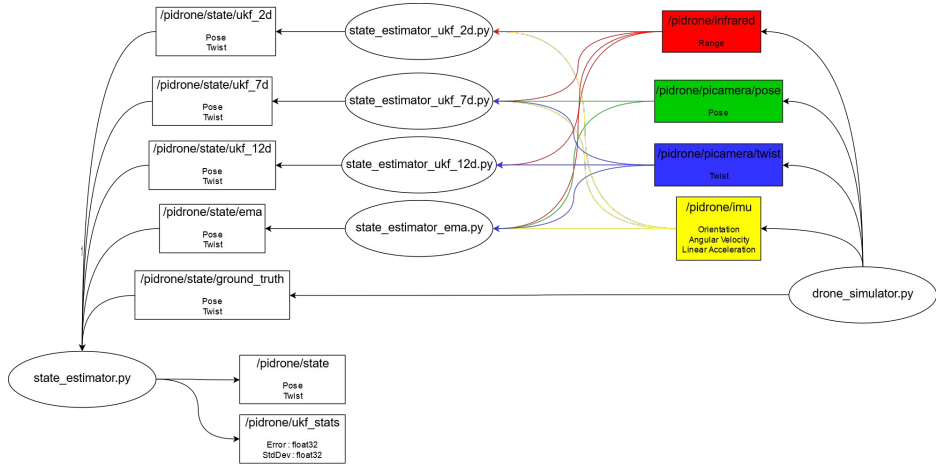
- Communication with Skyline32
- Sends the commands of Roll, Pitch, Yaw and Throttle
- Gets information about IMU and Battery
- If battery is low, disarm the drone

PID Controller



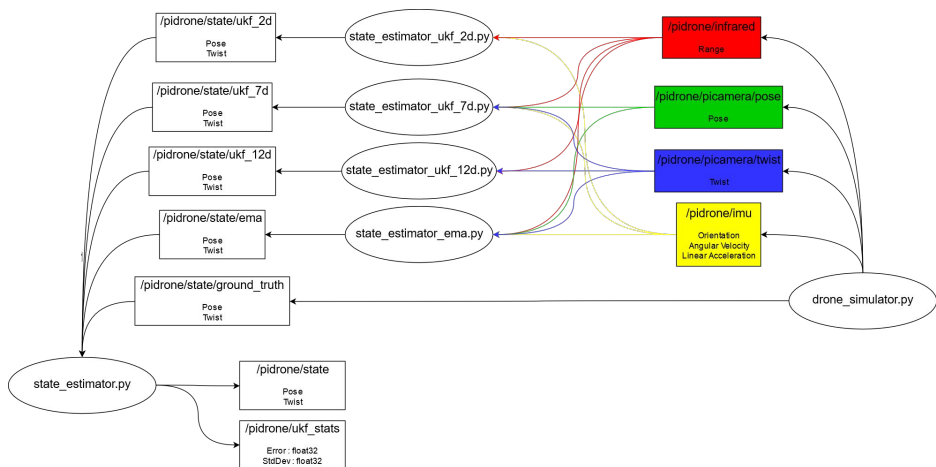
- Receives the current and desired states of the drone and calculates the Fly Commands (Roll, Pitch, Yaw and Throttle)
- It can cancel the position control if the camera is lost or if some other security problem is found
- Reset: make the current X and Y of the drone as the new origin and commands the drone to stop

State Estimator



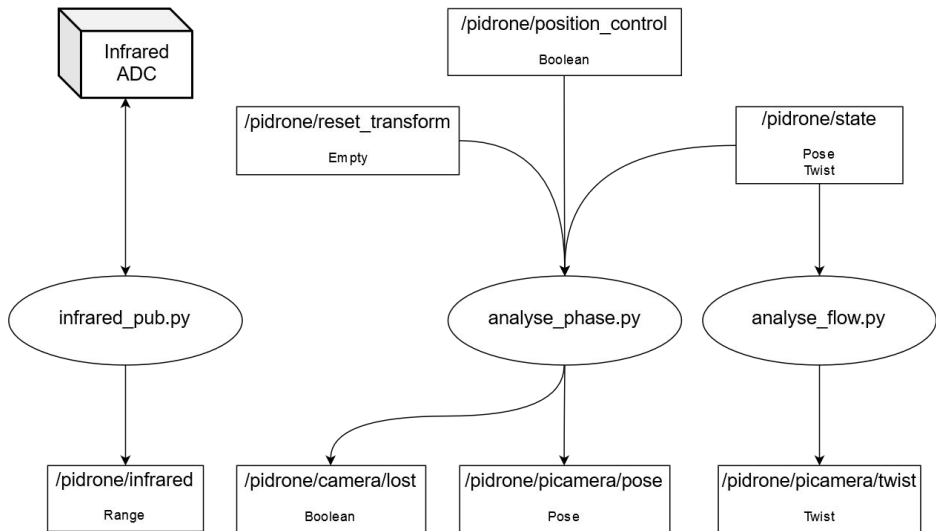
- A node that centralizes 4 state estimators.
- The choice is made by command line arguments, and other estimators can be activated for visualization or debug purposes
- ukf_stats is analytics info (error and standard deviation) sent to Web Interface if UKF and ground_truth are active

State Estimator



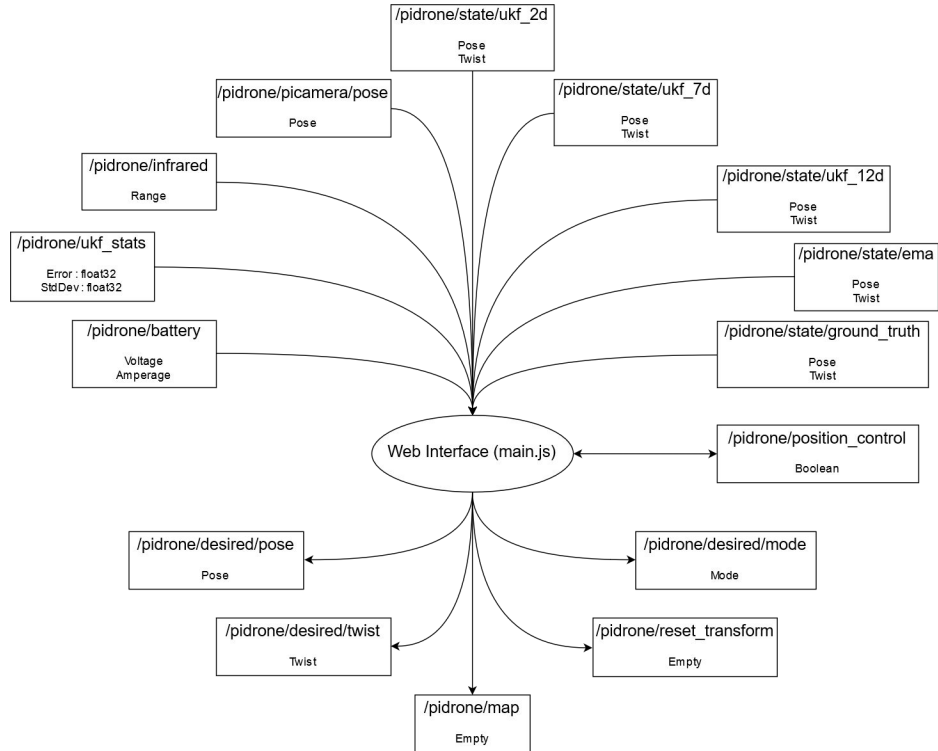
- EMA (Exponential Moving Average)
- UKF (Unscented Kalman Filter)
 - UKF 2D
 - UKF 7D
 - UKF 12D
- Simulation
 - provides ground_truth and dummy values of sensors and drone state

Camera and InfraRed



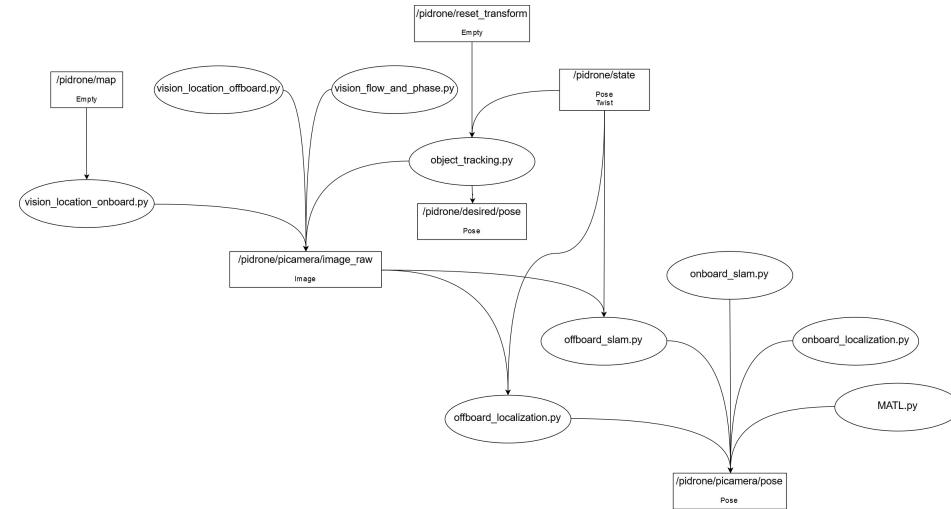
- infrared_pub makes the communication with the IR ADC, calculates the range and publish
- analyze_flow estimates the velocities by motion analyses of optical flow
- analyze_phase estimates the position with estimateRigidTransform

Web Interface



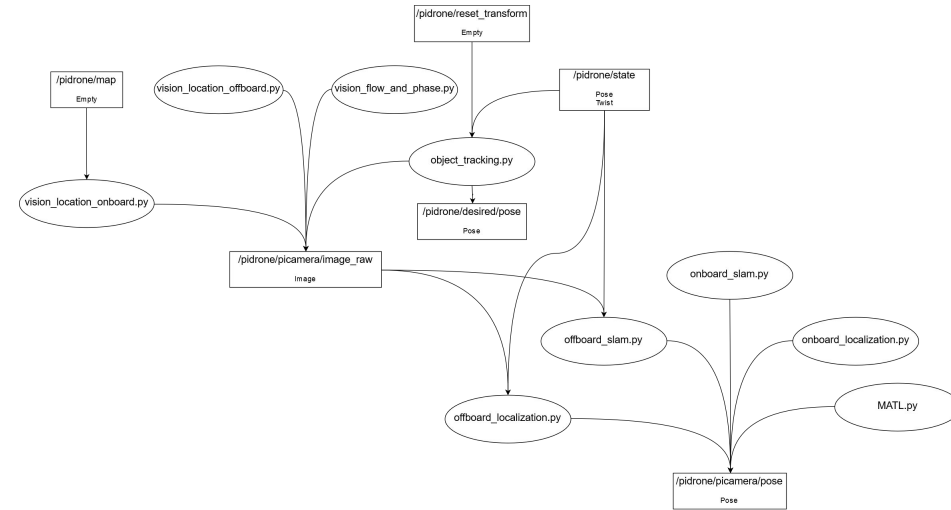
- Shows drone information to the user
- The user can control the mode (Disarmed, Armed and Flying)
- If in position control, the user can control the position of the drone
- Else, the user can control the linear velocities and yaw angular velocity

Mapping



- Use of SLAM (Simultaneous Localization and Mapping) or the customized algorithm MATL (Mapping and Localization)
- object_tracking have direct control of camera and send desired pose to follow some object

Mapping



- Support to use the algorithms embedded or not
- Some of them are not documented or seems unfinished, so maybe are yet on development or are residual code

Commands: idle, disarm, arm or /etc

