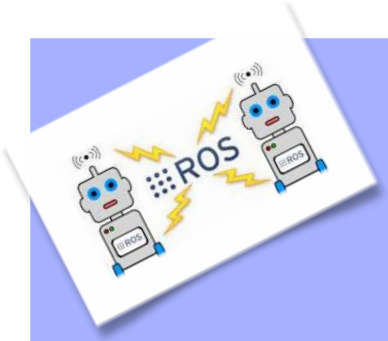


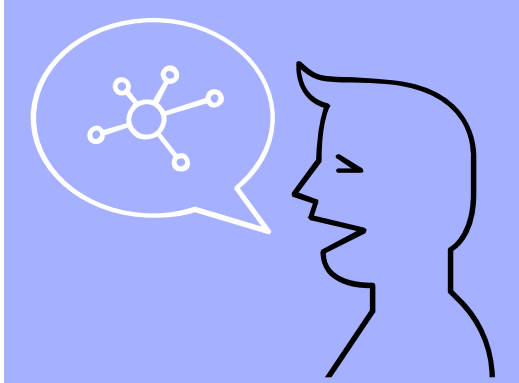
DOING COOL STUFF IN ROBOTICS USING ROS WITH PYTHON

Robot Operating System

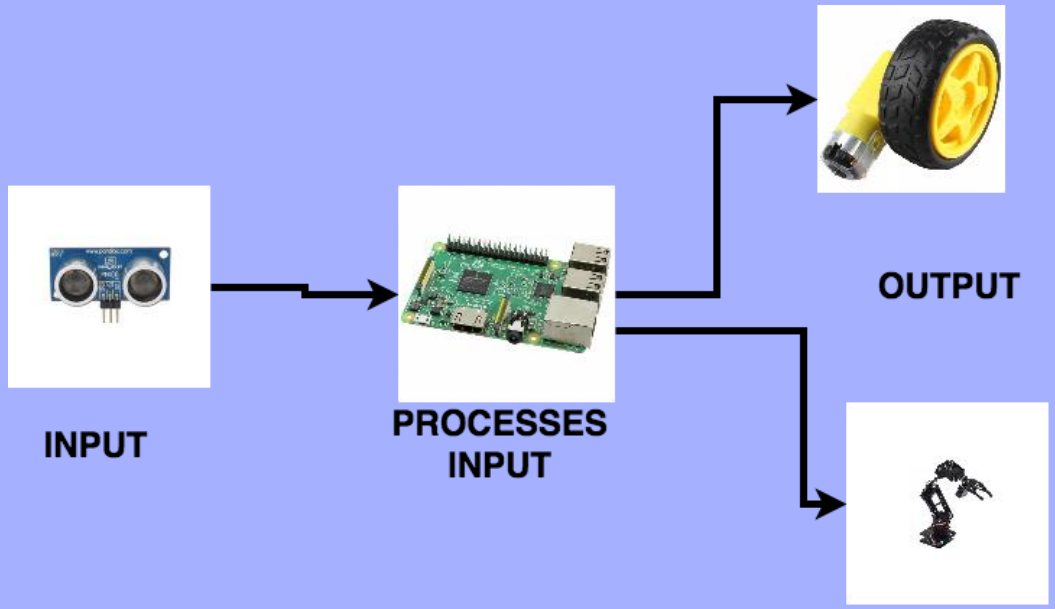
pyconNG | Kayode Emmanuel Oluwatobi

WHO BE THIS GUY

- Systems Engineering, University of Lagos.
- Software Engineer, Teamapt Ltd, creating financial happiness.
- Roboticist when not doing office work.

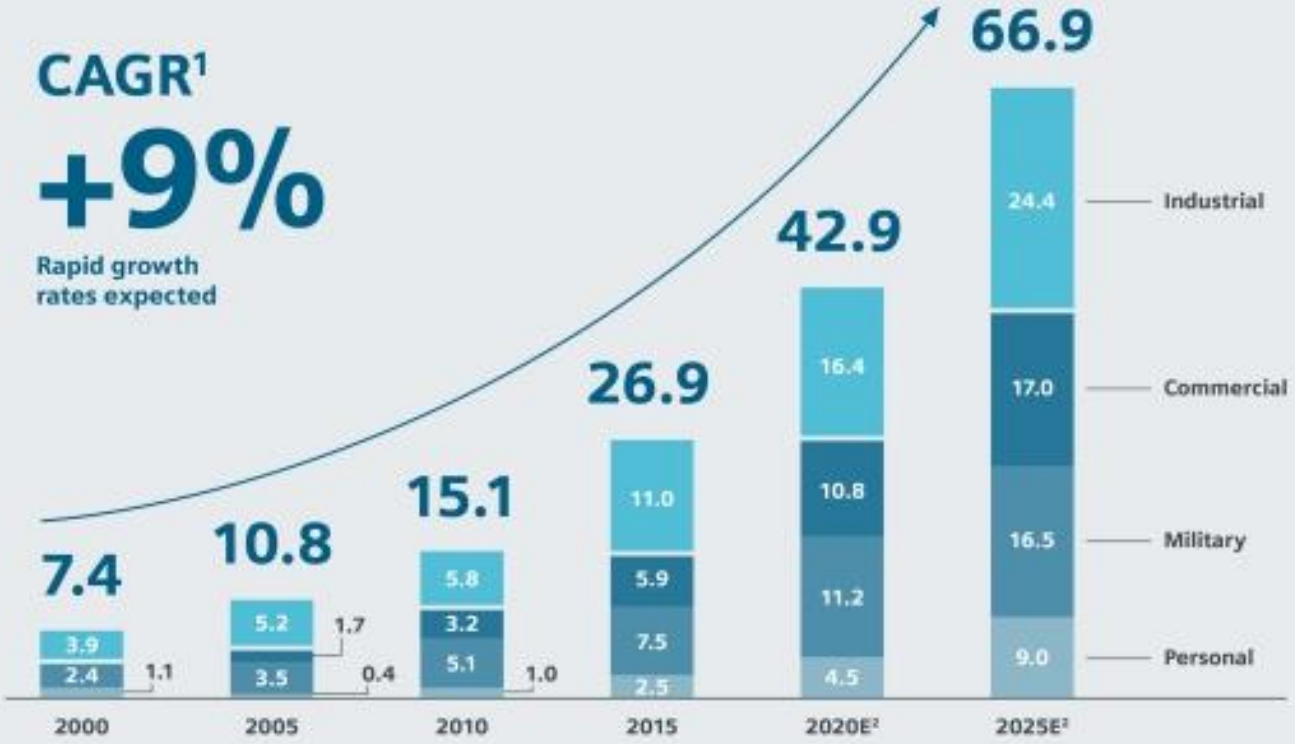


ROBOTICS



Worldwide Spending on Robotics is Expected to Reach US$ 67 Billion by 2025

Global robotics market (US$ Billions)



1 Compound Annual Growth Rate
2 E = Expected
Source: International Federation of Robotics, Japan Robot Association; Japan Ministry of Economy, Trade & Industry; euRobotics; company filings; BCG analysis.

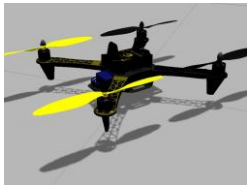
HARDWARE









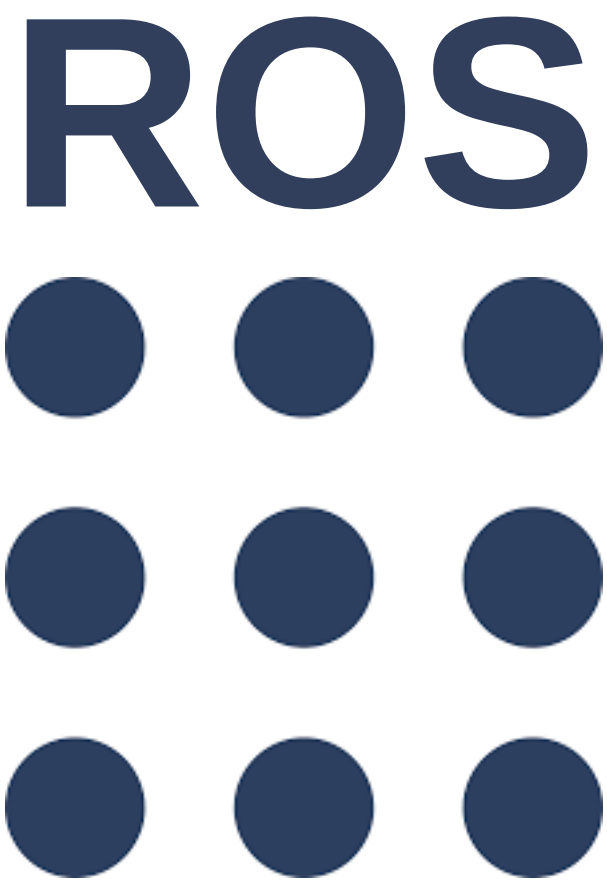


LOTS MORE

HOW DO I DO COOL STUFF WITH ALL THIS HARDWARE PART

- PERCEPTION
- NAVIGATION
- MAPPING
- Etc

- Distributed computation
- Software reuse
- Rapid testing
- Open source



ROS to the rescue

The Robot Operating System (ROS) is a flexible framework for writing robot software.

It is a collection of tools, libraries, and conventions that aim to simplify the task of creating complex and robust robot behavior across a wide variety of robotic platforms.

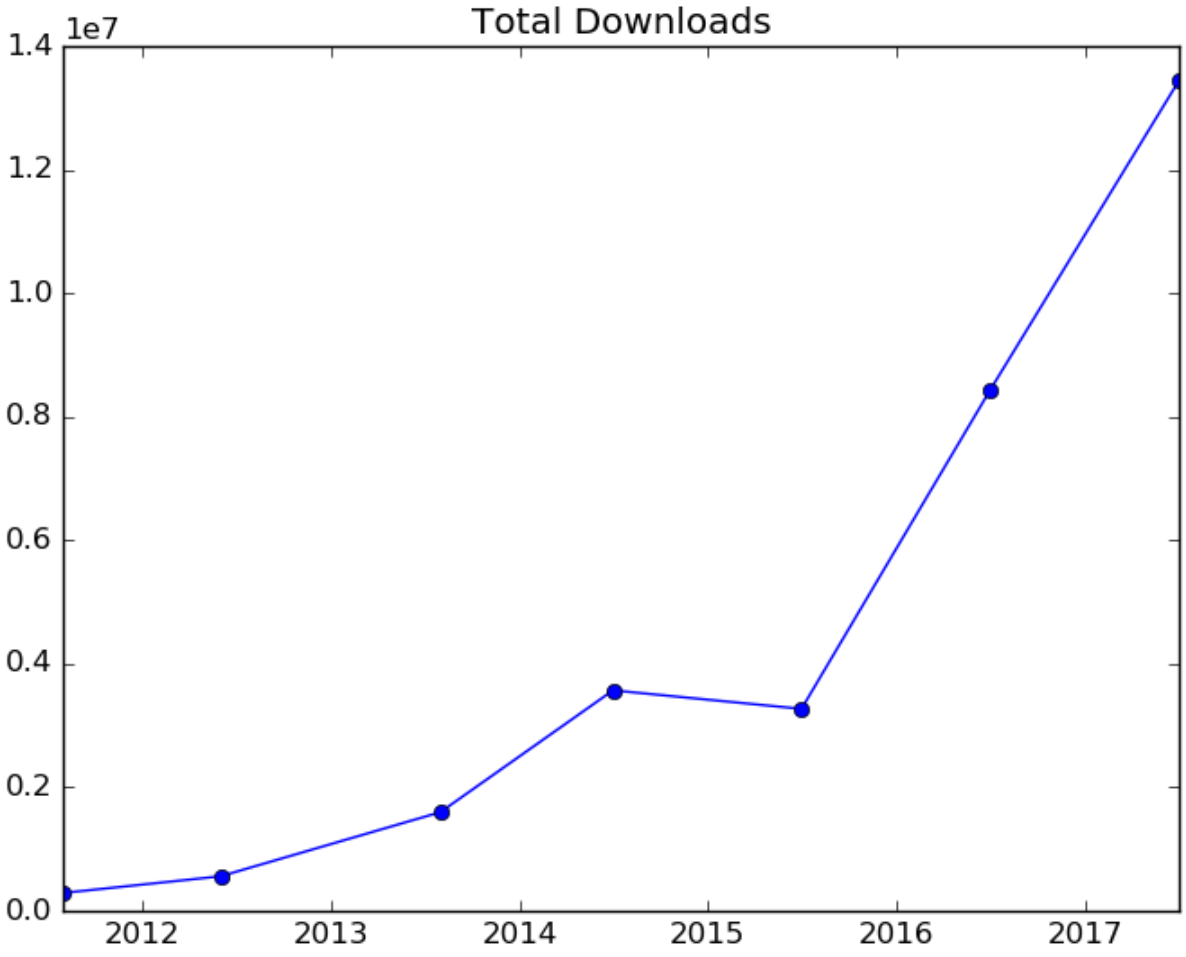
ROS

Why ROS

Because creating truly robust, general-purpose robot software is *hard*

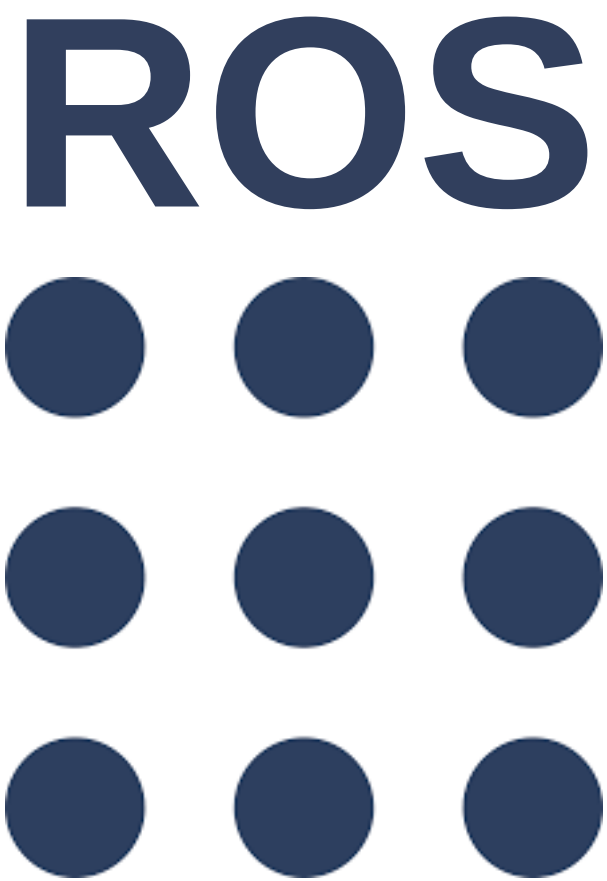
From the robot's perspective, problems that seem trivial to humans often vary wildly between instances of tasks and environments.

Dealing with these variations is so hard that no single individual, laboratory, or institution can hope to do it on their own.


Total Downloads
1e7
1.4
1.2
1.0
0.8
0.6
0.4
0.2
0.0
2012
2013
2014
2015
2016
2017

ROS

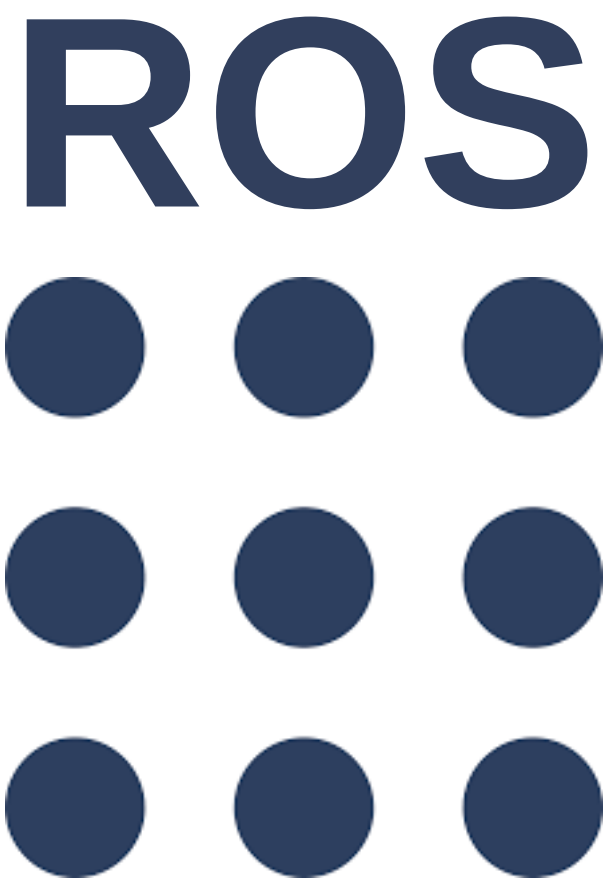
Robots using ROS



ROS concept

ROS Filesystem Level: Packages, Metapackages, Package Manifests, Repositories, Message (msg) types, Service (srv) types.

ROS Community Level: Distributions, Repositories, The ROS Wiki, Mailing Lists, ROS Answers, Blog.



ROS Computation Graph Level

Nodes,

Master,

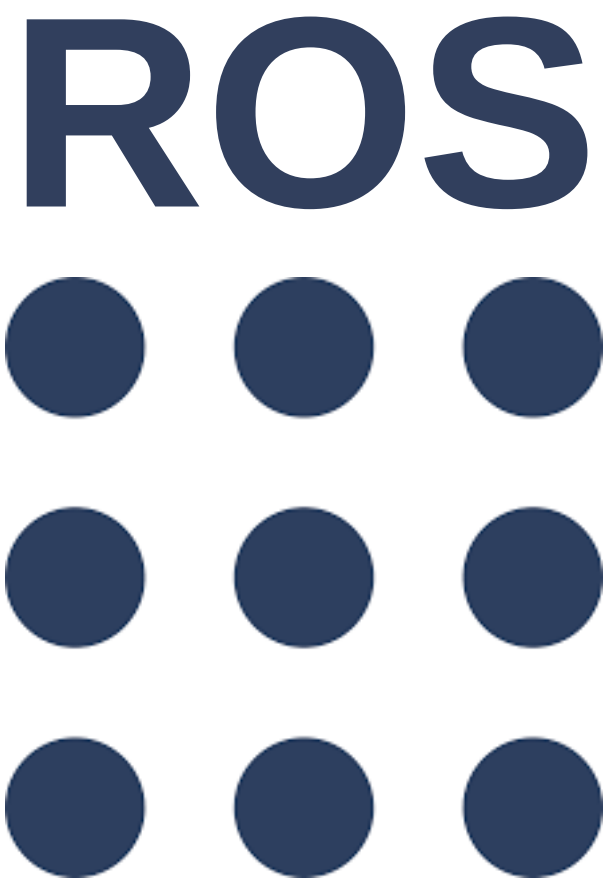
Parameter Server,

Messages,

Topics,

Services,

Bags.



ROS Computation Graph Level

Publisher code

```
import rospy
from std_msgs.msg import String
pub = rospy.Publisher('chatter', String, queue_size=10)
rospy.init_node('talker', anonymous=True)
rate = rospy.Rate(10) # 10hz
while not rospy.is_shutdown():
   hello_str = "hello world %s" % rospy.get_time()
   rospy.loginfo(hello_str)
   pub.publish(hello_str)
   rate.sleep()
```

Subscriber code

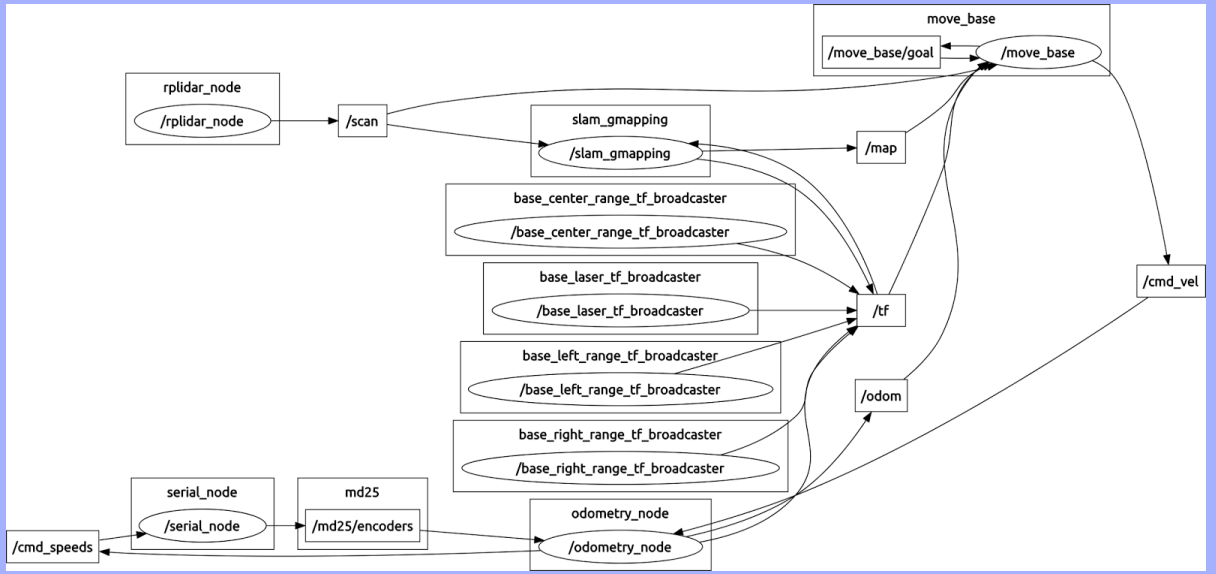
```
import rospy
from std_msgs.msg import String
def callback(data):
    rospy.loginfo(rospy.get_caller_id() + "I heard %s",  data.data)

rospy.init_node('listener', anonymous=True)
rospy.Subscriber("chatter", String, callback)
rospy.spin()
```

ROS

ROS Computation Graph Level

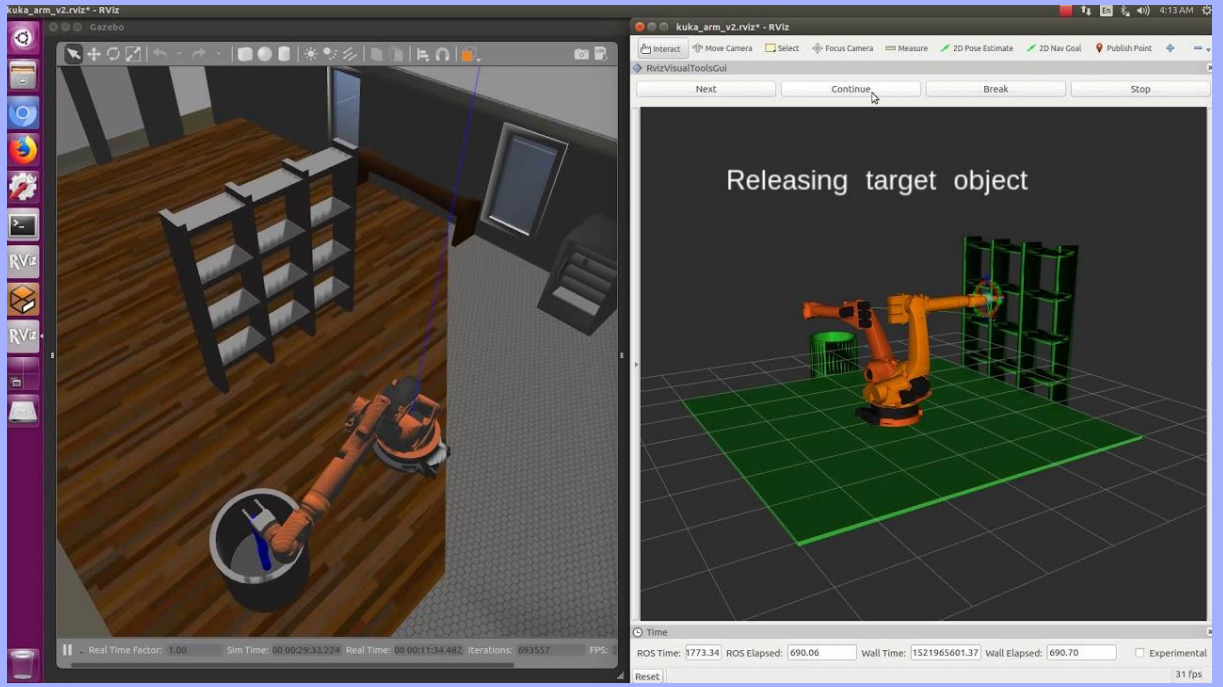
ROS

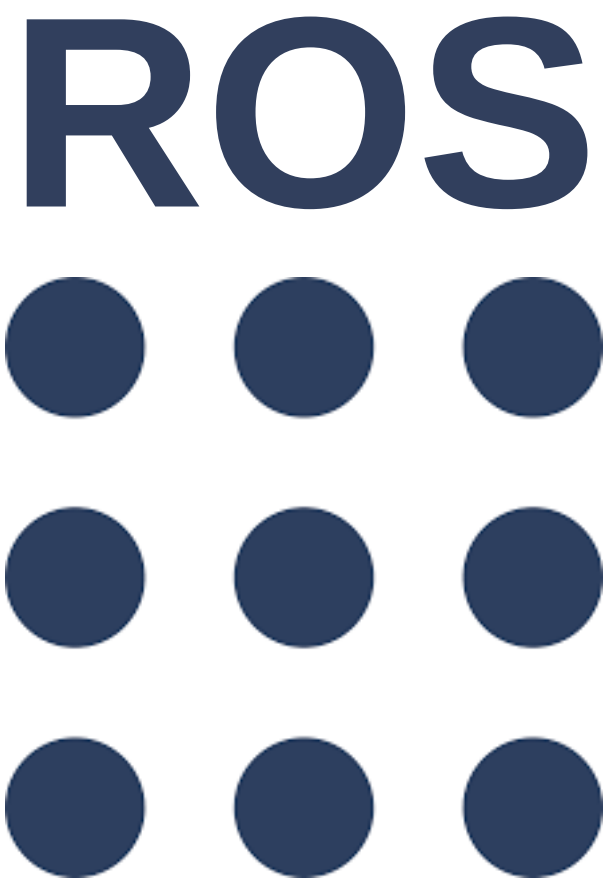
ROS Computation Graph Level



ROS

GAZEBO Simulator





HOW DO I START USING ROS

- Virtual Box
- Ubuntu (Bare Metal)
- ROSDS ros development studio

ROS

Stop Re-inventing the wheel, leverage the power in ROS

Our world depend on that your robotics Idea/project

Ameseginalehu
Thank You