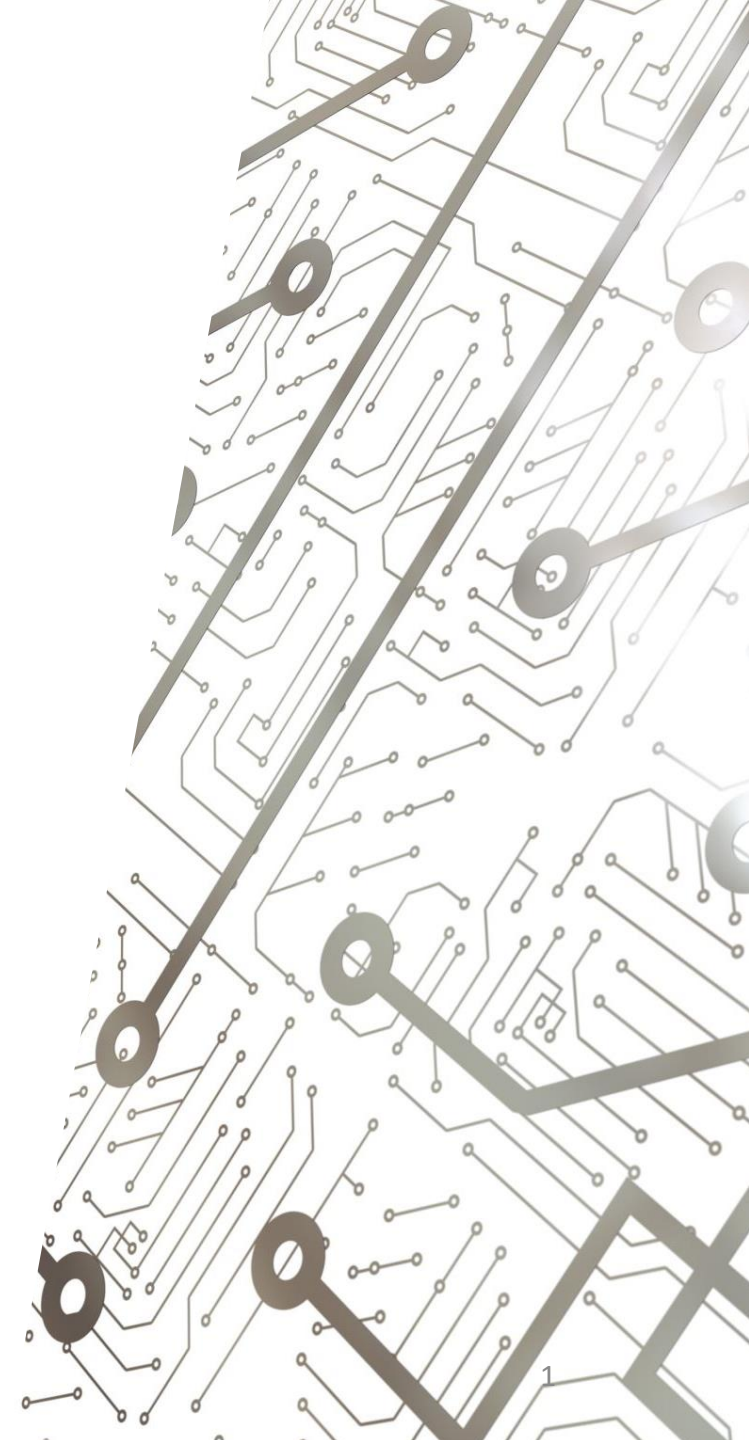




Software Lecture 1:

Introduction to ROS (And Linux!)

Jacques Cloete



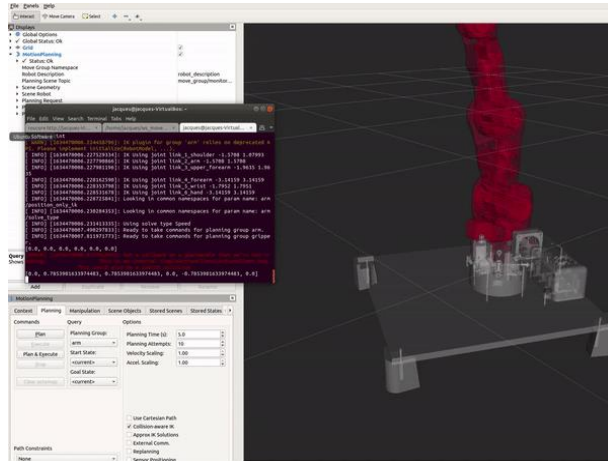
What is ROS?

- "Robot Operating System"

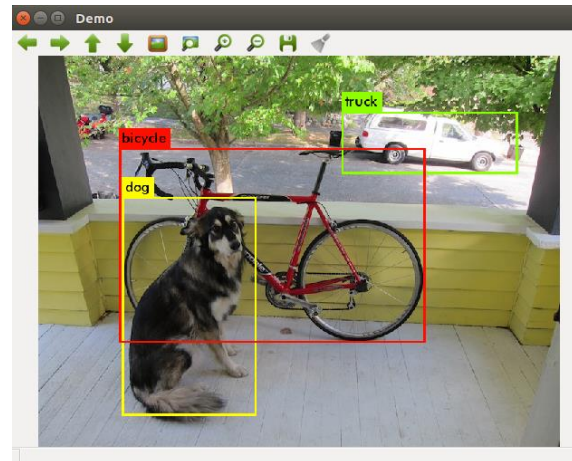
(Not really an operating system!)

- Framework that allows us to easily manage complex robotics-based systems
- Provides a wide array of useful software libraries and tools
- Entirely open source! (Good thing for us!)

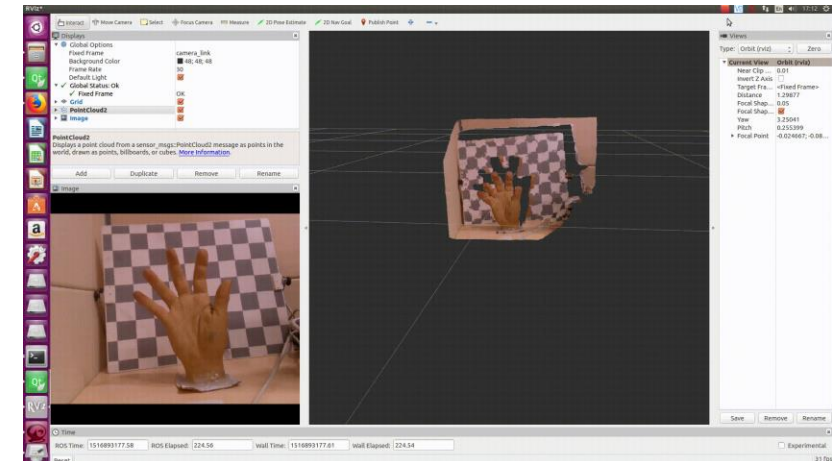
Example ROS applications?



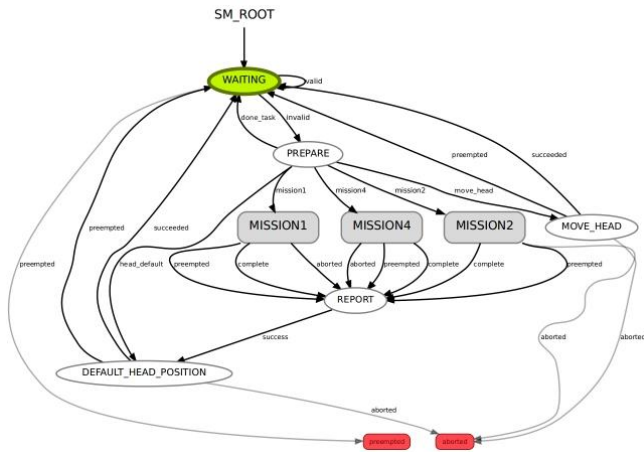
Motion-Planning



Object Detection



Analysing Point Cloud Data



State Machines



Communication with Hardware

...and so many more!

- ROS not only provides open-source packages for all these applications...
- ... but also lets them all simultaneously communicate with each other in a simple, streamlined manner!

Nodes, Publishers, Subscribers, Topics

- Each program is run as a **node** in the ROS network
- Nodes communicate by sending **messages** to each other
- Messages are communicated across **topics** (basically act like communication lines, or data buses)
- **Publisher** nodes **send out** messages onto a topic
- **Subscriber** nodes **listen to** messages on a topic

*ABSOLUTELY **CRITICAL** to learn and be familiar with this terminology! But don't worry, here's an analogy...*

Analogy – Sushi Conveyor Belt

Credit to Sneha!

- Sushi chefs put new plates of sushi on the conveyor belt
- Customers at the conveyor belt then take sushi from these plates



Analogy – Sushi Conveyor Belt

Credit to Sneha!

- Chefs and customers are **nodes**
- Chefs act as **publishers**
- Customers act as **subscribers**
- Conveyor belt is the **topic**
- Sushi itself is the **message**!



Some Further Notes

- A node can be both a publisher and/or a subscriber to multiple topics! You can also easily set up new topics!
- When a publisher publishes a message to a topic, **ALL** subscribers to that topic receive the message

ALL customers at the same conveyor belt get a piece of sushi from the new plate!

- Programs can also interact with each other using (very useful) ROS actions and services, but we will cover these later down the line...

Okay, Enough Theory (For Now...)

- To use ROS, you first need to have Linux on your machine
- If you have a Windows or Mac OS laptop, I suggest creating an Ubuntu virtual machine...



VirtualBox

Installing Virtualbox

<https://www.virtualbox.org/>

- Download the installer for your OS and follow the instructions to install VirtualBox



VirtualBox

Installing Ubuntu as a Virtual Machine

<https://releases.ubuntu.com/20.04.4/>

1. Download the 64-bit PC (AMD64) Ubuntu **20.04** desktop image from the above link

Important to specifically download the image for version 20.04!



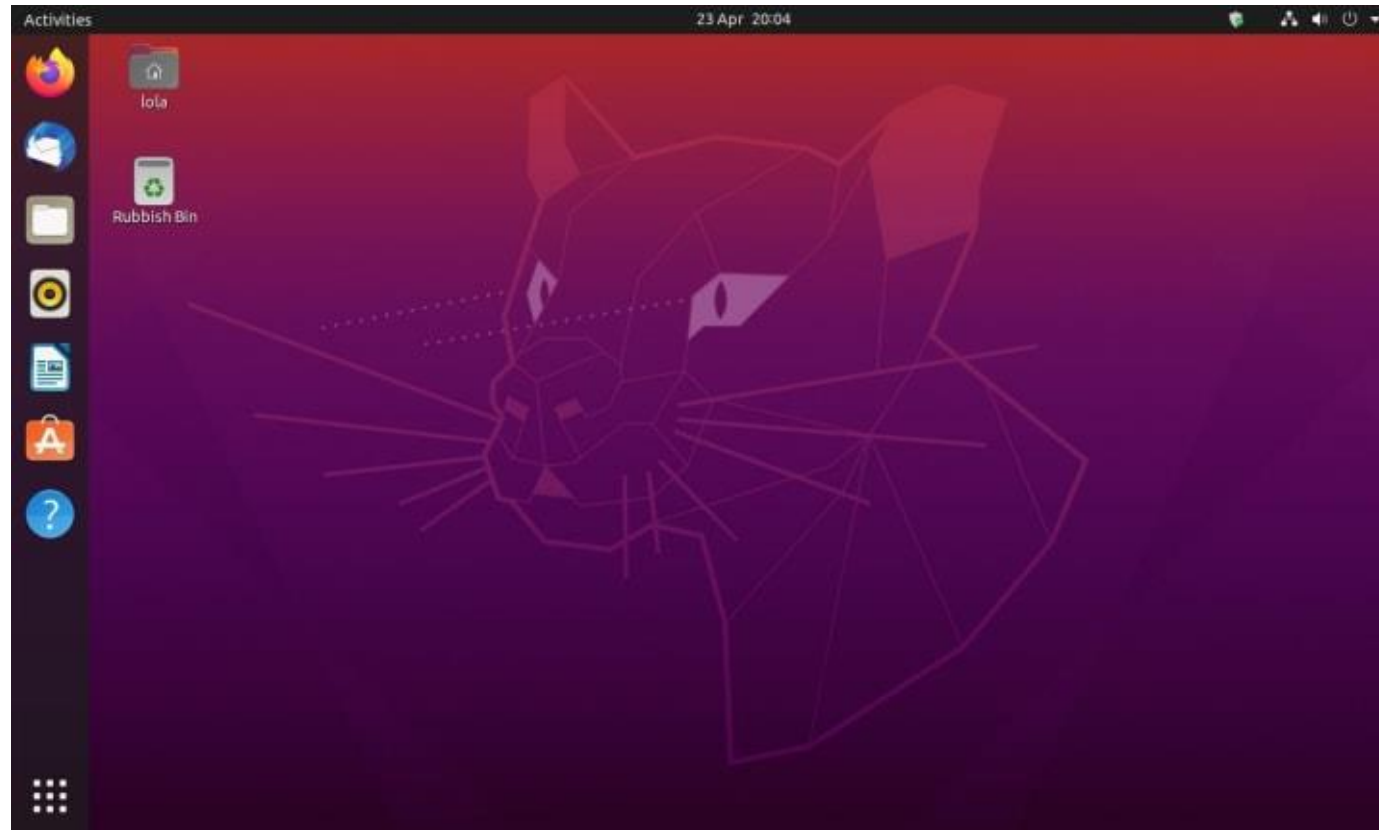
https://linuxhint.com/install_ubuntu_virtualbox_2004/

2. Open VirtualBox and follow the steps in the linked tutorial

*I recommend allocating your virtual machine at least **4GB** memory and **30GB** storage!*

Welcome to Ubuntu!

- Familiarise yourself with your Ubuntu system





The Command Line (or 'Terminal')

- Uses a text-based interface to control your computer

Commands are written in Bash, a command language

- You will be using it A LOT! We will practice now...

1. Click 'Show Applications' (bottom-left icon on the screen) and search for Terminal

When you find it, I suggest right-clicking and adding to favourites! It will now appear in the taskbar

2. Run Terminal

3. Type **cd Documents** and press Enter

This will navigate your active directory to Documents



4. Type `nano Hello_World.txt` and press Enter to start making a text document (named Hello_World)
5. Type `Hello World!` and press Ctrl+X
6. Type `Y` or `y` to agree to the changes made, and then press Enter to write the file.
7. To read the contents of the file, type `cat Hello_World.txt` and press Enter
8. If you want to edit the file, simply type `nano Hello_World.txt` again



9. Let's create a new folder and move our text file into it - type `mkdir MyFolder` and press Enter to create a new folder (named MyFolder)

10. Type `mv Hello_World.txt MyFolder` to move the text file into the folder

- To finish, let's clean up the mess we've made

11. Type `cd MyFolder` to navigate into the folder and then type `rm Hello_World.txt` to delete the text file

12. Type `cd ..` to exit out of the folder and then type `rm -rd MyFolder` to delete the folder

Basic Commands



- Navigate between folders: `cd <folder path>`

Note: `cd ..` exits the current folder

You can navigate to a specific directory in one go! Use `cd ~/<entire directory (from home)>`

- Opening files: `cat <file name>`
- Creating folder: `mkdir <folder name>`
- Creating/editing files: `nano <file name>`
- Copying files: `cp <source file> <target location>`
- Copying folders: `cp -r <source folder> <target location>`
- Deleting files: `rm <filename>`
- Deleting folders: `rm -rd <folder name>`

Many, MANY more commands exist!

<https://ubuntu.com/community/Beginners/BashScripting>

- Try the above link for more information on using the Command Line (and for much more practice!)
- I would recommend getting familiar with it sooner rather than later – you will really value this as we start working with ROS!

Summary

We covered:

- Overview of what ROS is
- How ROS works (nodes, publishers, subscribers, etc.)
- Introduction to VirtualBox and then Ubuntu
- The Command Line

Next time, we will install ROS and start using it!