



Ubuntu 101

chmod 777 what is security

Prerequisites

- A computer loaded with Ubuntu 20.04



Goals

What we're doing today!





- A basic intro to Linux
- Learning terminal commands
- Installing ROS and an overview

What is Linux?

Obey the penguin

- Open-source Operating System (OS)
- Commonly used by developers for software development, testing, etc.



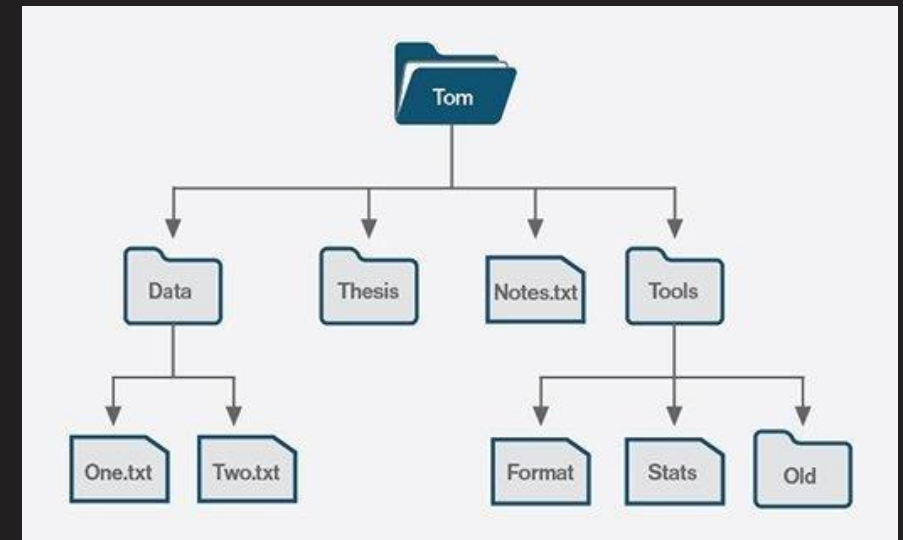
- Linux has many different distros:
 - Ubuntu 
 - Fedora 
 - Debian 
 - Arch Linux 
- 20.04 is the latest Long-Term Support (LTS) package
 - Also necessary for ROS Noetic



What the heck is Ubuntu?

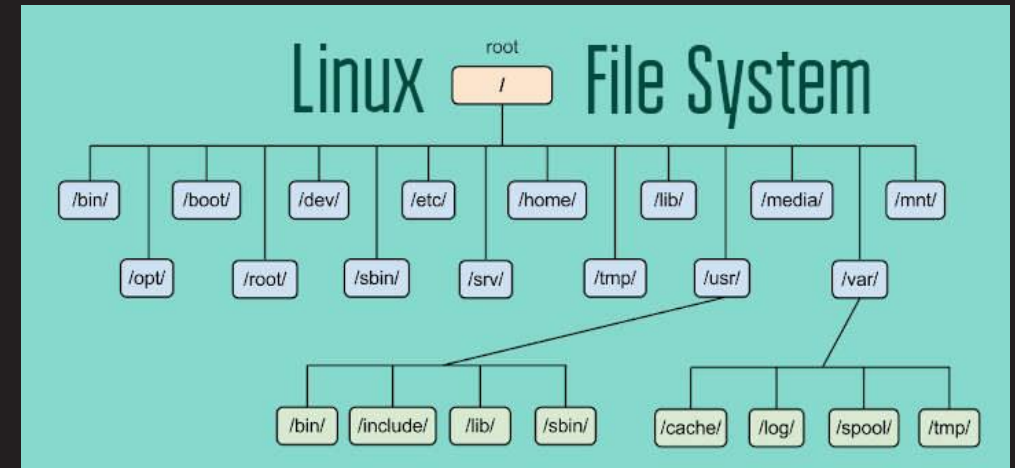
Filesystems

- The methods that an OS uses to organize its storage in the system
- Some common formats:
 - NTFS
 - FAT32
 - exFAT

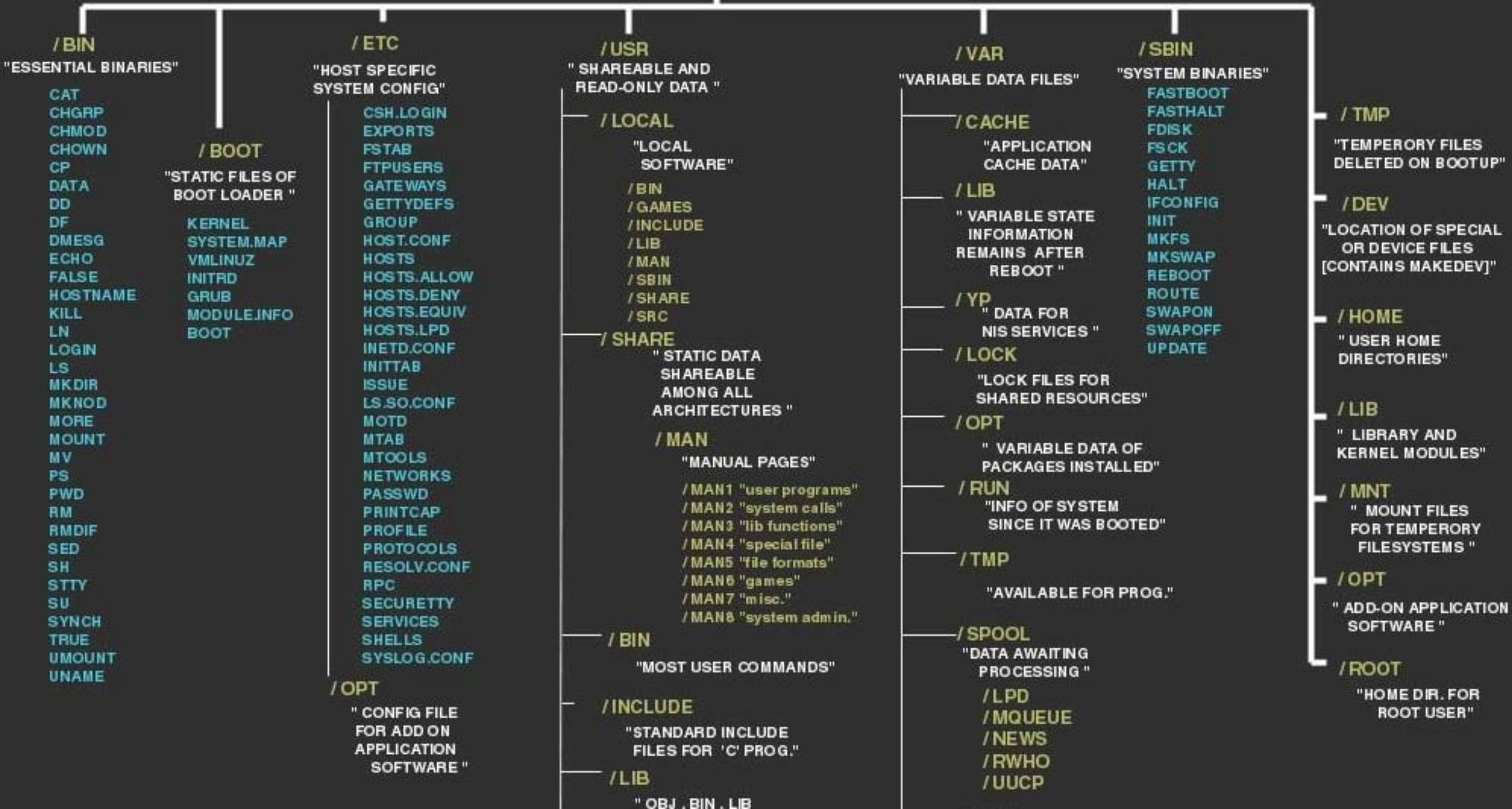


Linux File System Hierarchy

- Filesystem Hierarchy Standard (FHS)
- Standard layout for Linux
- Uses '/' for its path (vs '\', which Windows uses)
- Key paths:
 - /home/user/: Main directory (~)
 - /dev/: location of special or device files
 - /opt/: location of optional/add-on apps – ROS will be installed here!



/ "ROOT"



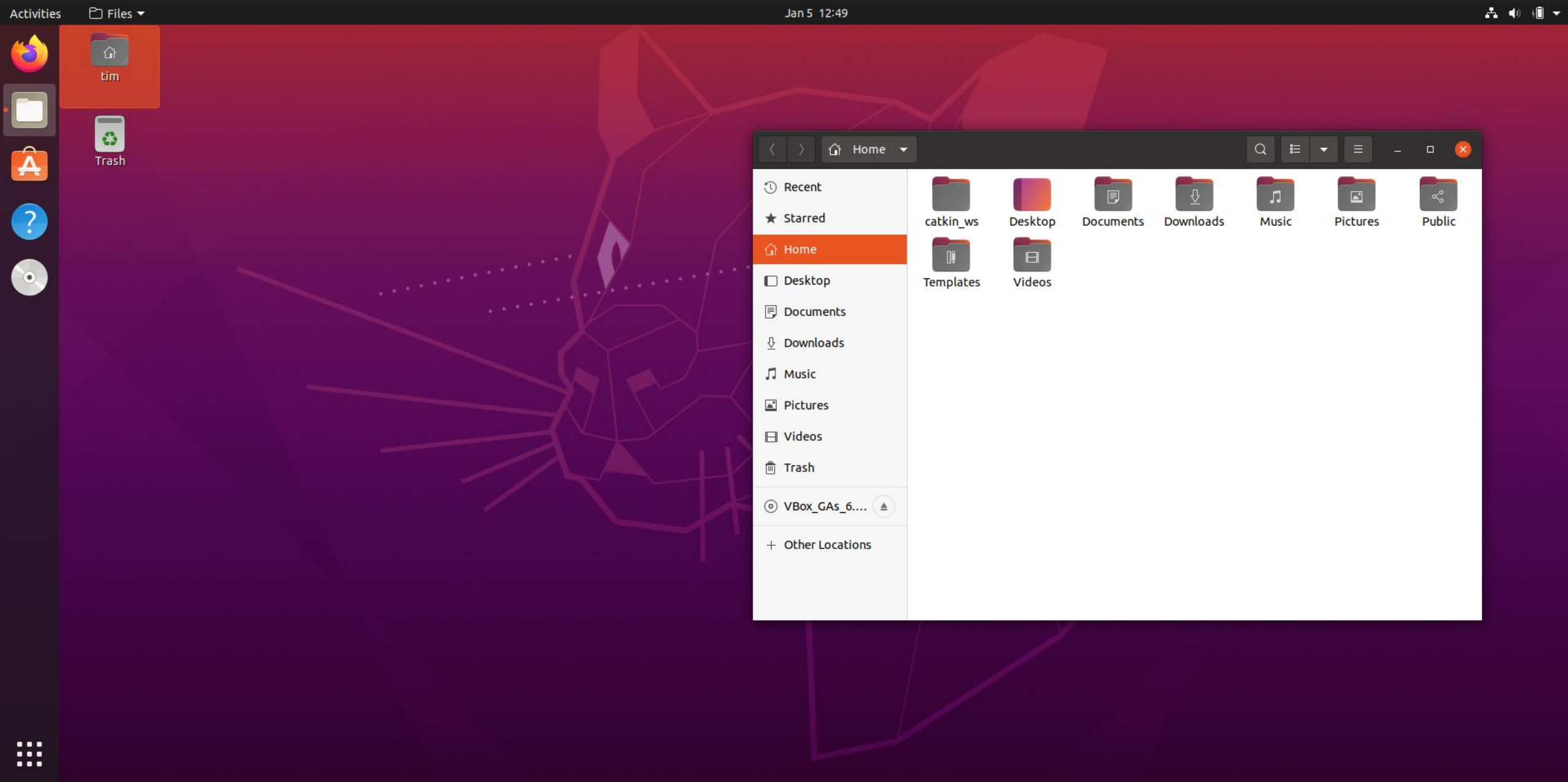

```
tim@tim-VirtualBox: ~$
```

(~ represents /home/USERNAME/)

```
tim@tim-VirtualBox: /dev$
```

Current path

Command Line Interface (CLI)



Common Console Commands

They're good to know!

Conventions

- `$` represents the terminal input, do not type it in!
- `#` is the start of a comment
- `[]` encloses optional arguments
 - If arguments only have specific inputs, they will be listed out

Examples:

`$ sudo apt update` -> Just type
“sudo apt update”

`$ ls [options] [path]` -> `ls`
can accept up to 2 arguments,
options and path

Superuser do (sudo)

- **Syntax:** `$ sudo [op]`
- Carries out whatever command is issued with administrator permissions
 - Usually requires admin password
- `[op]` is any command in the command line

General Commands

Checking for updates

apt update

- **Syntax: `$ sudo apt update`**
- Check for any updates (Update software repository list)
- May throw errors if the repositories for certain software are not properly linked/unlinked
- Does not carry out the update

apt upgrade

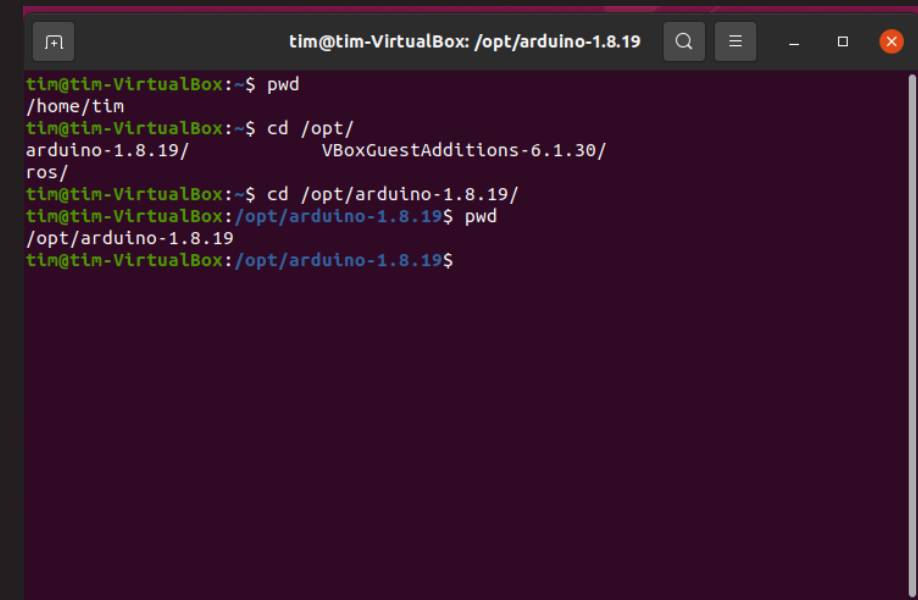
- **Syntax:** `$ sudo apt upgrade`
- Check for any updates AND carry out upgrade
- Note: always check whether any of the updates will affect your system/project dependencies!

Directory Navigation

Entering Directories, listing info

Print Working Directory

- Syntax: `$ pwd`
- Displays the full path of the current directory in the terminal window



```
tim@tim-VirtualBox: /opt/arduino-1.8.19
tim@tim-VirtualBox:~$ pwd
/home/tim
tim@tim-VirtualBox:~$ cd /opt/
arduino-1.8.19/
tim@tim-VirtualBox:~$ cd /opt/arduino-1.8.19/
tim@tim-VirtualBox:/opt/arduino-1.8.19$ pwd
/opt/arduino-1.8.19
tim@tim-VirtualBox:/opt/arduino-1.8.19$
```

change directory (cd)

- **Syntax:** `$ cd [path]`
- Goto the specified [path]
- If no path specified, goto home (~)
 - `$ cd` \leftrightarrow `cd ~`
- Shortcuts:
 - `..` \rightarrow go up a directory. `$ cd ..` brings you to the directory containing the current directory

list

- **Syntax:** `$ ls [options] [path]`
- Produces a list of filenames in the [path] specified
- If no path specified, does so for current directory
- Options
 - l: long list format → includes timestamps, filetypes and rwx perms
 - a: all files, including hidden ones
 - r: reverse order

ls (continued)

- **Syntax: `$ ls [options] [path]`**
- Can also be used for wildcard searching (*)
- Entering `$ ls /foo/bar/a*` will list all files which have path `/foo/bar/` and have a filename starting with 'a'
- Good for finding serial devices with
`$ ls /dev/tty*`

make directory (mkdir)

- **Syntax:** `$ mkdir [options] [name]`
 - Make a folder (directory) of name “name” in current directory
 - Options:
 - p: creates parent directories to generate the requested path as needed, does not throw an error if the directory already exists
- e.g.: `$ mkdir -p hello/hi`
Makes the directory “hello”, then create the directory “hi” in the directory “hello”

remove directory (rmdir)

- **Syntax:** `$ rmdir [options] [path]`
- Remove the directory(ies) supplied in [path] (you can specify multiple paths)
- The directory **MUST** be empty before removal!
- Options:
 - p: remove ALL ancestors in the given path up to current directory
e.g. carrying on from the previous examples, `$ rmdir -p hello/hi`
will remove both the directories “hello” and “hi”

Practice time!

1. Print out your current directory
2. Create the directory fish in the current directory
3. Enter the directory and further create 3 directories salmon, cod, eel
4. List all folders in fish in longlist format
5. Remove subfolder salmon

File Operations

Copy, create, delete, edit and more...

make file (touch)

- **Syntax:** `$ touch [file]`
- Create empty file(s) in directory (filename [file])
- Supports multiple file creation

e.g.

```
$ touch a.md
```

```
$ touch b.txt c.txt
```

move/rename (mv)

- **Syntax:** `$ mv [options] [src] [dest]`
- Move item from src to dest OR
- Rename item
- Options
 - r: recursive
 - i: interactive
 - n: prevent existing files in dest from being overridden

e.g.

```
$ touch a.md
```

```
$ mv a.md a.txt
```

```
$ mv a.txt salmon
```

copy (cp)

- **Syntax:** `$ cp [options] [src] [dest]`
- Copy item from src to dest OR
- Make a copy with a different name
- Options
 - r: recursive (copies all subdirectories of a directory)

e.g.

```
$ touch a.md
```

```
$ mv a.md a.txt
```

```
$ mv a.txt salmon
```

concatenate file (cat)

- **Syntax:** `$ cat [options] [file]`
- Concatenates file to console output (prints file in console)
- Options
 - n: show content with line number
 - s: don't show repeated empty lines in output
 - E: add a \$ at the end of each line

e.g.

```
$ echo "ax" > a.md #Writes "ax" into a.md
```

```
$ echo "chop" >> a.md #Append "chop" into a.md
```

```
$ cat a.md #Print on command line
```

```
$ cat -n a.md #Print with line number
```

Practice time!

1. Enter the cod directory
2. Create an empty textfile `swims.txt`
3. Rename the file to `action.txt`
4. Insert “swims” into `action.txt`
5. Copy `action.txt` into the eel directory (Hint: `../` represents one directory up)
6. Remove `action.txt` without changing directories

Viewing files

Preview

Get file type (file)

- **Syntax:** `$ file [file_name]`
- Prints file type in console output
- Common formats:
 - `.txt`: ASCII Text
 - `.md`: Markdown Text Format
 - `.py`: Python script, ASCII Text executable

Read file (less)

- **Syntax: `$ less [file_path]`**
- Prints file and paginates it
(Use the trackpad or up/down key to scroll through)
- Press 'q' to quit

File permissions

Read, Write, eXecute

Mod file permissions (chmod)

- Syntax: `$ chmod [permission] [file]`
- Usually done with a `sudo` prefix
- Permission(s):
 - Read (r): 4
 - Write (w): 2
 - eXecute (x): 1

```
# ls -l file
```

```
-rw-r--r-- 1 root root 0 Nov 19 23:49 file
```

Owner (rw-)

Group (r--)

Other (r--)

File type

r = Readable

w = Writeable

x = Executable

- = Denied

-: Regular File

D: Directory

Owner: you

Group: All users on this computer

Other: Users outside this computer (e.g. Remote access/clients)

u g o
754

	r	w	x	r	w	x	r	w	x
access									
binary	4	2	1	4	2	1	4	2	1
enabled	<u>1</u>	<u>1</u>	<u>1</u>	<u>1</u>	<u>0</u>	<u>1</u>	<u>1</u>	<u>0</u>	<u>0</u>
result	<u>4</u>	<u>2</u>	<u>1</u>	<u>4</u>	<u>0</u>	<u>1</u>	<u>4</u>	<u>0</u>	<u>0</u>
total	7			5			4		

Miscellaneous

Other useful stuff

Autocomplete

- **Syntax:** hit [Tab] key twice
- Your best friend for Linux console
- Will provide any existing suggestions for autocomplete

Execute (Order 66)

- **Syntax:** `$./[path_to_file]`
- Execute the file, used to run scripts or programs

Shell Execute

- **Syntax: `$ source [path_to_file]`**
- Runs all text in the file as if it was entered into the command line
- Important for running ROS!

Echo

- **Syntax:** `$ echo [var]`
- Prints specified variable in console output

e.g.

```
$ echo "lorem ipsum"
```

```
$ echo "time"
```

```
$ echo $HOME #System variable "$HOME"
```

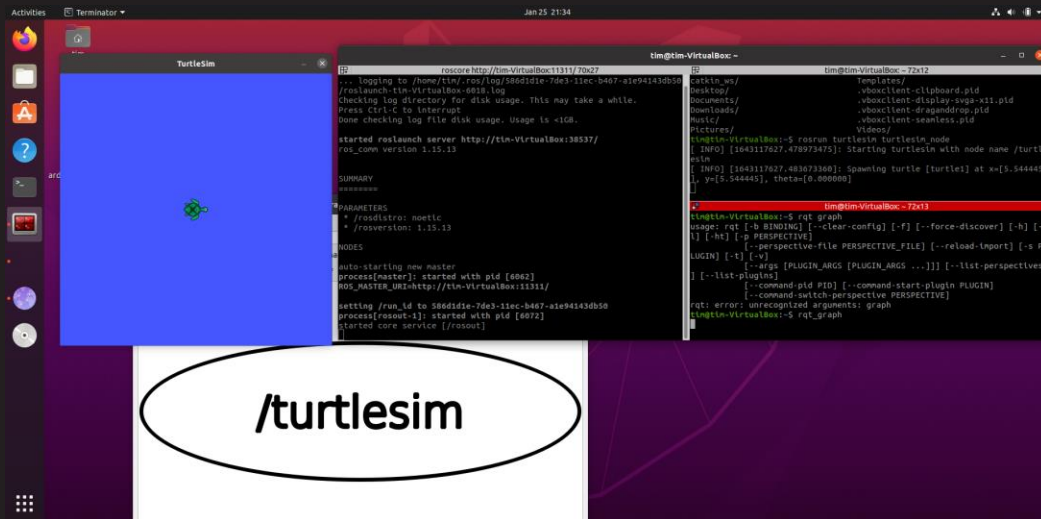
Clear terminal

- **Syntax: `$ clear`**
- Can't say much for this one

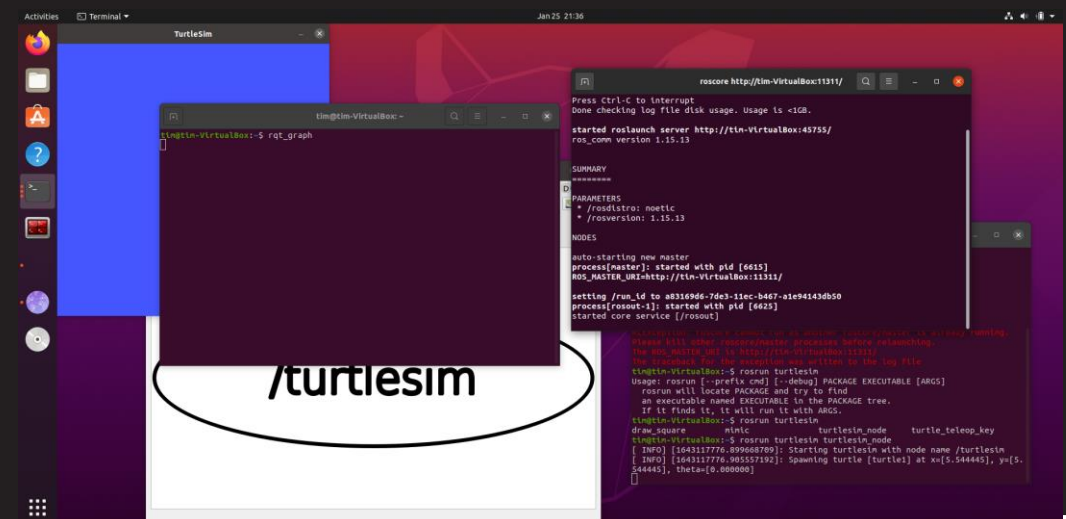
Get Terminator

- (optional step)

\$ sudo apt install terminator



With Terminator



Without Terminator

ROS

An overview

What is ROS?

- **Robot Operating System**
- Middleware – helps different programs/scripts communicate with one another
- A few key concepts:
 - Nodes
 - Topics
 - Publishers/Subscribers
 - Messages

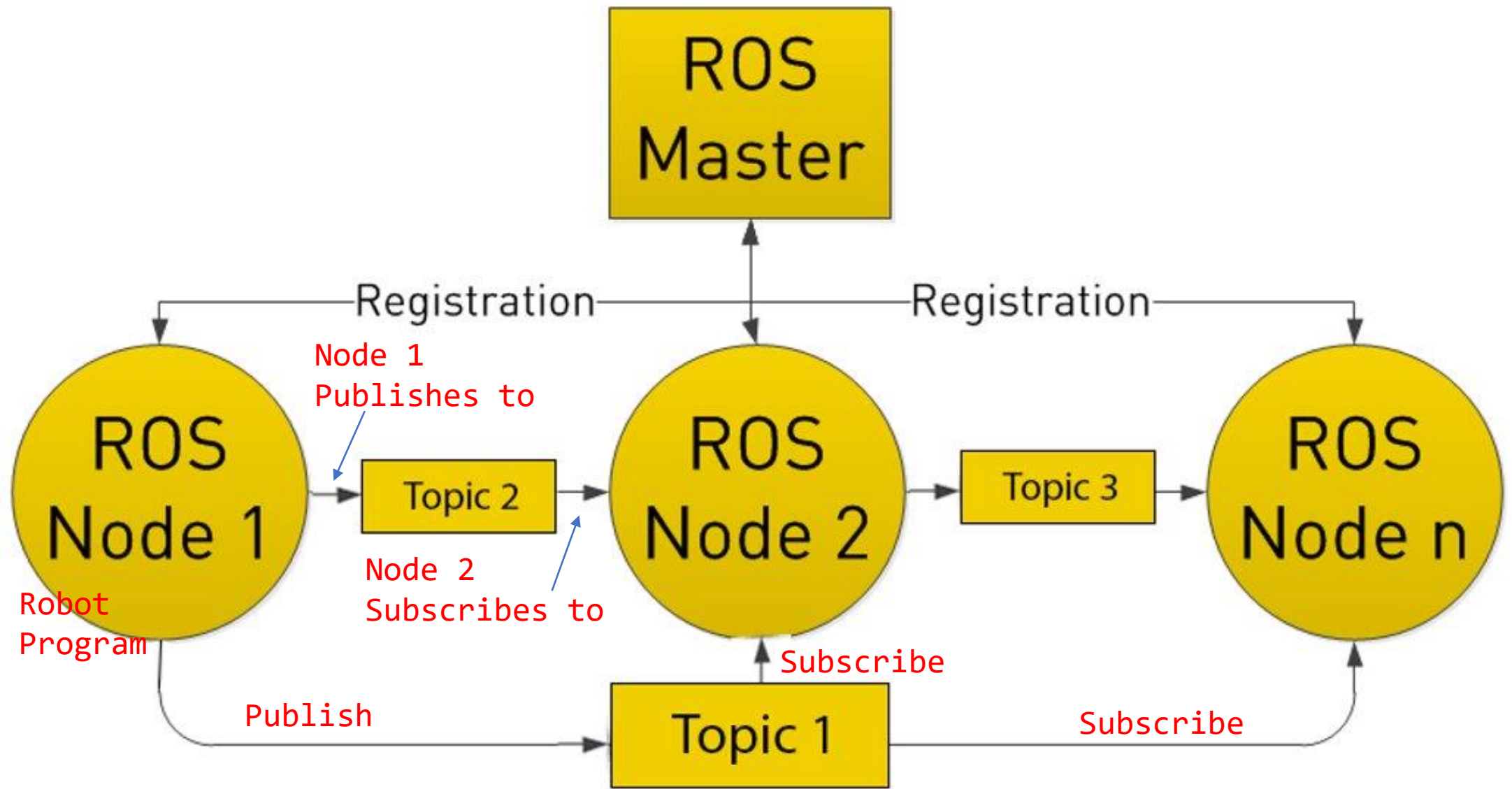
Why ROS?

- Premade Packages



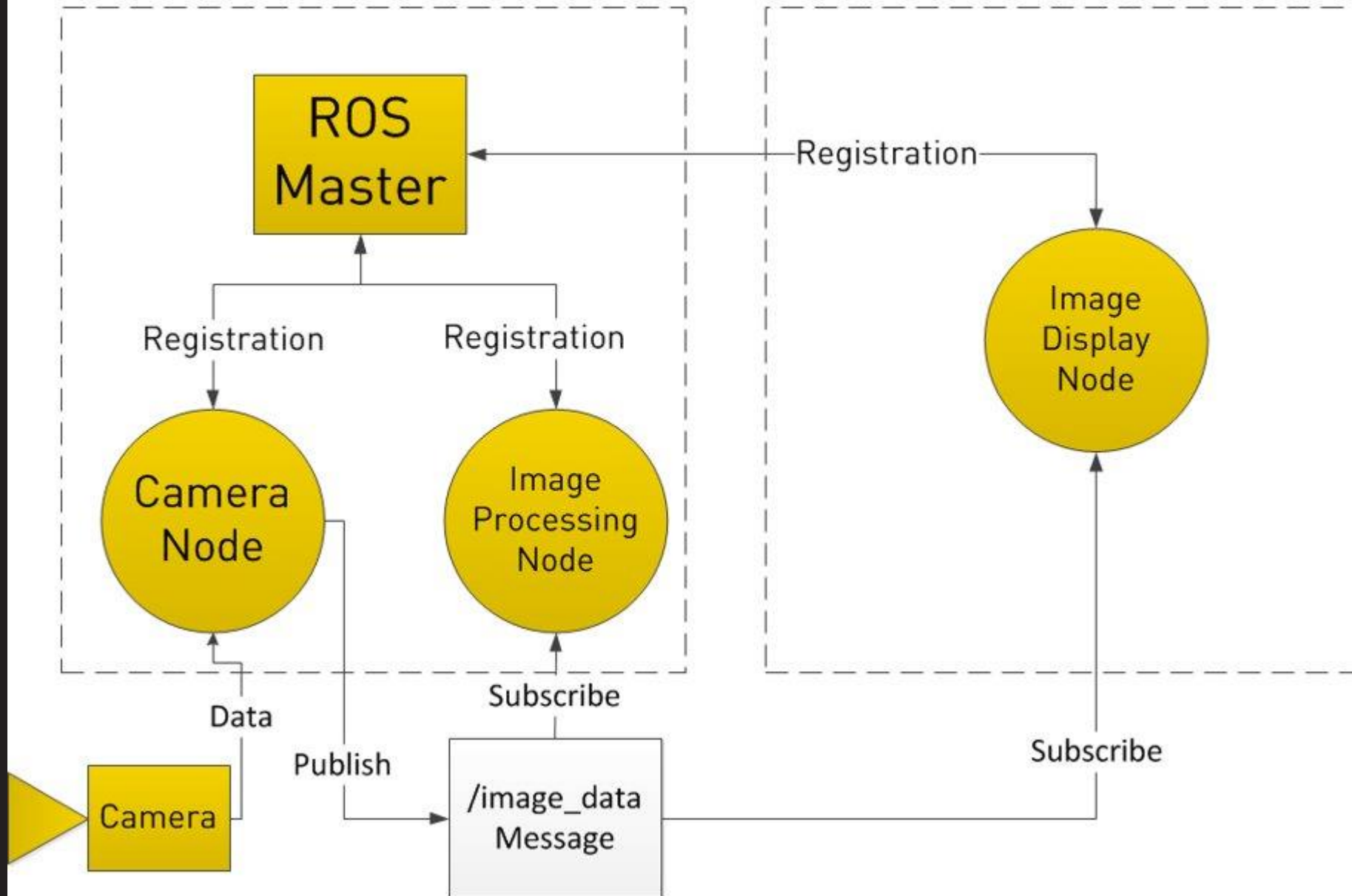
- Language Independence





Computer on the Robot

Laptop



Very Important Note

- Always remember to source

```
$ source /opt/ros/noetic/setup.bash
```

- Or automate it:

```
$ echo "source /opt/ros/noetic/setup.bash" >>  
~/.bashrc
```

TurtleSim

The less ugly side

Thank you!

You're ready for ROS next