AURORA ROBOTICS CORE WORKSHOP 1.0

INSTALLATION CLASS

GENERAL CLASS RULES

- All mics should be muted
- I am always looking at the chat box so feel free drop questions anytime
- Feel free to use the “raise your hand” option anytime
- I recommend you have a note (could be digital), I tend to drop random knowledge casually

All slides, code and materials will be shared in the training repo. Feel free to use and share but do not modify

(I am very good with lawsuits)

TOOLS WE'LL USE IN THE WORKSHOP

Core Robotics Tools

- ROS 2
- Gazebo + RViz

Programming & Languages

- Python 3.x
- C++
- Rust (bonus track)

Development Tools

- Git + GitHub
- VS Code

Extras (Fun & Later)

- Docker (not now)
- OpenCV (vision experiments)

THE ENVIRONMENT PROBLEM

- When I thought about how to make this training easy, I remembered my experiences:
  - IEEE RAS Summer School, Prague🇨🇿, everything ran in Docker on a website.
  - Greece🇬🇷 & Netherlands🇳🇱, same thing, all pre-configured for us.
- It was smooth, fast, and comfortable…

But Here’s the Truth

- If I gave you a pre-built Docker or website environment:
  - Easier for you in class
  - But unrealistic for the real world
- You’d leave this training without knowing how to set up your own robotics environment.
- And as a robotics engineer… as how naaa?? as howww??

SO I AM FORCING YOU GUYS TO GO DOWN THE HARD PATH TO LEARN! AND LEARN BETTER!

OPERATING SYSTEMS IN ROBOTICS

Operating Systems in Robotics

Why OS matters in Robotics

- The OS manages hardware, processes, and timing - critical for real-time performance.
- Determines which tools & frameworks you can run (e.g., ROS support).
- Impacts stability, drivers, and community support.

Common OS in Robotics

- Linux (Ubuntu) → [THE GOAT] Industry standard for ROS & robotics.
- Windows → [meehhhhh] okay for development tools, but limited ROS support.
- MacOS → [I don´t even know how to type on those] but it works for coding & simulation, but smaller robotics ecosystem.

⚠️ FRIENDLY WARNING! ⚠️

- In the robotics, Linux is king (especially Ubuntu).

- It’s 100% normal for robotics folks to throw shades at Windows😂. Just like I urge you’ll to throw your windows machine out the window!

- But don’t worry – no hard feelings!
- The truth is: we’ll all use whatever gets the job done (although it will stress me).

- Anyways, we joke about it, but we’re all here to learn together. 😎

A TRIP TO WILLOW GARGE, 2007

- While I was 5 years old crying because I didn’t want to wake up early to go to school, a company in Silicon Valley (Willow Garage) made a decision that would forever change robotics.
- They chose to make robotics software 100% open source and standardized. That decision gave birth to the Robot Operating System (ROS), a common language (not really) for robots everywhere.
- And since ROS was built and tested first on Linux (Ubuntu), it naturally became the most supported OS for robotics.

Disclaimer

So if you ever hear me throwing shades at Windows or Mac… just know this movement is bigger than me.
😇 I am innocent of the attacks, history already chose Linux.

MORE REASONS WHY LINUX

- Open Source Ecosystem
  - Robotics thrives on open-source collaboration.
  - Linux (Ubuntu) is free, customizable, and integrates easily with open-source tools.

- Real-Time & Low-Level Access
  - Linux allows direct access to hardware, device drivers, and kernel modifications.
  - Crucial for real-time performance in robotics.

- Package Management (apt)
  - Easy installation and updates of robotics dependencies (e.g., sudo apt install ros-humble-desktop).
  - No need to fight Windows installers.

MORE REASONS WHY LINUX

- Community & Documentation
  - The largest robotics community worldwide uses Ubuntu.
  - 90% of online help, GitHub repos, and forum answers assume Ubuntu.

- Industry Adoption
  - Drones (PX4, ArduPilot), self-driving cars, underwater robots, industrial arms → all tested on Ubuntu.
  - Major companies (NVIDIA, Intel, Boston Dynamics, Clearpath, etc.) release robotics SDKs and drivers primarily for Ubuntu.

OPTION 1:
SWITCH TO LINUX (UBUNTU)

You could ditch your OS

but I advise to first:

- Dual Boot for start with Your Current OS
  - Keep Windows/Mac for daily use
  - Boot into Ubuntu when working on robotics
  - Best of both worlds while you transition

OPTION 2 INSTALL UBUNTU IN WINDOWS (WSL!!)

Now, you can attempt to install our environment directly on windows, don’t get me wrong. But you see how I am dodging that non-deterministic operating system!

```
wsl --install -d Ubuntu-22.04
```

Then we will learn the art of reading documentations, for there is the wisdom of robotics💚

https://docs.ros.org/en/iron/Installation/Ubuntu-Install-Debs.html

OPTION 3 INSTALL DIRECTLY ON WINDOWS

Only try this in the event that option 2 does not work for you for some reasons best known to God

OPTION MAC!

Please see me in my office

WHO USES MAC??

- ROS 2 on macOS is possible
- RViz2 and Gazebo support is very limited
    - Because they are Qt/OpenGL heavy and not officially maintained for mac.

Most robotics devs I know that use mac (only 2!) only use macOS as an editor/IDE and run ROS 2 + Gazebo inside a Linux VM, Docker, or remote server.

You can install Ubuntu ARM64 (aarch64) inside Parallels/UTM or installed Asahi Linux natively.

POPULAR DEMAND

How to Linkedin

(Brief chat before we install)

THANK YOU FOR LISTENING

Contact Information

in Aurora Robotics

𝕏 aurora_robots