



# Introduction to Robot Operating System (ROS 1)

What is ros? How to install ros1 noetic?

Dr. Essa Alghannam

# (Robot Operating System)

## ROS



- an open source software development kit for robotics applications in research and industry.

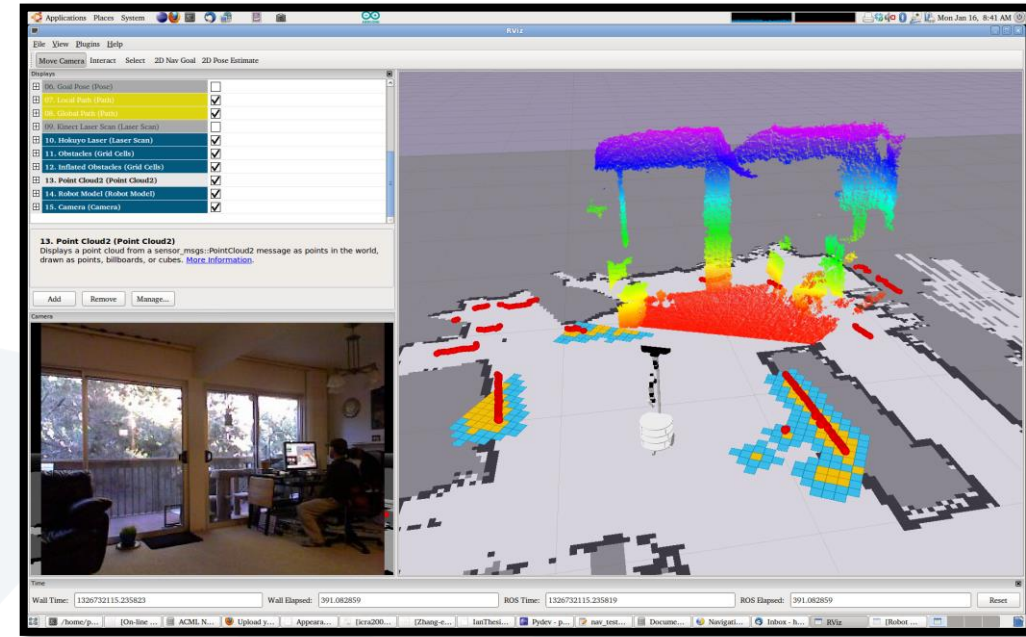
- **100% Open-source and Multi-platform**
- **Global Community**
- **Industry Support**
- **Proven in Use**
- **Shorten Time to Market**

# ROS Tools

## Rviz 3D visualization tool



- Visualize Robot Models: Display the 3D model of your robot, including its sensors, joints, and links.
- Show Sensor Data: Visualize data from sensors like cameras, lidar, and IMUs (Inertial Measurement Units).
- This includes point clouds, images, and sensor readings.
- Display Robot State: Show the robot's position, orientation, and joint configurations.
- Interact with the Robot: You can move the robot around, change its joints, and explore its environment in the RViz window.
- Simulations: While RViz is primarily for visualization, it can be used in conjunction with Gazebo for simulations. You can use RViz to visualize the robot's behavior within a Gazebo simulation.
- Example: You're working with a robot equipped with a laser scanner. RViz can display a point cloud generated by the laser scanner in real-time as your robot navigates its environment.

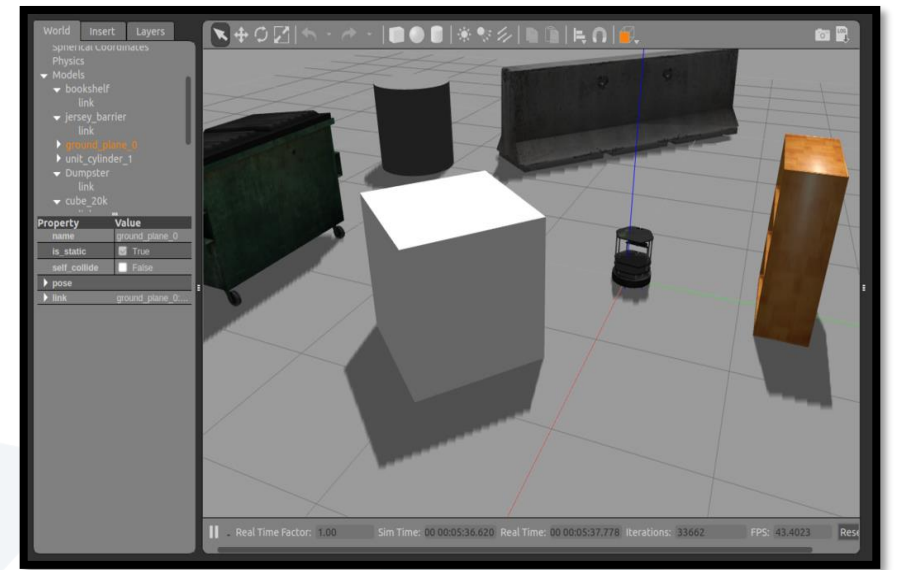


# ROS Tools

## Gazebo for Simulation



Gazebo is a powerful physics-based simulator for robots. It allows you to test and experiment with robot behavior in a safe and controlled environment.



- Create Simulated Worlds: Design realistic environments, including terrain, obstacles, and objects.
- Physically Simulate Robots: Simulate your robot's movements, interactions with the environment, and sensor data.
- Realistic Physics: Gazebo uses a physics engine to accurately model forces, collisions, and other physical interactions.
- Testing and Debugging: You can test your robot's algorithms and code in a virtual environment before deploying them on real hardware.

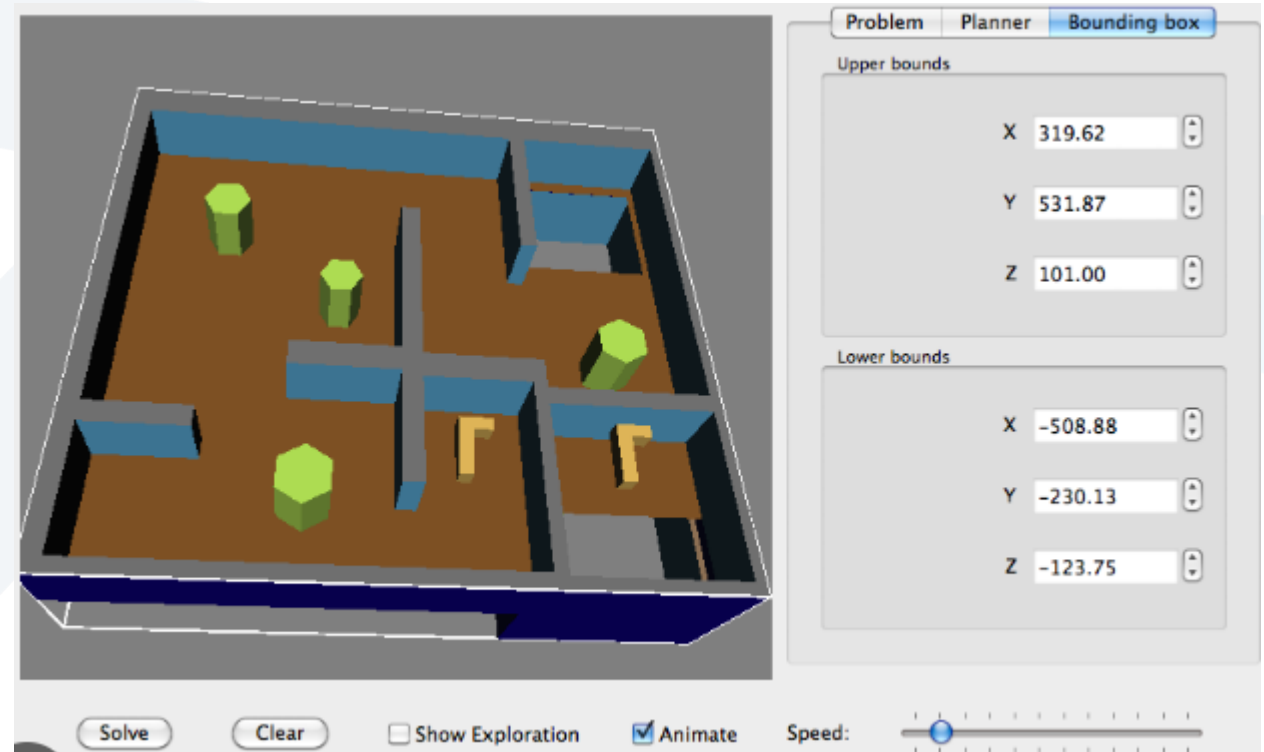
\* Example: You're developing a robot that navigates a warehouse. Gazebo allows you to create a virtual warehouse environment, including shelves, boxes, and obstacles. You can then simulate your robot's navigation algorithms in this environment and see how it responds to different situations.

Feature	Rviz your visual window into the robot's world	Gazebo is the virtual playground where you test and refine your robot's behavior
Purpose	Visualization	Simulation
Physics	No physics engine	Physics engine
Sensors	Displays sensor data	Simulates sensor data
Environments	Limited to visualization	Realistic environment creation
Hardware Interaction	None	Can interface with real robots

RViz and Gazebo are often used together. You might use Gazebo to simulate your robot's movements in a virtual environment, and then use RViz to visualize the robot's state, sensor data, and other important information during the simulation.

# ROS Tools

Graphical user interface



## What is a Distribution?



جامعة  
المنارة  
MANARA UNIVERSITY

- توزيع ROS عبارة عن مجموعة ذات إصدار من حزم ROS
- تشبه توزيعات Linux مثل Ubuntu
- الغرض من توزيعات ROS هو السماح للمطورين بالعمل على قاعدة بيانات ثابتة نسبياً حتى يصبحوا مستعدين لدفع كل شيء إلى الأمام. لذلك بمجرد إصدار التوزيع، نحاول قصر التغييرات على إصلاحات الأخطاء والتحسينات غير الفاصلة للحزم الأساسية كل شيء تحت ros-desktop-full

Technically you can still install and use ROS Noetic after 2025

- **Reduced Package Support:** Packages might stop working properly due to dependencies on other libraries or operating system features that are no longer actively supported.
- **Limited Community Support:** The community focus will shift towards newer ROS releases, meaning you'll likely find less help and resources available for Noetic-specific problems.
- **Compatibility Challenges:** As newer hardware and software technologies emerge, your system might encounter incompatibility issues with Noetic.

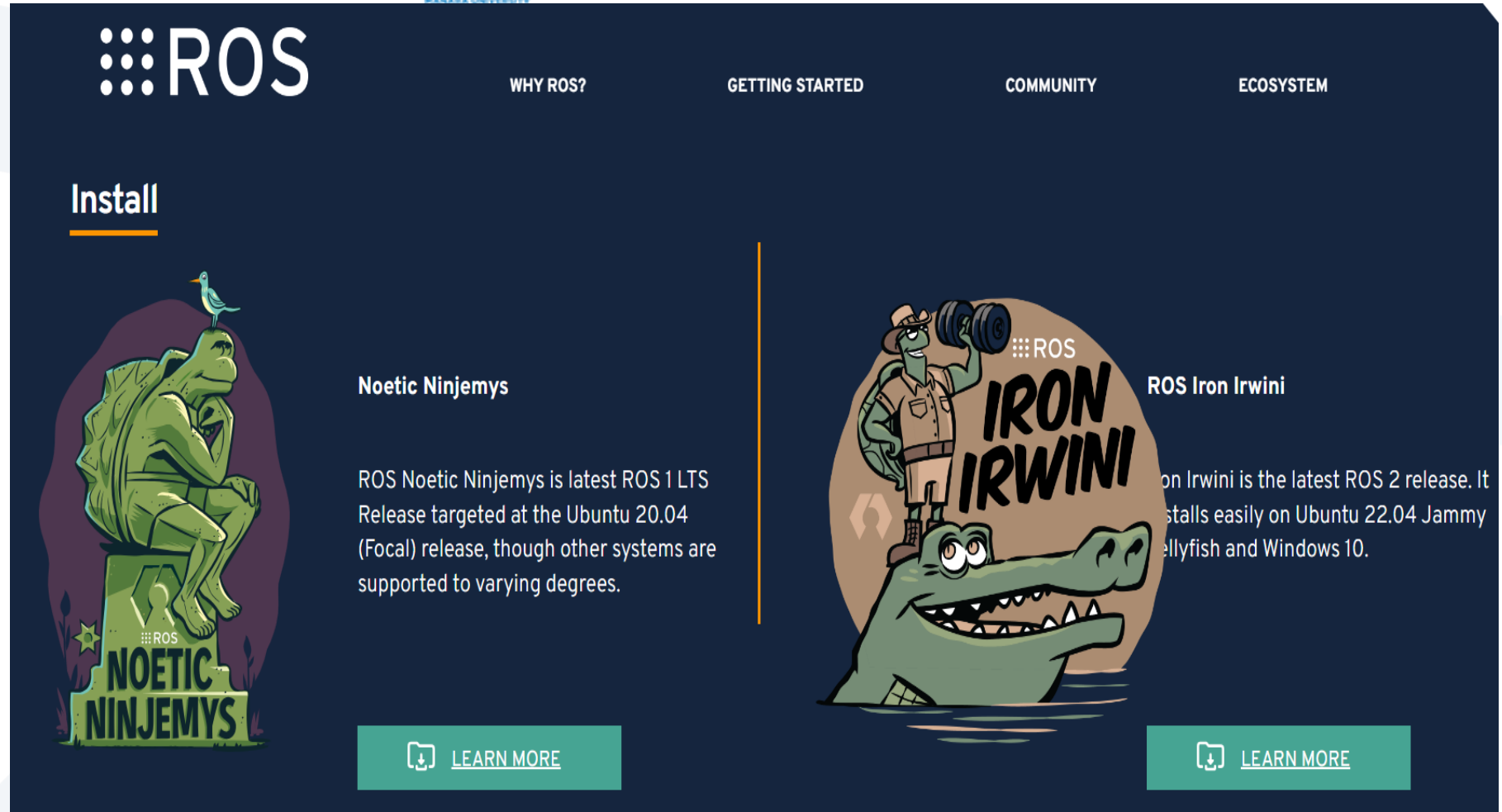
## ROS1 Distribution list

Distro	Release date	Poster	Tuturtle, turtle in tutorial	EOL date
ROS Noetic Ninjemys (Recommended)	May 23rd, 2020			May, 2025 (Focal EOL)
ROS Melodic Morenia	May 23rd, 2018			June 27, 2023 (Bionic EOL)
ROS Lunar Loggerhead	May 23rd, 2017			May, 2019
ROS Kinetic Kame	May 23rd, 2016			April, 2021 (Xenial EOL)
ROS Jade Turtle	May 23rd, 2015			May, 2017
ROS Indigo Igloo	July 22nd, 2014			April, 2019 (Trusty EOL)
ROS Hydro Medusa	September 4th, 2013			
ROS Groovy Galapagos	December 31, 2012			July, 2014

- light yellow: future release
- green: supported release
- grey: unsupported release (End of Life)

# Install ros1

- Support and Updates:  
Newer distributions generally have better support for newer operating systems, hardware, and software dependencies.
- Package Availability:  
Newer distributions have a larger package library,
- older distributions may still be sufficient for many applications.



The screenshot shows the ROS website's 'Install' page. At the top, the ROS logo is displayed, followed by navigation links: 'WHY ROS?', 'GETTING STARTED', 'COMMUNITY', and 'ECOSYSTEM'. The 'Install' section is highlighted with a red underline. Below this, there are two main options for installation:

- Noetic Ninjemys**: Accompanied by an illustration of a green turtle sitting on a pedestal with the text 'NOETIC NINJEMYS'. The text below reads: 'ROS Noetic Ninjemys is latest ROS 1 LTS Release targeted at the Ubuntu 20.04 (Focal) release, though other systems are supported to varying degrees.' A 'LEARN MORE' button is located below the text.
- ROS Iron Irwini**: Accompanied by an illustration of a green crocodile with a robot head and a person on its back holding binoculars, with the text 'IRON IRWINI'. The text below reads: 'Iron Irwini is the latest ROS 2 release. It installs easily on Ubuntu 22.04 Jammy Jellyfish and Windows 10.' A 'LEARN MORE' button is located below the text.



# Releases - Ubuntu Wiki



## List of releases

[Ubuntu Website release cycle page](#)

### Current

Version	Code name	Docs	Release	End of Standard Support	End of Life
Ubuntu 22.10	<a href="#">Kinetic Kudu</a>	<a href="#">Release Notes</a>	October 20, 2022	July 2023	July 2023
Ubuntu <b>22.04.1 LTS</b>	<a href="#">Jammy Jellyfish</a>	<a href="#">Release Notes</a>	August 11, 2022	April 2027	April 2032
Ubuntu <b>22.04 LTS</b>	<a href="#">Jammy Jellyfish</a>	<a href="#">Release Notes</a>	April 21, 2022	April 2027	April 2032
Ubuntu <b>20.04.5 LTS</b>	<a href="#">Focal Fossa</a>	<a href="#">Changes</a>	September 1, 2022	April 2025	April 2030
Ubuntu <b>20.04.4 LTS</b>	<a href="#">Focal Fossa</a>	<a href="#">Changes</a>	February 24, 2022	April 2025	April 2030
Ubuntu 20.04.3 LTS	<a href="#">Focal Fossa</a>	<a href="#">Changes</a>	August 26, 2021	April 2025	April 2030
Ubuntu 20.04.2 LTS	<a href="#">Focal Fossa</a>	<a href="#">Changes</a>	February 4, 2021	April 2025	April 2030
Ubuntu 20.04.1 LTS	<a href="#">Focal Fossa</a>	<a href="#">Changes</a>	August 6, 2020	April 2025	April 2030
Ubuntu 20.04 LTS	<a href="#">Focal Fossa</a>	<a href="#">Release Notes</a>	April 23, 2020	April 2025	April 2030
Ubuntu <b>18.04.6 LTS</b>	<a href="#">Bionic Beaver</a>	<a href="#">Changes</a>	September 17, 2021	April 2023	April 2028
Ubuntu 18.04.5 LTS	<a href="#">Bionic Beaver</a>	<a href="#">Changes</a>	August 13, 2020	April 2023	April 2028
Ubuntu 18.04.4 LTS	<a href="#">Bionic Beaver</a>	<a href="#">Changes</a>	February 12, 2020	April 2023	April 2028
Ubuntu 18.04.3 LTS	<a href="#">Bionic Beaver</a>	<a href="#">Changes</a>	August 8, 2019	April 2023	April 2028
Ubuntu 18.04.2 LTS	<a href="#">Bionic Beaver</a>	<a href="#">Changes</a>	February 15, 2019	April 2023	April 2028
Ubuntu 18.04.1 LTS	<a href="#">Bionic Beaver</a>	<a href="#">Changes</a>	July 26, 2018	April 2023	April 2028
Ubuntu 18.04 LTS	<a href="#">Bionic Beaver</a>	<a href="#">Release Notes</a>	April 26, 2018	April 2023	April 2028

- Ubuntu 20.04 was released in 2020 (20), April (04).
- Ubuntu 22.04 was released in 2022 (22), April (04).

•The last ".4" part of 20.04.4 is called the "point release." It means this is the 4 revision of the 20.04.

- Both 20.04 and 22.04 are [LTS](#). This means the version 20.04 gets long term support (5 years). Part of the long term support means support for new hardware.

LONG TIME STABLE



# ROS Noetic installation instructions

These instructions will install **ROS Noetic Ninjemys**, which is available for Ubuntu Focal (20.04), Debian Buster (10), and [other platform options](#).

To install our previous long-term support release, **ROS Melodic Morenia**, please see the [ROS Melodic installation instructions](#).

Release Date: May 2020

Key Features:

- Long-Term Support (LTS).
- Python 3 support (Python 2 is still supported)
- Improved build system performance
- Updated and enhanced ROS tools
- Support for operating systems (Ubuntu 20.04)
- packages for robotics and AI applications

## Select Your Platform

Supported:



Ubuntu Focal amd64 armhf arm64



Debian Buster amd64 arm64

[Source installation](#)

Experimental:



Windows 10 amd64



Arch Linux Any amd64 i686 arm armv6h armv7h aarch64

# Installation



<https://wiki.ros.org/noetic/Installation/Ubuntu>

## 1. Setup your sources.list:

Setup your computer to accept software from packages.ros.org.

```
sudo sh -c 'echo "deb http://packages.ros.org/ros/ubuntu $(lsb_release -sc) main" >
/etc/apt/sources.list.d/ros-latest.list'
```

**Sudo:** "superuser do." you're essentially running the command as the root user. Because Modifying system files like `/etc/apt/sources.list.d/ros-latest.list` requires root access.

`sh -c`: The `-c` flag tells the **shell** `sh` that the command to execute is provided **as an argument, not by reading from a file**. And this argument is provided within single quotes (`'...'`)

`echo ...` adding the ROS repository to your system's package sources

```
echo "deb http://packages.ros.org/ros/ubuntu $(lsb_release -sc) main" > /etc/apt/sources.list.d/ros-latest.list'
```

`echo`: This command prints text to the terminal, but with `>` redirection, it's used to write to a file.

`"deb http://packages.ros.org/ros/ubuntu $(lsb_release -sc) main"`: This is the actual text that is being written to the file.

- 1 `deb`: This specifies the **type of package source** (in this case, **Debian**).
- 2 `http://packages.ros.org/ros/ubuntu`: This is the URL of the ROS package repository.
- 3 `$(lsb_release -sc)`: This part dynamically gets your Ubuntu release codename (e.g., 'bionic', 'focal', etc.). This is important to ensure you're getting the correct ROS packages for your Ubuntu version.
- 4 `main`: This indicates the main repository of ROS packages.
- 5 `>`: This redirects the output of the `echo` command to the specified file.
- 6 `/etc/apt/sources.list.d/ros-latest.list`: This is the file where the ROS repository information is stored. This file is part of the `apt` system, which is the package manager on Ubuntu and Debian-based systems.

## In Summary



This command tells the system to use the `sudo` command with root privileges to execute a shell command (`sh -c`).

This shell command then writes the ROS repository information to a file within the `/etc/apt/sources.list.d` directory, allowing the `apt` system to access and install ROS packages.

# Installation

2. Set up your keys:

1. `sudo apt install curl # if you haven't already installed curl`
2. `curl -s https://raw.githubusercontent.com/ros/rosdistro/master/ros.asc | sudo apt-key add -`

```
`sudo apt install curl`
```



1. ``sudo``: - This stands for "superuser do." It allows you to run commands with elevated privileges, essentially running them as the root user. Installing software on your system usually requires root access to modify system files and directories.
2. ``apt``: - This is the command-line tool for managing software packages on Debian-based systems like Ubuntu. It acts as your package manager: It handles searching for, downloading, installing, upgrading, and removing software packages.
3. ``install``: - This is a subcommand of ``apt`` used to install new packages on your system.
4. ``curl``: This is the name of the software package you want to install.
  - curl stands for "client URL" and is a command-line tool for transferring data using URLs. It's extremely useful for downloading files, sending data to web servers, and working with APIs.

The command ``sudo apt install curl`` tells your system to use root privileges (``sudo``) to use the ``apt`` package manager to install the ``curl`` software package. Once the command is run, ``curl`` will be installed on your system, making it available to use from the command line.

curl command is a tool to download or transfer files/data from or to a server using FTP, HTTP, HTTPS, SCP, SFTP, SMB and other supported protocols on Linux or Unix-like system.

```
curl -s https://raw.githubusercontent.com/ros/rosdistro/master/ros.asc | sudo apt-key add -
```

This command is used to add the official ROS (Robot Operating System) public key to your system's list of trusted keys. This is crucial for verifying the authenticity of ROS packages you download later.

1. `curl -s https://raw.githubusercontent.com/ros/rosdistro/master/ros.asc`

- ❖ `curl`: This is the command-line tool for transferring data using URLs (as we discussed before).
- ❖ `-s`: This flag tells `curl` to run silently (no output to the terminal).
- ❖ `https://raw.githubusercontent.com/ros/rosdistro/master/ros.asc`: This is the URL of the ROS public key file. It's hosted on GitHub within the `rosdistro` repository.



## 2. `| sudo apt-key add -``

- ❖ `|``: This is the pipe operator. It takes the output of the command on the left and sends it as input to the command on the right.
- ❖ `sudo``: Again, this allows the command to run with root privileges.
- ❖ `apt-key``: This is a command-line tool specifically for **managing the list of trusted keys used by `apt` for package verification.**
- ❖ `add -``: This tells `apt-key`` to add the public key (which is being piped from `curl``) to the list of trusted keys. The `-`` indicates that the key data is being supplied from standard input (the pipe).

## In Summary

This command sequence first downloads the ROS public key from GitHub using ``curl`` and then uses ``sudo apt-key add -`` to add that key to the system's trusted key list.

## `sudo apt update`

First, make sure your Debian package index is up-to-date:

The command `sudo apt update` is a crucial command and step in managing software packages on Debian-based systems like Ubuntu.

1. `sudo`: As you know, this grants the command elevated privileges, allowing it to make changes that require root access.
2. `apt`: This is the command-line tool for managing software packages on your system. It's your package manager, handling tasks like searching for, downloading, installing, upgrading, and removing software.
3. `update`: This is a subcommand of `apt` that tells it to **refresh the package lists.**

## sudo apt install ros-noetic-desktop-full



Everything in Desktop plus 2D/3D simulators and 2D/3D perception packages

This command installs the full ROS Noetic desktop environment on your system. Let's break it down:

1. ``sudo``: As you know, this grants the command elevated privileges, allowing it to make changes that require root access. This is needed because installing software requires modifications to system files.

2. ``apt``: This is the command-line tool for managing software packages on Debian-based systems like Ubuntu. It's your package manager, handling tasks like searching for, downloading, installing, upgrading, and removing software.

3. ``install``: This is a subcommand of ``apt`` used to install new packages on your system.

4. ``ros-noetic-desktop-full``:

\* This is the name of the package you want to install. It represents the "full desktop" installation of ROS Noetic, a specific version of the Robot Operating System. Let's break it down further:

\* ``ros``: This prefix indicates that it's a ROS package.

\* ``noetic``: This refers to the specific ROS distribution version (Noetic Ninjemys).

\* ``desktop-full``: This indicates that it's the complete desktop environment, including tools for development, visualization, simulation, and more.

## Install a Specific Package Directly:



```
sudo apt install ros-noetic-slam-gmapping
```

install the `ros-noetic-slam-gmapping` package, which provides the GMapping SLAM (Simultaneous Localization and Mapping) algorithm within the ROS Noetic distribution.

1. `sudo`: This grants the command elevated privileges, allowing it to make changes that require root access. This is essential for installing software because it involves modifying system files and directories.
2. `apt`: This is the command-line tool used to manage software packages on Debian-based Linux systems like Ubuntu. It's your package manager, handling tasks like searching for, downloading, installing, upgrading, and removing software.
3. `install`: This is a subcommand of `apt` used to install new packages on your system.

#### 4. ``ros-noetic-slam-gmapping``:

This is the name of the specific package you want to install.

1 ``ros``: Indicates that it's a ROS package.

2 ``noetic``: Refers to the specific ROS distribution version (Noetic Ninjemys).

3 ``slam-gmapping``: This is the package name, signifying the GMapping SLAM algorithm package within ROS Noetic.

To find available packages, see [ROS Index](#) or use:



apt search ros-noetic

The command `apt search ros-noetic` is used to search for packages related to ROS (Robot Operating System) on your Ubuntu system.

```
essa@essa-VB: ~$ apt search ros-noetic
Sorting... Done
Full Text Search... Done
ros-noetic-abb-driver/focal 1.4.0-1focal.20230215.234845 amd64
  ROS-Industrial nodes for interfacing with ABB robot controllers.
ros-noetic-abb-driver-dbgsym/focal 1.4.0-1focal.20230215.234845 amd64
  debug symbols for ros-noetic-abb-driver
ros-noetic-abb-egm-msgs/focal 0.5.2-1focal.20220926.184335 amd64
  Provides ROS message definitions, representing Externally Guided Motion (EGM)
  data from ABB robot controllers.
ros-noetic-abb-rapid-msgs/focal 0.5.2-1focal.20220926.182609 amd64
  Provides ROS message definitions, representing RAPID data from ABB robot contr
  ollers.
ros-noetic-abb-rapid-sm-addin-msgs/focal 0.5.2-1focal.20220926.183002 amd64
  Provides ROS message and service definitions, representing interaction with AB
  B robot controllers using the RobotWare StateMachine Add-In.
ros-noetic-abb-robot-msgs/focal 0.5.2-1focal.20220926.184343 amd64
  Provides ROS message and service definitions, representing basic interaction w
  ith ABB robot controllers.
```

\* `apt`: This is the Advanced Packaging Tool, Ubuntu's primary package manager. It's responsible for installing, updating, and managing software packages.

\* `search`: This command tells `apt` to search for packages in the Ubuntu repositories that match your query.

\* `ros-noetic`: This is the specific version of ROS you are looking for.

- ros-noetic-ros-base - ROS Base Packages
- ros-noetic-ros-base-core - Core ROS Packages
- ros-noetic-ros-base-components - ROS Components
- ros-noetic-ros-base-devel - ROS Base Development Packages
- ros-noetic-ros-base-doc - ROS Base Documentation
- ...

# Redirect the output of `apt search ros-noetic` to a file:



1- using the redirection operator `>` in your terminal. It takes the output of a command and writes it to a file.

```
sudo apt search ros-noetic > output.txt
```

`output.txt` is the name of the file you want to create or overwrite. a file named `output.txt` will be created in your current directory.

2- Append to a file: Use `>>` instead of `>`.

3- Pipe to a command: You can use the pipe operator `|` to pass the output of `apt search` to another command, like `grep` to filter the results. For example:

```
sudo apt search ros-noetic | grep "navigation" > navigation_packages.txt
```



# Environment setup



You must source this script in every **bash** terminal you use ROS in.

bash

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

zsh

```
echo "source /opt/ros/noetic/setup.zsh" >> ~/.zshrc
source ~/.zshrc
```

# Dependencies for building packages



```
sudo apt install python3-rosdep python3-rosinstall python3-rosinstall-generator python3-wstool  
build-essential
```

To create and manage your own ROS workspaces, there are various tools and requirements that are distributed separately. To install this tool and other dependencies for building ROS packages, run:

## initialize rosdep:

- `sudo rosdep init`
- `rosdep update`

### 1. `sudo rosdep init`

This command initializes the `rosdep` tool, which is essential for managing system dependencies for ROS packages.

- Creates a configuration file (`/etc/ros/rosdep/sources.list.d/20-default.list`) that specifies where `rosdep` should look for information about dependencies.
- It also creates a directory structure for storing dependency information (`/etc/ros/rosdep`).
- Why you need it: Before you can use `rosdep` to install dependencies for your ROS packages, you need to initialize it.
- It's a one-time setup step.

## 2. `rosdep update`

- This command updates the dependency information stored by `rosdep`.
- Downloads the latest dependency data from the ROS repositories.
- This data includes information on which system packages are required for various ROS packages.
- Why you need it: `rosdep` uses this data to determine the correct system packages needed when you install ROS packages using tools like `catkin\_make`.
- It ensures your system has the most up-to-date information on dependencies, preventing potential problems during installation or runtime.

## Single-line installation and uninstall



**Single-line installation** The following line of command will install the latest ROS Noetic Ninjemys on Ubuntu Focal 20.04. Copy & Paste this line of command into the Ubuntu terminal.

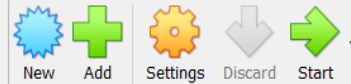
```
wget -c https://raw.githubusercontent.com/qboticslabs/ros_install_noetic/master/ros_install_noetic.sh && chmod +x ./ros_install_noetic.sh && ./ros_install_noetic.sh
```

The following line of command **will uninstall ROS Noetic Ninjemys**

```
wget -c https://raw.githubusercontent.com/qboticslabs/ros_install_noetic/master/ros_uninstall_noetic.sh && chmod +x ./ros_uninstall_noetic.sh && ./ros_uninstall_noetic.sh
```



Tools



64 ubuntu-20.04.5-focal fossa  
Powered Off

- Settings... Ctrl+S
- Clone... Ctrl+O
- Move...
- Export to OCI...
- Remove...
- Move to Group
- Start
- Pause
- Reset
- Stop
- Discard Saved State...
- Show Log... Ctrl+L
- Refresh
- Show in Explorer
- Create Shortcut on Desktop
- Sort
- Search Ctrl+F

**General**

Name: ubuntu-20.04.5-focal fossa  
Operating System: Ubuntu (64-bit)

**System**

Base Memory: 10025 MB  
Processors: 6  
Execution Cap: 54%  
Boot Order: Floppy, Optical, Hard Disk  
Acceleration: Nested Paging, KVM Paravirtualization

**Display**

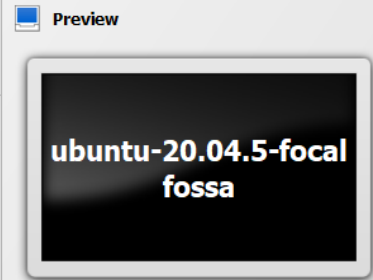
Video Memory: 60 MB  
Graphics Controller: VMSVGA  
Remote Desktop Server: Disabled  
Recording: Disabled

**Storage**

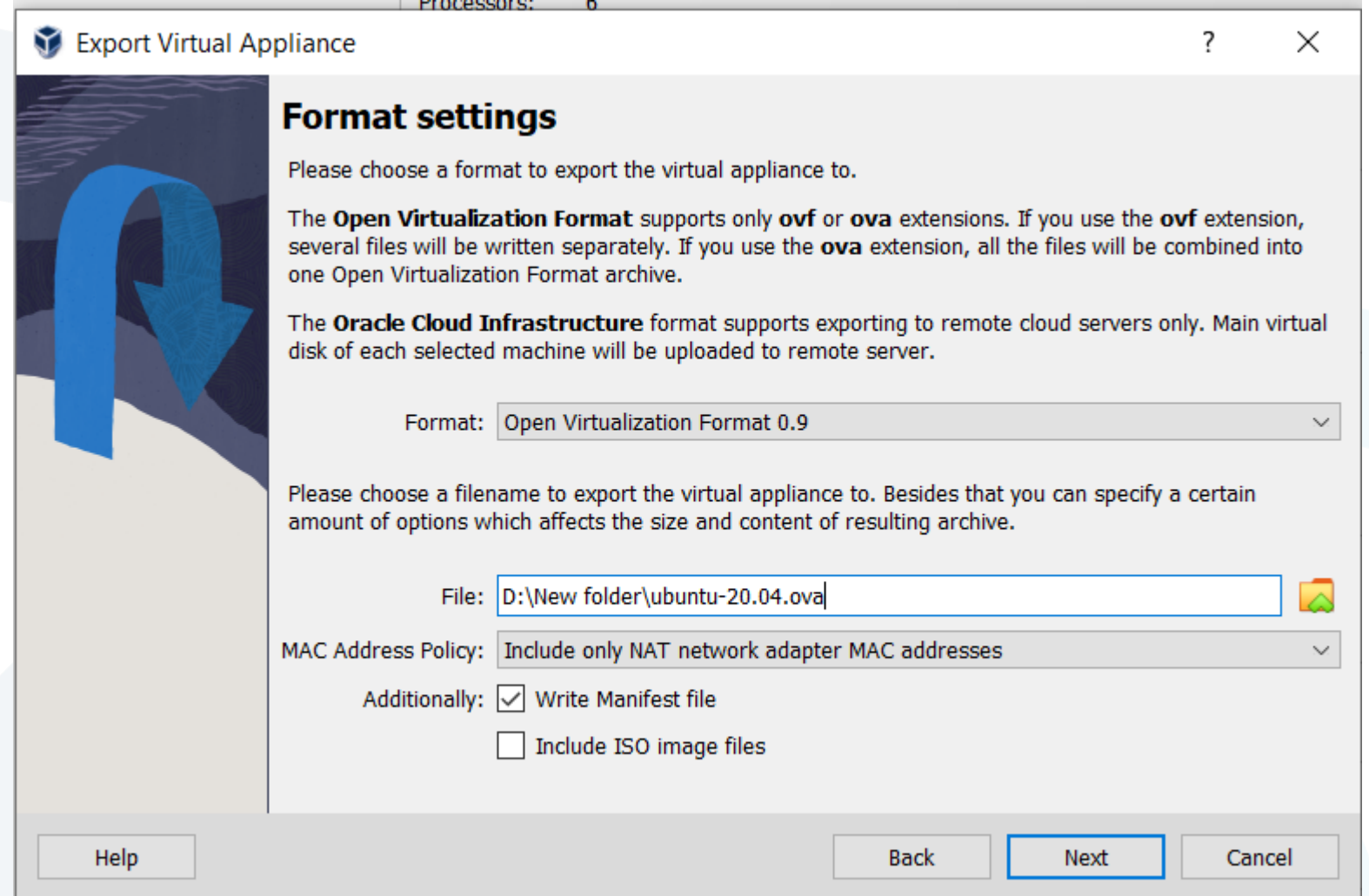
Controller: IDE  
IDE Secondary Device 0: [Optical Drive] Empty  
Controller: SATA  
SATA Port 0: ubuntu-20.04.5-focal fossa.vdi (Normal, 100.00 GB)

**Audio**

Host Driver: Default  
Controller: ICH AC97



The Open Virtualization Format (OVF) is just a container for Virtual Machine appliance. It literally is just a compressed file that includes a Virtual Hard Drive image (VDI, VHD, VMDK, or other) for the VM appliance, and some VM definition files.
















Export Virtual Appliance

## Appliance settings

This is the descriptive information which will be added to the virtual appliance. You can change it by double clicking on individual lines.

Virtual System 1

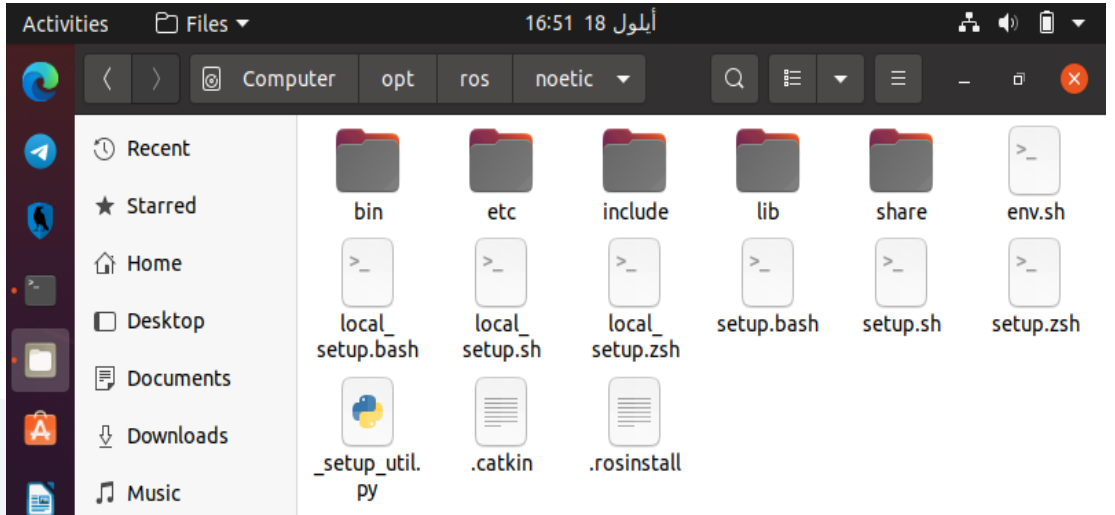
	Name	ubuntu-20.04.5-focal fossa
	Product	
	Product-URL	
	Vendor	
	Vendor-URL	
	Version	
	Description	
	License	
	Guest OS Type	Ubuntu (64-bit)
	CPU	6
	RAM	10025 MB
	USB Controller	<input checked="" type="checkbox"/>
	...	...

Help Back Finish Cancel



\$ rosversion -d  
noetic

# printenv | grep ROS

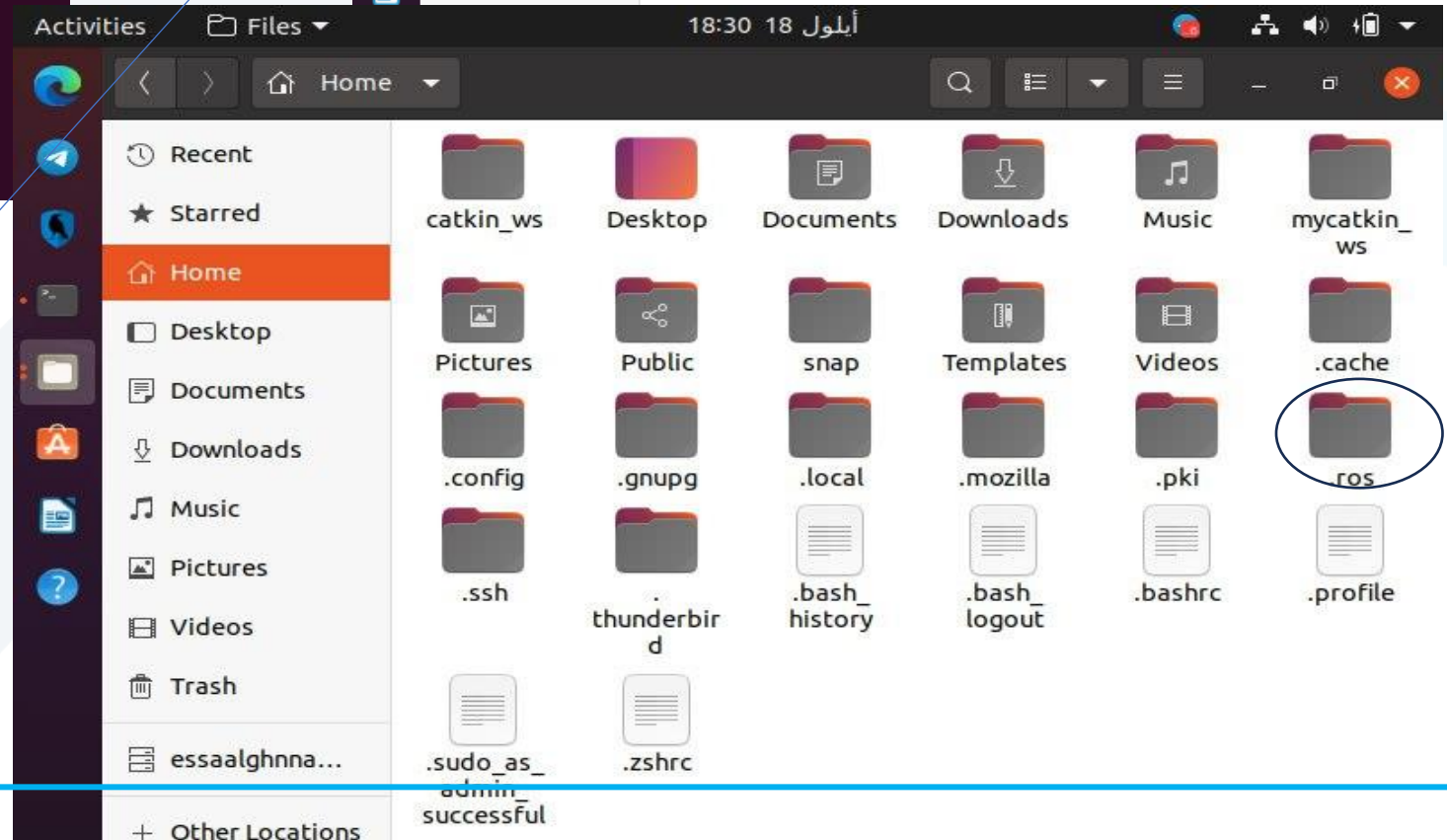


```
essa@essa:~/Desktop$ printenv | grep ROS
ROS_VERSION=1
ROS_PYTHON_VERSION=3
ROS_PACKAGE_PATH=/opt/ros/noetic/share
ROSLISP_PACKAGE_DIRECTORIES=
ROS_ETC_DIR=/opt/ros/noetic/etc/ros
ROS_MASTER_URI=http://localhost:11311
ROS_ROOT=/opt/ros/noetic/share/ros
ROS_DISTRO=noetic
essa@essa:~/Desktop$
```

If they are not then you might need to 'source' some setup.\*sh files.

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

- printenv: print environment variables
- | the pipe operator. It takes the output of the command on the left and sends it as input to the command on the right.
- grep: filter the ones which start with ROS



# Understanding Port 11311



```
ROS_MASTER_URI=http://localhost:11311
```

- `ROS_MASTER_URI` is This environment variable. it specifies the network address of the ROS Master.
- The Master is responsible for registering all the different nodes and facilitating communication between them.
- `http://`: This part specifies that the URI is using the HTTP protocol. The ROS Master is accessed over HTTP.
- `localhost`: This refers to the local machine. It tells ROS to look for the Master on the same machine where the command is run. If you were using a different machine, you would replace `localhost` with the IP address or hostname of that machine.
- `11311`: This is the default port on which the ROS Master listens for incoming connections.

# Tutorials

- <https://wiki.ros.org/ROS/Tutorials>

شكرا لحسن الاصغاء