

Resource R1 Setting up the development environment

A development environment for ROS can be installed on many personal computers and operating systems. In this text, we use the following stack of software for our development environment.

Resource R1.6 VirtualBox

virtual machines VirtualBox, by Oracle, is a free virtualizer that can install a *virtual machine* with a variety of operating systems installed. VirtualBox is cross-platform and can be installed on most modern operating systems (e.g. Windows, MacOS, Linux). This allows a host computer (your PC) with (say) Windows to run a virtual machine with (say) Linux—simultaneously!

Box 05.1 if your computer is resource-challenged

ROS can be resource-intensive, especially when running simulations. If your personal computer is resource-challenged, especially in RAM and processing, consider forgoing VirtualBox and installing Ubuntu in dual-boot mode. More on that in a moment.

Download and install the latest VirtualBox app for your host computer:

[virtualbox.org/wiki/Downloads](https://www.virtualbox.org/wiki/Downloads).

For greater functionality, consider installing the Extension Pack from the same downloads page.

Resource R1.7 Ubuntu Bionic (18.04.4) LTS

This popular Linux distribution is fully compatible with the version of ROS we will use, has a long-term maintenance schedule (LTS), is lightweight, and is free. Download the Desktop (64-bit) version here:

releases.ubuntu.com/18.04.4.

We will install this operating system as a virtual machine with VirtualBox.

Box 05.2 if your computer is resource-challenged: dual-boot

If you decide you need to dual-boot, skip Resource [Lecture R1.7.1](#) and see the installation guide:

help.ubuntu.com/lts/installation-guide/amd64.

The official installation guide above may not have some Windows 10-specific instructions; as a supplement, see:

tecmint.com/install-ubuntu-alongside-with-windows-dual-boot.

Resource R1.7.1 Preparing a virtual machine

First, we must prepare a virtual machine with VirtualBox. Use the following steps:

1. In the VirtualBox app, create a virtual machine. Name it (say) `ubuntu18`, select `Linux` and `Ubuntu (64-bit)`.
2. Allocate RAM of at least 4096 MB.
3. Select "Create a virtual hard disk now".
4. Select "VDI (VirtualBox Disk Image)".
5. Select "Dynamically allocated".
6. Allocate a maximum of 15–20 GB of virtual hard disk.
7. Select the new virtual machine `ubuntu18` and .
8. When prompted, select as the "virtual optical drive" the downloaded `Ubuntu .iso` file and .

Resource R1.7.2 Install Ubuntu

Pay special attention to the following during the Ubuntu installation process.

- .
- Download updates while installing Ubuntu.
- Install third-party software for graphics and Wi-Fi hardware, Flash, MP3 and other media.
- Erase disk and install Ubuntu.²

The Ubuntu Bionic documentation is available here:

help.ubuntu.com/lts/ubuntu-help.

Resource R1.8 Ubuntu VirtualBox Guest Additions

Installing VirtualBox Guest Additions *in Ubuntu* should improve the performance of your virtual machine. I recommend the following setup process.

1. Open a Terminal window.
2. Update your package manager.

```
sudo apt-get update
```

²If you're *not* installing to a virtual machine, be cautious here!

3. Install Ubuntu VirtualBox Guest Additions.

```
sudo apt-get install virtualbox-guest-additions-iso
```

4. Install VirtualBox Guest Utilities.

```
sudo apt-get install virtualbox-guest-utils
```

5. Restart the virtual machine.

Resource R1.8.1 Add a shared folder

Your virtual machine can mount a folder on your host machine such that files can be easily shared between them. In VirtualBox, in your virtual machine's **Settings** > **Shared Folders**, click the button that adds a new shared folder. Give it the path of the shared folder on the host machine and do auto-mount. This directory should now be available in the virtual machine as a mounted drive.

Resource R1.9 ROS Melodic Morenia

We will install the recent version ROS Melodic Morenia to our Ubuntu Bionic virtual machine. Follow the official instructions:

wiki.ros.org/melodic/Installation/Ubuntu.

Terminal It assumes you will enter the given commands in the *Terminal* in Ubuntu, which can be opened through the GUI or with the keyboard shortcut **Ctrl** + **Alt** + **T**.

Follow the recommended options and be sure to:

- install the “Desktop-Full” version;
- under “Environment setup,” follow the instructions to source your environment variables in your dotfile `~/.bashrc`; and
- under “Dependencies for building packages,” install the recommended tools, listed in the provided command.

Consider working through the first four ROS tutorials:

wiki.ros.org/ROS/Tutorials.

Use the recommended `catkin` option.

Resource R1.10 Python

Python 2 We will write most of our ROS code in *Python 2*. It's best practice not to mess with Ubuntu's Python installation and instead install our own.

The package `pyenv`³ will help us manage what will be multiple Python versions. Installing `pyenv` is easy in a Terminal. pyenv

```
curl https://pyenv.run | bash
```

Be sure to open your `~/.bashrc` file (e.g. `gedit ~/.bashrc`) and add the following lines.

```
export PATH="/home/picone/.pyenv/bin:$PATH"
eval "$ (pyenv init -)"
eval "$ (pyenv virtualenv-init -)"
```

To finalize the installation, either open a new Terminal or `exec $SHELL` (this will reload your `.bashrc` so that `pyenv` is available to the bash shell). Now create a fresh Python 2 installation using `pyenv`.

```
pyenv install 2.7.17
```

Now we can list installed Python versions with the following command.

```
pyenv versions
```

```
| system
| 2.7.17
```

To set the global default Python version, use the following.

```
pyenv global 2.7.17
```

You can also set local Python environments using `pyenv local`, which sets the environment in the current and sub-directories.

³See github.com/pyenv/pyenv for documentation.

