

Building ROOT and uproot with Python3 Compatibility

Prerequisites:

Make sure the following are installed:

1. A C++ compiler (e.g., g++)
2. python3, python3-dev (recommended version: 3.8 or above)
3. pip for Python packages
4. CMake (for source installation)
5. Git (for source or cloning)

You can use any one of the following options based on your system and preferences:

Option 1: # For Debian/Ubuntu via apt installer

```
$ sudo apt update
$ sudo apt install build-essential python3 python3-dev
python3-pip cmake git # prerequisite, not needed if already
installed

$ sudo apt install root-system
```

Option 2: # For Fedora/RHEL/CentOS

```
$ sudo dnf install gcc-c++ python3 python3-devel python3-pip
cmake git # (prerequisite, not needed if already installed)
$ sudo dnf install root
```

Option 3: # Compile from Source (All Linux platforms, and more control)

This method gives you the latest version and custom control.

a. Download source file and untar it: (You can replace v6-34-08 with another version if desired.)

```
$ wget https://root.cern/download/root_v6.34.08.source.tar.gz
```

```
$ tar -zxf root_v6.34.08.source.tar.gz
$ ls root-6.34.08 # (check if the folder exists after
extraction)
```

b. Create build directory:

```
$ mkdir root_build
$ cd root_build
```

c. Configure with CMake: (You can change ~/opt/root to another install location if desired.)

```
$ cmake ../root-6.34.08 \
  -DCMAKE_INSTALL_PREFIX=~/opt/root \
  -Dpython=ON \
  -Droofit=ON \
  -Dminuit2=ON \
  -Dx11=ON
```

d. Compile and install:

```
$ make -j 4 # (build with 4 parallel jobs. You can change to
something of your choice)
$ make install
```

Option 4: # Install from Binary Releases (No compilation required). Precompiled binary files are available only for a limited cases.

a. Check OS, architecture, and gcc version:

```
$ cat /etc/os-release
$ uname -m
$ gcc --version
```

b. Download from ROOT website:

Go to: <https://root.cern/install>

Choose a binary for your OS, architecture, and gcc version (e.g.,
root_v6.34.08.Linux-ubuntu22.04-x86_64-gcc11.4.tar.gz)

c. Extract and setup:

```
$ tar -xzf
https://root.cern/download/root_v6.34.08.Linux-ubuntu22.04-x86_64-gcc11.4.tar.gz
$ mv root ~/opt/root # (move the extracted folder to you
preferred installation location)
```

Option 5: Installation using conda.

a. Set conda environment and install:

```
$ conda create -n root_env -c conda-forge root
$ conda activate root_env
```

Final Step: Set Environment Variables (for **Option 3** and **Option 4**)

For any non-standard installation (source or binary), you must configure your shell to find ROOT.

If you installed in ~/opt/root:

Add the following line to your ~/.bashrc (for Bash) or ~/.zshrc (for Zsh):

```
source ~/opt/root/bin/thisroot.sh
```

Then reload your shell:

```
$ source ~/.bashrc
or
$ source ~/.zshrc
```

uproot Installation:

```
$ pip install uproot
or
$ conda install -c conda-forge uproot
```

awkward Installation:

```
$ pip install awkward
or
$ conda install -c conda-forge awkward
```

Now test:

From shell:

```
$ root
```

For Python:

```
$ python3
>>> import ROOT
>>> import uproot
>>> import awkward
>>> import ROOT.RDataFrame as RDF
```

(For RDataFrame, ROOT version has to be above 6.26)