

Installation guide of SciFiMatG4_v2 for Ubuntu

Data and information:

https://gitlab.cern.ch/bleverin/SciFiMatG4_v2

Disclaimer:

- Changes and additions to *readme.me* from Blake are marked in green
- Please excuse typing errors in terminal commands

1. Needed software in general (like gcc, the c++ compiler)

- `sudo apt-get update && sudo apt-get dist-upgrade`
//upgrades Ubuntu to 20.04.02 (or latest available version)
- `sudo apt install build-essential`
//this is for example gcc, check with 'gcc --version' for at least 4.9.3
- `sudo apt-get install libxerces-c-dev qt5-default freeglut3-dev libmotif-dev tk-dev cmake libxpm-dev libxmu-dev libxi-dev libqt5opengl5-dev libqt4-dev cmake libx11-dev xorg-dev libglu1-mesa-dev freeglut3-dev libglew1.5 libglew1.5-dev libglu1-mesa-dev libglu1-mesa-glx libgl1-mesa-dev libfontconfig1-dev libfreetype6-dev libx11-dev libxcursor-dev libxext-dev libxfixes-dev libxft-dev libxi-dev libxrandr-dev libxrender-dev libxmu-dev expat libexpat1-dev freeglut3 freeglut3-dev mesa-utils`
// optional: 'sudo apt-get install qt5-qmake qtbase5-dev-tools qt5-doc'

2. CMake latest version from source

- `cmake --version`
- `sudo apt remove cmake`
- // download latest bash script from <https://cmake.org/download/>, here `cmake-3.21.0-rc1-linux-x86_64.sh`, copy to folder you want e.g. `/home/path/to/your/folder/cmake-3.21.0/`
- `chmod +x /home/path/to/your/folder/cmake-3.21.0/cmake-3.21.0-rc1-linux-x86_64.sh`
- `sudo bash /home/path/to/your/folder/cmake-3.21.0/cmake-3.21.0-rc1-linux-x86_64.sh`
// need to press y twice
- `sudo ln -s /home/path/to/your/folder/cmake-3.21.0-rc1-linux-x86_64/bin/* /usr/local/bin`
//creates symbolic link
- `cmake --version` // for selfcontrol
- // Set environment variables in .bashrc (is found in /home/ with hidden files ON) add following lines:
 - `export PATH=/home/.../cmake-3.21.0-rc1-linux-x84_64/bin:$PATH`

3. CLHEP from source

- // Installation guide: <https://proj-clhep.web.cern.ch/proj-clhep/INSTALLATION/clhep-2.0.html>
- // download page: <https://proj-clhep.web.cern.ch/proj-clhep/clhep23.html>
- // unzip latest clhep to folder you want e.g. `/home/path/to/your/folder/clhep/`; in my case `CLHEP_2.4.4.2`
- `cd /home/path/to/your/folder/clhep/`
- `mkdir build install`
- `cd build`
- `cmake -DCMAKE_INSTALL_PREFIX= /home/path/to/your/folder/clhep/install /home/path/to/your/folder/clhep/CLHEP_2.4.4.2`
- `make -j2` // 2 is number of processor cores, therefore 4, 6 or 8 could also be possible
- `make test` // optional
- `make install`
- // Set environment variables in .bashrc (is found in /home/ with hidden files ON) add following lines:
 - `export CLHEP_DIR=/home/path/to/your/folder/clhep/install`
 - `export CLHEP_INCLUDE_DIR=${CLHEP_DIR}/include/`
 - `export CLHEP_LIBRARY=${CLHEP_DIR}/lib/`
 - `# export LD_LIBRARY_PATH=${CLHEP_LIBRARY}:${LD_LIBRARY_PATH}`
//found in install guide
<https://wwwcompass.cern.ch/compass/software/offline/TGeant/TGeantOldPage/na58-project-tgeant.web.cern.ch/content/step-step-installation-ubuntu.html>, but created problems later → hashed out
 - `export PATH=$CLHEP_DIR/bin:$PATH`

4. Root from source

- // restart terminal to activate new lines in .bashrc
- `cd /home/path/to/your/folder/root` //after creating a own folder for your root install
- `git clone --branch latest-stable https://github.com/root-project/root.git root_src`
// if git is missing (for Debian System): `sudo apt install git-all`
- `mkdir build-cxx17 install-cxx17`
- `cd build-cxx17`

- `cmake -Dpyroot=ON -Drofit=ON`
`-CMAKE_INSTALL_PREFIX:PATH=/home/path/to/your/folder/root/install-cxx17`
`-DCMAKE_CXX_STANDARD=17 ../root_src`
- `cmake --build . -- -j2`
- `root-config --features`
- `// terminal output should be:`
`cxx17 asimage builtin_afterimage builtin_clang builtin_gl2ps builtin_llvm builtin_lz4`
`builtin_nlohmannjson builtin_openui5 builtin_tbb builtin_vdt builtin_xrootd builtin_xxhash builtin_zstd`
`clad dataframe exceptions fftw3 fitsio gdml http imt mathmore mlp mysql opengl pyroot roofit webgui`
`root7 runtime_cxxmodules shared ssl tmva tmva-cpu tmva-pymva spectrum vdt x11 xml xrootd`
- `// Set environment variables in .bashrc (is found in /home/ with hidden files ON) add following lines:`
 - `source /home/path/to/your/folder/root/install-cxx17/bin/thisroot.sh`

5. Geant4 from source

- `// restart terminal to activate new lines in .bashrc`
- `// download page: https://geant4.web.cern.ch/support/download`
- `// unzip latest Geant4 to folder you want e.g. /home/path/to/your/folder/Geant4/ ; in my case`
`geant4.10.07.p02`
- `cd /home/path/to/your/folder/Geant4/`
- `mkdir geant4-build geant4-install`
- `cd geant4-build`
- `cmake -DCMAKE_INSTALL_PREFIX=/home/path/to/your/folder/Geant4/geant4-install -`
`DGEANT4_INSTALL_DATA=ON -DGEANT4_BUILD_MULTITHREADED=ON -`
`DGEANT4_USE_OPENGL_X11=ON -DGEANT4_USE_QT=ON -DGEANT4_BUILD_CXXSTD=17 -`
`DGEANT4_USE_FREETYPE=ON -DGEANT4_USE_SYSTEM_CLHEP_GRANULAR=ON ../geant4.10.07.p02`
- `// last lines in terminal should be:`
 - `-- Configuring done`
 - `-- Generating done`
 - `-- Build files have been written to: /path/to/geant4-build`
- `make -j2`
- `make install`
- `// Optional:`
`// Set environment variables in .bashrc (is found in /home/ with hidden files ON) add following lines:`
 - `source /home/path/to/your/folder/Geant4/geant4-install/bin/geant4.sh`

6. SciFiMatG4_v2 install

- `// download page: https://gitlab.cern.ch/bleverin/SciFiMatG4_v2`
- `// unzip to folder you want e.g. /home/path/to/your/folder/SciFiMatG4_v2/`
- `cd /home/path/to/your/folder/SciFiMatG4_v2/`
- `// edit the environment script setup.sh to point to your Geant4, Geant4 data, ROOT and clhep`
`installation:`
 - `source /home/path/to/your/folder/Geant4/geant4-install/share/Geant4-`
`10.7.2/geant4make/geant4make.sh`
 - `source /home/path/to/your/folder/root/install-cxx17/bin/thisroot.sh`
 - `G4DATA=/home/path/to/your/folder/Geant4/geant4-install/share/Geant4-10.7.2/data`
 - `export CLHEP_DIR=/home/path/to/your/folder/clhep/install`
- `source setup.sh`
`//required before running the software too if you open a new terminal`
`// OR add 'source /home/path/to/your/folder/SciFiMatG4_v2/setup.sh' to .bashrc`
- `mkdir -p build`
`//it is safe to delete this folder if you have problems during compilation`
- `cd build`
- `cmake ../SciFiSim`
- `cmake --build . // or make clean; make`
- `Run GUI:`
- `cd ../simulationData`
- `../build/scifiSim`