

# Geant4 Installation/Usage on Linux

(Ubuntu 16.04 Recommended)

Emily Tsai

12/20/17

# Background

- Geant4 is a simulation program that can be used to model the passage of particles through matter
- Covers physics processes such as: electromagnetic, hadronic, decay, optical
  - Long and short-lived particles
  - Energy ranging from tens of eV to trillions of eV
- It is used in high energy physics (ATLAS, ALICE, Fermilab, etc.), space and radiation, and medicine

# Installation (on Linux)

- Installing dependencies for Geant4

- `sudo apt-get update`
- `sudo apt-get upgrade`
- `sudo apt-get install libx11-dev libxmu-dev mesa-common-dev libglu1-mesa-dev -y libqt4-dev libxerces-c-dev cmake`

```
Setting up libxfixes-dev:amd64 (1:5.0.1-2) ...
Setting up x11proto-damage-dev (1:1.2.1-2) ...
Setting up libxdamage-dev:amd64 (1:1.1.4-2) ...
Setting up libxext-dev:amd64 (2:1.3.3-1) ...
Setting up x11proto-xf86vidmode-dev (2.3.1-2) ...
Setting up libxf86vm-dev:amd64 (1:1.1.4-1) ...
Setting up x11proto-dri2-dev (2.8-2) ...
Setting up x11proto-gl-dev (1.4.17-1) ...
Setting up libgl1-mesa-dev:amd64 (17.0.7-0ubuntu0.16.04.2) ...
Setting up libglu1-mesa-dev:amd64 (9.0.0-2.1) ...
Setting up libice-dev:amd64 (2:1.0.9-1) ...
Setting up libicu-dev:amd64 (55.1-0ubuntu0.3) ...
Setting up libqt4-designer:amd64 (4:4.8.7+dfsg-5ubuntu2) ...
Setting up libqt4-qtsupport:amd64 (4:4.8.7+dfsg-5ubuntu2) ...
Setting up libqt4-dev-bin (4:4.8.7+dfsg-5ubuntu2) ...
Setting up libqt4-help:amd64 (4:4.8.7+dfsg-5ubuntu2) ...
Setting up libqt4-scripttools:amd64 (4:4.8.7+dfsg-5ubuntu2) ...
Setting up libqt4-svg:amd64 (4:4.8.7+dfsg-5ubuntu2) ...
Setting up libqt4-test:amd64 (4:4.8.7+dfsg-5ubuntu2) ...
Setting up qt4-linguist-tools (4:4.8.7+dfsg-5ubuntu2) ...
Setting up qt4-qmake (4:4.8.7+dfsg-5ubuntu2) ...
Setting up libqt4-dev (4:4.8.7+dfsg-5ubuntu2) ...
Setting up libqt4-opengl-dev:amd64 (4:4.8.7+dfsg-5ubuntu2) ...
Setting up libqt4-opengl-dev (4:4.8.7+dfsg-5ubuntu2) ...
Setting up libsm-dev:amd64 (2:1.2.2-1) ...
Setting up libx11-doc (2:1.6.3-1ubuntu2) ...
Setting up libxerces-c3.1:amd64 (3.1.3+debian-1) ...
Setting up libxerces-c-dev (3.1.3+debian-1) ...
Setting up libxt-dev:amd64 (1:1.1.5-0ubuntu1) ...
Setting up libxmu-headers (2:1.1.2-2) ...
Setting up libxmu-dev:amd64 (2:1.1.2-2) ...
Processing triggers for libc-bin (2.23-0ubuntu9) ...
mango@mango:~$ █
```

# Installation (on Linux)

- Installing Geant4:

- Download source file from <http://geant4.web.cern.ch/geant4/support/download.shtml>
- Make a Geant4 directory (`mkdir Geant4`)
  - Moving the Geant4 package from Downloads to the Geant4 folder: `mv -v ~/path/Downloads/geant4.10.04.tar.gz ~/path/Geant4`
- Unpack source file in Geant4 directory (`tar -xvf geant4.10.04.tar.gz`)

```
mango@mango-VirtualBox:~/Geant4$ ls  
geant4.10.04  geant4.10.04.tar.gz
```

- If the Geant4 version changes, change the terminal commands to match the downloaded version of Geant4 (ex: `geant4.10.03.p03`, `geant4.10.05`)
- Make a `geant4-install` and `geant4-build` directory inside Geant4 directory: `mkdir geant4-install` and `mkdir geant4-build` (should have `geant4.10.04`, `geant4-install`, and `geant4-build` directories)

```
mango@mango-VirtualBox:~/Geant4$ ls  
geant4.10.04  geant4-build  geant4-install
```

# Installation (on Linux)

- Installing Geant4:

- In the geant4-build directory (`cd geant4-build`):  
`cmake -DCMAKE_INSTALL_PREFIX=~/path/Geant4/geant4-install  
-DGEANT4_INSTALL_DATA=ON -DGEANT4_USE_GDML=ON -DGEANT4_USE_QT=ON -DGEANT4_USE_OPENGL_X11=ON  
-DGEANT4_USE_RAYTRACER_X11=ON -DGEANT4_USE_SYSTEM_EXPAT=OFF ~/path/Geant4/geant4.10.04`
  - When using additional flags (from section 2.3.1 of the [Geant4 Installation Guide](#)), you have to run `cmake`, `make`, and `make install` again

```
-- Using unsigned short
-- Check if the system is big endian - little endian
-- Looking for off_t
-- Looking for off_t - not found
-- Looking for size_t
-- Looking for size_t - not found
-- Check size of off64_t
-- Check size of off64_t - done
-- Looking for fseeko
-- Looking for fseeko - found
-- Looking for unistd.h
-- Looking for unistd.h - found
-- Configuring download of missing dataset G4NDL (4.5)
-- Configuring download of missing dataset G4EMLOW (7.3)
-- Configuring download of missing dataset PhotonEvaporation (5.2)
-- Configuring download of missing dataset RadioactiveDecay (5.2)
-- Configuring download of missing dataset G4NEUTRONKS (1.4)
-- Configuring download of missing dataset G4PII (1.3)
-- Configuring download of missing dataset RealSurface (2.1)
-- Configuring download of missing dataset G4SAIDDATA (1.1)
-- Configuring download of missing dataset G4ABLA (3.1)
-- Configuring download of missing dataset G4ENSDFSTATE (2.2)
-- The following Geant4 features are enabled:
GEANT4_BUILD_CXXSTD: Compiling against C++ Standard '11'
GEANT4_USE_GDML: Building Geant4 with GDML support
GEANT4_USE_QT: Build Geant4 with Qt support
GEANT4_USE_RAYTRACER_X11: Build RayTracer driver with X11 support
GEANT4_USE_OPENGL_X11: Build Geant4 OpenGL driver with X11 support

-- Configuring done
-- Generating done
-- Build files have been written to: /home/mango/Geant4
mango@mango-VirtualBox:~/Geant4$
```

# Installation (on Linux)

- Installing Geant4:

- **make** (takes a very long time to complete)

```
[100%] Building CXX object source/visualization/OpenGL/CMakeFiles/G4OpenGL.dir/src/G4OpenGLVboDrawer.cc.o
[100%] Building CXX object source/visualization/OpenGL/CMakeFiles/G4OpenGL.dir/src/G4OpenGLQtViewer.cc.o
[100%] Building CXX object source/visualization/OpenGL/CMakeFiles/G4OpenGL.dir/src/G4OpenGLStoredQt.cc.o
[100%] Building CXX object source/visualization/OpenGL/CMakeFiles/G4OpenGL.dir/src/G4OpenGLStoredQtSceneHandler.cc.o
[100%] Building CXX object source/visualization/OpenGL/CMakeFiles/G4OpenGL.dir/src/G4OpenGLStoredQtViewer.cc.o
[100%] Building CXX object source/visualization/OpenGL/CMakeFiles/G4OpenGL.dir/include/moc_G4OpenGLQtExportDialog.cxx.o
.
[100%] Building CXX object source/visualization/OpenGL/CMakeFiles/G4OpenGL.dir/include/moc_G4OpenGLQtMovieDialog.cxx.o
[100%] Building CXX object source/visualization/OpenGL/CMakeFiles/G4OpenGL.dir/include/moc_G4OpenGLQtViewer.cxx.o
[100%] Linking CXX shared library ../../BuildProducts/lib/libG4OpenGL.so
[100%] Built target G4OpenGL
mango@mango-VirtualBox:~/Geant4$
```

- **make install**

```
-- Installing: /home/mango/Geant4/geant4-install/include/Geant4/G4OpenGLVboDrawer.hh
-- Installing: /home/mango/Geant4/geant4-install/include/Geant4/G4OpenGLImmediateQt.hh
-- Installing: /home/mango/Geant4/geant4-install/include/Geant4/G4OpenGLImmediateQtViewer.hh
-- Installing: /home/mango/Geant4/geant4-install/include/Geant4/G4OpenGLQt.hh
-- Installing: /home/mango/Geant4/geant4-install/include/Geant4/G4OpenGLQtExportDialog.hh
-- Installing: /home/mango/Geant4/geant4-install/include/Geant4/G4OpenGLQtMovieDialog.hh
-- Installing: /home/mango/Geant4/geant4-install/include/Geant4/G4OpenGLVboDrawer.hh
-- Installing: /home/mango/Geant4/geant4-install/include/Geant4/G4OpenGLQtViewer.hh
-- Installing: /home/mango/Geant4/geant4-install/include/Geant4/G4OpenGLStoredQt.hh
-- Installing: /home/mango/Geant4/geant4-install/include/Geant4/G4OpenGLStoredQtSceneHandler.hh
-- Installing: /home/mango/Geant4/geant4-install/include/Geant4/G4OpenGLStoredQtViewer.hh
mango@mango-VirtualBox:~/Geant4$
```

- I tested these instructions using a Linux virtual machine (VirtualBox VM)

# Sourcing geant4.sh

- Always do this before running Geant4 in a new terminal
  - Look in directory: `cd path/Geant4/geant4-install/bin`
  - In terminal, type: `source geant4.sh`
- Adding source geant4.sh to .profile
  - In home directory terminal, type: `emacs .profile` (`emacs` is a text editor)
  - Add this line to the end of `.profile`: `alias alias_name = "cd ~/path/geant4.10.04-install/bin; source geant4.sh; cd ~/path/Geant4"`
    - This will source `geant4.sh` and leave you in your Geant4 directory
- To use:
  - When you first open terminal (in home directory), type `source .profile`
  - Then, you can type `alias_name`

# Compiling and Running Example B1 (all examples)

- Move B1 to Geant4 directory (`mv -v ~/path/Geant4/geant4.10.04/examples/basic/B1 ~/path/Geant4`)
- `cd path/B1`
- `mkdir build`
- `cd build`
- `cmake ..`
  - If encountering an error, source `geant4.sh`

```
etsai@etsai-XPS-15-9550:~/Workspace/Geant4/B1/build$ cmake ..  
-- The C compiler identification is GNU 5.4.0  
-- The CXX compiler identification is GNU 5.4.0  
-- Check for working C compiler: /usr/bin/cc  
-- Check for working C compiler: /usr/bin/cc -- works  
-- Detecting C compiler ABI info  
-- Detecting C compiler ABI info - done  
-- Detecting C compile features  
-- Detecting C compile features - done  
-- Check for working CXX compiler: /usr/bin/c++  
-- Check for working CXX compiler: /usr/bin/c++ -- works  
-- Detecting CXX compiler ABI info  
-- Detecting CXX compiler ABI info - done  
-- Detecting CXX compile features  
-- Detecting CXX compile features - done  
-- Configuring done  
-- Generating done  
-- Build files have been written to: /home/etsai/Workspace/Geant4/B1/build  
etsai@etsai-XPS-15-9550:~/Workspace/Geant4/B1/build$ █
```

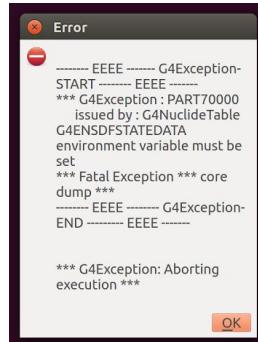
# Compiling and Running Example B1 (all examples)

- `make`

```
etsai@etsai-XPS-15-9550:~/Workspace/Geant4/B1/build$ make
Scanning dependencies of target exampleB1
[ 12%] Building CXX object CMakeFiles/exampleB1.dir/exampleB1.cc.o
[ 25%] Building CXX object CMakeFiles/exampleB1.dir/src/B1PrimaryGeneratorAction.cc.o
[ 37%] Building CXX object CMakeFiles/exampleB1.dir/src/B1DetectorConstruction.cc.o
[ 50%] Building CXX object CMakeFiles/exampleB1.dir/src/B1EventAction.cc.o
[ 62%] Building CXX object CMakeFiles/exampleB1.dir/src/B1SteppingAction.cc.o
[ 75%] Building CXX object CMakeFiles/exampleB1.dir/src/B1ActionInitialization.cc.o
[ 87%] Building CXX object CMakeFiles/exampleB1.dir/src/B1RunAction.cc.o
[100%] Linking CXX executable exampleB1
[100%] Built target exampleB1
etsai@etsai-XPS-15-9550:~/Workspace/Geant4/B1/build$
```

- How to run examples: `./exampleB1`

- If encountering this error when trying to run executables, fix by sourcing `geant4.sh`



- After making changes to an example, run `cmake` and `make` again to implement the changes

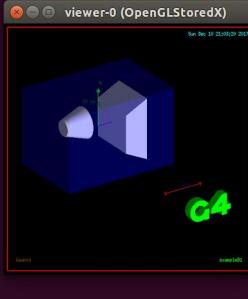
# Other Helpful Sources

- Geant4 Installation Guide:  
<http://geant4.web.cern.ch/geant4/UserDocumentation/UsersGuides/InstallationGuide/fo/BookInstalGuide.pdf>
- Geant4 Forum: <http://hypernews.slac.stanford.edu/HyperNews/geant4/cindex>

# Tutorials

- Geant4:  
<http://geant4.cern.ch/support/training.shtml>
- How to use the visualizer:  
<http://geant4.slac.stanford.edu/Presentations/vis/G4OpenGLTutorial/G4OpenGLTutorial.html>
- B1:  
[http://geant4.web.cern.ch/geant4/UserDocumentation/Doxygen/examples\\_doc/html/ExampleB1.htm](http://geant4.web.cern.ch/geant4/UserDocumentation/Doxygen/examples_doc/html/ExampleB1.htm)

```
etsai@etsai-XPS-15-9550: ~/Workspace/Geant4/B1-build
# Attach text to one corner of Shape2, with a small, fixed offset
/vis/scene/add/text 6 7 10 cm 18 4 4 Shape2
/vis/scene/notifyHandlers
#
# To get nice view
# Make the "World" box invisible
/vis/geometry/set/visibility World 0 false
/vis/scene/notifyHandlers
# "Envelope" is transparent blue to represent water
/vis/geometry/set/colour Envelope 0 0 0 1 .3
/vis/scene/notifyHandlers
/vis/viewer/set/style surface
/vis/viewer/set/hiddenMarker true
/vis/viewer/set/viewpointThetaPhi 120 150
#
# Re-establish auto refreshing and verbosity:
/vis/viewer/set/autoRefresh true
/vis/viewer/refresh
/vis/verbose warnings
Visualization verbosity changed to warnings (3)
#
# For file-based drivers, use this to create an empty detector view:
#/vis/viewer/flush
Idle> 
```



# Sources

- <http://geant4.web.cern.ch/geant4/applications/index.shtml>
- <https://web.fnal.gov/project/Geant4/SitePages/Home.aspx>
- <https://kkandyli.wordpress.com/2017/01/25/installation-guide-of-geant4/>