

Bardzo szczegółowa instrukcja instalacji pakietu geant4

Aleksandra Fijałkowska

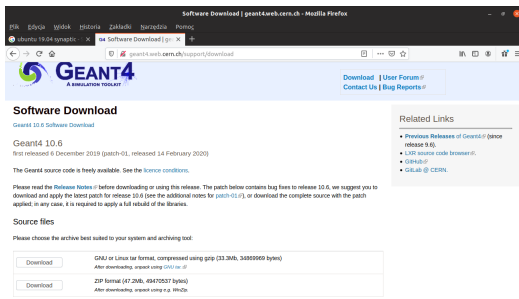
Wydział Fizyki, Uniwersytet Warszawski

aleksandra.fijalkowska@fuw.edu.pl

16 marca 2020

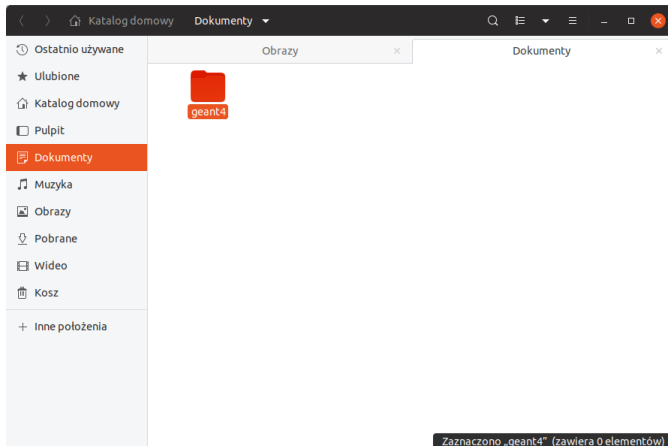
Aby w pełni doświadczyć trudów instalacji pakietu GEANT4 zainstalowałam go ponownie, na świeżo zainstalowanym systemie Ubuntu 19.04. Poniżej opisuję szczegółowo każdy krok, który wykonałam. Zainstalowana wersja to Geant4.10.06.p01 (najnowsza).

Źródła najnowszej wersji pakietu geant są do ściągnięcia na stronie <http://geant4.web.cern.ch/support/download>.

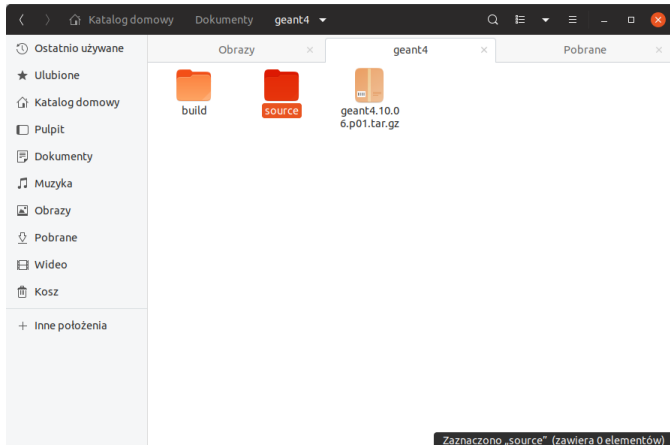


Aby zapisać wersję działającą na systemie Linux naciśnij przycisk *Download* znajdujący się obok *GNU or Linux tar format, compressed using gzip (33.3Mb, 34869969 bytes)*. Starsze wersje są dostępne po wybraniu linka z prawego menu Previous Releases of Geant4. Link prowadzi do strony zawierającej archiwalne wydania (https://geant4.web.cern.ch/support/download_archive)

Warto wygospodarować sobie jakieś szlachetne miejsce na źródła i pliki instalacyjne. Niech to nie będzie katalog „Pobrane”!



Źródło rozpakowujemy w wybranym katalogu. Powstały katalog będzie się nazywał geant4.10.06.p01. U siebie zmieniłam mu nawet na „source”. Ponadto stworzyłam katalog „build”, w którym zbuduję i zainstaluję geant'a.



Budowanie

Zanim zainstalujemy geant'a musimy wybrać interesujące nas opcje np. umożliwić pracę wielowątkową, albo określić biblioteki do wizualizacji. Opcje te można ustalić jako parametry wywołania programu cmake, albo skorzystać z ccmake, czyli prostego GUI do cmake.

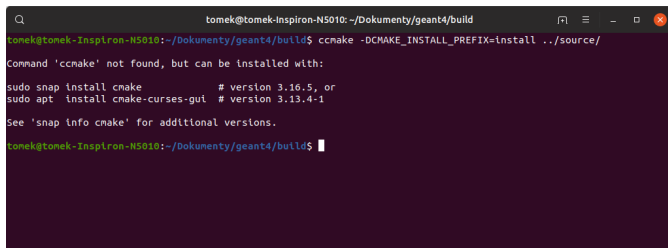
Weszłam do katalogu build i stamtąd wywołałam polecenie:

```
ccmake -DCMAKE_INSTALL_PREFIX=install ../source
```

Opcja `-DCMAKE_INSTALL_PREFIX=install` mówi o tym, gdzie mają się znaleźć pliki instalacyjne. W moim przypadku geant stworzył katalog `install` wewnątrz katalogu `build`. Gdybym instalowała geant'a na nowo, kazałabym stworzyć katalog `install` obok katalogu `build` i `source`, czyli zmieniła opcję na `-DCMAKE_INSTALL_PREFIX=../install`

Opcja `../source` mówi o położeniu pliku `CMakeList.txt`, który znajduje się w katalogu `source`.

Mój system nie miał zainstalowanego cmake (ani ccmake), o czym zostałam poinformowana wraz z instrukcją instalacji (wybrałam drugie z wymienionych poleceń):



```
tomek@tomek-Inspiron-N5010: ~/Dokumenty/geant4/build
tomek@tomek-Inspiron-N5010:~/Dokumenty/geant4/build$ ccmake -DCMAKE_INSTALL_PREFIX=install ../source/

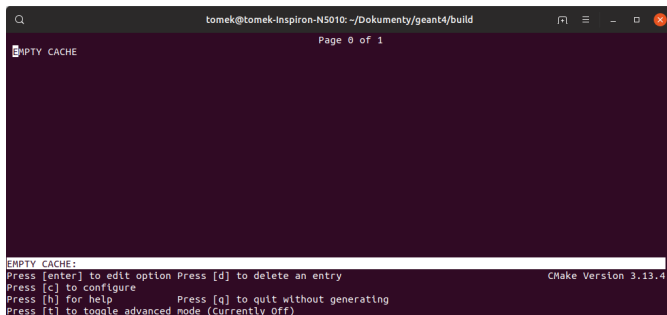
Command 'ccmake' not found, but can be installed with:

sudo snap install cmake          # version 3.16.5, or
sudo apt install cmake-curses-gui # version 3.13.4-1

See 'snap info cmake' for additional versions.

tomek@tomek-Inspiron-N5010:~/Dokumenty/geant4/build$
```

Zainstalowałam cmake zgodnie z sugestią systemu i powtórzyłam polecenie `ccmake -DCMAKE_INSTALL_PREFIX=install ../source` Otworzył się program ccmake, co wygląda następująco:



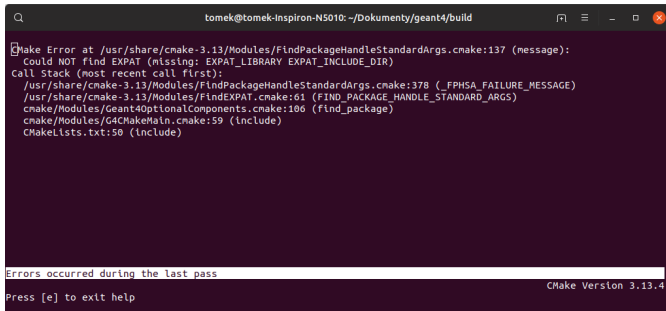
```
tomek@tomek-Inspiron-N5010: ~/Dokumenty/geant4/build
Page 0 of 1

EMPTY CACHE

EMPTY CACHE:
Press [enter] to edit option Press [d] to delete an entry
Press [c] to configure
Press [h] for help          Press [q] to quit without generating
Press [t] to toggle advanced mode (Currently Off)

CMake Version 3.13.4
```

Kliknęłam przycisk „C”, co zgodnie z menu na dole oznacza konfigurację.
Otrzymałam następujący błąd:



```
tomek@tomek-Inspiron-N5010: ~/Dokumenty/geant4/build

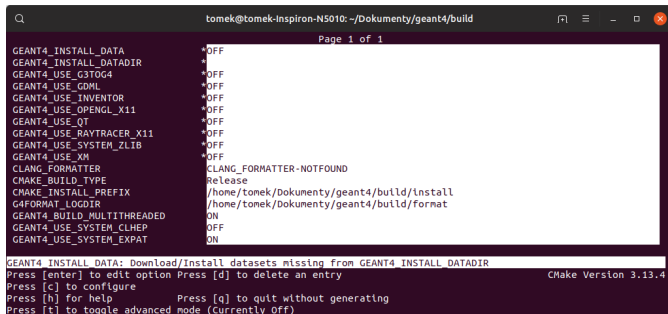
CMake Error at /usr/share/cmake-3.13/Modules/FindPackageHandleStandardArgs.cmake:137 (message):
  Could NOT find EXPAT (missing: EXPAT_LIBRARY EXPAT_INCLUDE_DIR)
Call Stack (most recent call first):
  /usr/share/cmake-3.13/Modules/FindPackageHandleStandardArgs.cmake:378 (_FPHSA_FAILURE_MESSAGE)
  /usr/share/cmake-3.13/Modules/FindEXPAT.cmake:61 (FIND_PACKAGE_HANDLE_STANDARD_ARGS)
  cmake/Modules/Geant4OptionalComponents.cmake:106 (find_package)
  cmake/Modules/G4CMakeMain.cmake:59 (include)
  CMakeLists.txt:50 (include)

Errors occurred during the last pass

Press [e] to exit help                                CMake Version 3.13.4
```

Zgodnie z komunikatem brakuje mi biblioteki EXPAT. W nowym terminalu wpisałam w konsoli polecenie (znalezione z pomocą google):
`sudo apt-get install -y libexpat1-dev`

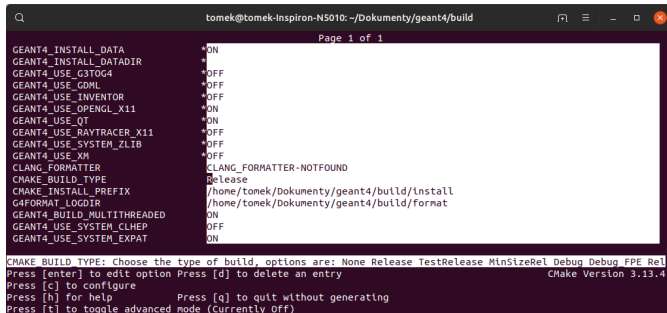
Wróciłam do konsoli z otwartym programem cmake, kliknęłam „e”
otrzymując panel do zmiany konfiguracji budowania geant'a



```
Q tomek@tomek-Inspiron-N5010: ~/Dokumenty/geant4/build
Page 1 of 1
GEANT4_INSTALL_DATA          *OFF
GEANT4_INSTALL_DATADIR      *
GEANT4_USE_G3TOG4            *OFF
GEANT4_USE_GDML              *OFF
GEANT4_USE_INVENTOR          *OFF
GEANT4_USE_OPENGL_X11        *OFF
GEANT4_USE_QT                *OFF
GEANT4_USE_RAYTRACER_X11     *OFF
GEANT4_USE_SYSTEM_ZLIB       *OFF
GEANT4_USE_XM                *OFF
CLANG_FORMATTER              CLANG_FORMATTER-NOTFOUND
CMAKE_BUILD_TYPE             Release
CMAKE_INSTALL_PREFIX         /home/tomek/Dokumenty/geant4/build/install
G4FORMAT_LOGDIR              /home/tomek/Dokumenty/geant4/build/format
GEANT4_BUILD_MULTITHREADED    ON
GEANT4_USE_SYSTEM_CLHEP      OFF
GEANT4_USE_SYSTEM_EXPAT      ON

GEANT4_INSTALL_DATA: Download/Install datasets missing from GEANT4_INSTALL_DATADIR
Press [enter] to edit option Press [d] to delete an entry          CMake Version 3.13.4
Press [c] to configure
Press [h] for help          Press [q] to quit without generating
Press [t] to toggle advanced mode (Currently Off)
```

Nawigując strzałkami i przyciskiem „Enter” zmieniałam konfigurację cmake na moją ulubioną:

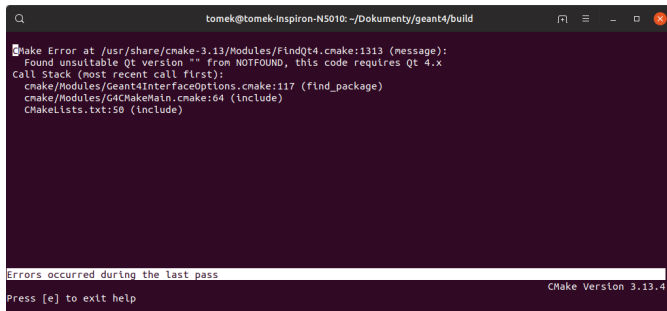


```
tomek@tomek-Inspiron-N5010: ~/Dokumenty/geant4/build
Page 1 of 1
GEANT4_INSTALL_DATA *ON
GEANT4_INSTALL_DATADIR *
GEANT4_USE_G3TOG4 *OFF
GEANT4_USE_GDML *OFF
GEANT4_USE_INVENTOR *OFF
GEANT4_USE_OPENGL_X11 *ON
GEANT4_USE_QT *ON
GEANT4_USE_RAYTRACER_X11 *OFF
GEANT4_USE_SYSTEM_ZLIB *OFF
GEANT4_USE_XM *OFF
CLANG_FORMATTER CLANG_FORMATTER-NOTFOUND
CHAKE_BUILD_TYPE Release
CHAKE_INSTALL_PREFIX /home/tomek/dokumenty/geant4/build/install
G4FORMAT_LOGDIR /home/tomek/dokumenty/geant4/build/format
GEANT4_BUILD_MULTITHREADED ON
GEANT4_USE_SYSTEM_CLHEP OFF
GEANT4_USE_SYSTEM_EXPAT ON

CHAKE BUILD TYPE: Choose the type of build, options are: None Release TestRelease MinSizeRel Debug Debug FPE Rel
Press [enter] to edit option Press [d] to delete an entry CMake Version 3.13.4
Press [c] to configure
Press [h] for help Press [q] to quit without generating
Press [t] to toggle advanced mode (currently Off)
```

NAJWAŻNIEJSZE JEST WŁĄCZENIE OPCJI GEANT4_USE_QT ORAZ
GEANT4_USE_OPENGL_X11 !!!! BEZ TEGO SYMULACJA NIE BĘDZIE
MIAŁA PIĘKNEJ WIZUALIZACJI.

Po ustawieniu opcji budowania kliknęłam przycisk „c” otrzymując komunikat informujący o braku bibliotek QT:



```
tomek@tomek-Inspiron-N5010: ~/Dokumenty/geant4/build

CMake Error at /usr/share/cmake-3.13/Modules/FindQt4.cmake:1313 (message):
  Found unsuitable Qt version "" from NOTFOUND, this code requires Qt 4.x
Call Stack (most recent call first):
  cmake/Modules/Geant4InterfaceOptions.cmake:117 (find_package)
  cmake/Modules/G4CMakeMain.cmake:64 (include)
  CMakeLists.txt:50 (include)

Errors occurred during the last pass

Press [e] to exit help

CMake Version 3.13.4
```

Wyszukałam polecenia instalujące QT na ubuntu i wpisałam je w nowej konsoli (nie muszę wychodzić z ccmake). Nie instalowałam dokumentacji QT (można to zrobić, jeśli ktoś ma ochotę).

ubuntu 19.04 synaptic | X | Software Download | X | Building and Installing | X | Building and Installing | X | Using apt-get Commu | X | c++ - Install Qt on Ubuntu | X

https://stackoverflow.com/questions/48147356/install-qt-on-ubuntu

stackoverflow Products Customers Use cases Search... Login Sign up

Home PUBLIC Stack Overflow Tags Users Jobs TEAMS What's this? Free 30 Day Trial

Install Qt

69

```
sudo apt-get install build-essential
sudo apt-get install qtcreator
sudo apt-get install qt5-default
```

Install documentation and examples if Qt Creator is installed thanks to the Ubuntu Software Center or thanks to the synaptic package manager, documentation for Qt Creator is not installed. Hitting the F1 key will show you the following message: "No documentation available". This can easily be solved by installing the Qt documentation:

```
sudo apt-get install qt5-doc
sudo apt-get install qt5-doc-html qtbase5-doc-html
sudo apt-get install qtbase5-examples
```

Restart Qt Creator to make the documentation available.

Error while loading shared libraries

Problem:

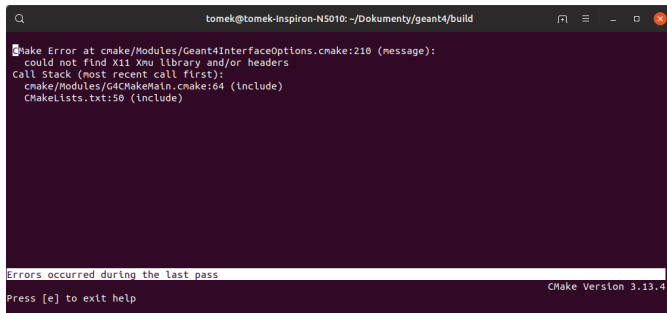
Related

- ModelSim Altera error
- How do I install the OpenSSL libraries on Ubuntu?
- sqlite3-ruby install error on Ubuntu
- "Unable to find remote helper for https" during git clone
- How to install Boost on Ubuntu
- How to install the JDK on Ubuntu Linux
- Qt project release ubuntu - error while loading shared libraries: libQt5Webkit.so.5
- libz.so.1: cannot open shared object file
- Geany motion Error in Ubuntu 14.04 LTS
- Resolve error while loading shared libraries: libselinux.so.1 in ubuntu 14.04 LTS

Hot Network Questions

By using our site, you acknowledge that you have read and understand our [Cookie Policy](#), [Privacy Policy](#), and our [Terms of Service](#).

Wróciłam do konsoli z otwartym cmake, kliknęłam przycisk „e” i jeszcze raz spróbowałam skonfigurować projekt (przycisk „c”). Tym razem otrzymałam komunikat o braku X11



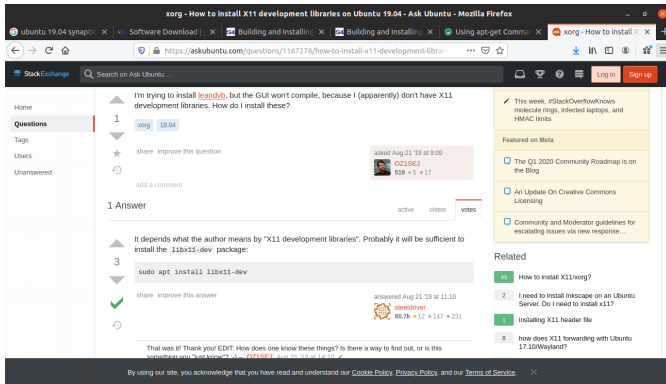
```
tomek@tomek-Inspiron-N5010: ~/Dokumenty/geant4/build

CMake Error at cmake/Modules/Geant4InterfaceOptions.cmake:210 (message):
  could not find X11 Xmu library and/or headers
Call Stack (most recent call first):
  cmake/Modules/G4CMakeMain.cmake:64 (include)
  CMakeLists.txt:50 (include)

Errors occurred during the last pass

Press [e] to exit help                                     CMake Version 3.13.4
```

Ponownie skorzystałam z pomocy google (przepisałam komunikat z ccmake do wyszukiwarki) otrzymując następującą wskazówkę:



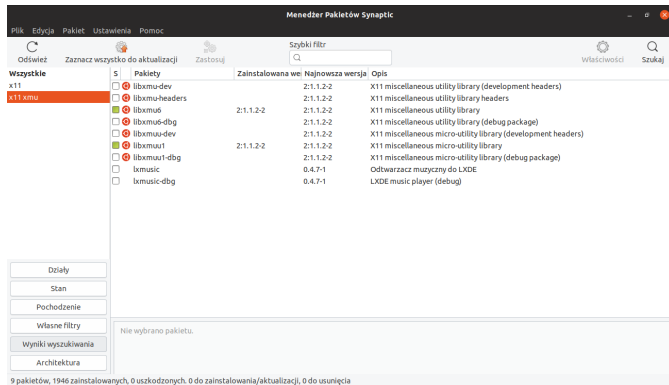
Polecenie wpisałam w nowej konsoli.

Budowanie

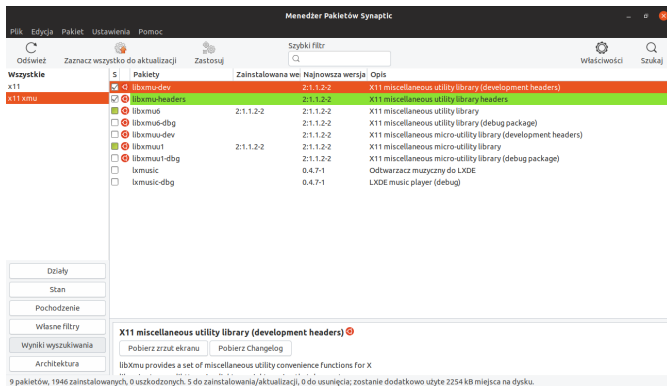
Po raz pierwszy napotkałam jakiś problem. Okazało się, że libx11-dev już było zainstalowane (być może razem z QT), czyli to rozwiązanie nie pomoże. Jako, że nie znalazłam mądrzejszej podpowiedzi zainstalowałam sobie SYNAPTIC (w starszych Ubuntu jest zainstalowany domyślnie, w wersji 19.04 trzeba go samemu dodać). Wpisałam w konsoli:

```
sudo apt-get install synaptic
```

Uruchomiłam synaptic, kliknęłam przycisk wyszukiwania (lupkę) i wpisałam x11 xmu.



Zaznaczyłam do instalacji lubxmu-dev oraz libxmu-headers: zaznaczyłam kwadraty znajdujące się obok nazwy pakietu i kliknęłam „Zastosuj”.



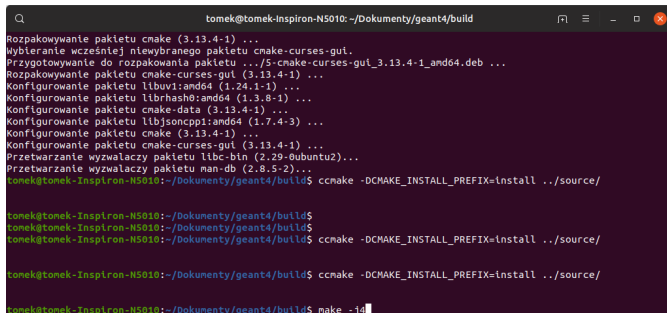
Kopilowanie

Wróciłam do konsoli z otwartym cmake, kliknęłam „e” aby wyjść z okienka z błędem, potem „c” aby dokończyć konfigurację, a potem „g” aby wygenerować pliki do kompilacji i instalacji. Tym razem nie dostałam więcej błędów, cmake zakończył pracę.

W konsoli, w której uprzednio uruchamiałam cmake (czyli otwartej w katalogu build) wpisałam:

```
make -j4
```

Opcja -j umożliwia wykorzystanie więcej niż jednego wątku. Mój procesor jest czterowątkowy dlatego wybrałam 4.



```
tomek@tomek-Inspiron-N5010: ~/Dokumenty/geant4/build
Rozpakowywanie pakietu cmake (3.13.4-1) ...
Wybieranie wcześniej niewybranego pakietu cmake-curses-gui.
Przygotowywanie do rozpakowania pakietu .../5-cmake-curses-gui_3.13.4-1_and64.deb ...
Rozpakowywanie pakietu cmake-curses-gui (3.13.4-1) ...
Konfigurowanie pakietu libuv1:amd64 (1.24.1-1) ...
Konfigurowanie pakietu libhash0:amd64 (1.3.8-1) ...
Konfigurowanie pakietu cmake-data (3.13.4-1) ...
Konfigurowanie pakietu libjsoncpp1:amd64 (1.7.4-3) ...
Konfigurowanie pakietu cmake (3.13.4-1) ...
Konfigurowanie pakietu cmake-curses-gui (3.13.4-1) ...
Przetwarzanie wyzwalaczy pakietu libc-bin (2.29-0ubuntu2)...
Przetwarzanie wyzwalaczy pakietu man-db (2.8.5-2)...
tomek@tomek-Inspiron-N5010:~/Dokumenty/geant4/build$ cmake -DCMAKE_INSTALL_PREFIX=install ../source/

tomek@tomek-Inspiron-N5010:~/Dokumenty/geant4/build$
tomek@tomek-Inspiron-N5010:~/Dokumenty/geant4/build$
tomek@tomek-Inspiron-N5010:~/Dokumenty/geant4/build$ cmake -DCMAKE_INSTALL_PREFIX=install ../source/

tomek@tomek-Inspiron-N5010:~/Dokumenty/geant4/build$ cmake -DCMAKE_INSTALL_PREFIX=install ../source/

tomek@tomek-Inspiron-N5010:~/Dokumenty/geant4/build$ make -j4
```

Poleceniem make rozpoczęłam długą kompilację geant'a. U mnie trwało to ze 2 godziny, ale na lepszej maszynie na pewno będzie dużo szybciej.

Po zakończonej sukcesem kompilacji zostało już tylko pakiet zainstalować wpisując polecenie `make install`. Instalacja powinna trwać już tylko chwilę

```
tomek@tomek-Inspiron-N5010: ~/Dokumenty/geant4/build
[ 98%] Building CXX object source/physics_lists/CMakeFiles/G4physicslists.dir/lists/src/LBE.cc.o
[ 98%] Building CXX object source/visualization/OpenGL/CMakeFiles/G4OpenGL.dir/include/moc_G4OpenGLQtExportDialog.cpp.o
[ 98%] Building CXX object source/visualization/OpenGL/CMakeFiles/G4OpenGL.dir/include/moc_G4OpenGLQtMovieDialog.cpp.o
[ 98%] Building CXX object source/visualization/OpenGL/CMakeFiles/G4OpenGL.dir/include/moc_G4OpenGLQtViewer.cpp.o
[ 98%] Building CXX object source/physics_lists/CMakeFiles/G4physicslists.dir/lists/src/NuBeam.cc.o
[ 98%] Building CXX object source/physics_lists/CMakeFiles/G4physicslists.dir/lists/src/QBBC.cc.o
[100%] Building CXX object source/physics_lists/CMakeFiles/G4physicslists.dir/lists/src/QGS_BIC.cc.o
[100%] Linking CXX shared library ../BuildProducts/lib/libG4OpenGL.so
[100%] Built target G4OpenGL
[100%] Building CXX object source/physics_lists/CMakeFiles/G4physicslists.dir/lists/src/QGSP_BERT.cc.o
[100%] Building CXX object source/physics_lists/CMakeFiles/G4physicslists.dir/lists/src/QGSP_BERT_HP.cc.o
[100%] Building CXX object source/physics_lists/CMakeFiles/G4physicslists.dir/lists/src/QGSP_BIC_AllHP.cc.o
[100%] Building CXX object source/physics_lists/CMakeFiles/G4physicslists.dir/lists/src/QGSP_BIC.cc.o
[100%] Building CXX object source/physics_lists/CMakeFiles/G4physicslists.dir/lists/src/QGSP_BIC_HP.cc.o
[100%] Building CXX object source/physics_lists/CMakeFiles/G4physicslists.dir/lists/src/QGSP_FTFP_BERT.cc.o
[100%] Building CXX object source/physics_lists/CMakeFiles/G4physicslists.dir/lists/src/Shieldng.cc.o
[100%] Building CXX object source/physics_lists/CMakeFiles/G4physicslists.dir/util/src/G4PhysicsListUtil.cc.o
[100%] Building CXX object source/physics_lists/CMakeFiles/G4physicslists.dir/util/src/G4WarnPLStatus.cc.o
[100%] Linking CXX shared library ../BuildProducts/lib/libG4physicslists.so
[100%] Built target G4physicslists
tomek@tomek-Inspiron-N5010: ~/Dokumenty/geant4/build$ make install
```

Program zostanie zainstalowany w miejscu podanym na początku w zmiennej `-DCMAKE_INSTALL_PREFIX`. Jak już wspomniałam, w moim przypadku w katalogu build stworzył się katalog `install` z plikami instalacyjnymi.

Proszę zwrócić uwagę, że instalacja odbyła się bez uprawnień `sudo`, czyli wszystko zrobiło się lokalnie, bez ingerencji w pliki systemowe. Aby móc korzystać z `geanta` musimy powiedzieć systemowi, gdzie go szukać. W tym celu każdorazowo po uruchomieniu konsoli, w której chcemy pracować z `geantem`, powinniśmy wywołać skrypt `geant4.sh` ustawiający `geantowe` zmienne systemowe. Skrypt znajduje się w katalogu instalacyjnym (u mnie `install`) w podkatalogu `bin`. W konsoli należy wpisać: `source`
`ściezka_do_katalogu_instalacyjnego/install/bin/geant4.sh`

Każdorazowe uruchamianie skryptu jest dość niewygodne w sytuacji, w której intensywnie i często pracujemy z wykorzystaniem geant'a. Proponuję aby polecenie wpisać do skryptu `.bashrc`. Jest to skrypt automatycznie uruchamiany po każdym włączeniu terminala. Znajduje się w katalogu domowym. Jest to skrypt ukryty, więc domyślnie nie widać go w nautilus-ie (menadżerze plików w Ubuntu).

Aby otworzyć `.bashrc` otwórz konsolę w katalogu domowym i wpisz:
`gedit .bashrc`

Zjedź na koniec pliku i dopisz linię:

`source sciezka_do_katalogu_instalacyjnego/install/bin/geant4.sh`

U mnie wygląda to tak:

```
# enable programmable completion features (you don't need to enable
# this, if it's already enabled in /etc/bash.bashrc and /etc/profile
# sources /etc/bash.bashrc).
if ! shopt -oq posix; then
  if [ -f /usr/share/bash-completion/bash_completion ]; then
    . /usr/share/bash-completion/bash_completion
  elif [ -f /etc/bash_completion ]; then
    . /etc/bash_completion
  fi
fi

source /home/tonek/Dokumenty/geant4/build/install/bin/geant4.sh
```

sh ▼ Szerokość tabulacji: 8 ▼ Wrsz 1, kol 1 ▼ WST

Sprawdzenie

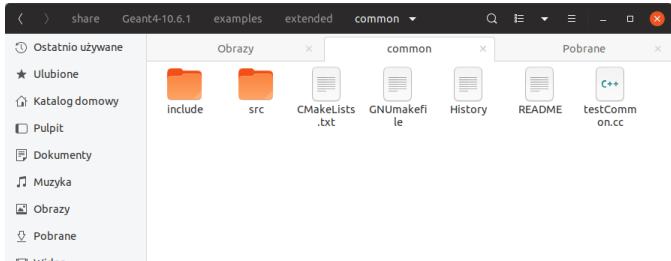
Po zainstalowaniu geanta warto sprawdzić, czy wszystko działa. W tym celu najlepiej skorzystać z projektu, który na pewno nie ma błędów. Wraz z kodem pakietu ściągnęliśmy szereg przykładów dostępnych katalogu instalacyjnym, podkatalogu `share/Geant4-10.06.1/examples`

(u mnie

`home/Dokumenty/geant4/build/install/share/Geant4-10.06.1/examples`).

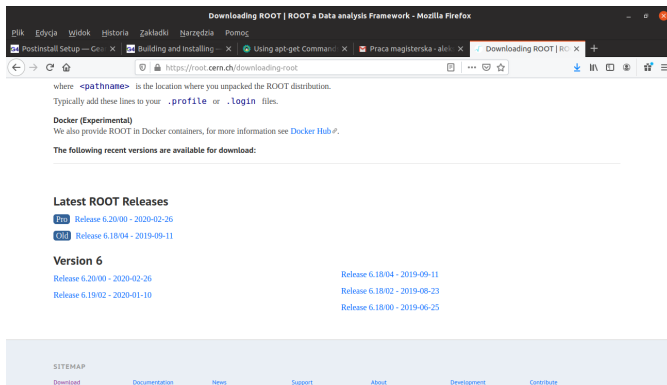
Dostępne są 3 typy przykładów: proste (basic), zaawansowane (advanced) oraz rozszerzone (extended). W celu przetestowania instalacji pakietu można skorzystać z dowolnego przykładu (prawie, o czym za chwilę).

Ja skompilowałam i uruchomiłam przykład z `extended/common`. Wszłam do katalogu przykładu. Stworzyłam katalog `build`. W katalogu `build` otworzyłam terminal, wpisałam polecenie `cmake ..`, potem `make`. Kod się skompilował, powstał plik wykonywalny `testCommon`. Uruchomiłam go wpisując `./testCommon`. Program się włączył, a następnie wyłączył. Nie był to efekt, którego oczekiwałam, ale nie chciało mi się drażnić tej sprawy. Wyjaśniła się chwilę później.



Idąc za ciosem postanowiłam zainstalować też ROOT-a. ROOT będzie nam potrzebny do zapisywania wyników symulacji oraz ich analizy. Pakietu ROOT się nie instaluje, wystarczy go ściągnąć, rozpakować i ustawić zmienne wskazujące na root-a.

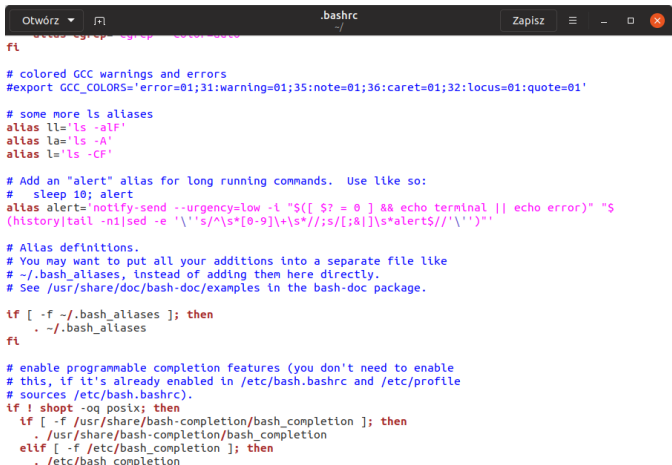
Proszę udać się do strony <https://root.cern.ch/download-root>, kliknąć w link „Release 6.20/00 - 2020-02-26”



ROOT

Wybieramy wersję binarną dostosowaną do naszego systemu operacyjnego. W moim przypadku jest to Ubuntu 19.. Ściągamy plik na dysk. Robimy mu jakieś stosowne miejsce (np. obok katalogu geant możemy stworzyć katalog root). Rozpakowujemy paczkę. Powinniśmy mieć w środku katalog bin, a w nim skrypt thisroot.sh. Podobnie jak w przypadku geanta musimy skrypt uruchomić każdorazowo po uruchomieniu konsoli lub wpisać takie polecenie do pliku .bashrc

U mnie to wygląda tak:



```
.bashrc
~/.bashrc

# colored GCC warnings and errors
#export GCC_COLORS='error=01;31:warning=01;35:note=01;36:caret=01;32:locus=01:quote=01'

# some more ls aliases
alias ll='ls -alF'
alias la='ls -A'
alias l='ls -CF'

# Add an "alert" alias for long running commands. Use like so:
# sleep 10; alert
alias alert='notify-send --urgency=low -i "${[ $? = 0 ]} && echo terminal || echo error)" "$
(history|tail -n1|sed -e '\''s/^\[0-9\]\+\s*//;s/[]\&[]\s*alert$/'\''\`"'

# Alias definitions.
# You may want to put all your additions into a separate file like
# ~/.bash_aliases, instead of adding them here directly.
# See /usr/share/doc/bash-doc/examples in the bash-doc package.

if [ -f ~/.bash_aliases ]; then
    . ~/.bash_aliases
fi

# enable programmable completion features (you don't need to enable
# this, if it's already enabled in /etc/bash.bashrc and /etc/profile)
# sources /etc/bash.bashrc).
if ! shopt -oq posix; then
    if [ -f /usr/share/bash-completion/bash_completion ]; then
        . /usr/share/bash-completion/bash_completion
    elif [ -f /etc/bash_completion ]; then
        . /etc/bash_completion
    fi
fi
```

Finał

Na koniec postanowiłam uruchomić któryś z moich starszych kodów, aby upewnić się, czy nie muszę dokonać jakichś zmian pomiędzy wersją 10.04 a 10.06. Kod się pięknie skompilował, jednak po uruchomieniu okazało się, że program znów się otwiera i zamyka, bez włączenia wizualizacji.

Okazuje się, że wersja 10.06 ma pewne istotne zmiany w stosunku do wersji poprzednich. Większość przykładów dostarczanych wraz z paczką geant'a jest napisana na starsze wersje i nie będzie działać (też uważam, że to granda, powinni przepisać wszystkie przykłady tak, aby działały w nowej wersji). Przykłady napisane pod wersję 10.06 to:

Geant4 10.6 Release Notes - Mozilla Firefox

Overview | Software Download | **Geant4 10.6 Release Notes** | Building and Installing | Using apt-get | Praca magistra | Release 6.20/1 | G4VIS_USE

geant4-data.web.cern.ch/geant4-data/ReleaseNotes/ReleaseNotes4.10.6.html

• A new stopping physics constructor (G4StoppingPhysicsFritiofWithBinaryCascade) has been introduced, which uses Fritiof (FTF) coupled with Binary cascade (BIC) for the anti-baryon annihilation at rest. It is used only in the physics lists FTF_BIC and QGS_BIC.

Visualization and Interfaces

- New set of UI commands and improvements in visualization. Added new UI commands for view interpolation and centering.
- Introducing cloud drawing style in visualization, using kernel algorithms for generating points on the surface of volumes, by-passing polyhedral representations; the solid is being visualised by a polymarker of dots.

Examples

- *GB07*: new example (in extended/biasing) to demonstrate the usage of the leading particle biasing functionality, introduced in the generic biasing package.
- *Hadr08*: new example (in extended/hadronic) to demonstrate the possibility of emulating hadronic-model-per-region capability (not foreseen by the hadronic framework) via the use of generic biasing with usual, unbiased weights of 1.0.
- Extended GFlash examples set by moving existing *gflash* example to *gflash1* and added: *gflash2* with envelope in parallel world; *gflash3* with SD in parallel world; *gflash4* with histogramming of shower profiles.
- *dndamage1*: DNA simulation of damage on a chromatin fiber.
- *microprox*: DNA computation of proximity functions in liquid water.

5. Expected effects on physics and computing performance

Electromagnetic physics

- Expected unchanged response for calorimeter simulations.
- General speedup of electro-magnetic physics by 4-8% depending on the software platform. Additional speedup if the *general gamma process* mode is enabled (disabled by default).

Hadronic physics

- Increased energy response and more compact hadronic showers mostly in the projectile energy range between 5 and 20 GeV, mainly due to the

Mam nadzieję, że każdemu z Państwa udało się zainstalować geanta. Jeśli nie, proszę do mnie pisać/wysyłać screeny błędów, a jeśli ktoś na prawdę nie ma na czym go zainstalować, proszę dać mi znać. Może uda mi się coś wymyślić.