IX SEMINAR ON SOFTWARE FOR NUCLEAR, SUBNUCLEAR AND APPLIED PHISICS

Porto Conte, Alghero, Italy
28th May - 1th June 2012

How to install Geant 4

Geant 4 tutorial course

INFN
Geant4

Outline

- Supported platforms & compilers
- Required software
- Where to download the packages
- Geant4 toolkit installation *(release 9.5.p01)*
  - Configuring the environment manually
  - Using *CMake*
- CLHEP full version installation *(optional)*

Supported platforms & compilers

- Linux systems
  - Scientific Linux CERN SLC5, with gcc 4.1.2
    - G4SYSTEM: Linux-g++
- MacOSX systems
  - MacOSX 10.7(Lion) and 10.6(Leopard), with gcc 4.2.1
    - G4SYSTEM: Darwin-g++
- Windows systems
  - Windows 7 and XP, with Visual Studio 9 and 10
    - G4SYSTEM: WIN32-VC

Check current Geant4 supported platforms in http://cern.ch/geant4

Required software

- A UNIX shell and related basic UNIX commands
- C++ compiler
  - gcc is usually installed on your Linux. If not, you need to install it (*not shown here*)
- Cmake 2.6.4 or higher
- The Geant4 toolkit source code
- CLHEP library
  - an internal version is now supplied with the geant4 source (since 9.5 version)
- The Geant4 data files
  - an automatic procedure can retrieve them (with cmake)

External software packages I

Visualization/GUI tools (optional):

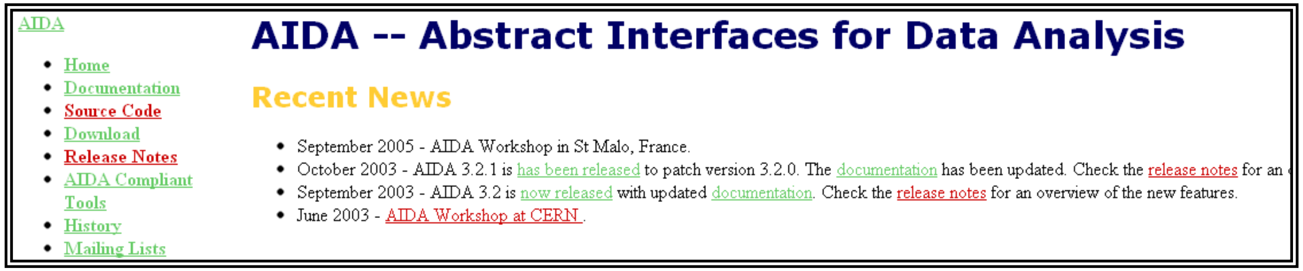
- X Windows
- OpenGL or MesaGL
- VRML browser
- DAWN (PostScript renderer)
- Open Inventor or HEP Inventor
- WIRED4 JAS Plug-In (HepRep browser)
  - Uses the HepRep built-in graphics driver
- Qt graphics toolkit
- Open Scientist
  - interactive environment, including GUI
- Momo
  - a Java-based GUI environment, GGE, GPE ...

Alternatively, you can produce an ascii file for VRML or DAWN

External software packages II

Software for analysis and histogramming (optional):

- AIDA (Abstract Interfaces for Data Analysis)
  - iAIDA (an implementation of AIDA in C++)
  - JAS (Java Analysis Studio)
  - Open Scientist (Interactive Analysis Environment)
  - rAIDA (a Root implementation of AIDA)



AIDA

- Home
- Documentation
- Source Code
- Download
- Release Notes
- AIDA Compliant Tools
- History
- Mailing Lists

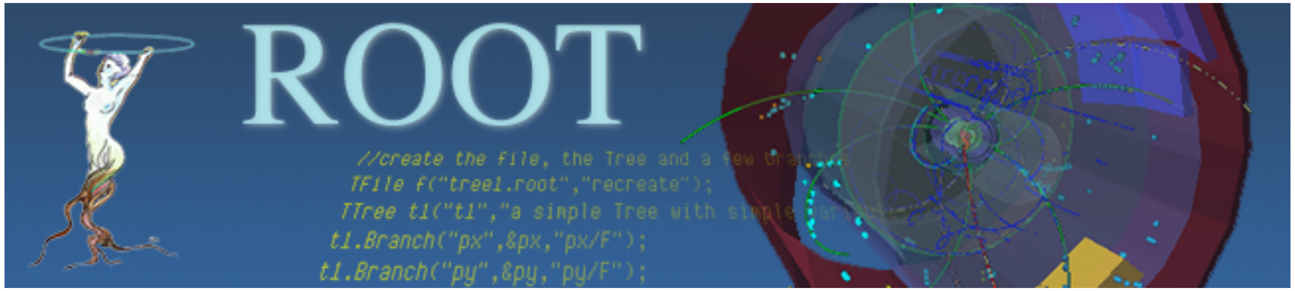
AIDA -- Abstract Interfaces for Data Analysis

Recent News

- September 2005 - AIDA Workshop in St Malo, France.
- October 2003 - AIDA 3.2.1 is has been released to patch version 3.2.0. The documentation has been updated. Check the release notes for an
- September 2003 - AIDA 3.2 is now released with updated documentation. Check the release notes for an overview of the new features.
- June 2003 - AIDA Workshop at CERN .

http://aida.freehep.org/

- ROOT (a data analysis framework)

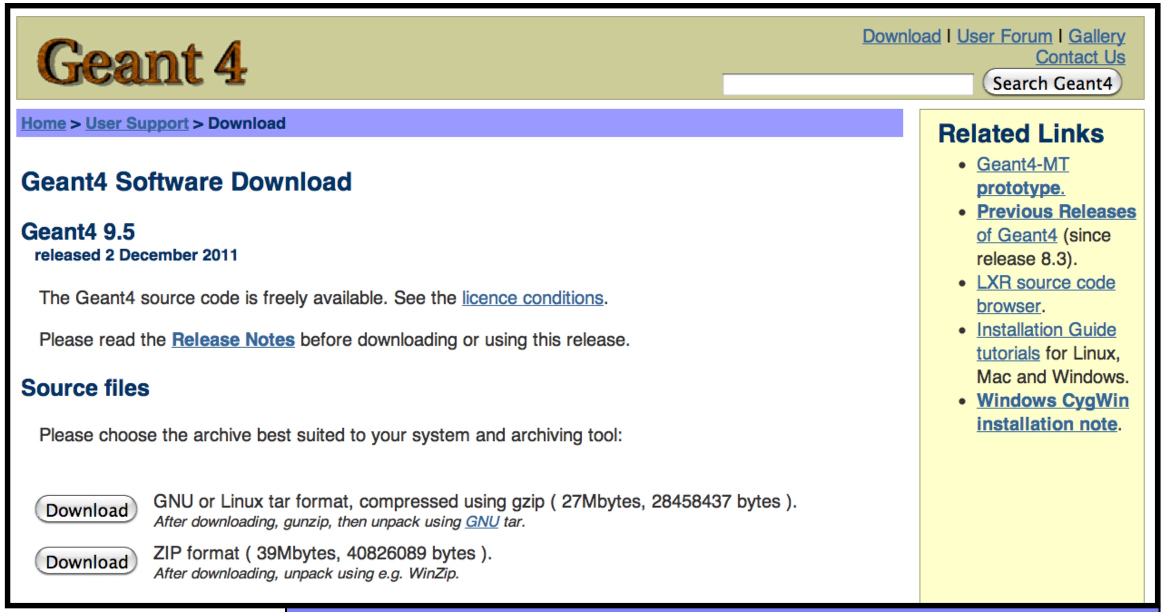


```
//create the file, the Tree and a few branches
TFile f("tree1.root","recreate");
TTree t1("t1","a simple Tree with simple vari
t1.Branch("px",&px,"px/F");
t1.Branch("py",&py,"py/F");
```

http://root.cern.ch/

Where to download the packages

- Geant4



Geant 4
Download | User Forum | Gallery
Contact Us
Search Geant4
Home > User Support > Download

Geant4 Software Download

Geant4 9.5
released 2 December 2011

The Geant4 source code is freely available. See the licence conditions.

Please read the **Release Notes** before downloading or using this release.

Source files

Please choose the archive best suited to your system and archiving tool:

Download — GNU or Linux tar format, compressed using gzip (27Mbytes, 28458437 bytes).
After downloading, gunzip, then unpack using GNU tar.

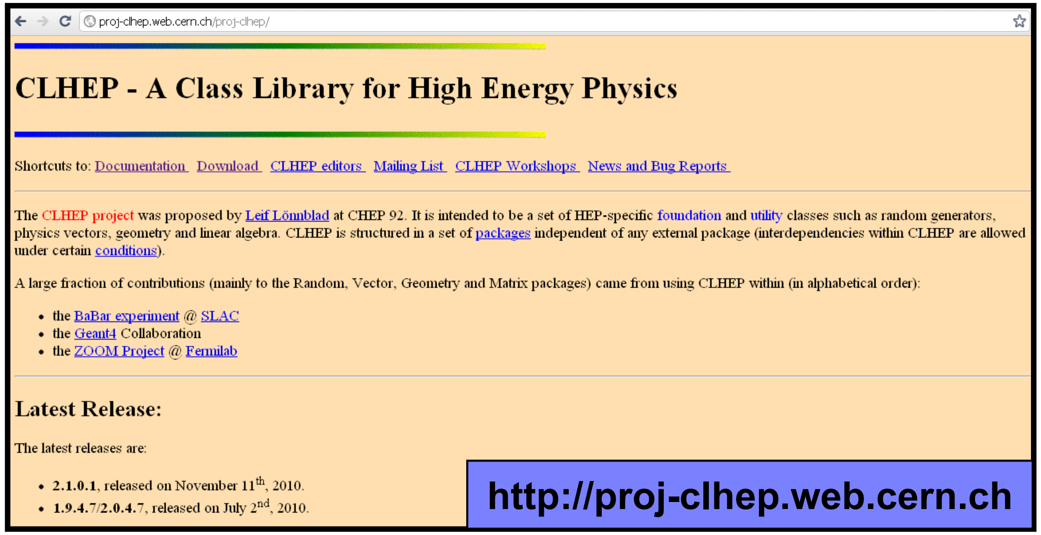
Download — ZIP format (39Mbytes, 40826089 bytes).
After downloading, unpack using e.g. WinZip.

Related Links

- Geant4-MT **prototype.**
- **Previous Releases** of Geant4 (since release 8.3).
- LXR source code browser.
- Installation Guide tutorials for Linux, Mac and Windows.
- **Windows CygWin installation note.**

http://geant4.cern.ch/support/download.shtml

- CLHEP



CLHEP - A Class Library for High Energy Physics

Shortcuts to: Documentation Download CLHEP editors Mailing List CLHEP Workshops News and Bug Reports

The CLHEP project was proposed by Leif Lönnblad at CHEP 92. It is intended to be a set of HEP-specific foundation and utility classes such as random generators, physics vectors, geometry and linear algebra. CLHEP is structured in a set of packages independent of any external package (interdependencies within CLHEP are allowed under certain conditions).

A large fraction of contributions (mainly to the Random, Vector, Geometry and Matrix packages) came from using CLHEP within (in alphabetical order):

- the BaBar experiment @ SLAC
- the Geant4 Collaboration
- the ZOOM Project @ Fermilab

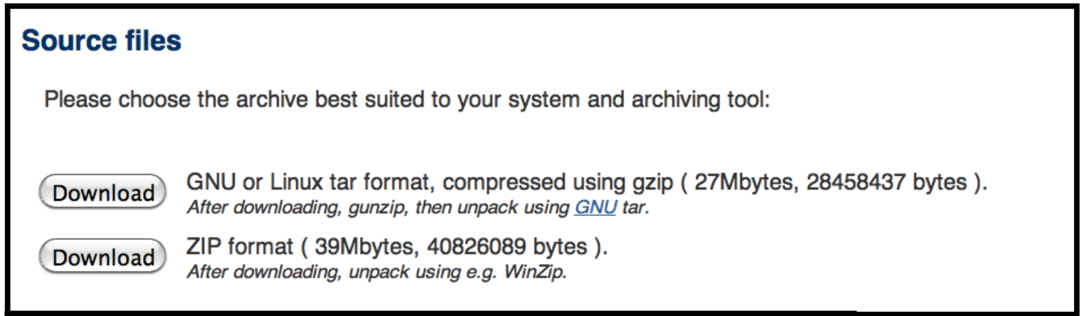
Latest Release:

The latest releases are:

- **2.1.0.1**, released on November 11th, 2010.
- **1.9.4.7/2.0.4.7**, released on July 2nd, 2010.

http://proj-clhep.web.cern.ch

Downloading Geant4 and data files

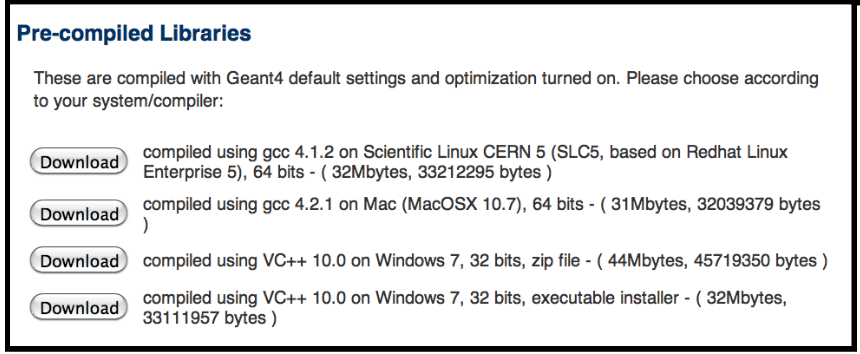


Source files

Please choose the archive best suited to your system and archiving tool:

Download GNU or Linux tar format, compressed using gzip (27Mbytes, 28458437 bytes).
After downloading, gunzip, then unpack using GNU tar.

Download ZIP format (39Mbytes, 40826089 bytes).
After downloading, unpack using e.g. WinZip.



Pre-compiled Libraries

These are compiled with Geant4 default settings and optimization turned on. Please choose according to your system/compiler:

Download compiled using gcc 4.1.2 on Scientific Linux CERN 5 (SLC5, based on Redhat Linux Enterprise 5), 64 bits - (32Mbytes, 33212295 bytes)

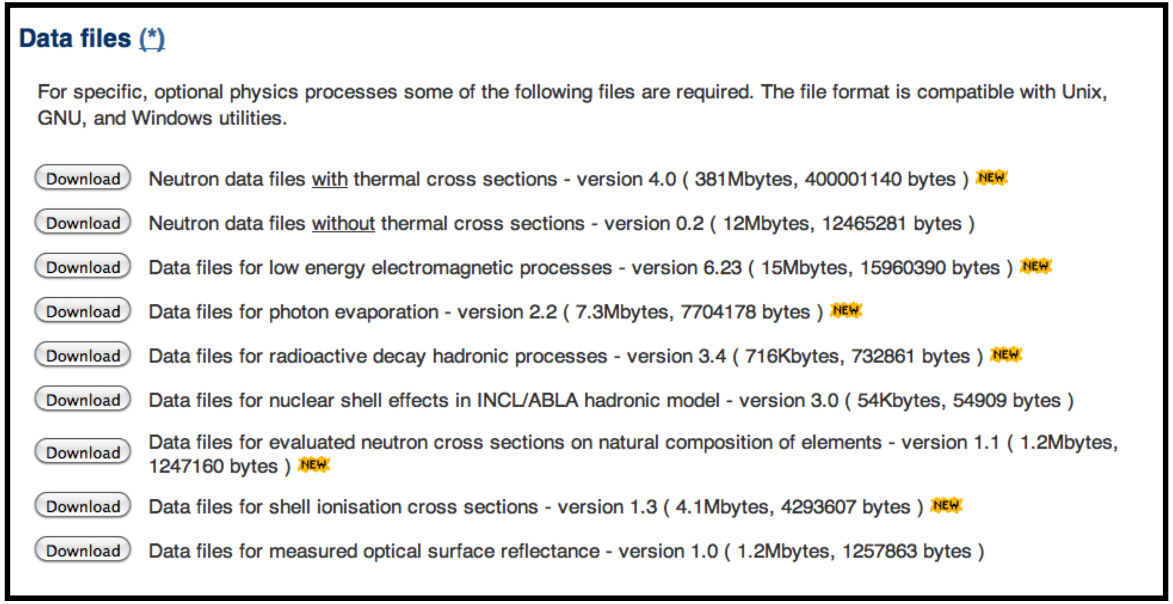
Download compiled using gcc 4.2.1 on Mac (MacOSX 10.7), 64 bits - (31Mbytes, 32039379 bytes)

Download compiled using VC++ 10.0 on Windows 7, 32 bits, zip file - (44Mbytes, 45719350 bytes)

Download compiled using VC++ 10.0 on Windows 7, 32 bits, executable installer - (32Mbytes, 33111957 bytes)

Geant4 source
or
pre-compiled libraries

data files



Data files (*)

For specific, optional physics processes some of the following files are required. The file format is compatible with Unix, GNU, and Windows utilities.

Download Neutron data files with thermal cross sections - version 4.0 (381Mbytes, 400001140 bytes) NEW

Download Neutron data files without thermal cross sections - version 0.2 (12Mbytes, 12465281 bytes)

Download Data files for low energy electromagnetic processes - version 6.23 (15Mbytes, 15960390 bytes) NEW

Download Data files for photon evaporation - version 2.2 (7.3Mbytes, 7704178 bytes) NEW

Download Data files for radioactive decay hadronic processes - version 3.4 (716Kbytes, 732861 bytes) NEW

Download Data files for nuclear shell effects in INCL/ABLA hadronic model - version 3.0 (54Kbytes, 54909 bytes)

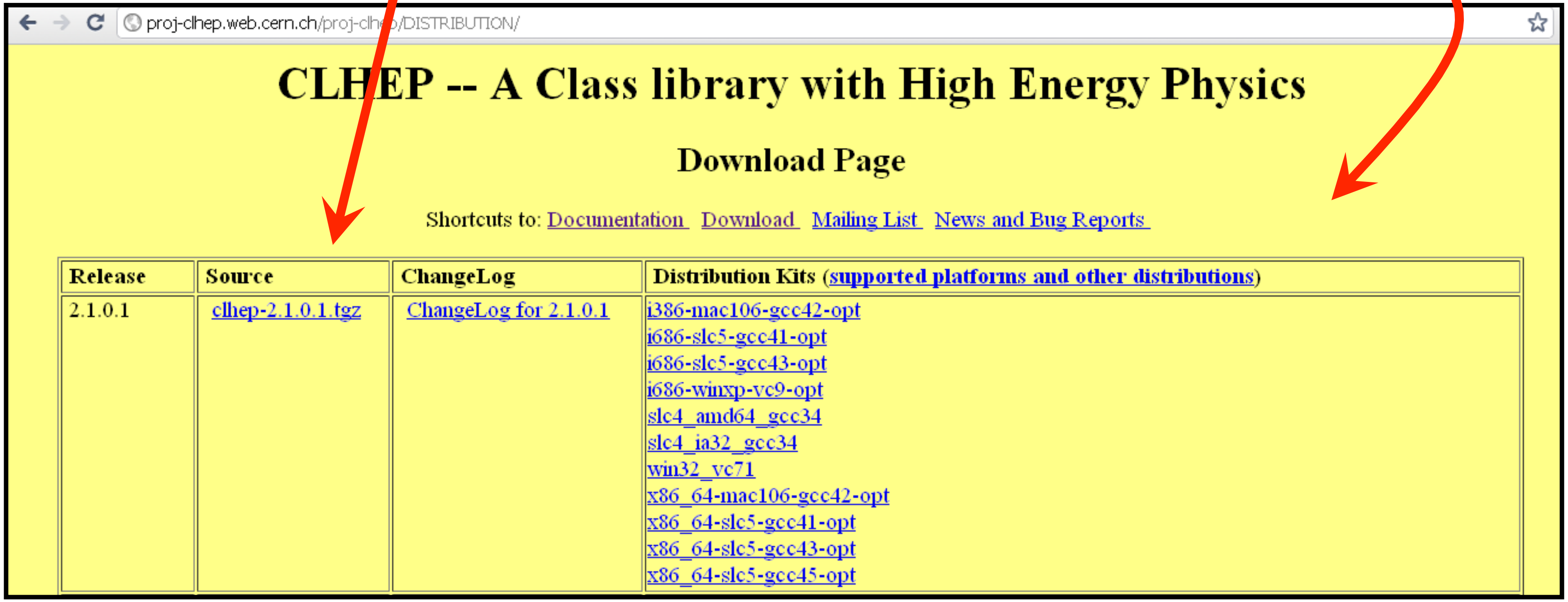
Download Data files for evaluated neutron cross sections on natural composition of elements - version 1.1 (1.2Mbytes, 1247160 bytes) NEW

Download Data files for shell ionisation cross sections - version 1.3 (4.1Mbytes, 4293607 bytes) NEW

Download Data files for measured optical surface reflectance - version 1.0 (1.2Mbytes, 1257863 bytes)

Downloading CLHEP (optionally)

Source code *or* ***pre-compiled libraries***



proj-clhep.web.cern.ch/proj-clhep/DISTRIBUTION/

CLHEP -- A Class library with High Energy Physics

Download Page

Shortcuts to: Documentation Download Mailing List News and Bug Reports

Release	Source	ChangeLog	Distribution Kits (supported platforms and other distributions)
2.1.0.1	clhep-2.1.0.1.tgz	ChangeLog for 2.1.0.1	i386-mac106-gcc42-opt i686-slc5-gcc41-opt i686-slc5-gcc43-opt i686-winxp-vc9-opt slc4_amd64_gcc34 slc4_ia32_gcc34 win32_vc71 x86_64-mac106-gcc42-opt x86_64-slc5-gcc41-opt x86_64-slc5-gcc43-opt x86_64-slc5-gcc45-opt

Geant4 installation (9.5 version)

Working area & installation area

- Why two different areas ?
  - To allow centralized installation of the Geant4 kernel libraries and related sources in a multi-user environment
  - To decouple user-developed code and applications from the kernel
  - To allow an easy integration of the Geant4 software in an existing software framework
- They are controlled by two environment variables:

  `G4WORKDIR` and **`G4INSTALL`**

Two ways to proceed:

- Manually installing by environment variables definition
- Using **CMake** *(recommended)*

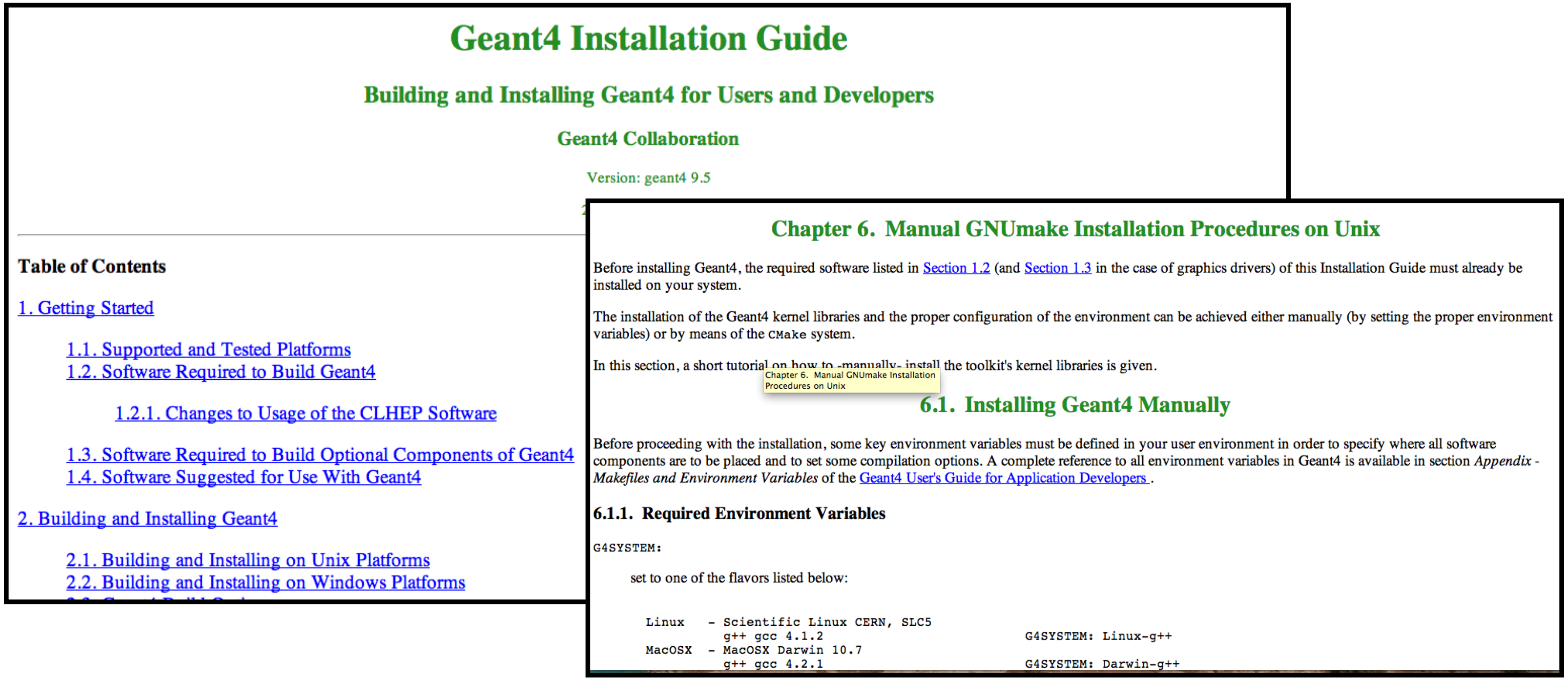
Installing Geant4 manually

- Identify the system used for the installation
  - **`G4SYSTEM`**
- Identify the area of installation (i.e. path where the source code and the kernel libraries should be based)
  - **`G4INSTALL`**
  - Optionally, specify a different path for the kernel libraries and/or the temporary object files
    - **`G4LIB`**, **`G4TMP`**
  - Optionally, specify a different path for exporting of source header files
    - **`G4INCLUDE`**
- Specify the path of installation of CLHEP
  - **`CLHEP_BASE_DIR`**

- Specify all the optional environment variables you need
  - **G4WORKDIR**
  - **G4DEBUG**
  - …

This part is not covered here. For a detailed guide:

http://geant4.web.cern.ch/geant4/UserDocumentation/UsersGuides/InstallationGuide/html/



Geant4 Installation Guide

Building and Installing Geant4 for Users and Developers

Geant4 Collaboration

Version: geant4 9.5

Table of Contents

Chapter 6. Manual GNUmake Installation Procedures on Unix

Before installing Geant4, the required software listed in Section 1.2 (and Section 1.3 in the case of graphics drivers) of this Installation Guide must already be installed on your system.

The installation of the Geant4 kernel libraries and the proper configuration of the environment can be achieved either manually (by setting the proper environment variables) or by means of the CMake system.

In this section, a short tutorial on how to -manually- install the toolkit's kernel libraries is given.

6.1. Installing Geant4 Manually

Before proceeding with the installation, some key environment variables must be defined in your user environment in order to specify where all software components are to be placed and to set some compilation options. A complete reference to all environment variables in Geant4 is available in section *Appendix - Makefiles and Environment Variables* of the Geant4 User's Guide for Application Developers .

6.1.1. Required Environment Variables

`G4SYSTEM:`

set to one of the flavors listed below:

```
Linux    - Scientific Linux CERN, SLC5
           g++ gcc 4.1.2                        G4SYSTEM: Linux-g++
MacOSX   - MacOSX Darwin 10.7
           g++ gcc 4.2.1                        G4SYSTEM: Darwin-g++
```

Installing Geant4 with *CMake*

CMake installation *(if not provided)*

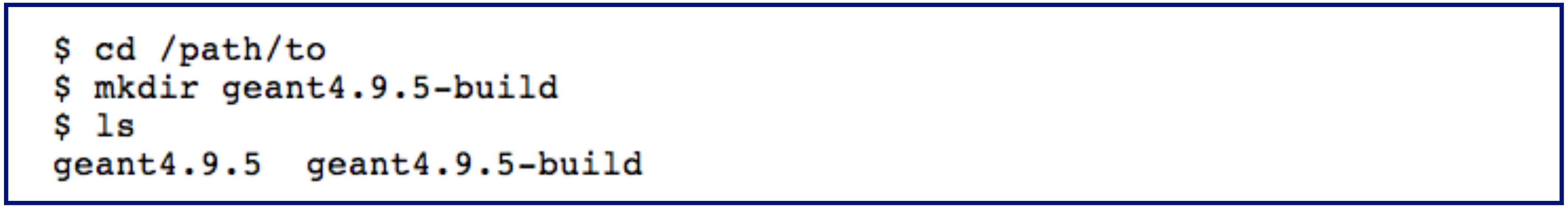
- Depending on the OS installation, CMake may not be installed by default. In that case you have to install it:
  - On Linux: it is recommended to use the CMake provided by the package management system of your distribution.

    In case it does not meet the minimum version requirement:

    1. download the latest version *(http://www.cmake.org/)*
    2. unzip the tar-ball
    3. **`./bootstrap, make, make install`**
  - On Mac: install it using the Darwin64 dmg installerpackage
  - On Windows: install it using the Win32 exe installerpackage

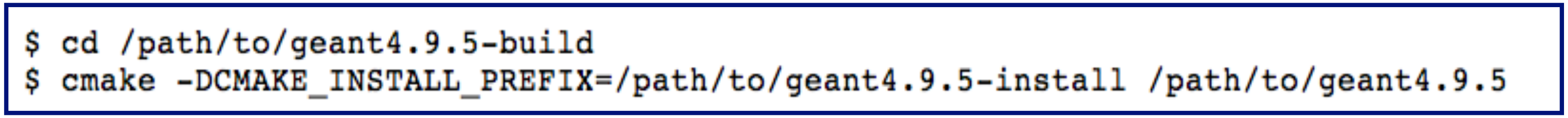
Geant4 installation with CMake

- Unpack the geant4 source package geant4.9.5.tar.gz to a location of your choice:
  - ex.: /path/to/geant4.9.5 → *source directory*
- Create a directory in which to configure and run the build and store the build products (not inside the source dir!)
  - ex.: /path/to/geant4.9.5-build → *build directory*



```
$ cd /path/to
$ mkdir geant4.9.5-build
$ ls
geant4.9.5  geant4.9.5-build
```

- To configure, change into the build directory and run CMake:

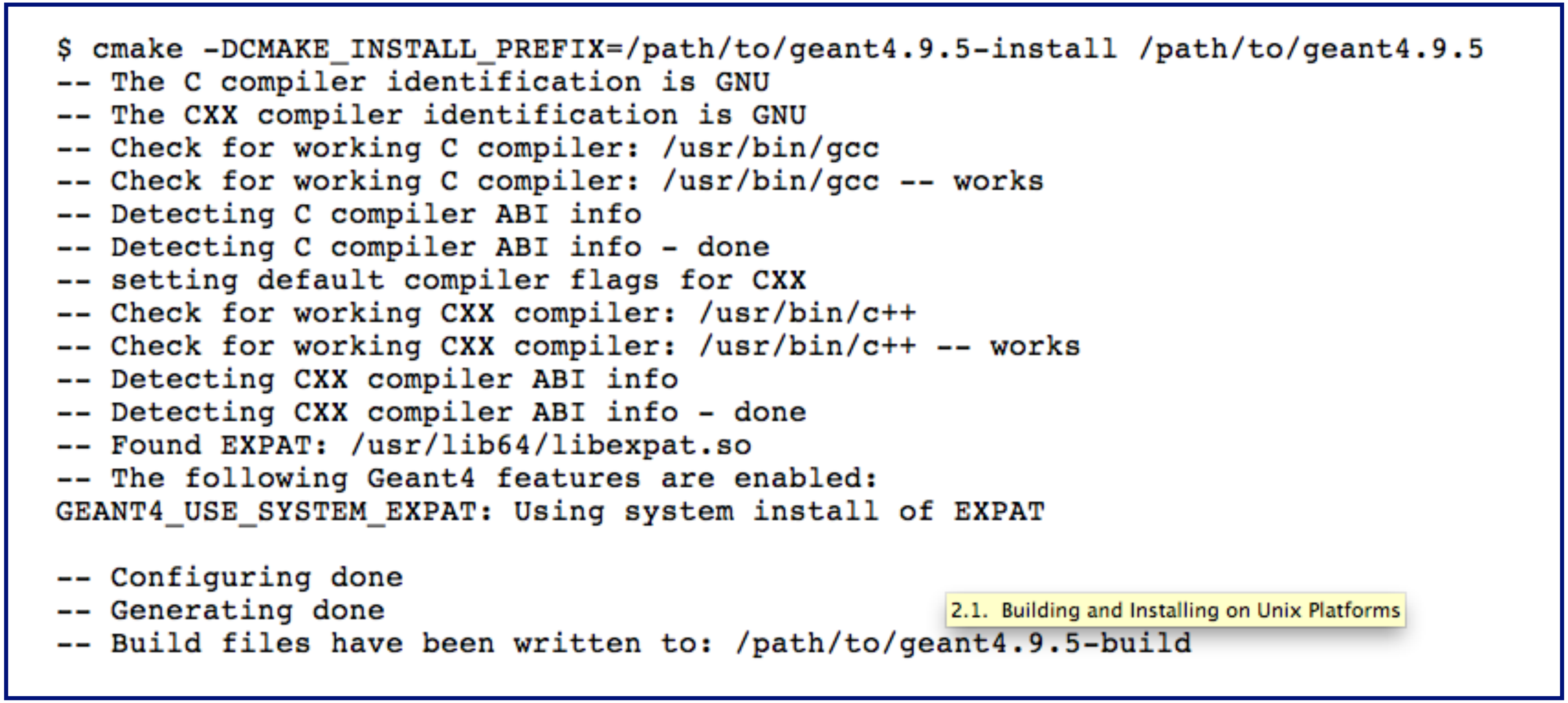


```
$ cd /path/to/geant4.9.5-build
$ cmake -DCMAKE_INSTALL_PREFIX=/path/to/geant4.9.5-install /path/to/geant4.9.5
```

  - CMAKE_INSTALL_PREFIX option is used to set the *install directory*
  - The second argument to CMake is the path to the source directory.

Geant4 installation with CMake

- CMake configures the build and generates Unix Makefiles to perform the actual build:



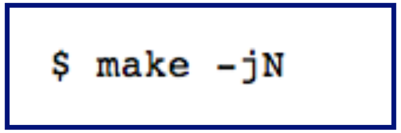
```
$ cmake -DCMAKE_INSTALL_PREFIX=/path/to/geant4.9.5-install /path/to/geant4.9.5
-- The C compiler identification is GNU
-- The CXX compiler identification is GNU
-- Check for working C compiler: /usr/bin/gcc
-- Check for working C compiler: /usr/bin/gcc -- works
-- Detecting C compiler ABI info
-- Detecting C compiler ABI info - done
-- setting default compiler flags for CXX
-- 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
-- Found EXPAT: /usr/lib64/libexpat.so
-- The following Geant4 features are enabled:
GEANT4_USE_SYSTEM_EXPAT: Using system install of EXPAT

-- Configuring done
-- Generating done
-- Build files have been written to: /path/to/geant4.9.5-build
```

  - If you see any errors at this point, carefully check the messages output by CMake

Geant4 installation with CMake

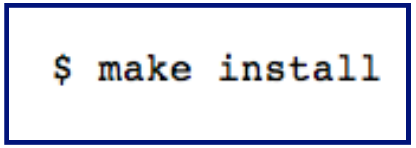
- After the configuration has run, CMake have generated Unix Makefiles for building Geant4. To run the build, simply execute make in the build directory:



```
$ make -jN
```

  - where N is the number of parallel jobs you require. The build will now run, and will output information on the progress of the build and current operations

- When build has completed, you can install Geant4 to the directory you specified earlier in CMAKE_INSTALL_PREFIX by running:



```
$ make install
```

Geant4 installation with CMake

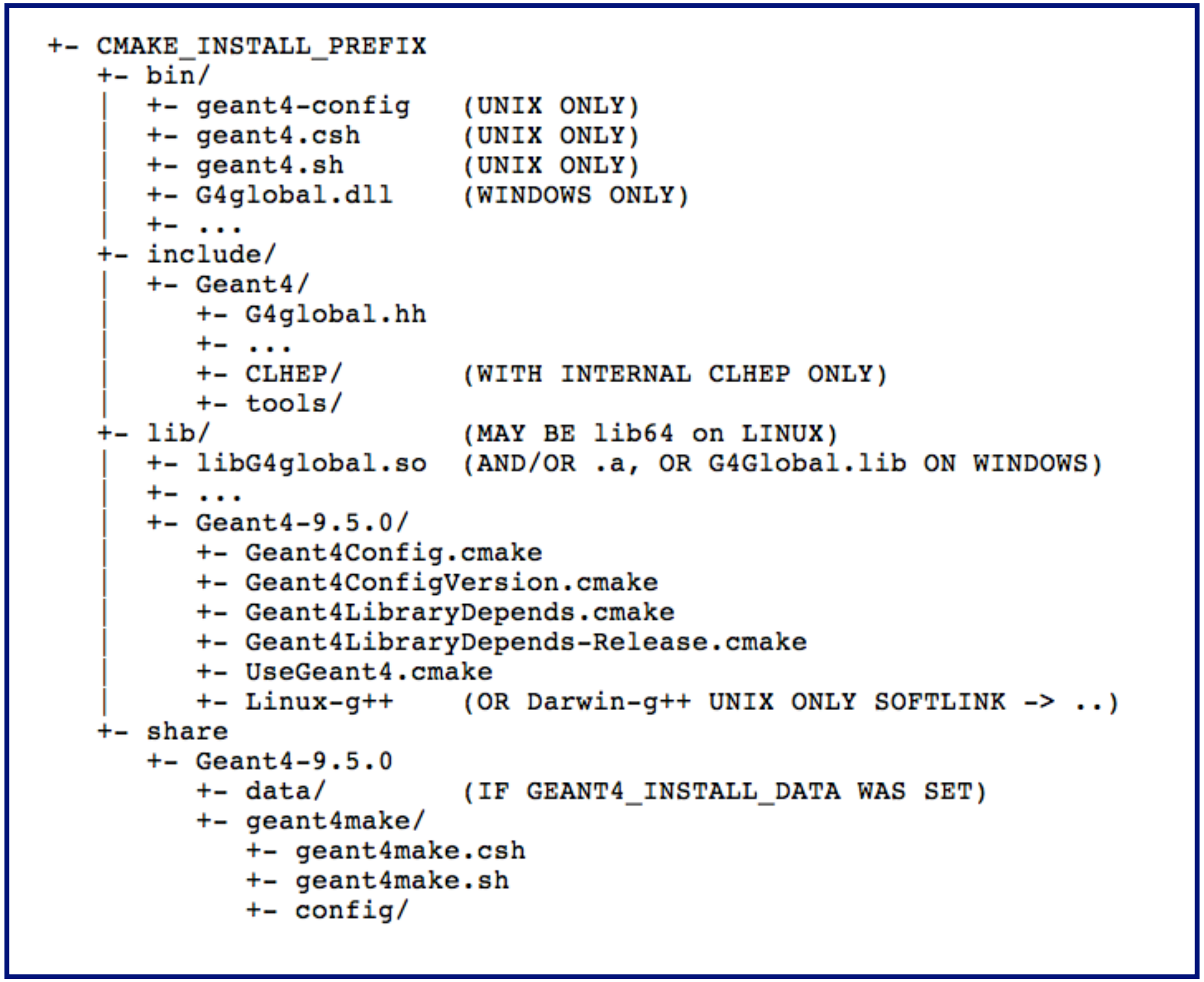
- Additional arguments can be passed to CMake to activate optional components of Geant4 (*standard* and *advanced* options):
  - **-DGEANT4_INSTALL_DATA=ON** *(recommended)*
    the additional external data libraries are automatically downloaded
  - **-DGEANT4_INSTALL_EXAMPLES=ON** *(recommended)*
    examples are installed
  - **-DGEANT4_USE_OPENGL_X11=ON** *(recommended)*
    build the X11 OpenGL visualization driver
  - -DGEANT4_USE_SYSTEM_CLHEP=ON *(optional)*
    external CLHEP are required

You can directly include the options since the beginning:

```
cmake -DCMAKE_INSTALL_PREFIX=/path/to/geant4.9.5-install  -DGEANT4_INSTALL_DATA=ON
-DGEANT4_USE_OPENGL_X11=ON -DGEANT4_INSTALL_EXAMPLES=ON /path/to/geant4.9.5
```

Geant4 installation with CMake

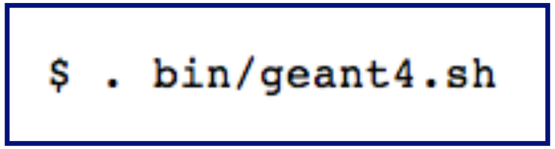
- If default installation paths have been chosen, the install of Geant4 is contained under the directory chosen (CMAKE_INSTALL_PATH), with the following structure:



```
+- CMAKE_INSTALL_PREFIX
   +- bin/
   |  +- geant4-config   (UNIX ONLY)
   |  +- geant4.csh      (UNIX ONLY)
   |  +- geant4.sh       (UNIX ONLY)
   |  +- G4global.dll    (WINDOWS ONLY)
   |  +- ...
   +- include/
   |  +- Geant4/
   |     +- G4global.hh
   |     +- ...
   |     +- CLHEP/       (WITH INTERNAL CLHEP ONLY)
   |     +- tools/
   +- lib/               (MAY BE lib64 on LINUX)
   |  +- libG4global.so  (AND/OR .a, OR G4Global.lib ON WINDOWS)
   |  +- ...
   |  +- Geant4-9.5.0/
   |     +- Geant4Config.cmake
   |     +- Geant4ConfigVersion.cmake
   |     +- Geant4LibraryDepends.cmake
   |     +- Geant4LibraryDepends-Release.cmake
   |     +- UseGeant4.cmake
   |     +- Linux-g++    (OR Darwin-g++ UNIX ONLY SOFTLINK -> ..)
   +- share
      +- Geant4-9.5.0
         +- data/        (IF GEANT4_INSTALL_DATA WAS SET)
         +- geant4make/
            +- geant4make.csh
            +- geant4make.sh
            +- config/
```

Geant4 installation with CMake

- To make Geant4 binaries and libraries available on your PATH and library path:



```
$ . bin/geant4.sh
```

- Now you are able to configure your environment to build your own application, by sourcing the setup script:

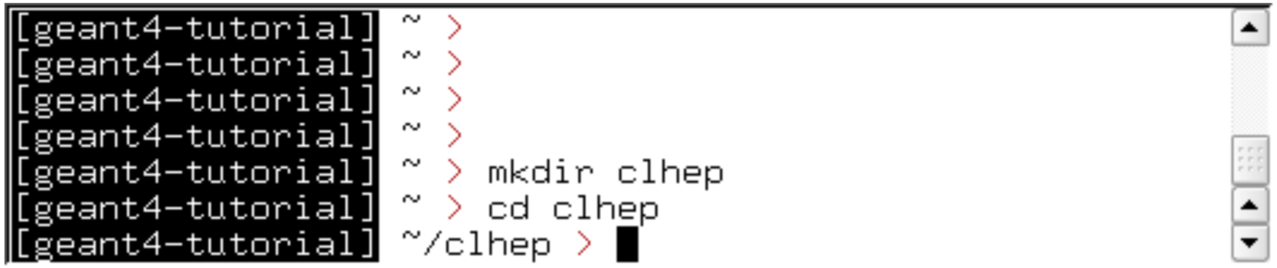
```
source geant4.9.5.p01-install/share/Geant4-9.5.1/geant4make/geant4make.sh
```

- Go inside the directory which contains your own application and try to compile it with *make*

Installing CLHEP full version

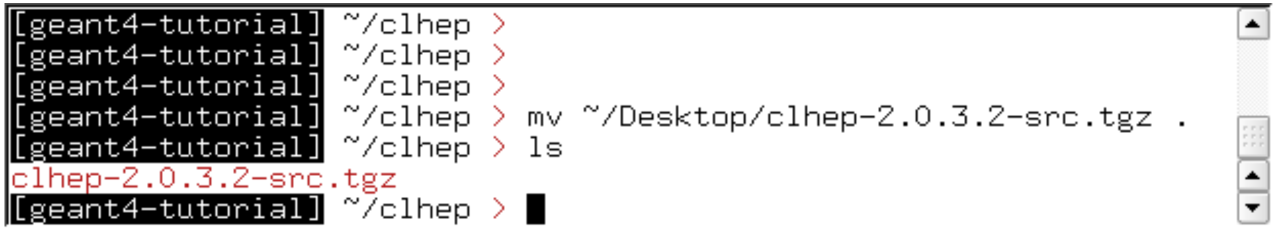
(not mandatory)

- Create a directory for the installation procedure (ex.:clhep)



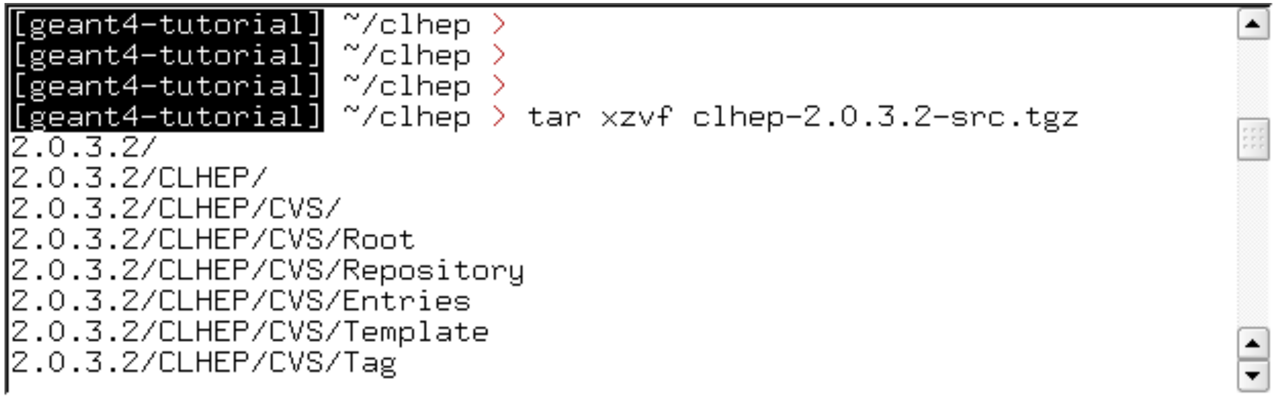
```
[geant4-tutorial] ~ >
[geant4-tutorial] ~ >
[geant4-tutorial] ~ >
[geant4-tutorial] ~ >
[geant4-tutorial] ~ > mkdir clhep
[geant4-tutorial] ~ > cd clhep
[geant4-tutorial] ~/clhep > █
```

- Move the downloaded tar-ball into this directory



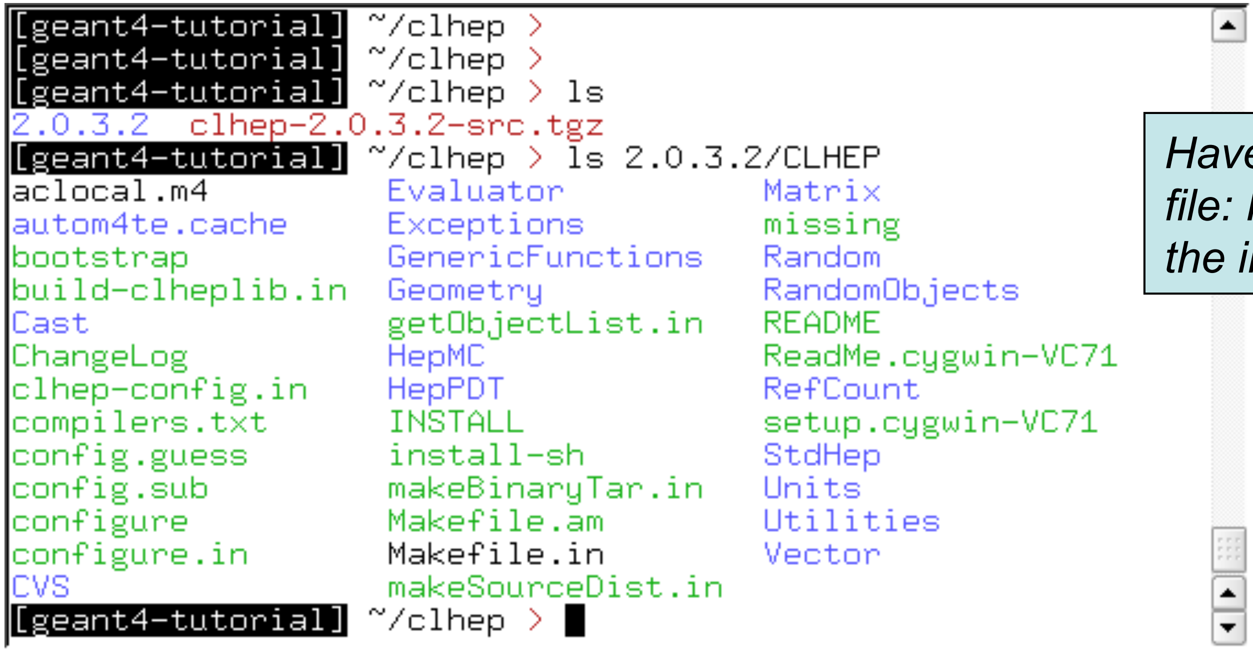
```
[geant4-tutorial] ~/clhep >
[geant4-tutorial] ~/clhep >
[geant4-tutorial] ~/clhep >
[geant4-tutorial] ~/clhep > mv ~/Desktop/clhep-2.0.3.2-src.tgz .
[geant4-tutorial] ~/clhep > ls
clhep-2.0.3.2-src.tgz
[geant4-tutorial] ~/clhep > █
```

- Unzip the extract tar-ball into this directory



```
[geant4-tutorial] ~/clhep >
[geant4-tutorial] ~/clhep >
[geant4-tutorial] ~/clhep >
[geant4-tutorial] ~/clhep > tar xzvf clhep-2.0.3.2-src.tgz
2.0.3.2/
2.0.3.2/CLHEP/
2.0.3.2/CLHEP/CVS/
2.0.3.2/CLHEP/CVS/Root
2.0.3.2/CLHEP/CVS/Repository
2.0.3.2/CLHEP/CVS/Entries
2.0.3.2/CLHEP/CVS/Template
2.0.3.2/CLHEP/CVS/Tag
```

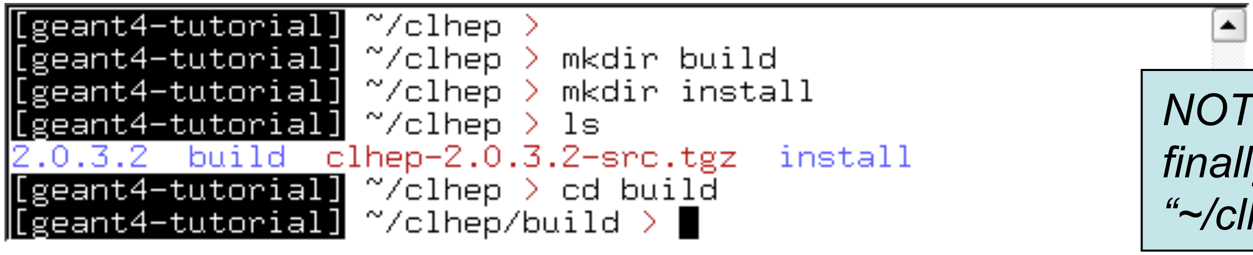
- The extracted CLHEP package can be found in the subdirectory 2.0.3.2/CLHEP”. Have a look at the content:



```
[geant4-tutorial] ~/clhep >
[geant4-tutorial] ~/clhep >
[geant4-tutorial] ~/clhep > ls
2.0.3.2  clhep-2.0.3.2-src.tgz
[geant4-tutorial] ~/clhep > ls 2.0.3.2/CLHEP
aclocal.m4          Evaluator           Matrix
autom4te.cache      Exceptions          missing
bootstrap           GenericFunctions    Random
build-clheplib.in   Geometry            RandomObjects
Cast                getObjectList.in    README
ChangeLog           HepMC               ReadMe.cygwin-VC71
clhep-config.in     HepPDT              RefCount
compilers.txt       INSTALL             setup.cygwin-VC71
config.guess        install-sh          StdHep
config.sub          makeBinaryTar.in    Units
configure           Makefile.am         Utilities
configure.in        Makefile.in         Vector
CVS                 makeSourceDist.in
[geant4-tutorial] ~/clhep > █
```

Have a look in the “INSTALL” file: It contains more details on the installation procedure

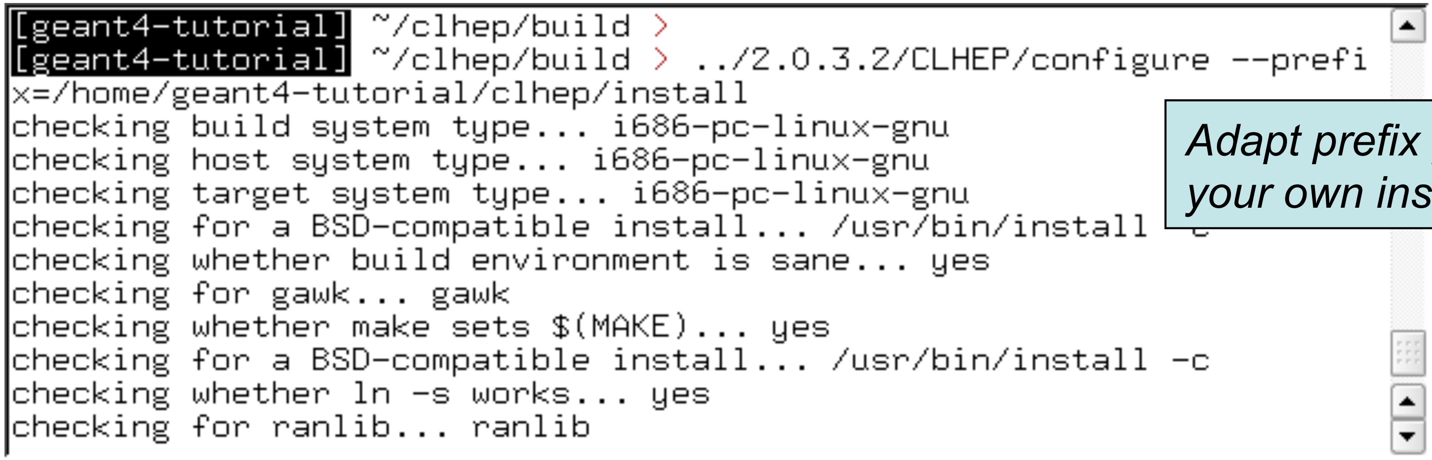
- Create two directories (inside our “clhep” directory), which are used for building and installing the package:



```
[geant4-tutorial] ~/clhep >
[geant4-tutorial] ~/clhep > mkdir build
[geant4-tutorial] ~/clhep > mkdir install
[geant4-tutorial] ~/clhep > ls
2.0.3.2  build  clhep-2.0.3.2-src.tgz  install
[geant4-tutorial] ~/clhep > cd build
[geant4-tutorial] ~/clhep/build > █
```

NOTE: The package will be finally installed in the directory “~/clhep/install”

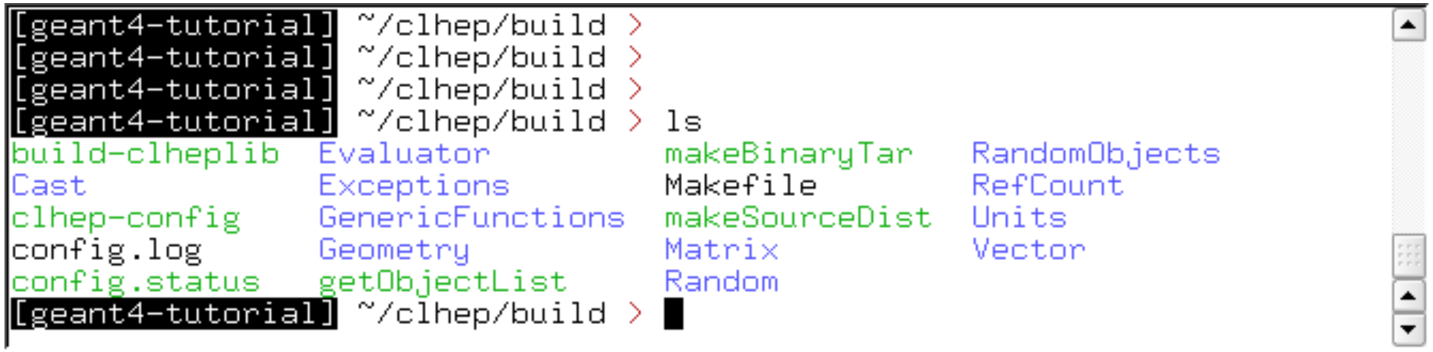
- Inside the “build” directory, call the CLHEP configure script (which is contained in the “2.0.3.2/CLHEP” directory).
  NOTE: As argument you need to specify the directory, where CLHEP should be installed. Thus the full command to be called is: ../2.0.3.2/CLHEP/configure --prefix=/home/geant4-tutorial/clhep/install



```
[geant4-tutorial] ~/clhep/build >
[geant4-tutorial] ~/clhep/build > ../2.0.3.2/CLHEP/configure --prefi
x=/home/geant4-tutorial/clhep/install
checking build system type... i686-pc-linux-gnu
checking host system type... i686-pc-linux-gnu
checking target system type... i686-pc-linux-gnu
checking for a BSD-compatible install... /usr/bin/install -c
checking whether build environment is sane... yes
checking for gawk... gawk
checking whether make sets $(MAKE)... yes
checking for a BSD-compatible install... /usr/bin/install -c
checking whether ln -s works... yes
checking for ranlib... ranlib
```

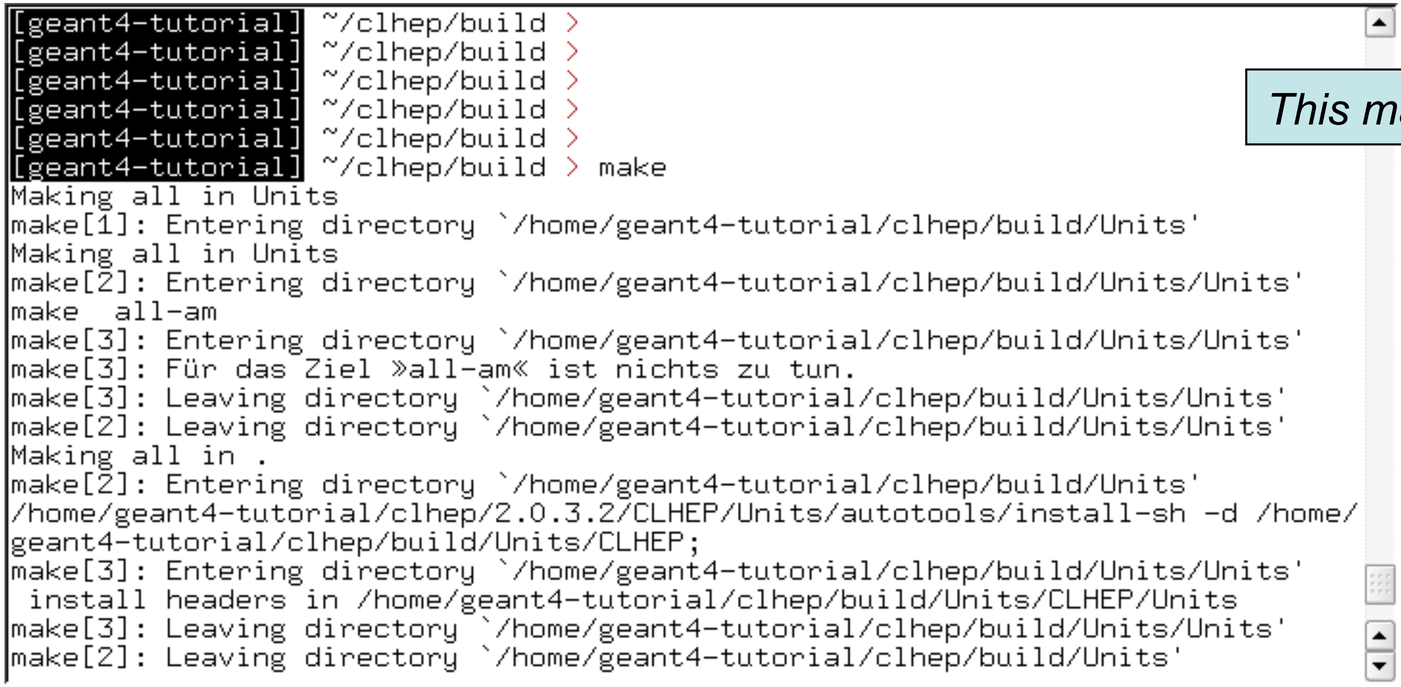
Adapt prefix path according to your own installation directory!

- The configure script checks for required programs and libraries,and creates some files, e.g. makefiles, and directories:



```
[geant4-tutorial] ~/clhep/build >
[geant4-tutorial] ~/clhep/build >
[geant4-tutorial] ~/clhep/build >
[geant4-tutorial] ~/clhep/build > ls
build-clheplib  Evaluator         makeBinaryTar   RandomObjects
Cast            Exceptions        Makefile        RefCount
clhep-config    GenericFunctions  makeSourceDist  Units
config.log      Geometry          Matrix          Vector
config.status   getObjectList     Random
[geant4-tutorial] ~/clhep/build > █
```

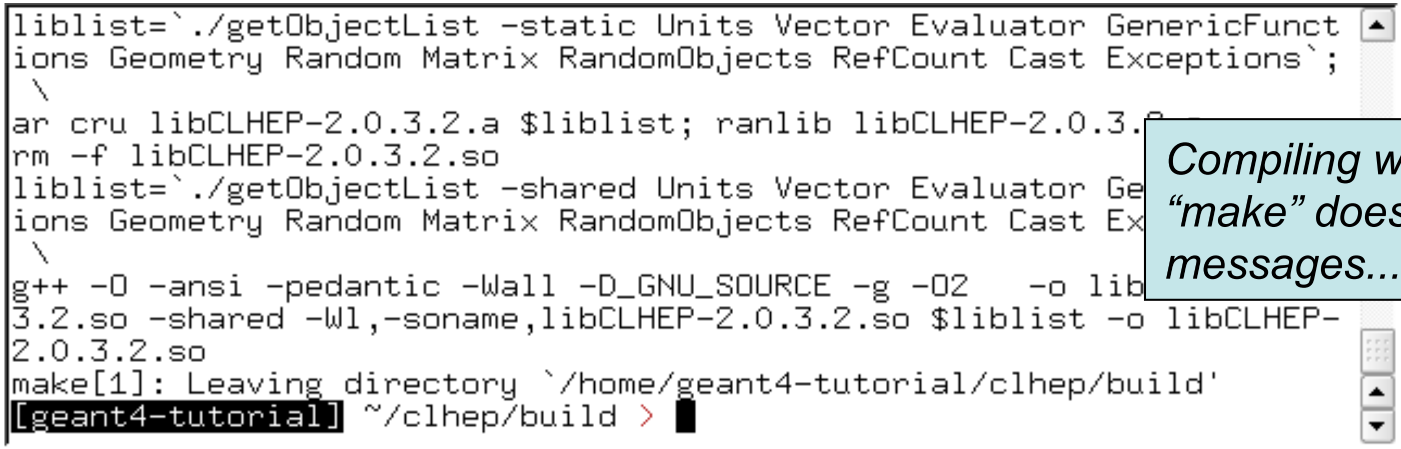
- If no error occured in the configure process, one can start to build the CLHEP package using the “make” command:



```
[geant4-tutorial] ~/clhep/build >
[geant4-tutorial] ~/clhep/build >
[geant4-tutorial] ~/clhep/build >
[geant4-tutorial] ~/clhep/build >
[geant4-tutorial] ~/clhep/build >
[geant4-tutorial] ~/clhep/build > make
Making all in Units
make[1]: Entering directory `/home/geant4-tutorial/clhep/build/Units'
Making all in Units
make[2]: Entering directory `/home/geant4-tutorial/clhep/build/Units/Units'
make  all-am
make[3]: Entering directory `/home/geant4-tutorial/clhep/build/Units/Units'
make[3]: Für das Ziel »all-am« ist nichts zu tun.
make[3]: Leaving directory `/home/geant4-tutorial/clhep/build/Units/Units'
make[2]: Leaving directory `/home/geant4-tutorial/clhep/build/Units/Units'
Making all in .
make[2]: Entering directory `/home/geant4-tutorial/clhep/build/Units'
/home/geant4-tutorial/clhep/2.0.3.2/CLHEP/Units/autotools/install-sh -d /home/
geant4-tutorial/clhep/build/Units/CLHEP;
make[3]: Entering directory `/home/geant4-tutorial/clhep/build/Units/Units'
 install headers in /home/geant4-tutorial/clhep/build/Units/CLHEP/Units
make[3]: Leaving directory `/home/geant4-tutorial/clhep/build/Units/Units'
make[2]: Leaving directory `/home/geant4-tutorial/clhep/build/Units'
```

This may take a while...

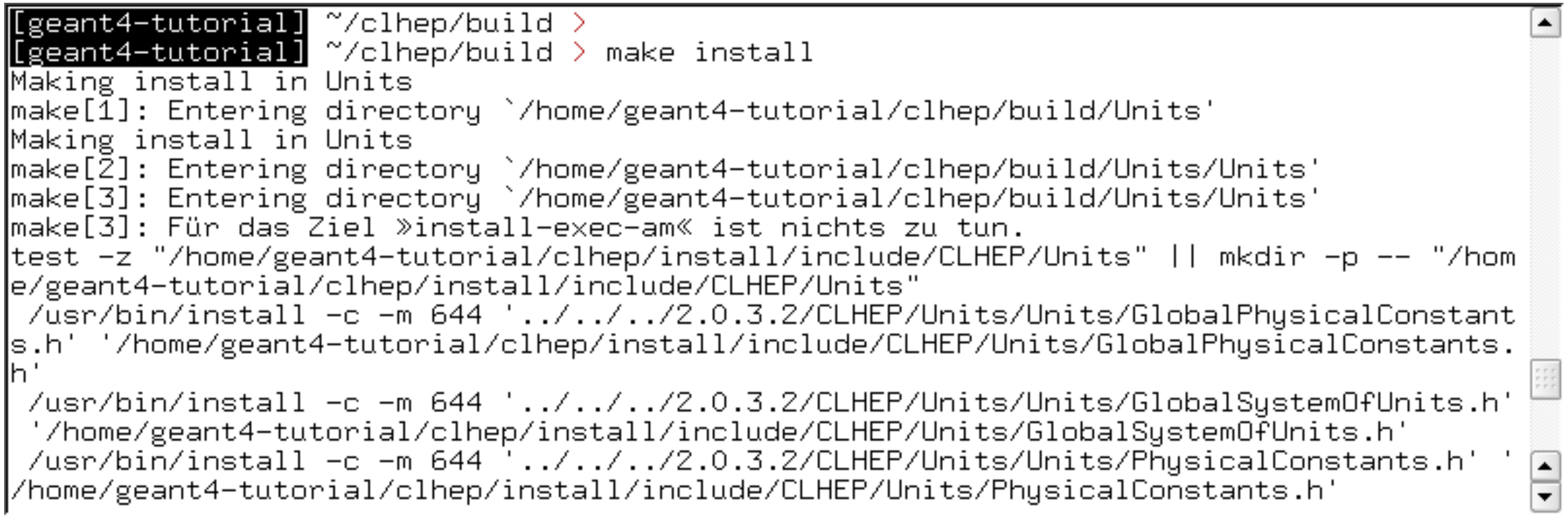
Only the initial and last output messages of the make command are shown



```
liblist=`./getObjectList -static Units Vector Evaluator GenericFunct
ions Geometry Random Matrix RandomObjects RefCount Cast Exceptions`;
 \
ar cru libCLHEP-2.0.3.2.a $liblist; ranlib libCLHEP-2.0.3.
rm -f libCLHEP-2.0.3.2.so
liblist=`./getObjectList -shared Units Vector Evaluator Ge
ions Geometry Random Matrix RandomObjects RefCount Cast Ex
 \
g++ -O -ansi -pedantic -Wall -D_GNU_SOURCE -g -O2   -o lib
3.2.so -shared -Wl,-soname,libCLHEP-2.0.3.2.so $liblist -o libCLHEP-
2.0.3.2.so
make[1]: Leaving directory `/home/geant4-tutorial/clhep/build'
[geant4-tutorial] ~/clhep/build > █
```

Compiling was successful if “make” does not exit with error messages...

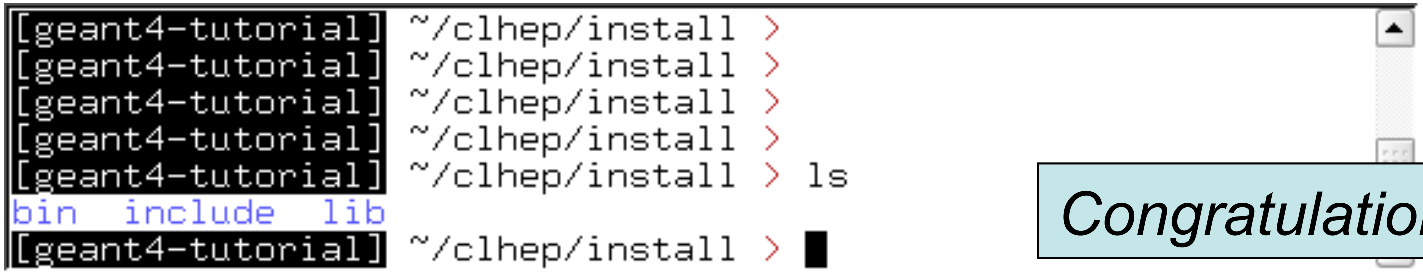
- Once the package was compiled successfully, CLHEP can be installed using the "make install" command:



```
[geant4-tutorial] ~/clhep/build >
[geant4-tutorial] ~/clhep/build > make install
Making install in Units
make[1]: Entering directory `/home/geant4-tutorial/clhep/build/Units'
Making install in Units
make[2]: Entering directory `/home/geant4-tutorial/clhep/build/Units/Units'
make[3]: Entering directory `/home/geant4-tutorial/clhep/build/Units/Units'
make[3]: Für das Ziel »install-exec-am« ist nichts zu tun.
test -z "/home/geant4-tutorial/clhep/install/include/CLHEP/Units" || mkdir -p -- "/hom
e/geant4-tutorial/clhep/install/include/CLHEP/Units"
 /usr/bin/install -c -m 644 '../../../2.0.3.2/CLHEP/Units/Units/GlobalPhysicalConstant
s.h' '/home/geant4-tutorial/clhep/install/include/CLHEP/Units/GlobalPhysicalConstants.
h'
 /usr/bin/install -c -m 644 '../../../2.0.3.2/CLHEP/Units/Units/GlobalSystemOfUnits.h'
 '/home/geant4-tutorial/clhep/install/include/CLHEP/Units/GlobalSystemOfUnits.h'
 /usr/bin/install -c -m 644 '../../../2.0.3.2/CLHEP/Units/Units/PhysicalConstants.h' '
/home/geant4-tutorial/clhep/install/include/CLHEP/Units/PhysicalConstants.h'
```

- The CLHEP libraries are now installed in the directory "~/clhep/install"

(NOTE: We specified the installation directory in the configure process; see the previous slides)

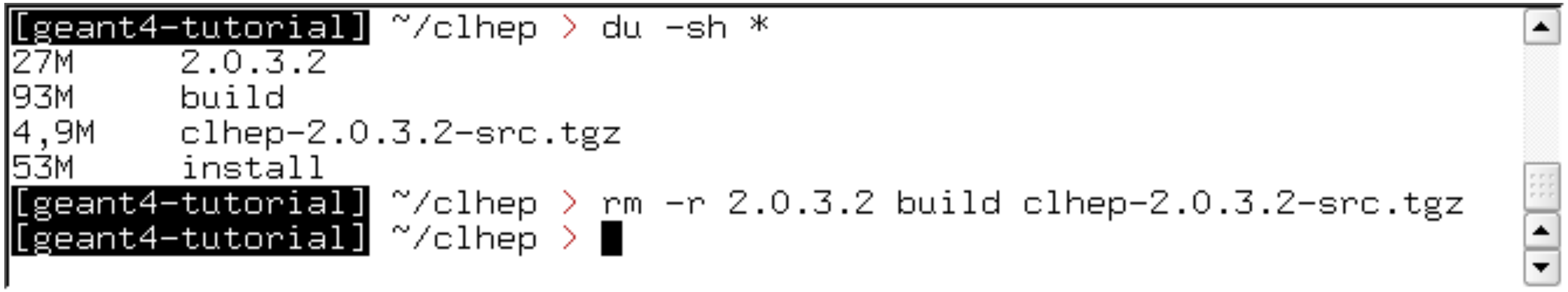


```
[geant4-tutorial] ~/clhep/install >
[geant4-tutorial] ~/clhep/install >
[geant4-tutorial] ~/clhep/install >
[geant4-tutorial] ~/clhep/install >
[geant4-tutorial] ~/clhep/install > ls
bin  include  lib
[geant4-tutorial] ~/clhep/install > 
```

Congratulations!

- What do the subdirectories in “~/clhep/install” contain?
  - include: Contains (in a defined directory tree structure) the C++ header files of CLHEP
  - lib: Contains the (static and shared) CLHEP libraries
  - bin: Contains configure scripts and the very useful “clhep- config” script

- Finally, to save some disk space, you can remove the “build” directory, as well as the tar-ball and the source package



```
[geant4-tutorial] ~/clhep > du -sh *
27M     2.0.3.2
93M     build
4,9M    clhep-2.0.3.2-src.tgz
53M     install
[geant4-tutorial] ~/clhep > rm -r 2.0.3.2 build clhep-2.0.3.2-src.tgz
[geant4-tutorial] ~/clhep > █
```

Thanks for your attention