

# SERESSA 2022

5<sup>th</sup> to 9<sup>th</sup> of December at CERN, Geneva

## G4SEE lecture preparation guideline

Dávid Lucsányi, CERN



# Contents



- ❑ 1) Installing Docker on your computer
- ❑ 2) Downloading the G4SEE Docker image
- ❑ 3) Starting a G4SEE Docker container
- ❑ 4) Running a G4SEE simulation inside a G4SEE Docker container
- ❑ [optional] Displaying visualization/GUI
- ❑ [optional] Building G4SEE from source
- ❑ User support – Links & Contact

# 1) Installing Docker

- Docker is an easy-to-install, cross-platform application that enables you to build, share and run **containerized applications** in loosely isolated environments called Docker containers. It's an ideal tool for sharing and running the G4SEE toolkit without building Geant4 or G4SEE from source!



## Docker installation steps



[Docker Desktop for Linux](#)



[Docker Desktop for Windows](#)



[Docker Desktop for Mac](#)

## 2) Downloading G4SEE Docker image



- A Docker container uses a custom and isolated filesystem, provided by a container image. Such an **image** contains everything needed to run the G4SEE application: Linux (Debian 11) environment, Python3.10, all dependencies incl. Geant4, scripts, compiled executables, source code, etc.

To download image using *docker pull* via CLI

G4SEE **v0.5 release** with Geant4 v11.0.3 (current release):

```
$ docker pull gitlab-registry.cern.ch/g4see/g4see:v0.5_G4-11.0.3
```

or

G4SEE **v0.5.1 release** with Geant4 v11.0.3 (next release):

```
$ docker pull gitlab-registry.cern.ch/g4see/g4see:v0.5.1_G4-11.0.3
```

CLI = Command Line Interface

### 3) Starting a G4SEE Docker container



- ❑ A **container** is a lightweight, runnable instance of an image. You can create, start, stop or delete a container. Everyone gets the same G4SEE container that works in the same way and has the same content inside.
- ❑ Sharing a folder (using docker's `-v` argument) between the host machine and the container is recommended, e.g. to share macro and output files.

To start a container using *docker run* via CLI

```
$ docker run -it -h g4see -v /host/path/to/shared_folder:/home \
gitlab-registry.cern.ch/g4see/g4see:v0.5_G4-11.0.3
```

Use a **real, absolute path of host machine** you want to access also from the container!  
After next release, replace **v0.5** with **v0.5.1**!

# 4) Running a G4SEE simulation



- ❑ In the container, copy example macros and create a new folder for outputs.
- ❑ Let's run a short example simulation with G4SEE to test it!
- ❑ After a successful multi-threaded run, merge the histogram files together!

## To run a simulation inside the container via CLI

```
root@g4see:/home# cp -r $G4SEE_SRC/examples /home
root@g4see:/home# mkdir output
root@g4see:/home# g4see -h
root@g4see:/home# g4see examples/SRAM_example.mac -o output/
root@g4see:/home# mergeHistograms output/
```

Merged histogram files in home dir.:

```
root@g4see:/home# ls
Edep_0_histogram.out
Edep_1_histogram.out
Ekin_0_histogram.out
Ekin_1_histogram.out
Ekin_2_histogram.out
examples
output
root@g4see:/home#
```

To quit a container press Ctrl+d keys, to detach it instead press Ctrl+p and Ctrl+q.  
To attach again a detached (but running) container use **docker attach** command.

# Displaying visualization/GUI

[optional]

- You might need to install the *X Window System* on your host machine to forward any visualization or GUI display: [Xming](#) (Win) or [XQuartz](#) (Mac)

## To start a container via CLI with visualization/GUI forwarding

On Linux hosts (For Windows and Mac hosts, [find steps here](#)):

```
$ export DISPLAY=:0.0
$ xhost +local:docker
$ docker run -it -h g4see -e DISPLAY=$DISPLAY \
    -v /tmp/.X11-unix:/tmp/.X11-unix \
    -v /host/path/to/shared_folder:/home \
    gitlab-registry.cern.ch/g4see/g4see:v0.5_G4-11.0.3
```

Depending on your display settings, the **0.0** value might be different for you.  
After next release, replace **v0.5** with **v0.5.1!**

# Building G4SEE from source



[optional, NOT recommended]

- ❑ Building without Docker, however using Docker is the recommended option
- ❑ Dependencies: Geant4 ( $\geq 10.7.0$ ), CMake ( $\geq 3.17$ ), Python ( $\geq 3.8$ )

## To clone and build G4SEE toolkit via CLI on Linux

Clone recursively the G4SEE GitLab repositories:

```
$ git clone --recursive https://gitlab.cern.ch/g4see/g4see.git
$ cd g4see
```

In the main repo's root dir., build the app with your Geant4:

```
$ mkdir build && cd build
$ export G4LIB=<Geant4_install_path>/lib64/Geant4-<version>/
$ cmake -DGeant4_DIR=$G4LIB ..
$ make -j <jobs>
$ sudo make install
```

(the last step is optional)



# User support – Links & Contact



- ❑ G4SEE Website: <https://g4see.web.cern.ch>
- ❑ G4SEE Documentation: <https://g4see-docs.web.cern.ch>
- ❑ G4SEE User Forum: <https://g4see-forum.web.cern.ch>
- ❑ G4SEE Main GitLab repository: <https://gitlab.cern.ch/g4see/g4see>
  
- ❑ G4SEE Developers' email: [g4see.toolkit@cern.ch](mailto:g4see.toolkit@cern.ch)

Contact us for additional help, or if you have any questions about G4SEE!