

The OpenCMT Castor Simulator

Requirements

The environment must be set up according to the [guide](#)

OpenCMT Castor Simulator: building the code

The source code for the Image Builder is stored in [Baltig](#):

```
git clone https://baltig.infn.it/muontomography/castor-simulator.git
cd castor-simulator
```

These are the steps required for building the Image Builder and install it in the system:

```
mkdir build
cd build
cmake3 -DCMAKE_INSTALL_PREFIX=/usr -DCMAKE_BUILD_TYPE=RelWithDebInfo \
-DPACKAGE_INSTALL_LIB_DIR=/usr/lib64 \
-DPACKAGE_INSTALL_INC_DIR=/usr/include \
-DPACKAGE_INSTALL_DATA_DIR=/usr/share \
-DENABLE_MT=ON ..
make
sudo make install
```

The libraries and header files are installed in the standard locations for the CentOS 7 system, so there's no need for further configurations.

The option **ENABLE_MT** requires the GEANT4 suite to be built with multi-thread support.

OpenCMT Castor Simulator: testing the application

The simulator requires two different configuration files: **g4config.in** and **muCastor.ini**. The templates for those files with basic definitions are located in the **setup** directory of the castor-simulator project.

The configuration files must be located in the working directory. The application stores all the files in the directory **output** created in the working directory.

A simple test is:

```
simulate_muCastor -run 1 -events 100000
```

The Castor visualization tool

The simulator suite provides a visualization tool based on the Event Visualization Environment for ROOT. The tool is encapsulated in a ROOT macro, installed in the standard location for ROOT macros:

```
$ root
-----
| Welcome to ROOT 6.22/02                               https://root.cern |
| (c) 1995-2020, The ROOT Team; conception: R. Brun, F. Rademakers |
| Built for linuxx86_64gcc on Aug 17 2020, 12:46:52 |
| From tags/v6-22-02@v6-22-02 |
| Try '.help', '.demo', '.license', '.credits', '.quit'/.q' |
-----

root [0] .X runCastorEve.C(1)
```

The argument of the macro is the run number of the simulation. The tool requires the file **g4config.in** to be placed in the working directory, and the data files to be stored in the **root** subdirectory.



The macro needs also a symbolic link to the root subdirectory:

```
ln -s $PWD/root $PWD/../root
```