

# eAST Documentation: a quick update (plus a few other thoughts)

---

Maxim Potekhin

*Nuclear and Particle Physics Software Group*



*Project eAST Meeting*

10/12/2021




# Summary

---

- The prototype of the eAST documentation website was introduced ~3 weeks ago
- To conserve development effort, we chose Jekyll/GitHub pages as the solution
- Streamlined design, easy to edit, contribute to etc
- It's not mobile-friendly yet, we may revisit this issue later and change the layout
- Adjustments and additions were made recently (next two slides)

# The Updates



**eAST**

- Home
- Installation **TOC**
- User Manual
- Project proposal
- Open Tasks
- The Team
- Contribute to this site

- eAST on GitHub
- Geant 4
- EIC Software Group

Site built  
2021-10-12 00:24:57 +0000

## Installation Procedure

- [GEANT](#)
- [The Build](#)
- [Configuration options](#)
- [HepMC support](#)

## GEANT

eAST runs on top of the latest public version of Geant4 (currently [version 10.7.p02](#)), so before building eAST you need to install Geant4. Please refer to [the Geant4 installation guide](#) and make sure you set the following options:

Option	Comment
GEANT4_BUILD_MULTITHREADED	-
GEANT4_USE_GDML	<b>Mandatory.</b> Requires <a href="#">Xerces-C++</a> parser.
GEANT4_USE_OPENGL_X11	Needed if you use the event display. Requires <a href="#">X11</a> and <a href="#">OpenGL</a> libraries.
GEANT4_USE_QT	<b>Mandatory.</b> Requires <a href="#">Qt5</a> and <a href="#">OpenGL</a> libraries.
GEANT4_INSTALL_DATA	Install datasets (recommended).


A command line for `cmake` using some of these options might look like this:

```
cmake -DCMAKE_INSTALL_PREFIX=/install/path -DGEANT4_INSTALL_DATA=ON -DGEANT4_USE_GDML=ON -DGEANT4_USE_QT=ON /path/to/geant/directories
```

If the install path is outside the user folder e.g. under "opt", one may need to use root identity or "sudo" to get the necessary privileges.

Building is done via `cmake`, using the `CMakeLists.txt` file at the top level. It is recommended to use a separate build directory. Please make sure all the necessary Geant4 environment variables are set in advance.

# The “Open Tasks” page



Home

Installation

User Manual

Project proposal

Open Tasks

The Team

Contribute to this site

eAST on GitHub

Geant 4

EIC Software Group

Site built  
2021-10-11 21:15:16 +0000

## Open Tasks

- [CAD import](#)
- [Fast Simulations](#)
- [Full Simulations](#)
- [Documentation](#)
- [Promotion](#)

---

## CAD import

- MOU with INFN as we plan to use their converter
- Verify that we can import the latest CAD files from the EIC project
- Make the latest GDML files available as part of the east-geometry repository with Git LFS support
- Document the CAD import
- Test the silicon tracker CAD model by James Fast

## Fast Simulations

- Create an example calorimeter geometry in eAST
- Give an example for fast simulations in eAST, perhaps for one region (Kolja, Maxim, Makoto)

## Full Simulations

- Afterburner (Markus will discuss with Jin)
  - Including option to turn off pre-assigned decay procedure
- Implementation of detector components
  - Separate file (either C++ or GDML) for each component
  - Associated sensitive detector is preferred but not mandatory
  - Define detector regions
- Import detector components from ATHENA or ECCE in eAST, including sensitive detectors if available
- Physics List

# A few thoughts about the content

---

- The website is targeted mainly towards users, and to a smaller extent, developers
  - ... a guess - in 80% to 20% ratio
- For better engagement, the users should quickly see the goals, design principles and interface features - all of this is yet to be developed
  - The existing text of the eAST proposal explains some of the CAD interface aspects, but that's not all that the users would want
- Tutorials (can host on GitHub), screenshots need to be created
- There is plenty of space on the front page to place links and intros
- Do we need a dedicated page for Fast Simulations? CVMFS guidance?

# User engagement: containers

---

- Do we expect users to build from scratch on their workstations/laptops? To what extent we expect users to leverage CVMFS? Are we going to stay with the OSG?
- Should we consider containerization even now, as a test of capability?
  - Flatten the learning curve for the users to encourage adoption
  - NB. container-based development has become fairly standard
  - Capturing simulation workflows in REANA is a definite bonus
- Observation - solid support for Docker on Windows/WSL2 and Mac
  - As a bonus, a few Linux flavors are available under WSL2 (Debian etc)
- Side effect of creating an image is a working and complete example of building eAST, encapsulated in a Dockerfile... there is relevant experience in this group

# User engagement: interactive tools and graphics

---

- Jupyter notebooks are gradually becoming mainstream
- Both Python and Julia are used, with a degree of interoperability
- Powerful visualization tools are available in either language
- Packages exist to parse GDML
- Potential exists to make eAST more accessible to users and developers