Framework for a future μ→eγ experiment

This talk

… is not

- a presentation of a definite solution
- about bashing existing solutions

… is

- about the “state of the art”
- trying to give an input to a discussion

Existing frameworks

MEG

- Relies heavily on rome3
  - rome3 is more or less abandoned (occasionally receives life support from Yusuke)
- Relatively high barriers to entry:
  - In a year I haven’t managed to
    - set up a development environment on my laptop
    - reliably use incremental compilation (probably skill isue)
    - **really** compile with multiple threads (`warning: jobserver unavailable: using -j1`)

Mu3e

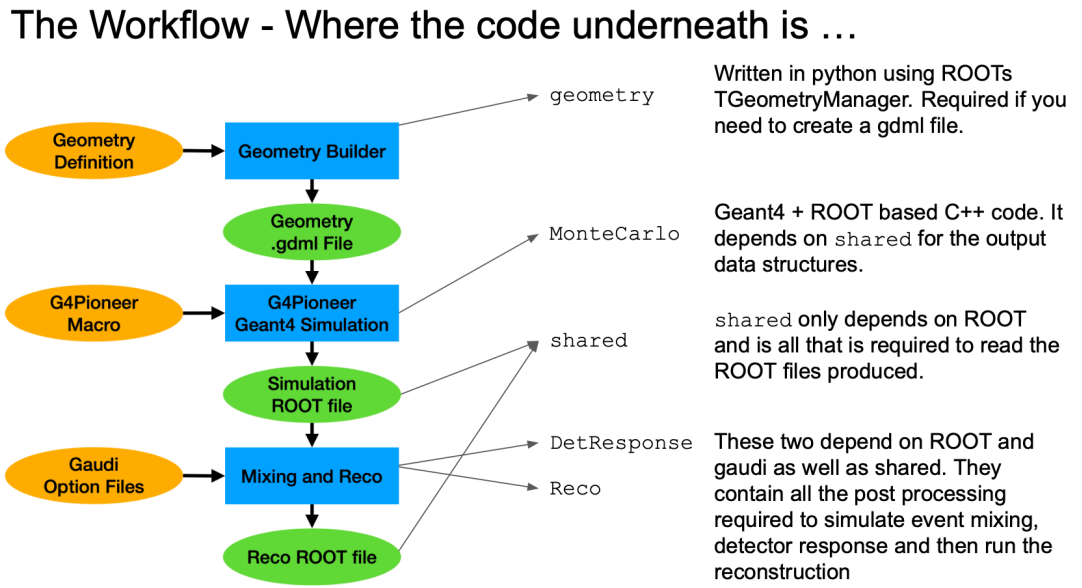
- Continuous readout is reflected in simulation and analysis

⇒ More or less custom everything (Geant4 based simulation)

- Probably not much to learn for event-based experiments

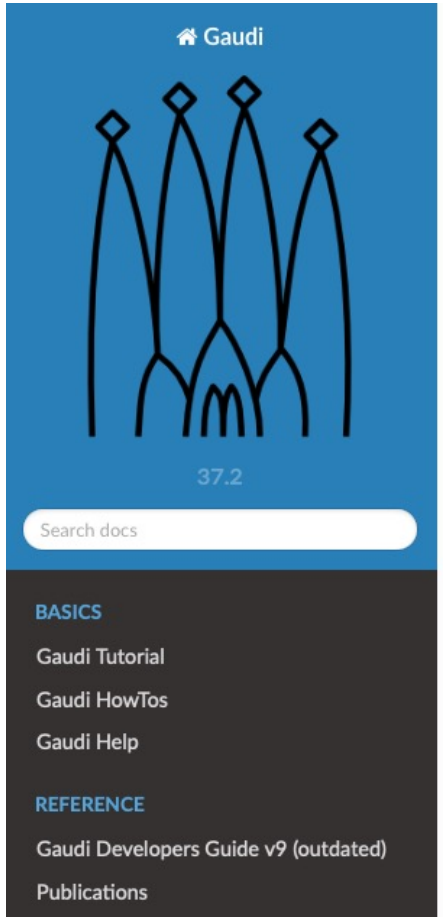
PIONEER

- Patrick explained their framework to me:
- Seems reasonable
- Gaudi was the most important news (to me)



More information: https://indico.psi.ch/event/15146/overview

Gaudi: Widely used (and well-maintained)



» Welcome to the Gaudi Project documentation View page source

Welcome to the Gaudi Project documentation

Gaudi is a framework software package that is used to build data processing applications for High-Energy Physics experiments. It contains all of the components and interfaces to allow you to build event data processing frameworks for your experiment.

Gaudi scales to the needs of the most demanding experiments at the LHC, but is simple enough to get started quickly and have an application running in just a short time.

Gaudi has been in production for the ATLAS and LHCb experiments and others for many years and is also the framework used by the *Key4hep* common software for Future Collider studies such as FCC, CLIC/ILC and CEPC. A (non-exhaustive) list of framework users:

- LHCb Computing
- ATLAS Athena framework
- HARP Gaudino framework
- Fermi (previously GLAST)
- MINERvA
- BESIII BOSS framework
- LBNE (Long Baseline Neutrino Detector, WCD group), see also GARPI project
- Key4hep (common software for FCC, CLIC/ILC and CEPC)

Key4hep

turnkey

adjective [before noun] US

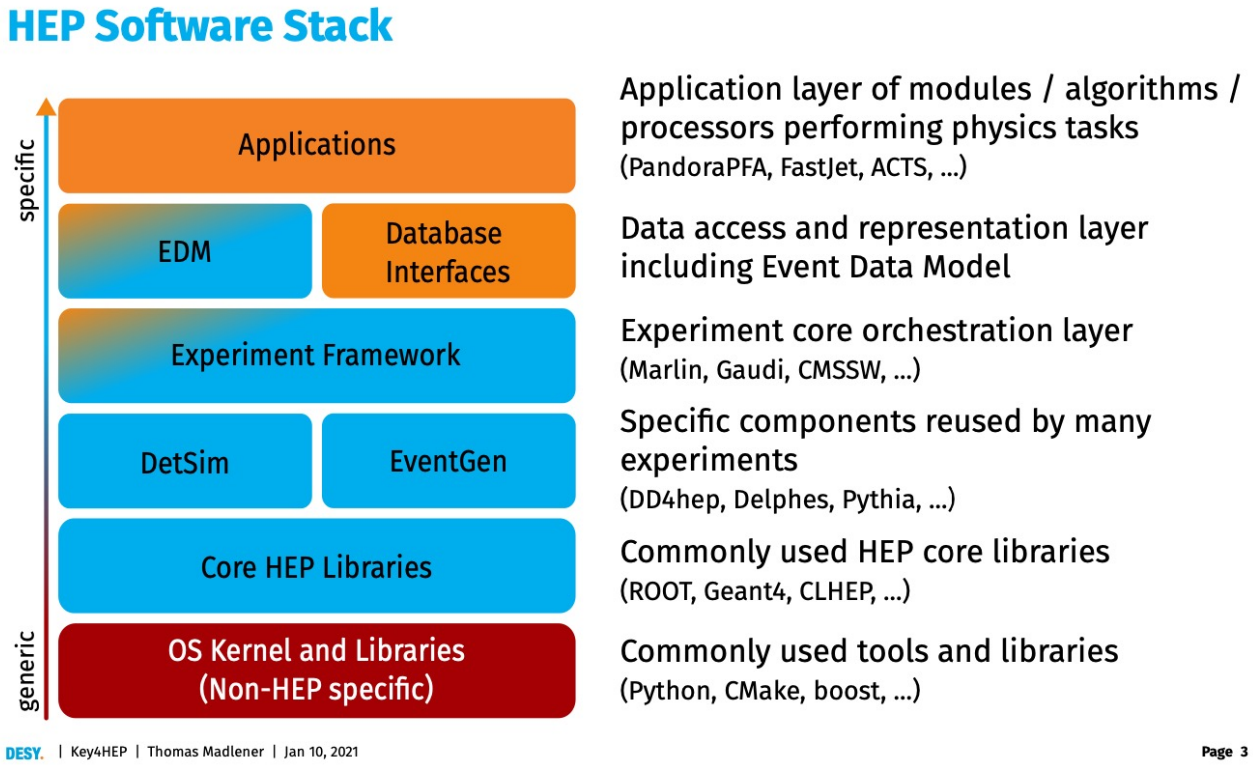
UK /ˈtɜːnˌkiː/ US /ˈtɝːnˌkiː/

Add to word list

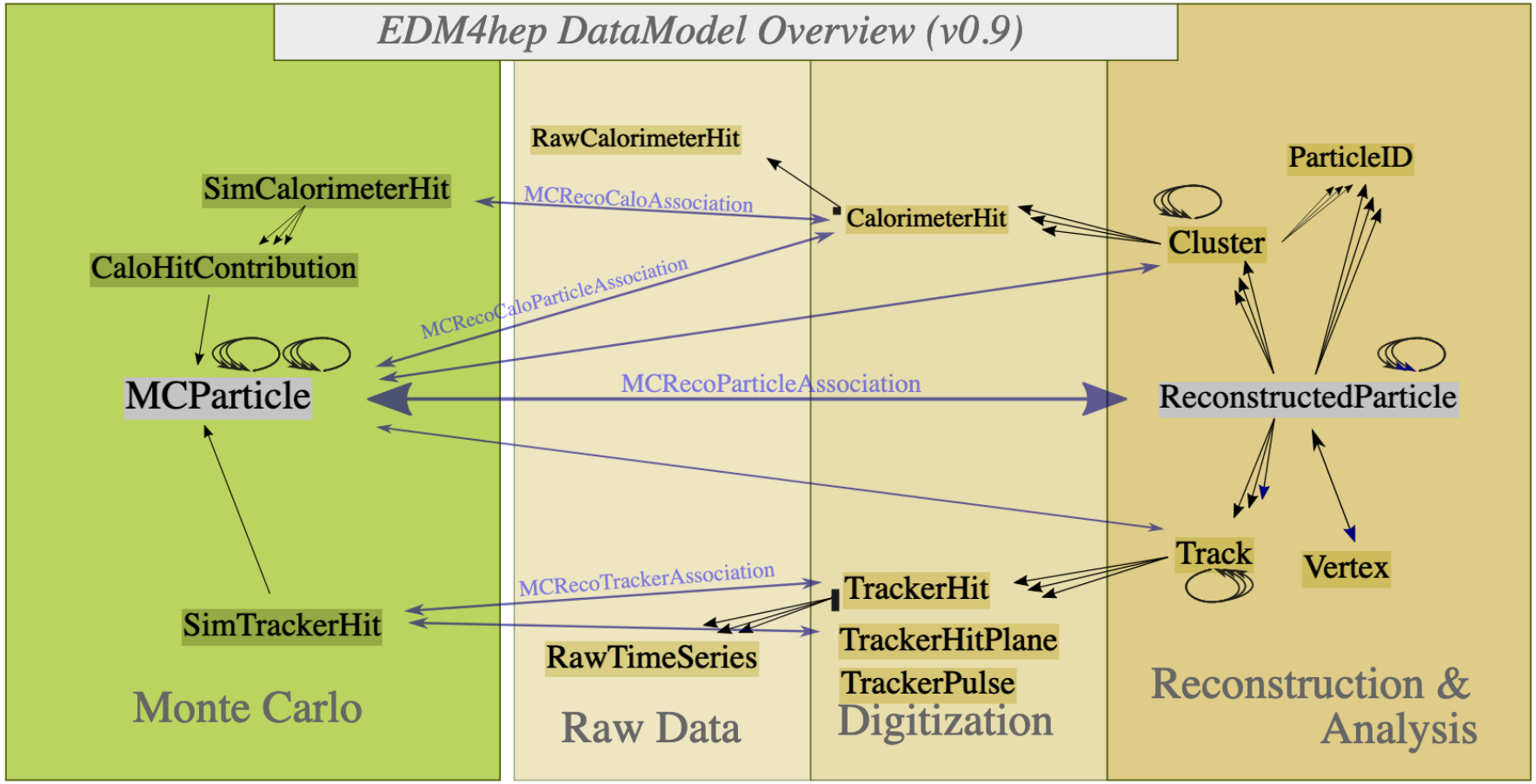
(of a property or a piece of equipment) ready to be used immediately by the person who is buying or renting it, or relating to this arrangement:

- *Some people hire a contractor to build turnkey houses - ready to move into with no renovations or repairs required.*
- *We have developed turnkey solutions designed specifically for small business owners.*

- Gaudi is actually part of Key4hep, “turnkey software for future accelerators”



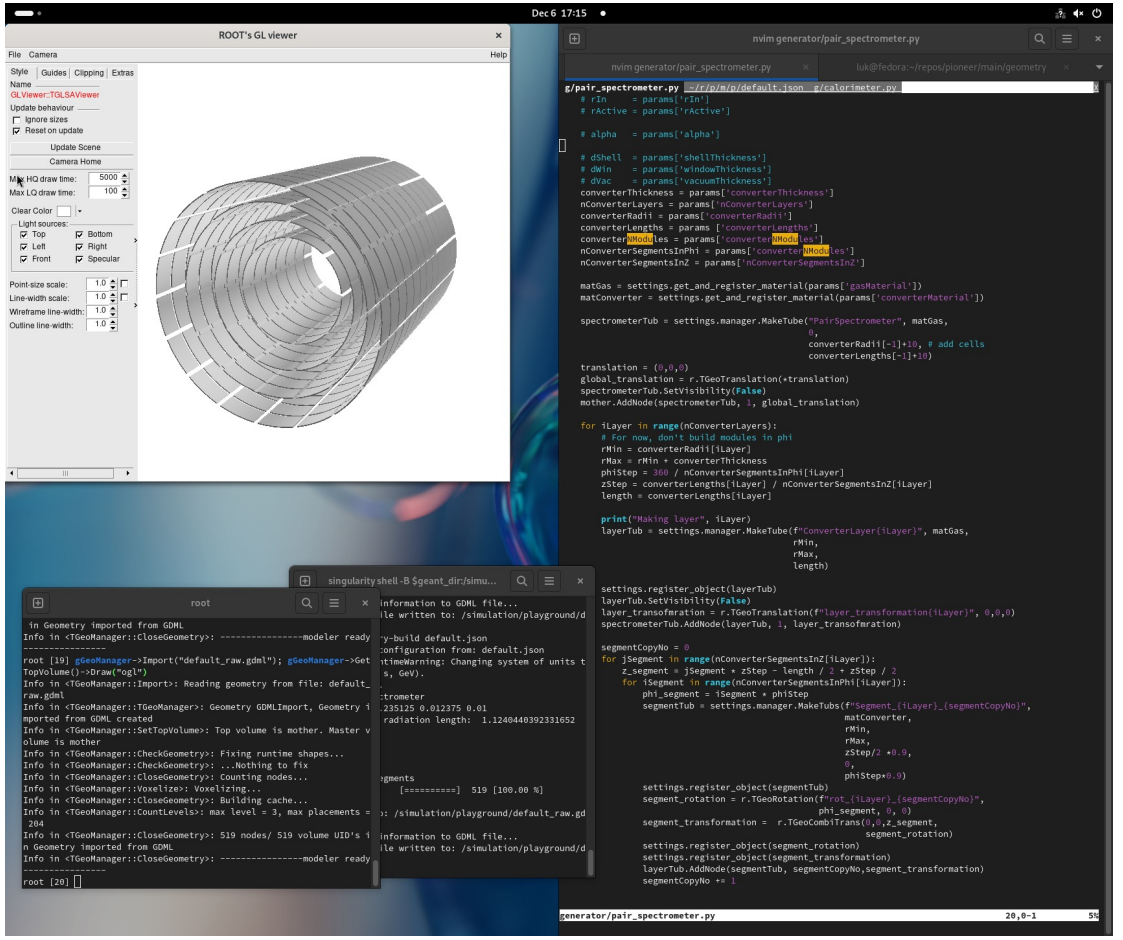
EDM4HEP (Event Data Model)



Is it useful for us?

My personal experience of the recent past

- Playing around with PIONEER's framework (implementing MEG III stuff)
- Geometry is quite easy
- MC looks like straightforward G4 code (we can probably port a lot of what we already did)
- Reco relies on Gaudi



Discussion: How much HEP do we want?

Other points:

- Development environment? (Containers? Might be good to start with singularity early on)
- Hosting: are we happy with bitbucket? (PIONEER uses github)
- Development: In the photon analysis, we tried out Jira, seems nice, but adds overhead
  - We should have an issue system either way, currently disabled for meg2 (why?)
- Even more technical discussions, should we call for a workshop for this?