Software Situation for EIC Detector Simulation

Yue Shi Lai

LBNL RNC

UC Consortium

Overview

- BNL EicRoot
  - `https://wiki.bnl.gov/eic/index.php/Eicroot` (outdated)
  - `https://git.racf.bnl.gov/gitea/EIC/EicRoot.git`
  - `https://hub.docker.com/u/ayk1964/`
  - GEANT 3, FairSoft/FairRoot, PANDA-based
  - Has an implementation of BeAST
  - By A. Kiselev, but development stopped $\approx$ 2016
  - Used/named as the preferred tool by eRD16, eRD18, eRD21
- BNL Fun4All
  - GEANT 4-based
  - "ePHENIX" probably exists, but no recipe for EIC detector study (yet)

Overview

- JLab eJANA
  - GEANT 4-based
  - `https://gitlab.com/eic/ejana`
  - `https://hub.docker.com/u/electronioncollider/`
- ANL TOPSiDE
  - `https://hub.docker.com/u/electronioncollider/` (very recently)
  - Any instruction is unknown to me

Tracking situation

- EicRoot/FairSoft/PANDA
  - Single track (but can group them to an event boundary)
  - No vertex reconstruction
  - Kalman filter is “cheated”, i.e. truth information is used to associate hits
  - Digitization is Gaussian smear, no pixel occupancy effect
  - Non-helical tracks probably breaks reco
  - No specialized electron tracking
- GENFIT/RAVE
  - Event reconstruction
  - RAVE is **not the actual CMS vertex fitter**
  - RAVE is from the same Vienna group, but CMS does not endorse all their algorithms (and favors their older, conservative ones)
  - RAVE does not talk to GENFIT, unlike actual CMSSW with vertex refit
  - No specialized electron tracking (also unlike CMS)

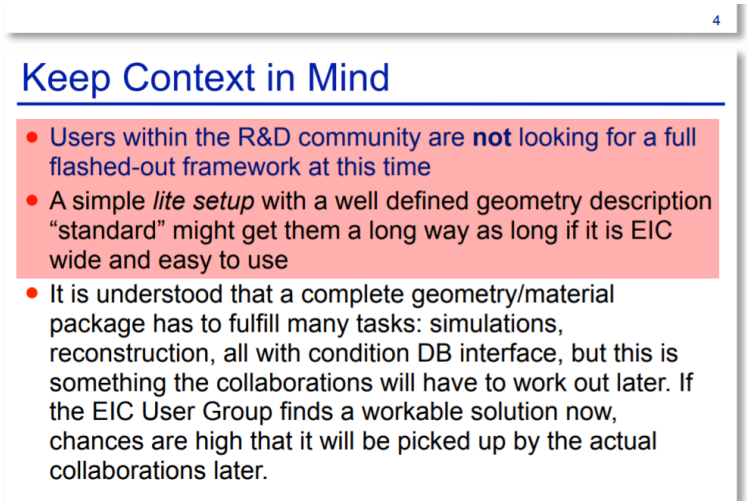
Tracking situation

- ACTS
  - Event reconstruction
  - Does vertex refit
  - Gaussian sum filter for electrons
  - There is no such thing as a TPC
  - Requires C++17 (e.g. issues with TGeo exchange with old C++ FairRoot)

Observation

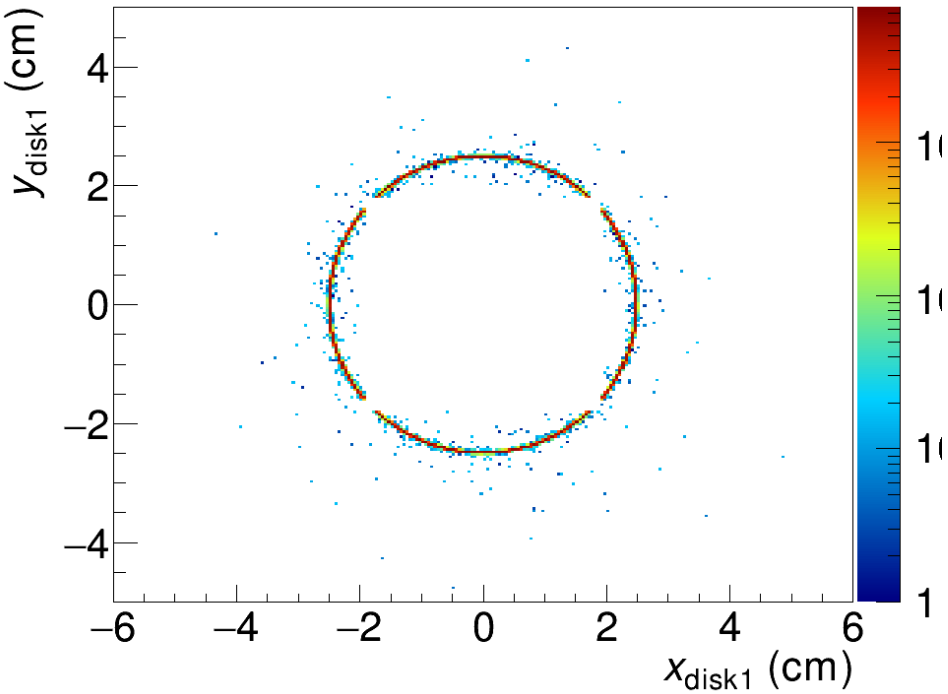
- Most packages are site-specific (e.g. no BNL BeAST geometry in JLab's eJANA and vice versa)
- Plain C++ GEANT 4 not user-friendly for R&D, e.g. geometry scans
- Example:
  - EicRoot contains ALICE-like MAPS staves
  - Allows the construction of additional layers with few lines
- Only geometry exchange via (C++ GEANT 4) reached consensus between BNL/JLab(/ANL)
- Demand for "lightweight framework for R&D" lacks interest/funding from the software groups

T. Ullrich, eRD20 Meeting, July 2019



Installing EicRoot

- Method 1: Docker
- Method 2:
  - Install FairSoft/FairRoot (`https://fairroot.gsi.de/`)
  - (Install eic-smear to process event structure)
  - Get an BNL RACF account and access the Gitea
  - Follow `https://wiki.bnl.gov/eic/index.php/Eicroot#Compiling_the_Code`
- eRD16 used method 2 on NERSC's Cori, and local (laptop) VMware, possibly transition to Docker in the future (eic-smear dependency issue)

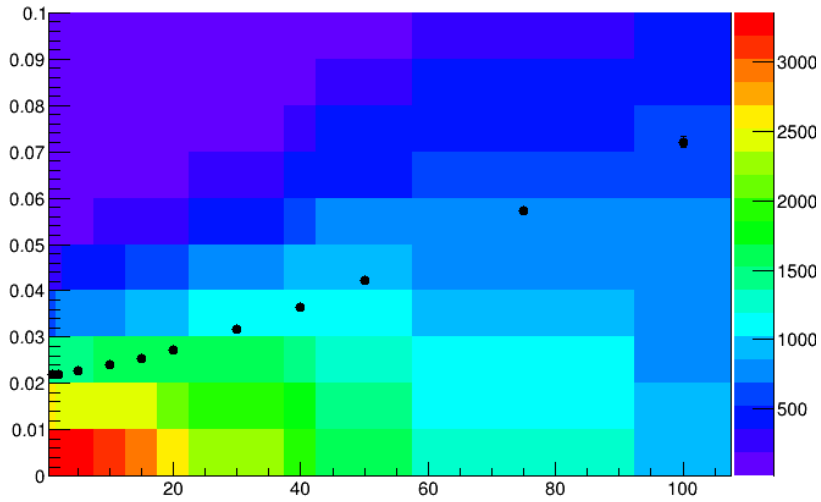


1.7 M tracks "imaging" of the first FBST disk, taking advantage of Cori's 512 GB RAM/node, despite memory leak in EicRoot

Simulation/Visualization

BeAST workflow:

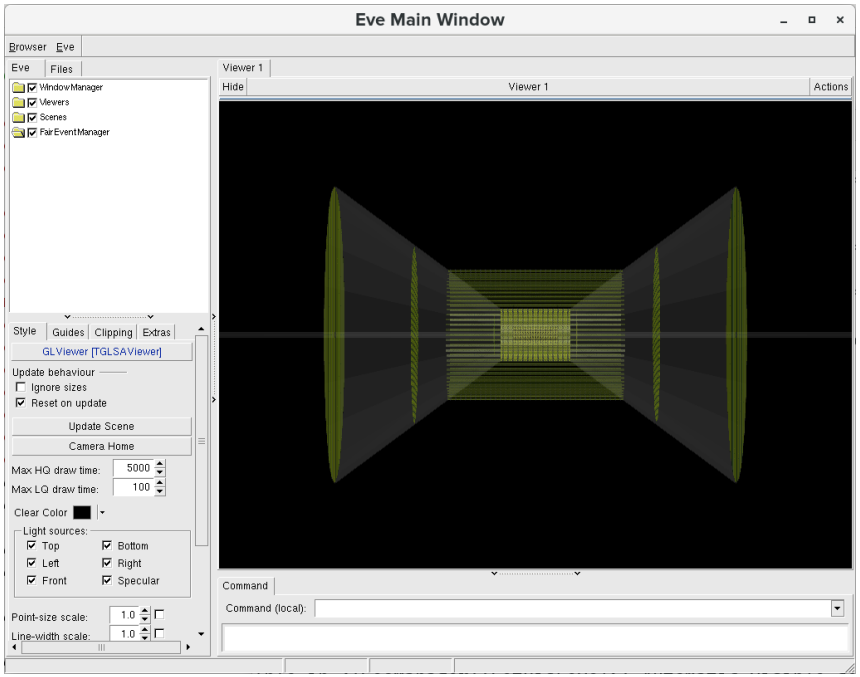
- `simulation.C` (GEN-SIM)
  - ROOT output contains hits and TGeo
- `digitization.C` (DIGI)
- `reconstruction.C` (RECO)
  - ROOT output containing truth vs. reco tracks
- `analysis.C`
- GEANT hit data structure lacks example, hence most studies done by instrumenting the GEANT code



Simulation/Visualization

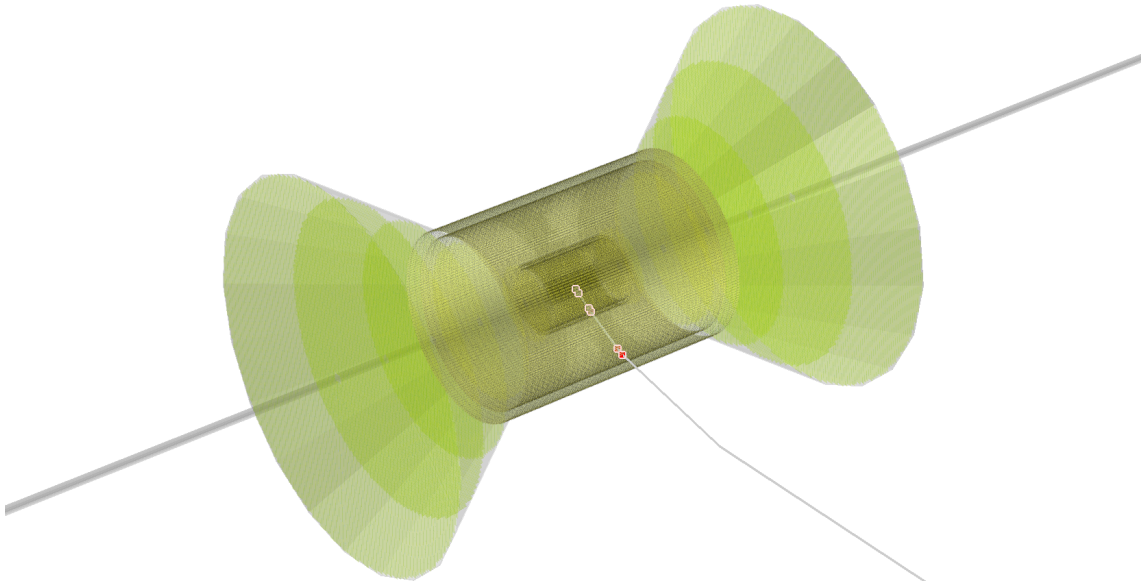
BeAST workflow:

- `simulation.C` (GEN-SIM)
  - ROOT output contains hits and TGeo
- `eventDisplay.C`



All-Si tracking in EicRoot

- Adding barrel layers (≈ ALICE radii) to VST
- Adding variable FBST disk size
- Adding "support structure" as TGeoPcon
  - Beware EicRoot is in mm, GEANT/TGeo is in cm



Perspective view of the fully projective all-Si tracking

What's next (eRD16 perspective)

- Integration of a full event track reconstruction/vertexing
  - Scenario: "Marry" EicRoot with eJANA/GENFIT
  - Scenario: Interface with ACTS
- Ramp up vertexing studies in the coming month(s)
- Short term: Do what ever is pragmatic (e.g. "marrying" existing EIC tools)
- Possible long term preference towards ACTS