EicRoot software framework

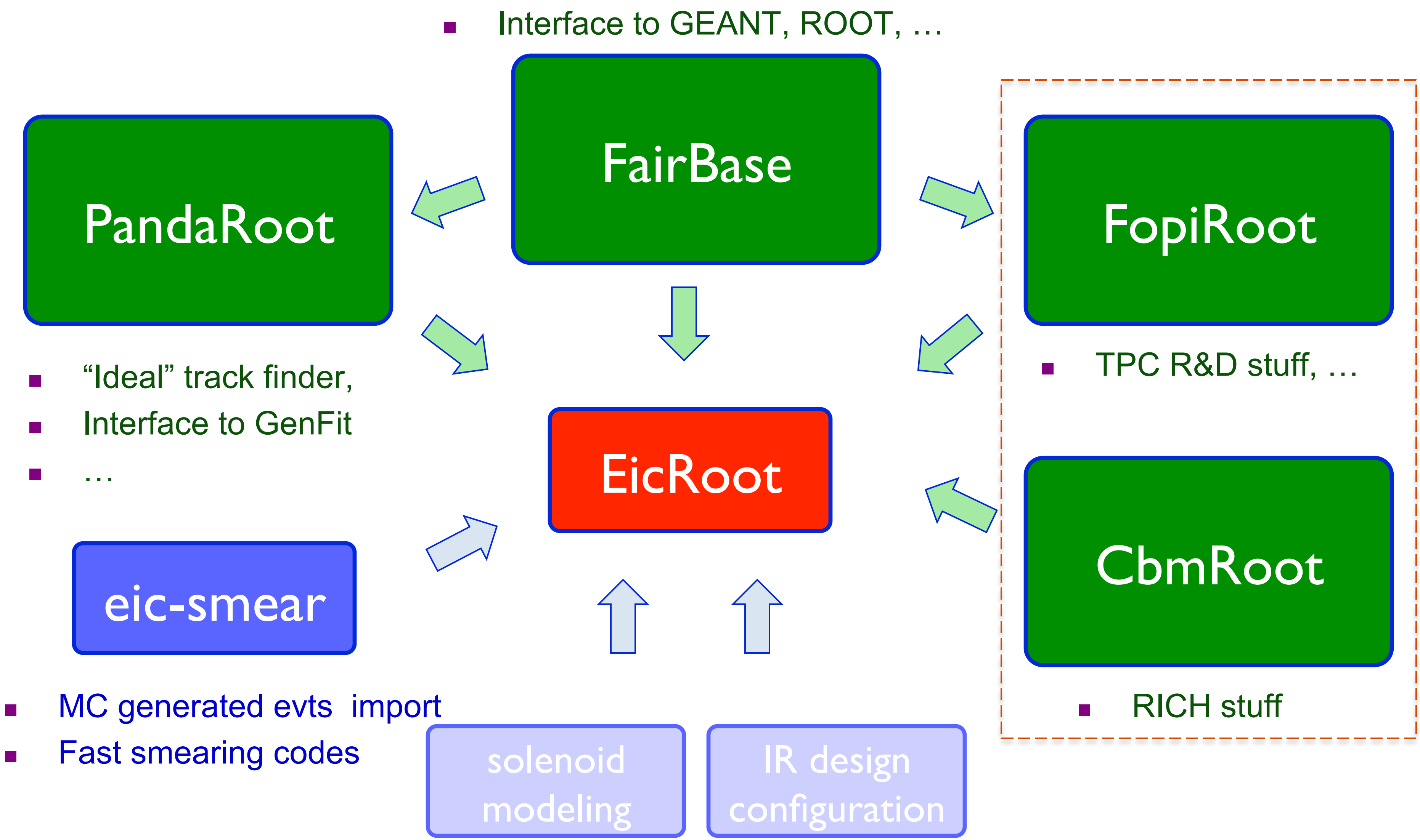
Alexander Kiselev

EIC Software Meeting

~~Jefferson Lab September,24 2015~~

Brookhaven Lab July,10 2019

EicRoot framework building blocks



- ROOT-based I/O, VGM, VMC (e.g. easy switch between G3 & G4)

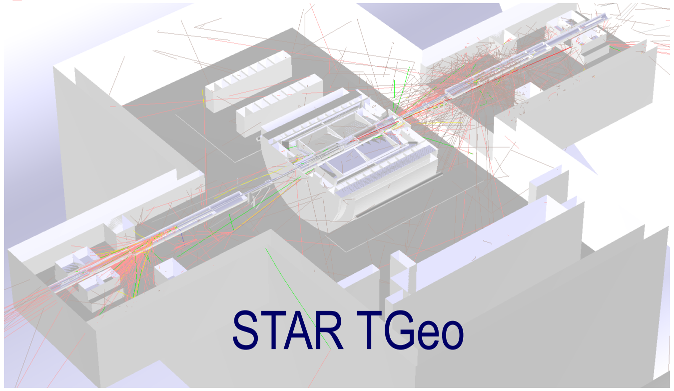
End user view

- No executable (steering through ROOT macro scripts)

simulation	digitization	reconstruction	PID; assembly
-> MC points	-> Hits	-> Tracks, clusters	-> Events

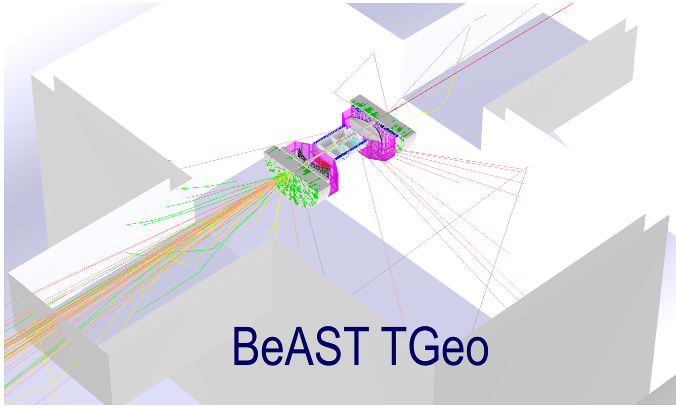
- ROOT files for analysis available after each step
- C++ class structure is well defined at each I/O stage

Geometry description



- Input formats:
  - ROOT TGeo (with mapping)
  - GEANT GDML
  - Old HADES .geo files

  - CAD design drawings (.stp, .stl)



- Output format:
  - ROOT TGeo (geometry modules used for a particular simulation run are assembled on-the-fly into a TGeoManager instance and written out into the output simulation.root file used by digitization and reconstruction)

Typical applications

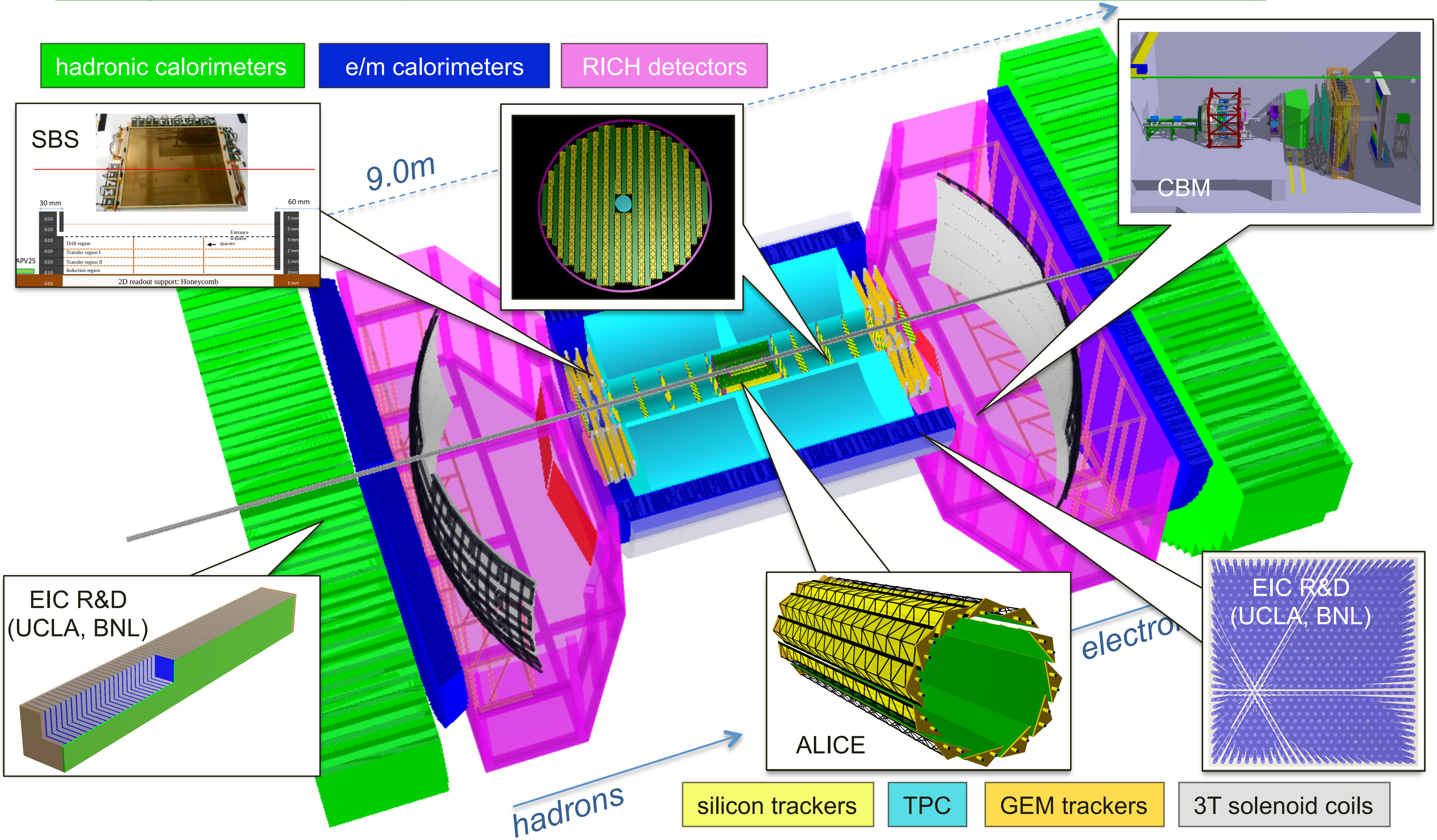
- Physics analysis
- Test beam data analysis
- R&D studies
  - Tracking
  - Calorimetry
  - RICH simulations (FopiRoot codes adapted)
  - TPC modeling (CbmRoot codes adapted)

(?)

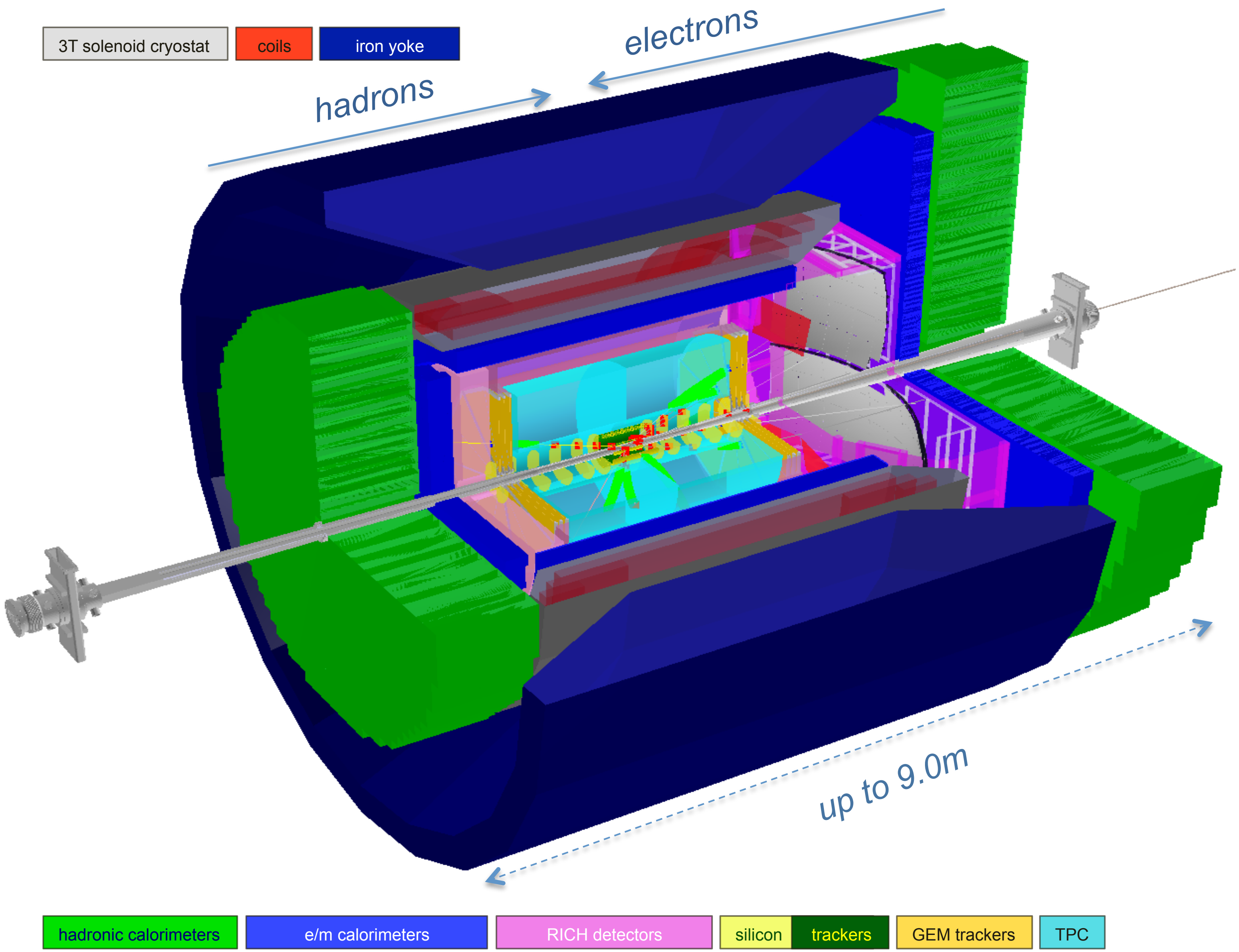
- Lately: eRHIC IR design

eRHIC model detector layout (2015)

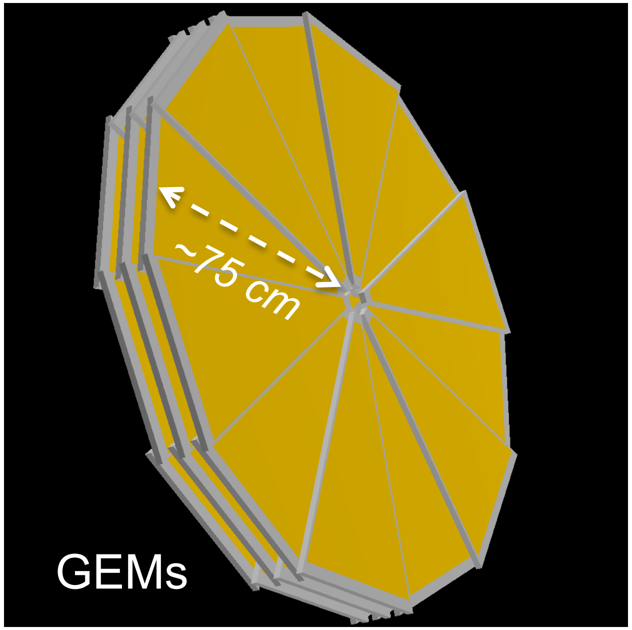
-4<η<4: Tracking & e/m Calorimetry (hermetic coverage)

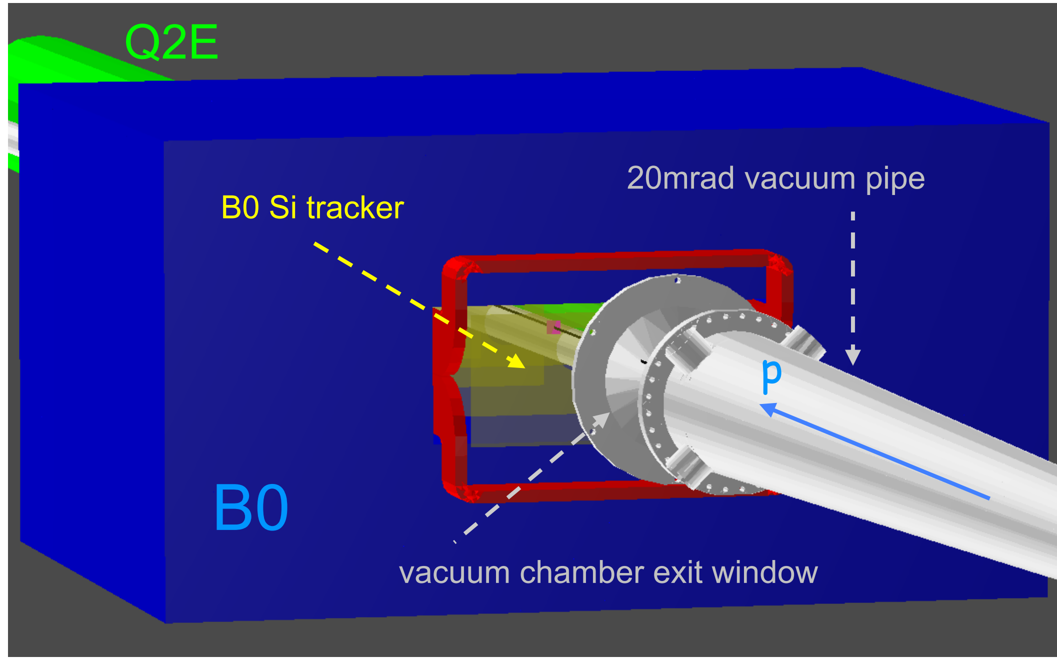


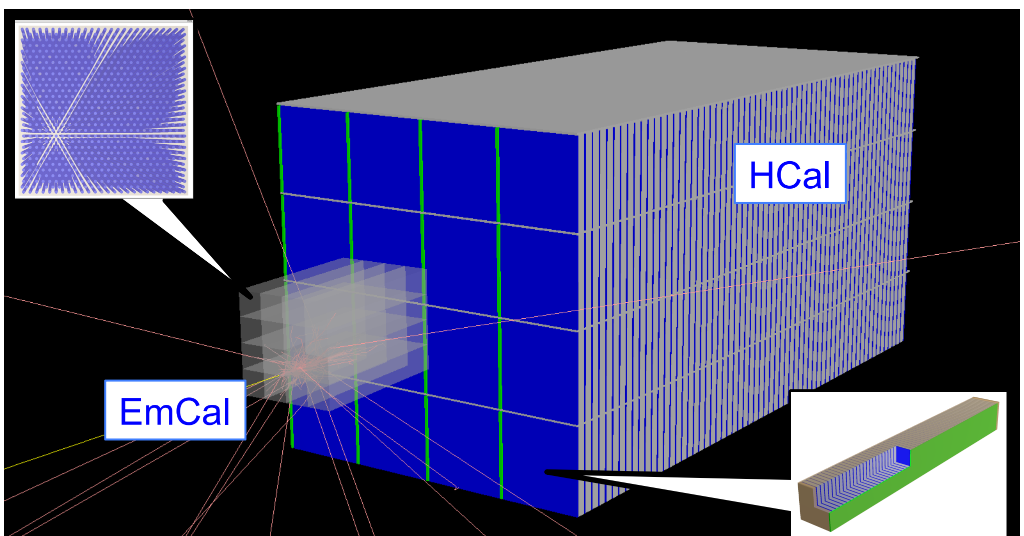
eRHIC model detector layout (2018)

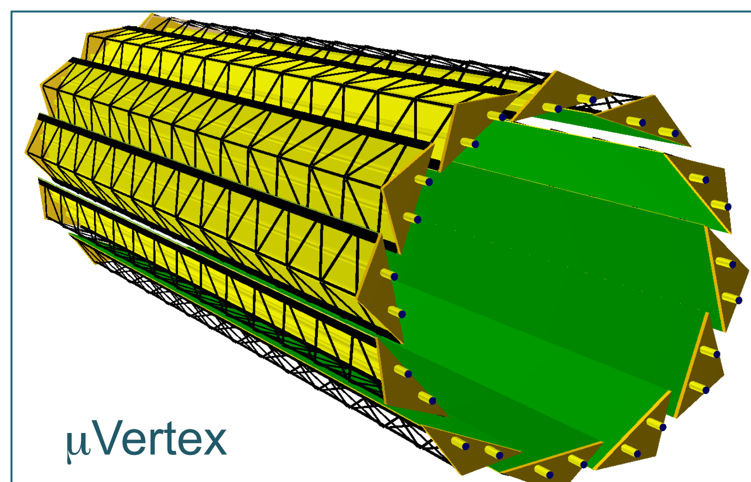


Examples of geometry building blocks

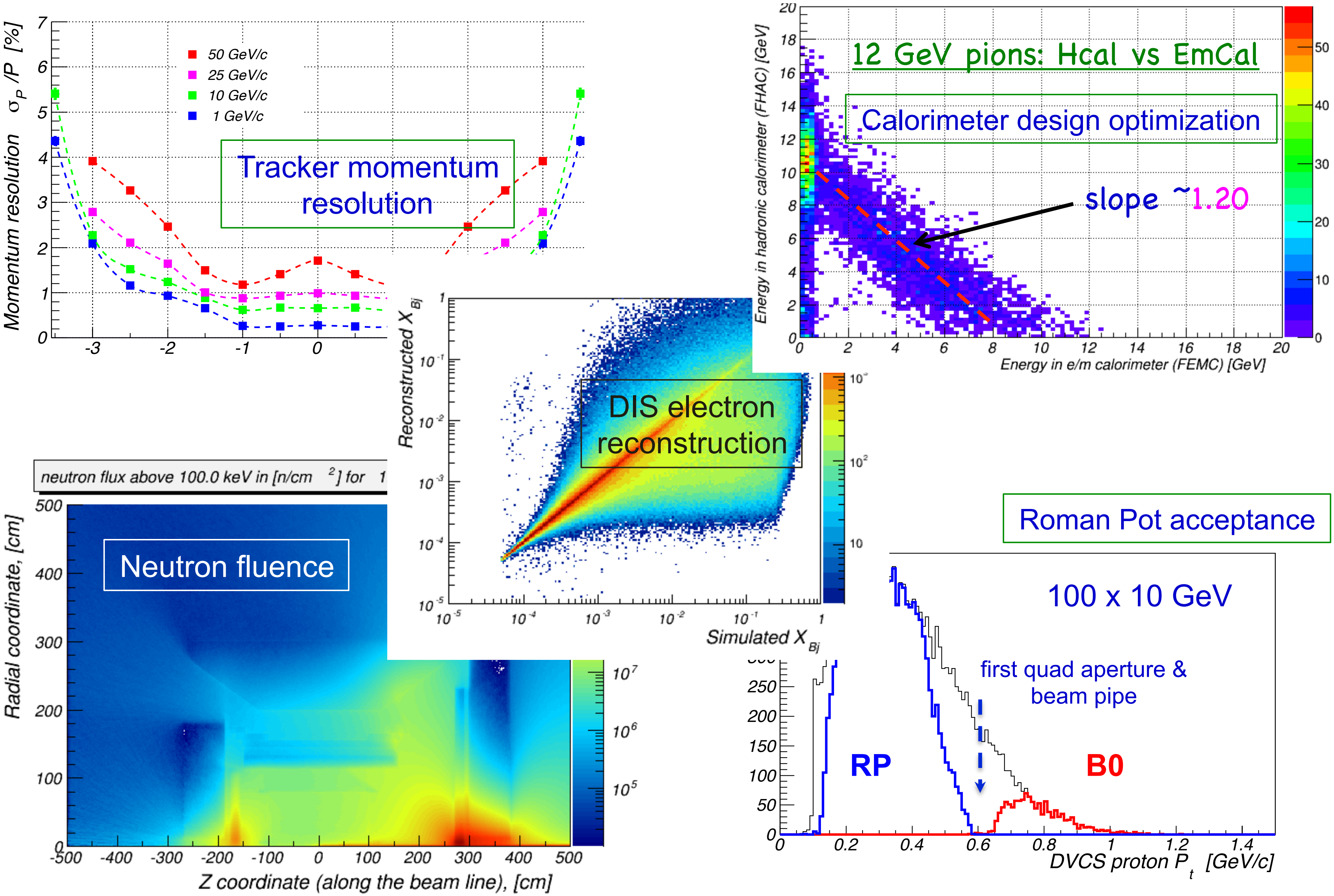








Example case studies



EIC smearing generator interface

MC generator ASCII output → **Tree code:** Build ROOT tree containing events → **Smearer:** Perform fast detector smearing

PYTHIA | DPMJet
PEPSI | Milou | gmc_trans
Rapgap | LEPTO | Djangoh

Large number of EIC Monte Carlo generators with standard ASCII format

- Both event import and smearing functionality is supported in EicRoot

Calorimeter code implementation

- Written from scratch (use ideas rather than codes)
  - ATLAS fast simulation (“frozen” showers)
  - CMS topological cluster search
- Unified interface (geometry definition, digitization, clustering) for all EIC calorimeter types
- Rather detailed digitization:
  - configurable light yield
  - exponential decay time; light collection in a time window
  - attenuation length; possible light reflection on one “cell” end
  - SiPM dark counting rate; APD gain, ENF, ENC
  - configurable thresholds

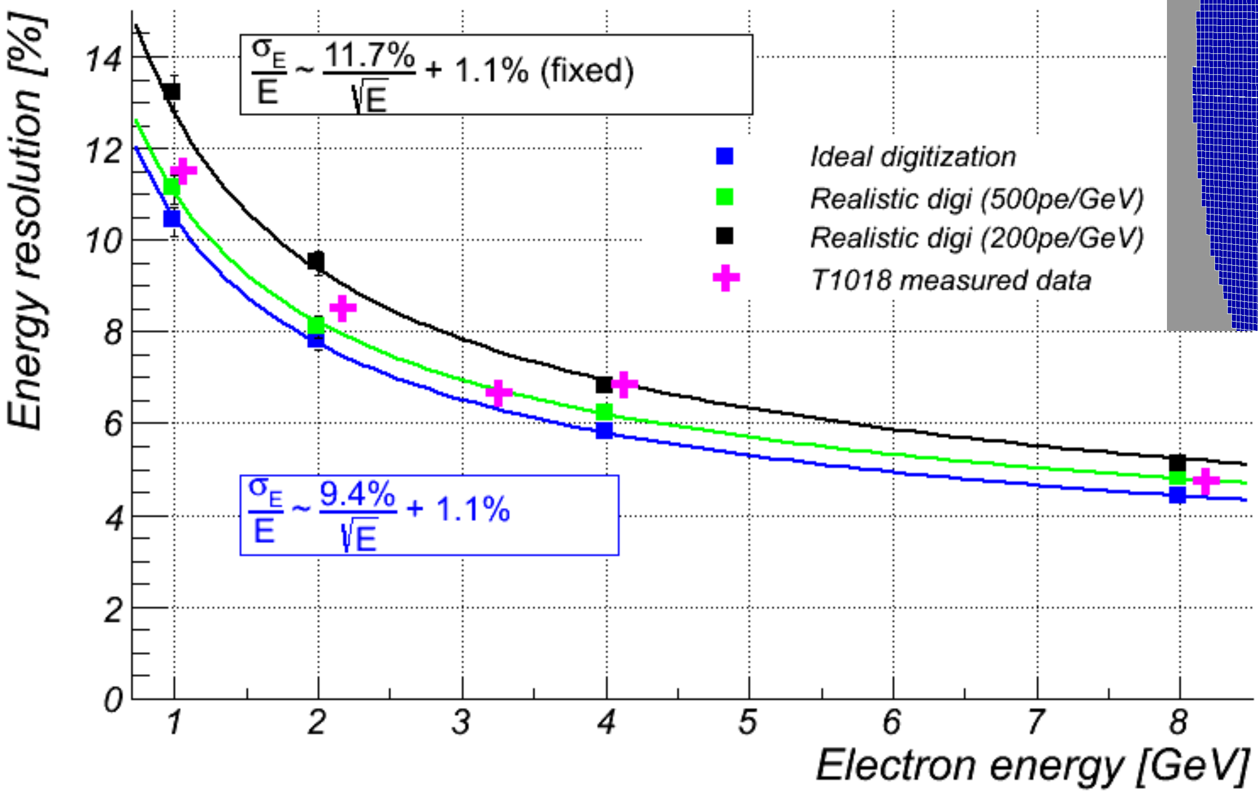
Calorimetry “designer” tools

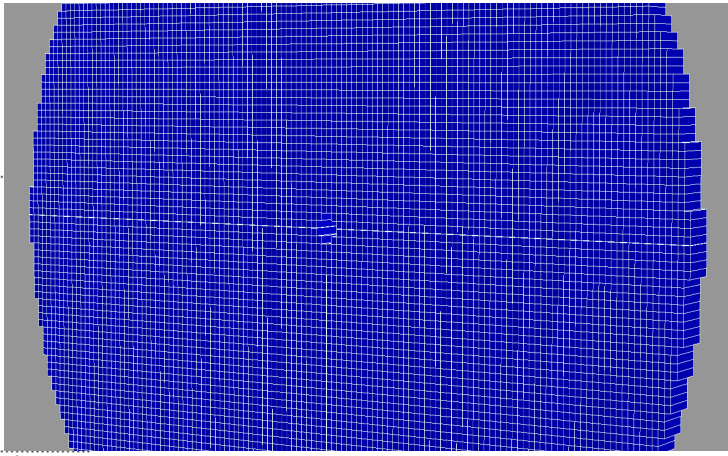
- As long as the following is true:
  - your dream calorimeter is a logical 2D matrix …
  - … composed of “long cells” as elementary units,
  - all the game is based on (known) light output per energy deposit,
  - energy resolution after “ideal” digitization suffices as a result
- … one can with a moderate effort (99% of which is writing a ROOT C macro with geometry and mapping description) build custom EicRoot-friendly calorimeter which can be used for both standalone resolution studies and/or as an optional EIC device (and internal cell structure does not matter)

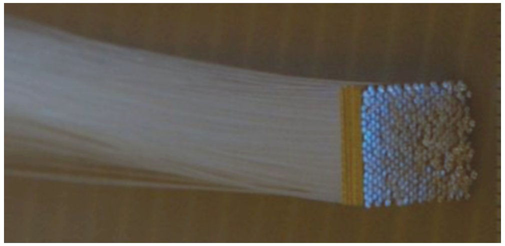
-> see examples/calorimetry directory for details

STAR EmCal upgrade simulations

3 degree track-to-tower-axis incident angle





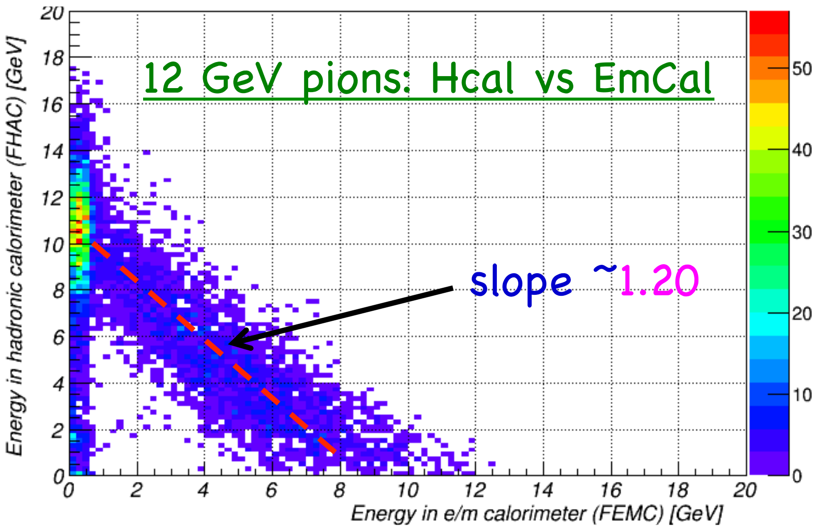


- "Exact" geometry description

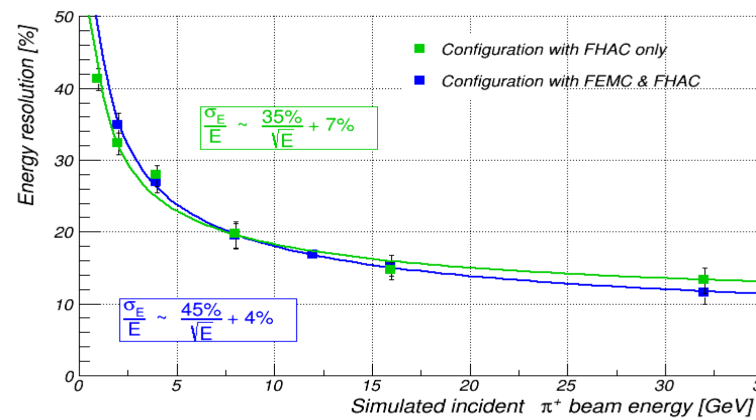
- "Realistic" digitization stands for: 40MHz SiPM noise in 50ns gate; 4m attenuation length; 5 pixel single tower threshold; 70% light reflection on upstream fiber end;

-> good agreement with original MC studies and measured data

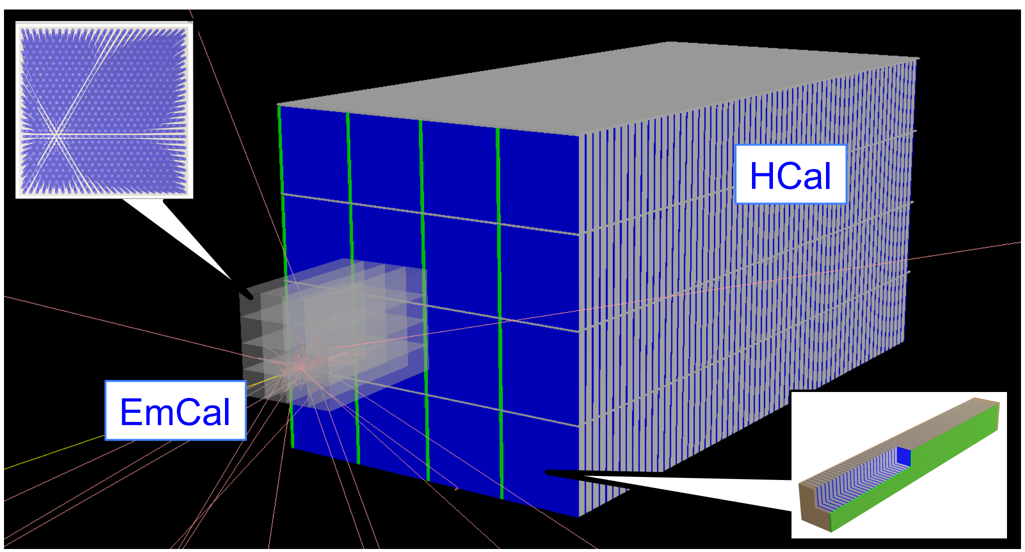
STAR HCal upgrade simulations

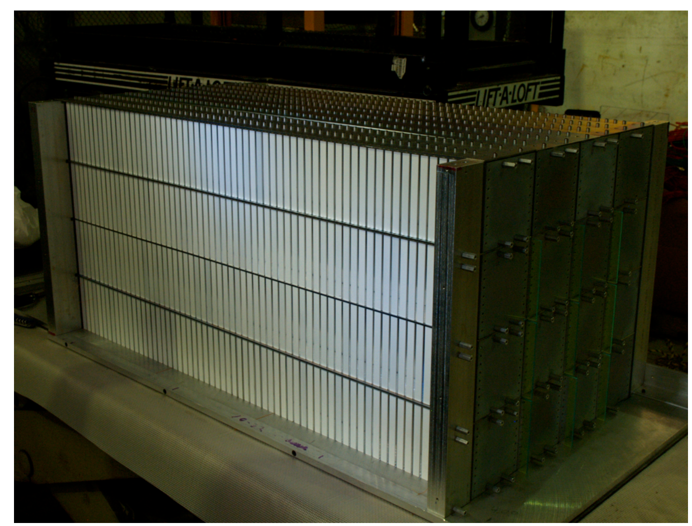


- GEANT 4, FTFP_BERT physics list
- Birk's correction accounted by hand



-energy resolution comparable to ZEUS 1987 paper

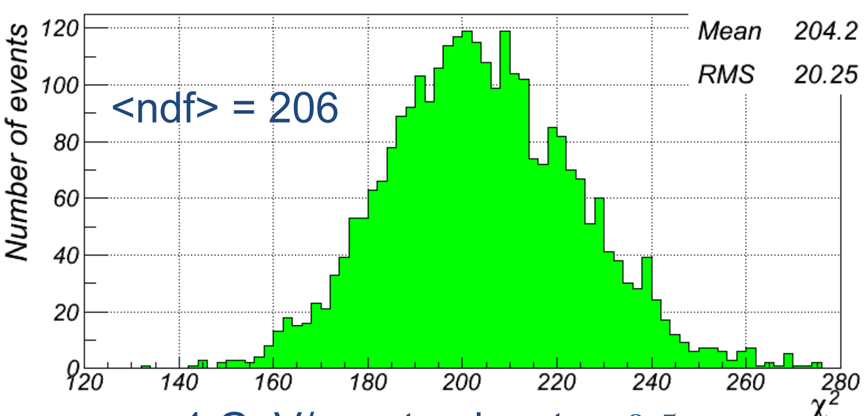




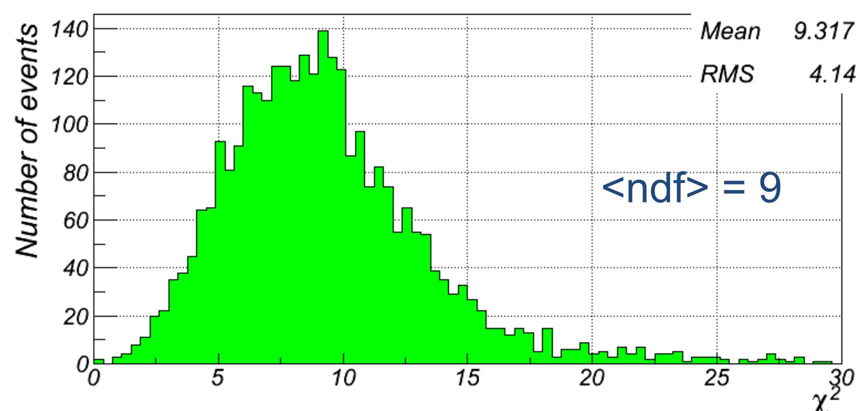
Tracking code implementation

- Large parts of other experiment's codes adapted:
  - PandaRoot: "ideal" track finder, GenFit fitter, etc
  - FopiRoot: TPC digitization, realistic track finders (Hough transform; Riemann sphere fit), GenFit fitter, RAVE vertex builder, etc
  - HERMES: linearized Kalman filter for forward spectrometers

Kalman filter fit quality for two "extreme" track configurations



1 GeV/c π^+ tracks at $\eta=0.5$:



32 GeV/c π^+ tracks at $\eta=3.0$:

Feature list & restrictions

- Modularity and flexibility in geometry description
- Several detector templates available in digitization:
  - Flat 1D (strip) and 2D (pixel) sensors; gaussian and sqrt(12) smearing
  - Flat 2D $\{r,\phi\}$ sensors (endcap-like)
  - Volume 3D (TPC-like)
  - Volume 1+2D (axial symmetric along the track)
- Kalman filter fitting through all hits at once (via GenFit)

- No coding required rather than detector geometry description and steering macros (all in ROOT C++)

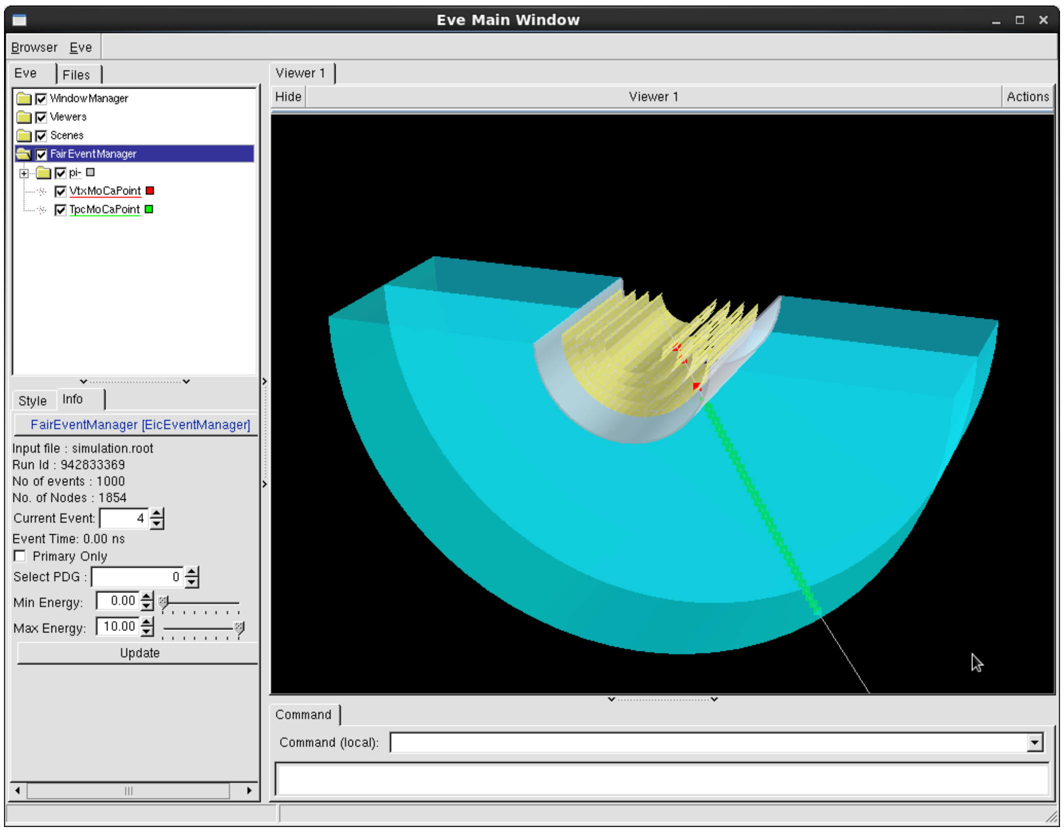
Feature list & restrictions

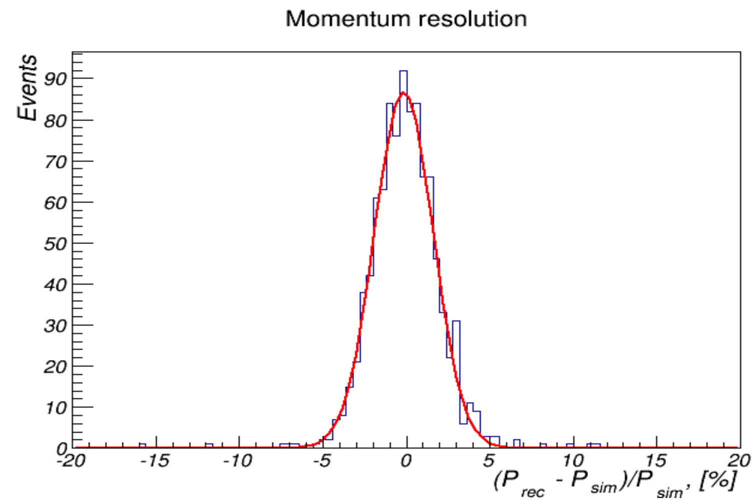
- Only ROOT TGeo geometry is supported
- Mapping information should be bundled in the same ROOT files with TGeo geometry description and it is directly available for digitization and reconstruction codes (so for each MC hit the information about physical volume it belongs to - shape, 3D transformation, etc - can be polled on-the-fly at any time)

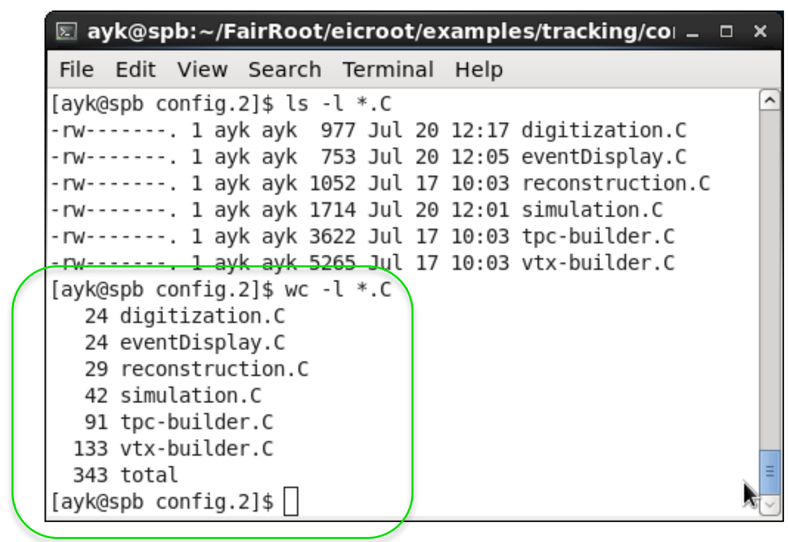
- “Ideal” (known a-priori) hit-to-track association is typically assumed (suffices for resolution studies)

Example R&D study: vertex+barrel tracker

Consider vertex tracker + TPC in 3T field; shoot 10 GeV/c pions at θ=75°





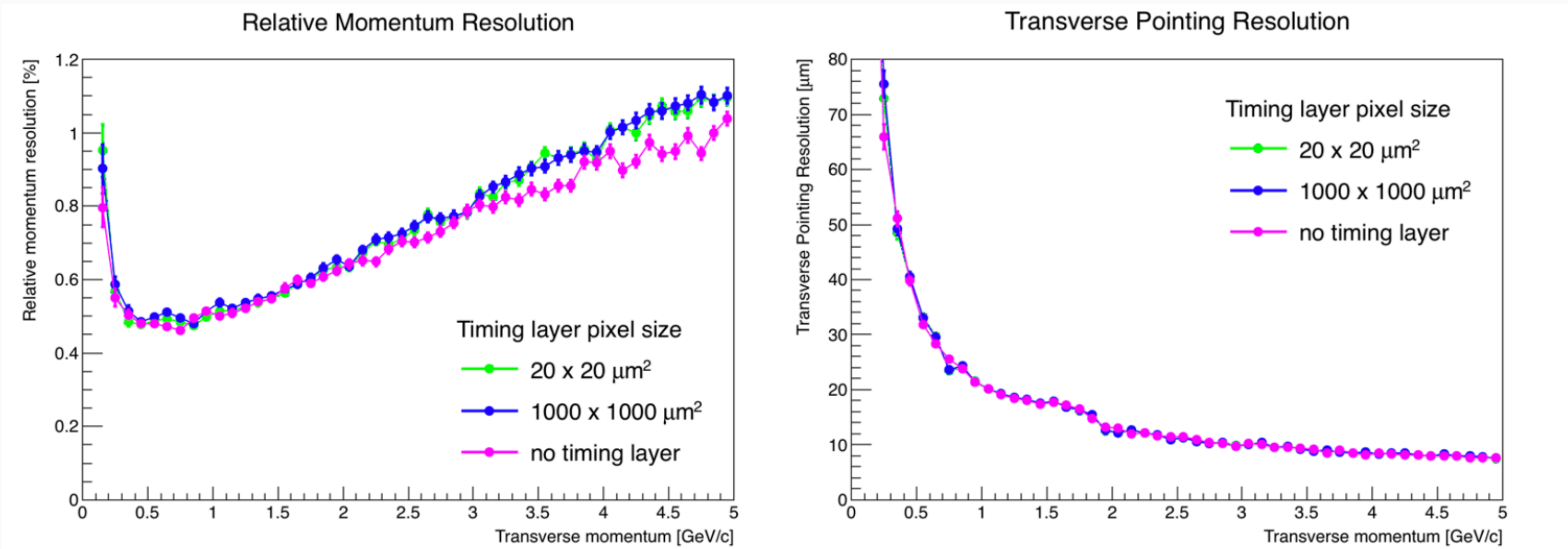


-> see examples/tracking/config.2 directory for details

- Once Docker image is downloaded it takes <5 minutes to generate this plot

EicRoot via Docker container

WP2: Barrel timing layer, pixel size study

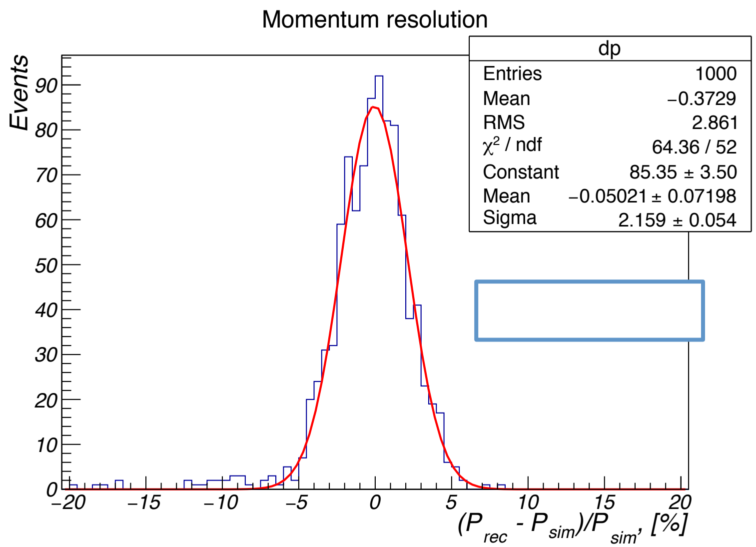


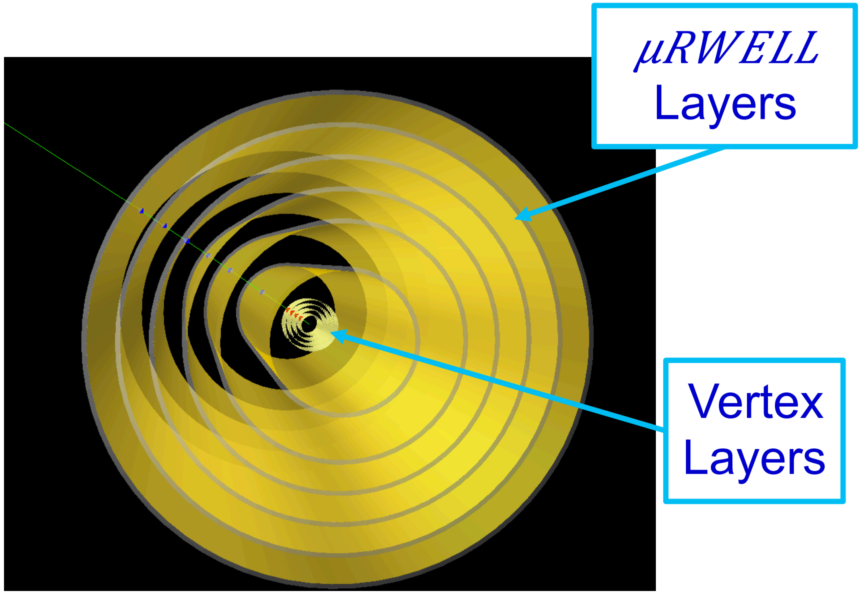
- Pions, $0 \leq p_T \leq 5$ GeV/c, $|\eta| \leq 0.5$
- 4-layer BeAST tracker, plus timing layer
- Timing layer pixel size does <u>not</u> affect detector performance

by Hakan Wennloef

EicRoot via Docker container

- Initial simulation Investigates central tracking system consisting of silicon vertex detector and cylindrical μRWELL operating in μTPC mode within EicRoot.
  - μTPC mode will allow reconstruction of Z track and could reduce material budget from more traditional central tracking solutions.
- Silicon vertex detector
  - **Four** silicon layers each with **X-Y** pixel resolution of **20 μm – 20 μm**
- Cylindrical μRWELL Barrel Tracker

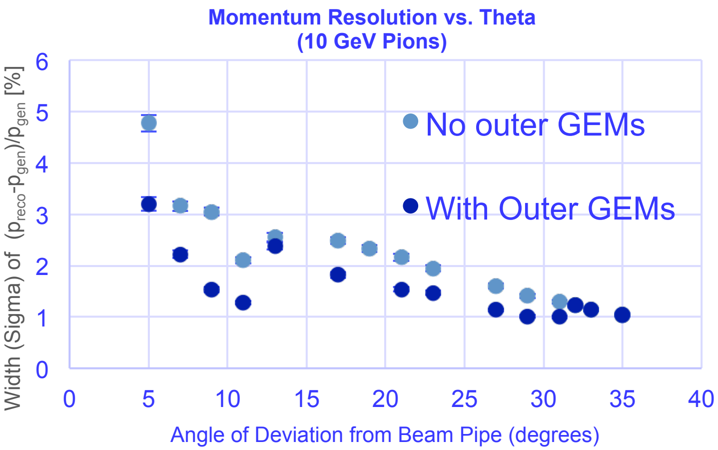


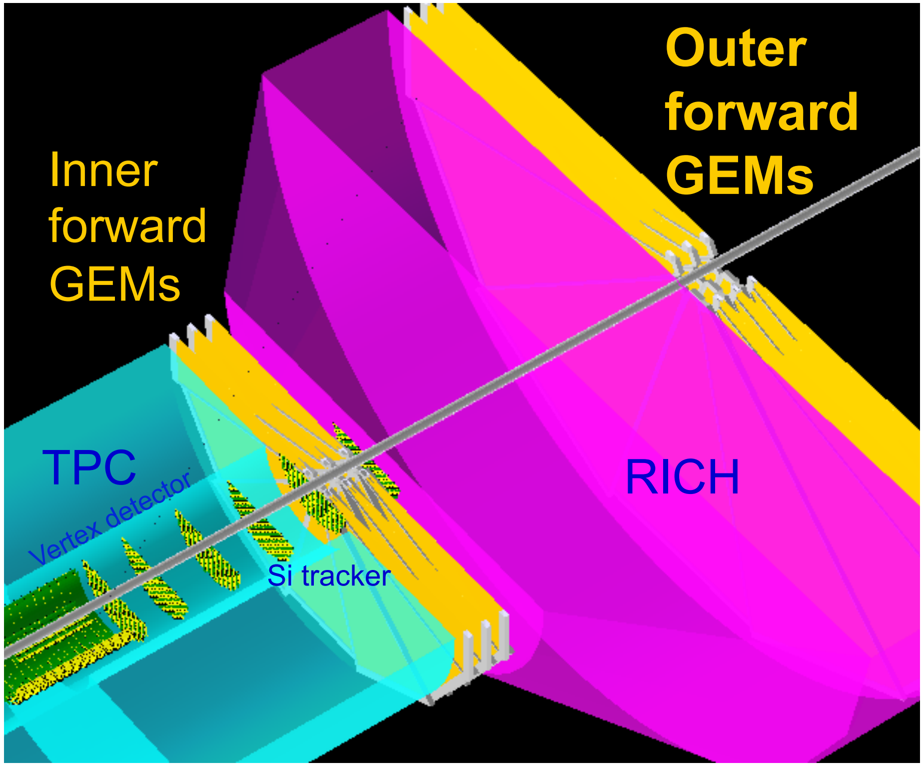


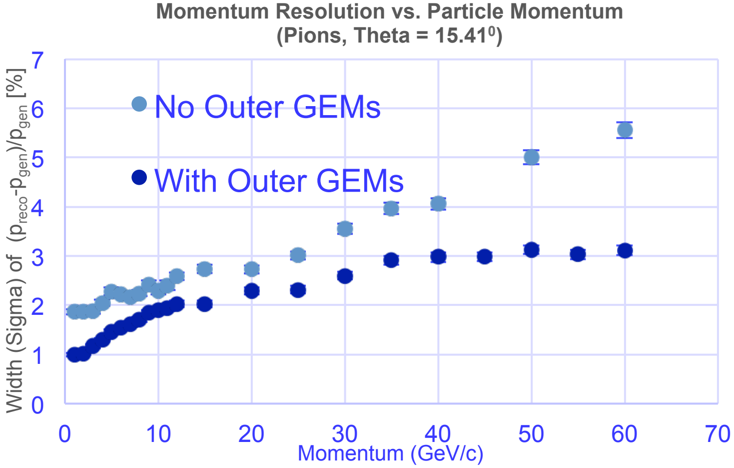
by Matt Posik

EicRoot via Docker container

- Impact of the outer forward GEM detectors on seeding the RICH ring reconstruction in BeAST geometry







by Matt Bomberger

Take away message

- EicRoot can be used “as is” for standalone R&D studies
- Several ready-to-go examples exist

- Codes available for download from BNL ~~SVN~~ Git server

❑ ... as well as in Docker container image(s)

❑ ... and can seemingly be incorporated in a more generic packaging scheme if needed