March 12th, 2024

IMCC and MuCol MDI Workshop

Software for BIB propagation & mitigation

Funded by the European Union

Views and opinions expressed are however those of the author only and do not necessarily reflect those of the EU or European Research Executive Agency (REA). Neither the EU nor the REA can be held responsible for them.

N. Bartosik [a]

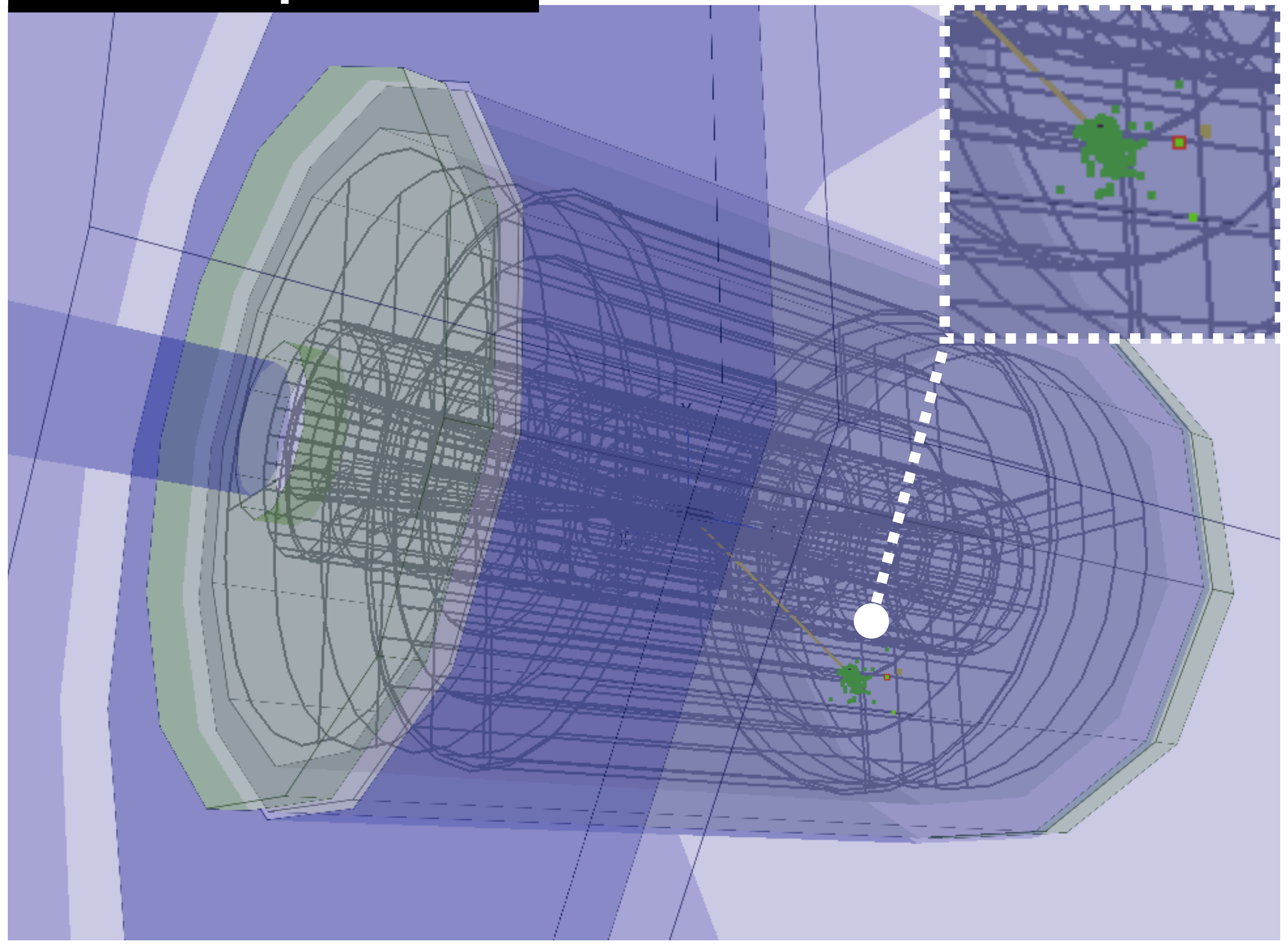
for the Muon Collider Physics and Detector Group

[a] INFN Torino *(Italy)*

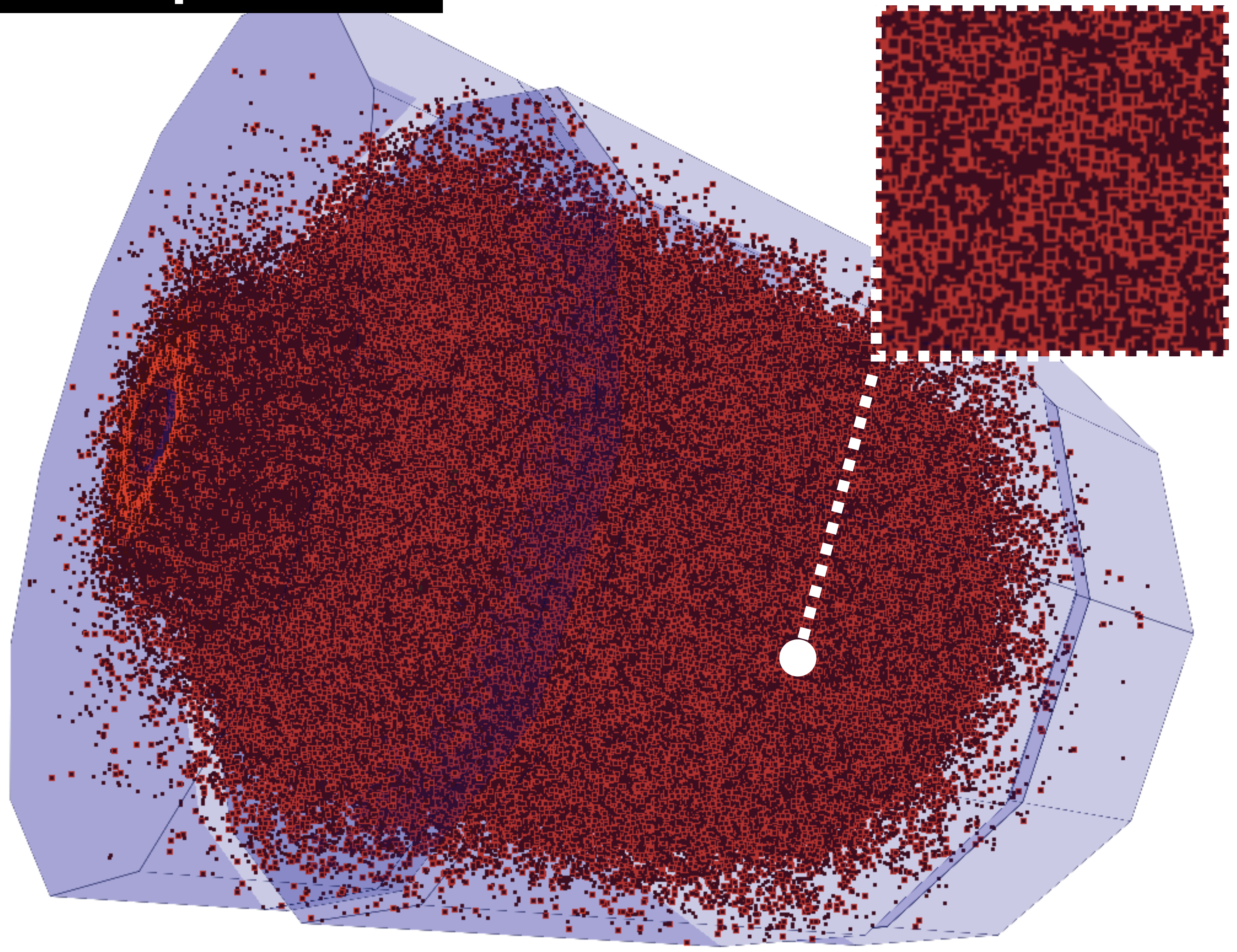
We want our simulation studies to be representative of what it will look like in the actual experiment

↳ all the relevant BIB effects have to be included during detector design for the target performance

10 GeV photon

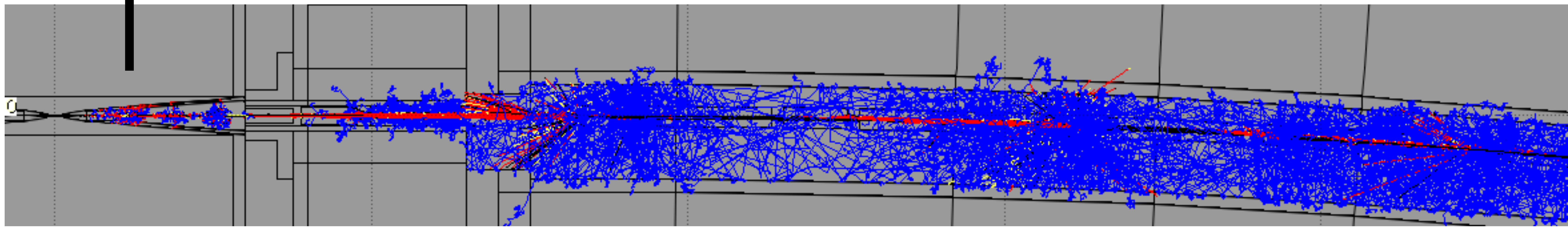


10 GeV photon + **BIB** in ECAL



BIB sample generated
by the MDI team
propagated to the detector

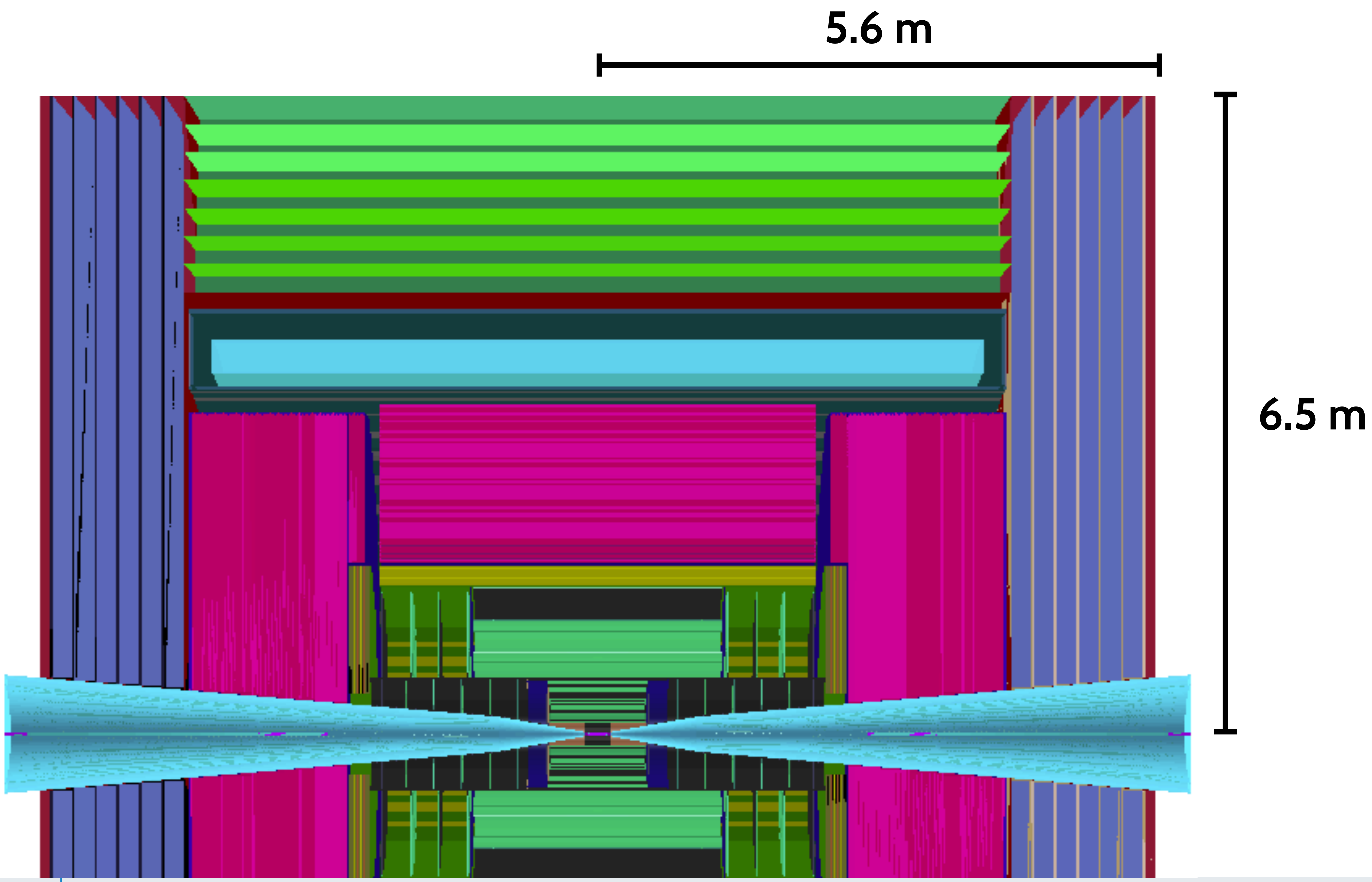
Simulated in FLUKA ▶



Muon Collider detector

Our detector spans 5.6 m along the beam and 6.5 m in radius

↳ each sub-detector has its own scale in terms of dimensions, sensitivity to BIB



Propagation steps

Propagation through the detector includes a number of steps:

- interaction of BIB particles with the detector's passive/active material
  ↳ eventual production of secondary particles

- application of detector effects
  (segmentation, resolution, noise, threshold effects, etc.)

- execution of reconstruction algorithms
  (tracking, just clustering, flavor-tagging, particle-flow, etc.)

Propagation steps

Propagation through the detector includes a number of steps:

- interaction of BIB particles with the detector's passive/active material

  ↳ eventual production of secondary particles

  GEANT4 interfaced via **DD4hep** - commonly used framework to simplify geometry description

- application of detector effects
  (segmentation, resolution, noise, threshold effects, etc.)

  Collection of digitization processors in the Marlin framework + simplified filters, etc.

- execution of reconstruction algorithms
  (tracking, just clustering, flavor-tagging, particle-flow, etc.)

  Collection of libraries interfaced with the Marlin framework: ACTS, PandoraPFO, etc.

Software details

We have the baseline software stack: ILCSoft + an ongoing migration to Key4hep

ILCSoft is well tested and readily available: installed on CERN lxplus machines (CVMFS) + containers

- uses **LCIO** data format to store MCParticles, SimHits, RecHits and higher-level objects
- includes **LCTuple** processor for storing objects in flat ROOT TTrees

  ↳ easy to interprete by other people (e.g. MDI team)

Migration to Key4hep has several advantages:

- integrated support of parallelization → much better computing performance;
- shared tools with other future experiments → easy integration of external developments.

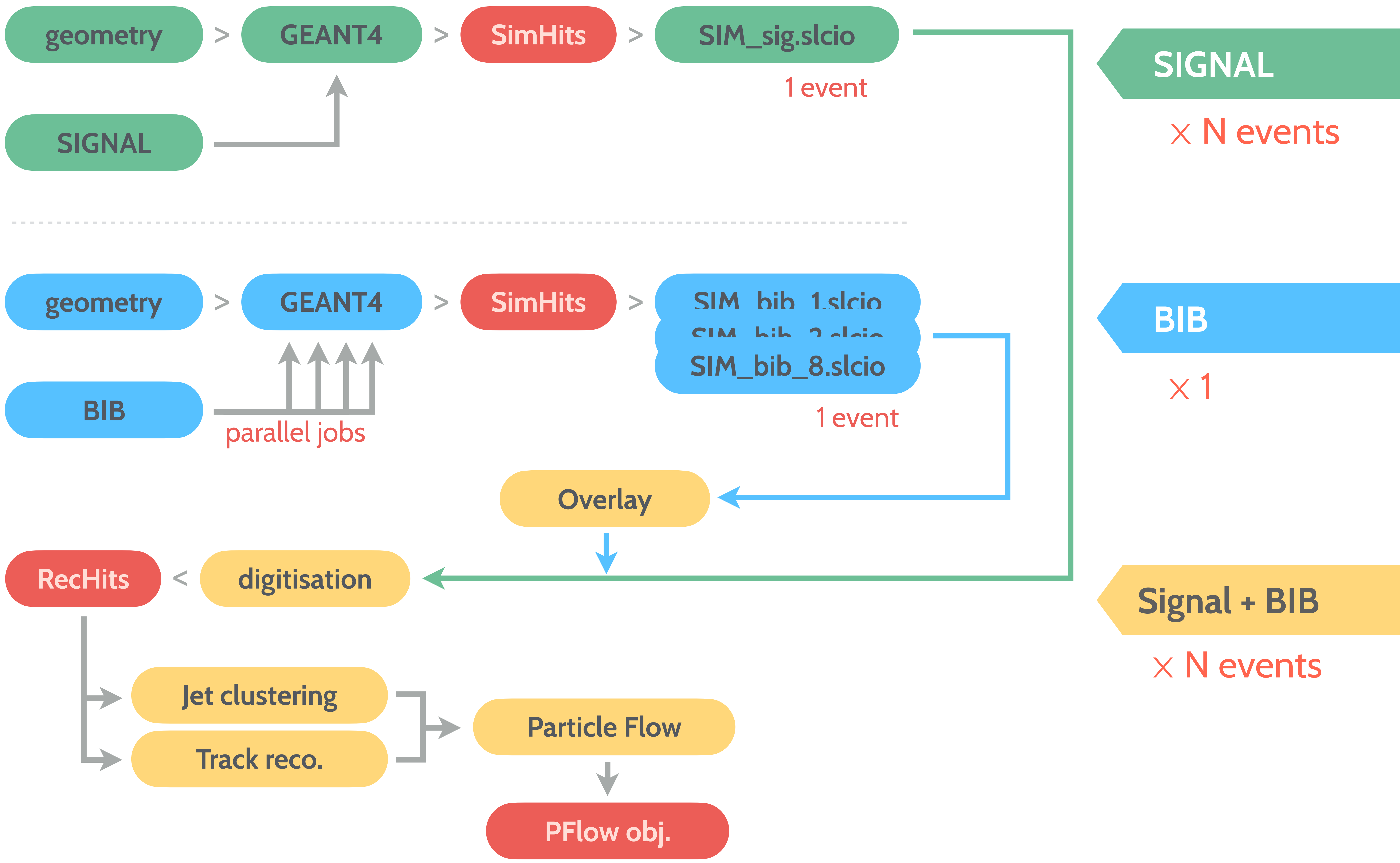
The biggest showstopper:

- different data format: **LCIO** → **edm4hep** - requires adapting a lot of the existing code

See the practical guide to our software in the latest tutorial session

Simulation workflow

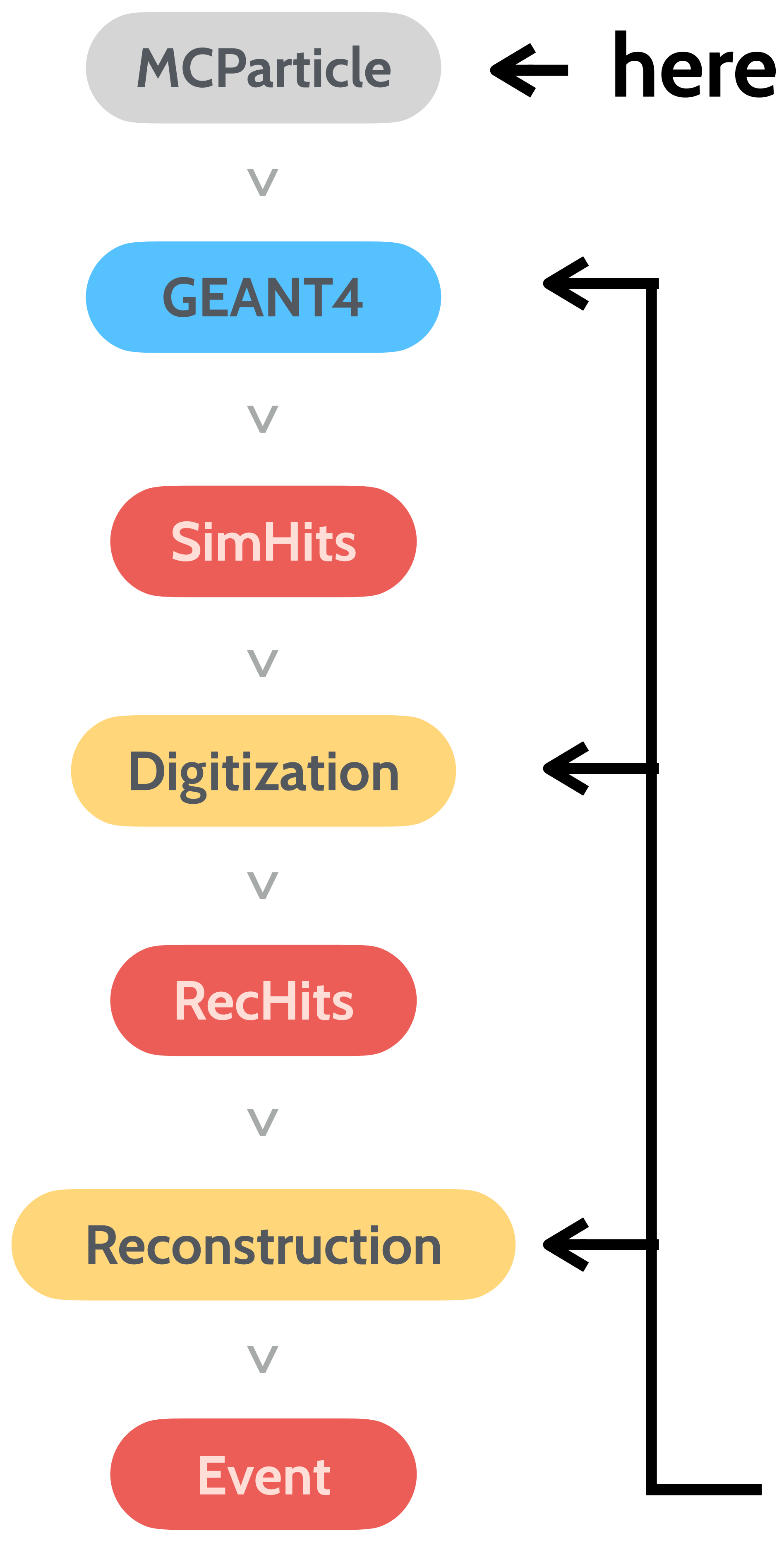
Fully simulating + reconstructing a single event at Muon Collider:

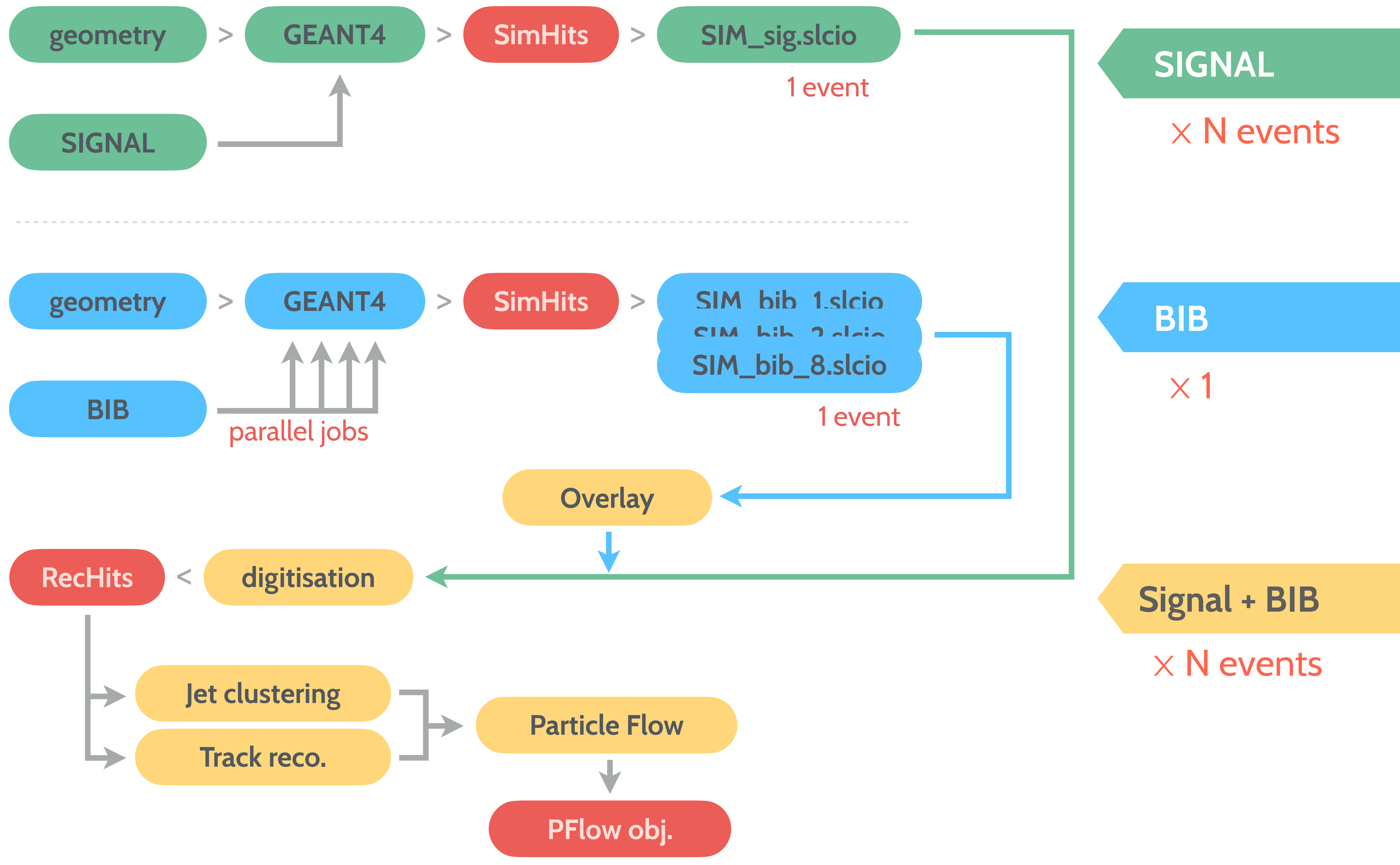


Simulation workflow

Fully simulating + reconstructing a single event at Muon Collider:

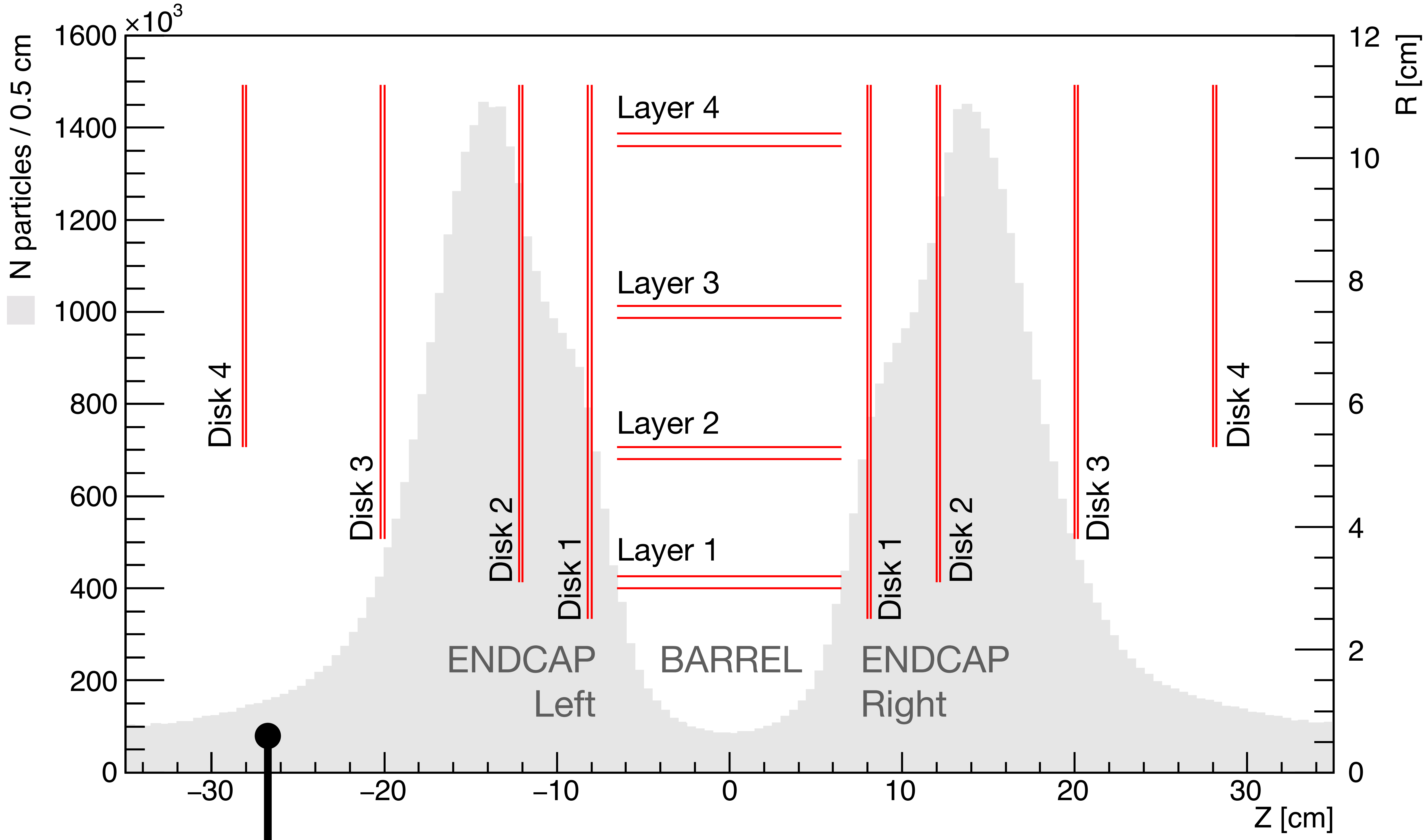
FLUKA stops here



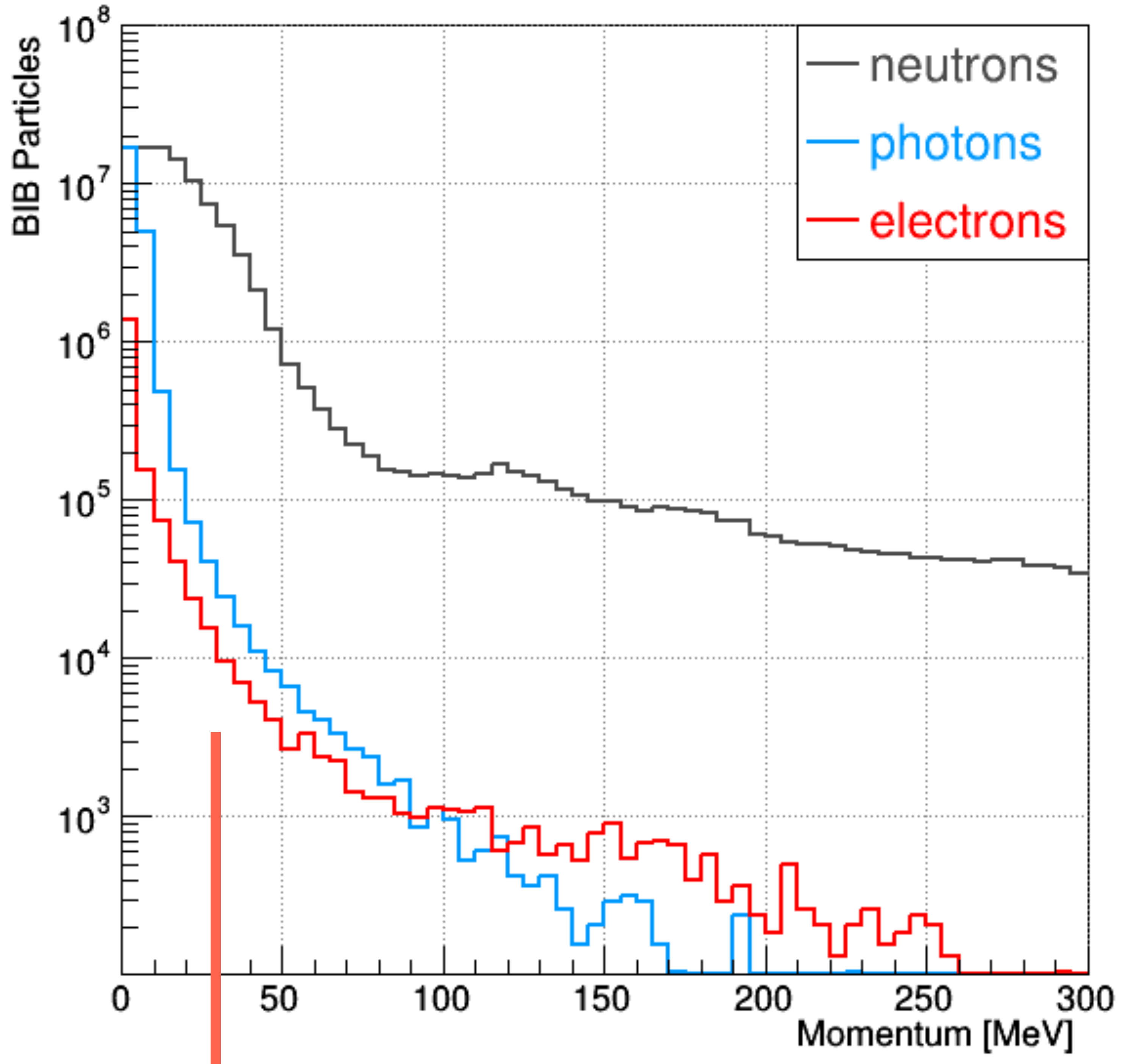


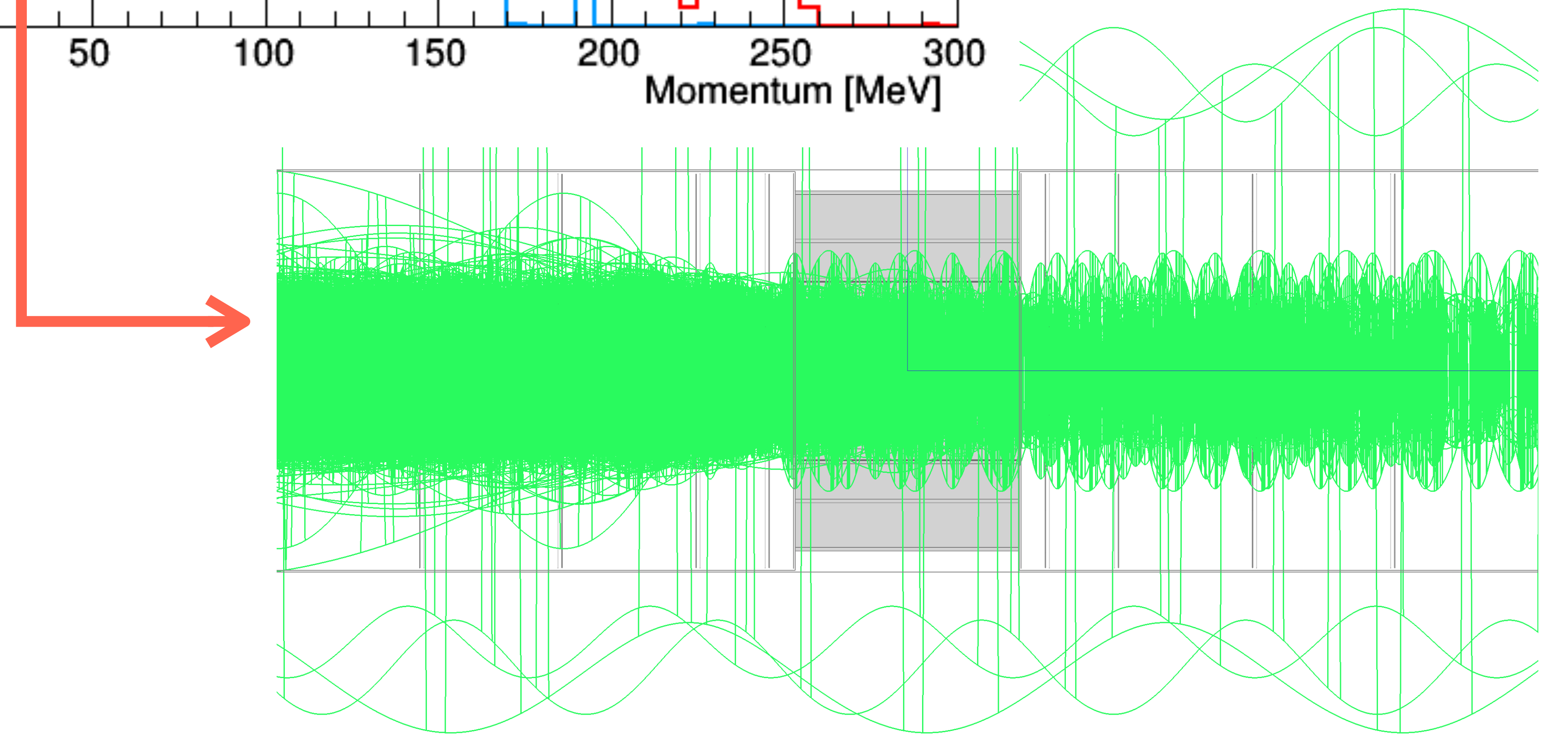
BIB-induced effects can make a significant impact at any stage

Classical example of non-obvious BIB impact: Vertex Detector



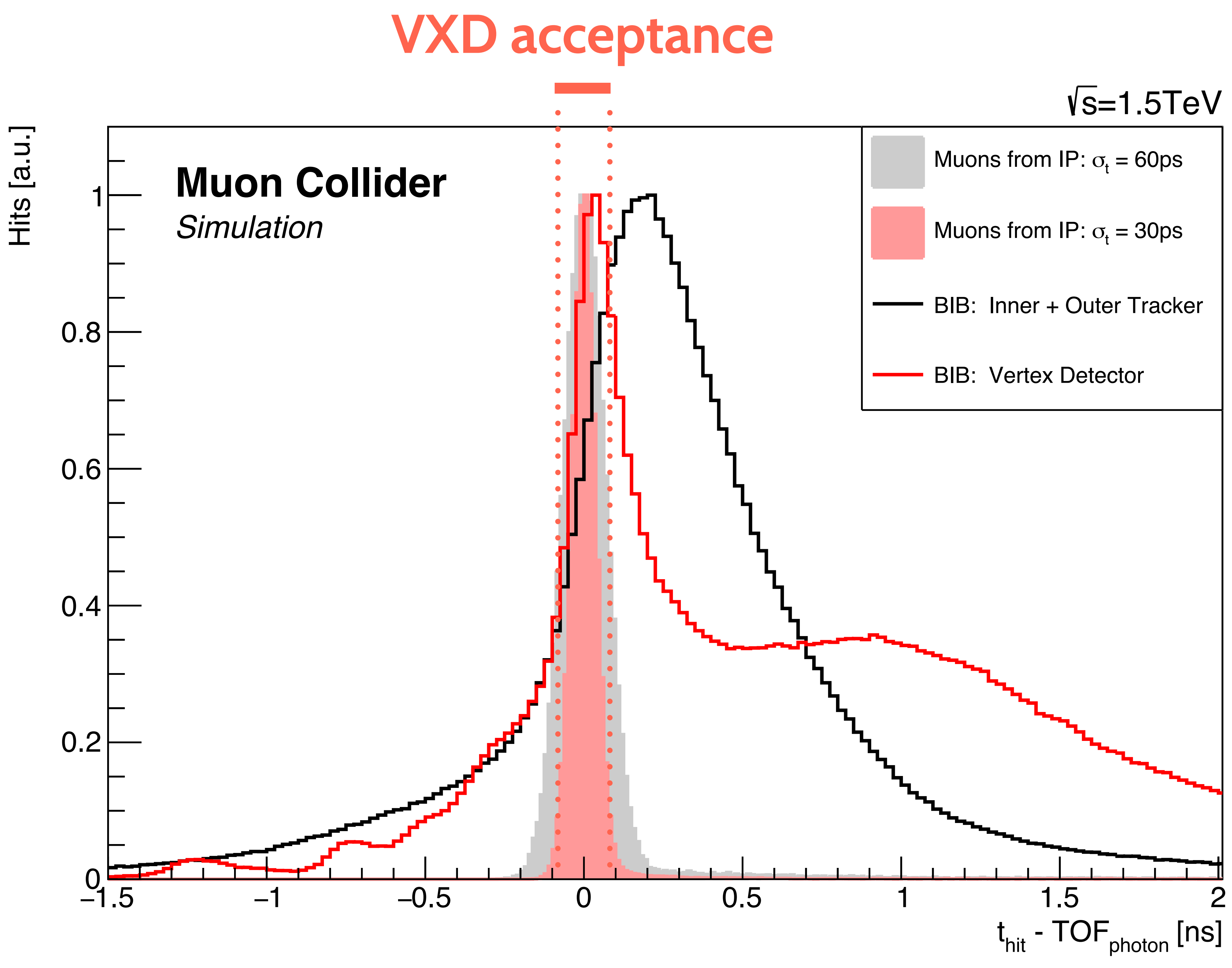
distribution of BIB particles' exit points along the beam pipe

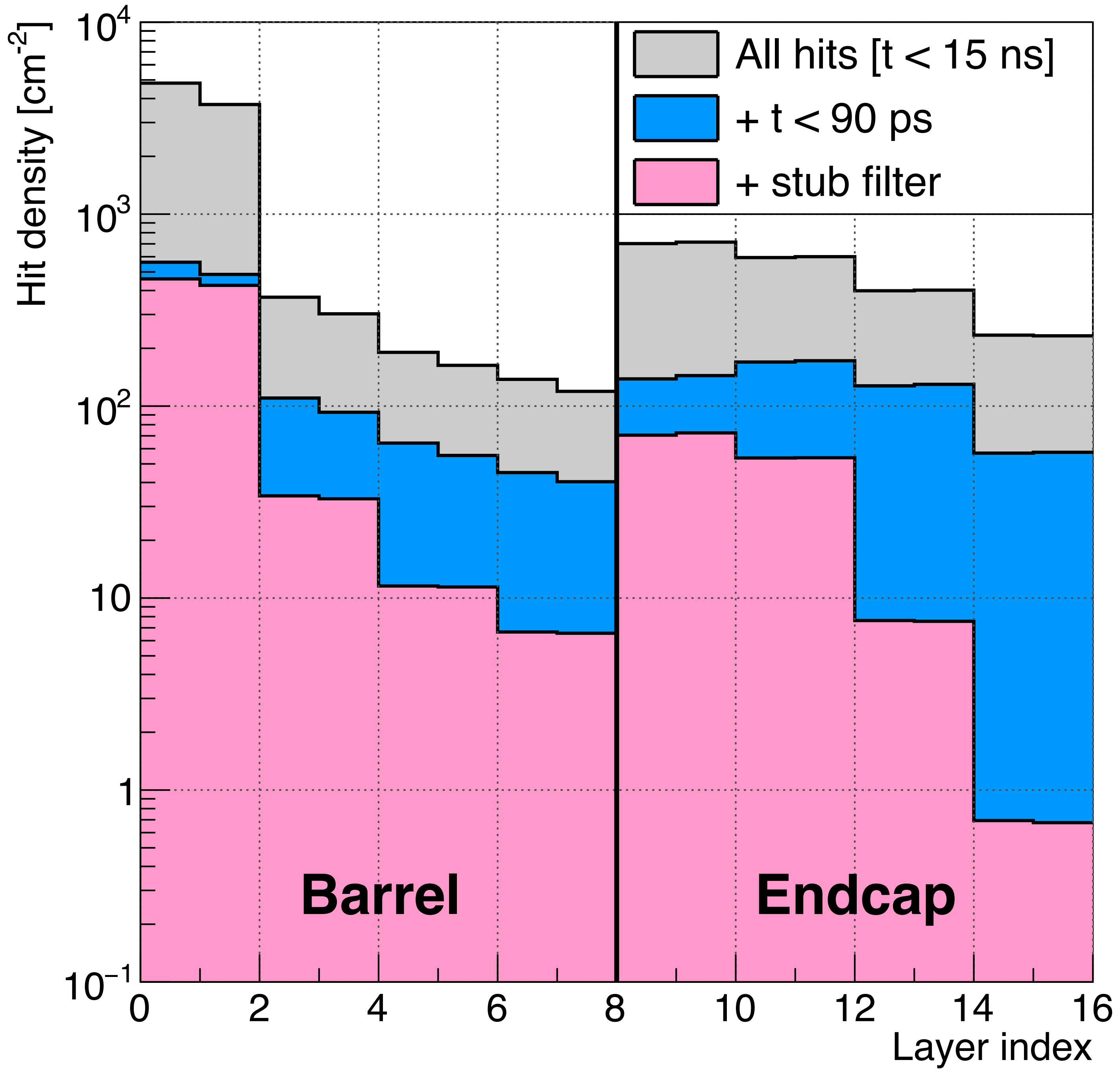




p_T of the BIB electrons defines the size of the loopers cloud

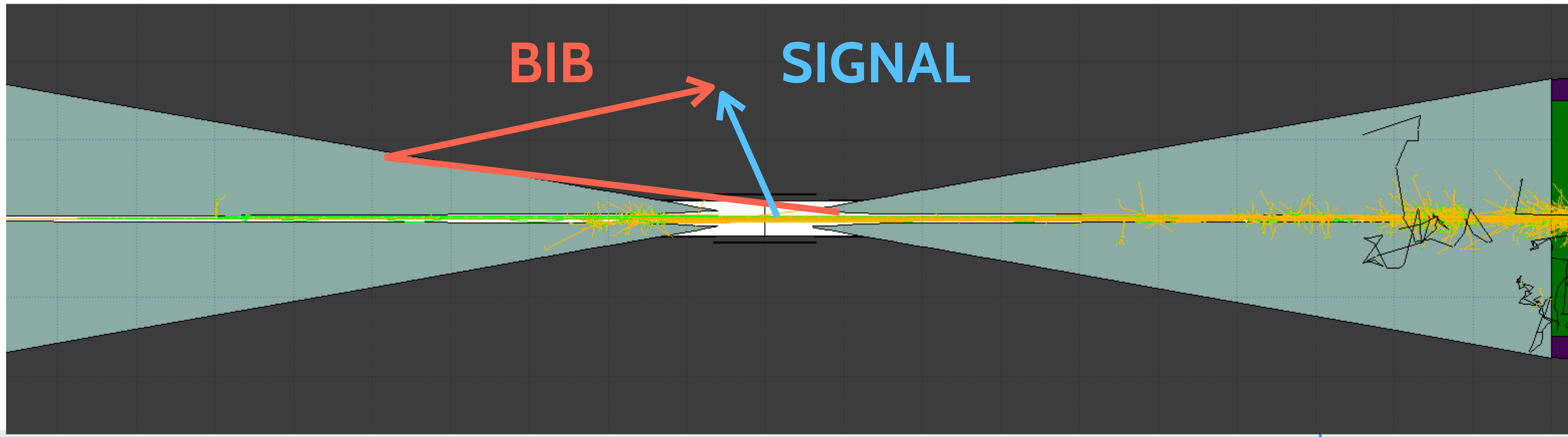
Timing is also a very sensitive parameter → allows to suppress hits in the 1st layer the most



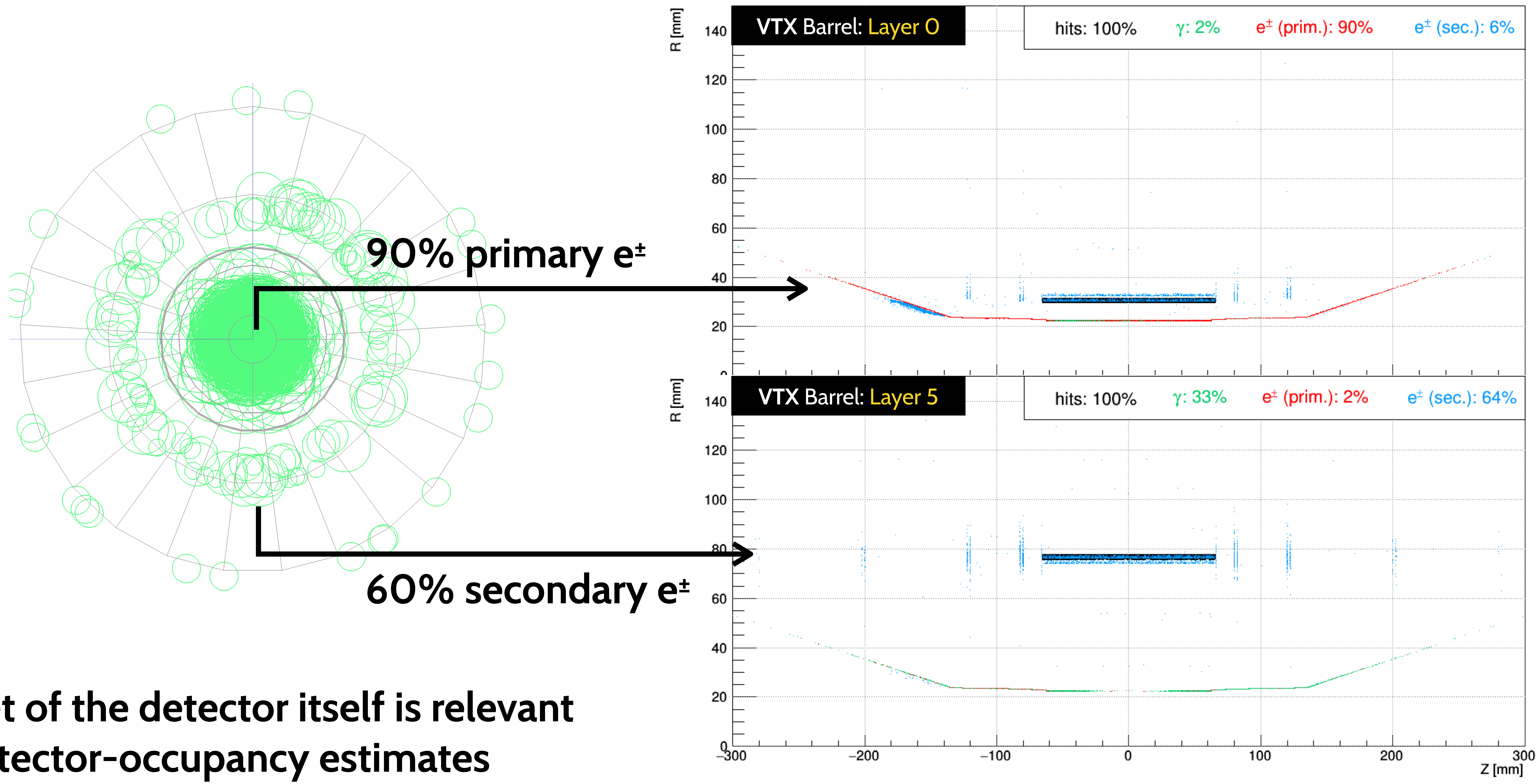


Increasing the flight distance for BIB electrons by 1 cm could shift the TOF by 30 ps (1 σ_t)

BIB distributions at MDI level are typically viewed at the scale of nanoseconds → ×100 too high

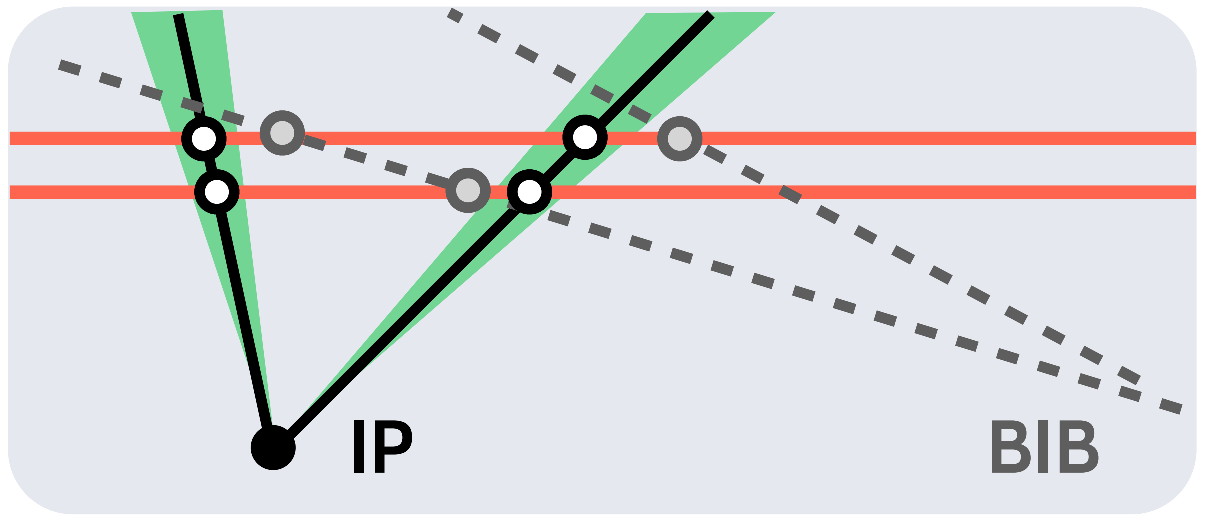


Primary BIB particles from FLUKA do not tell the whole story either

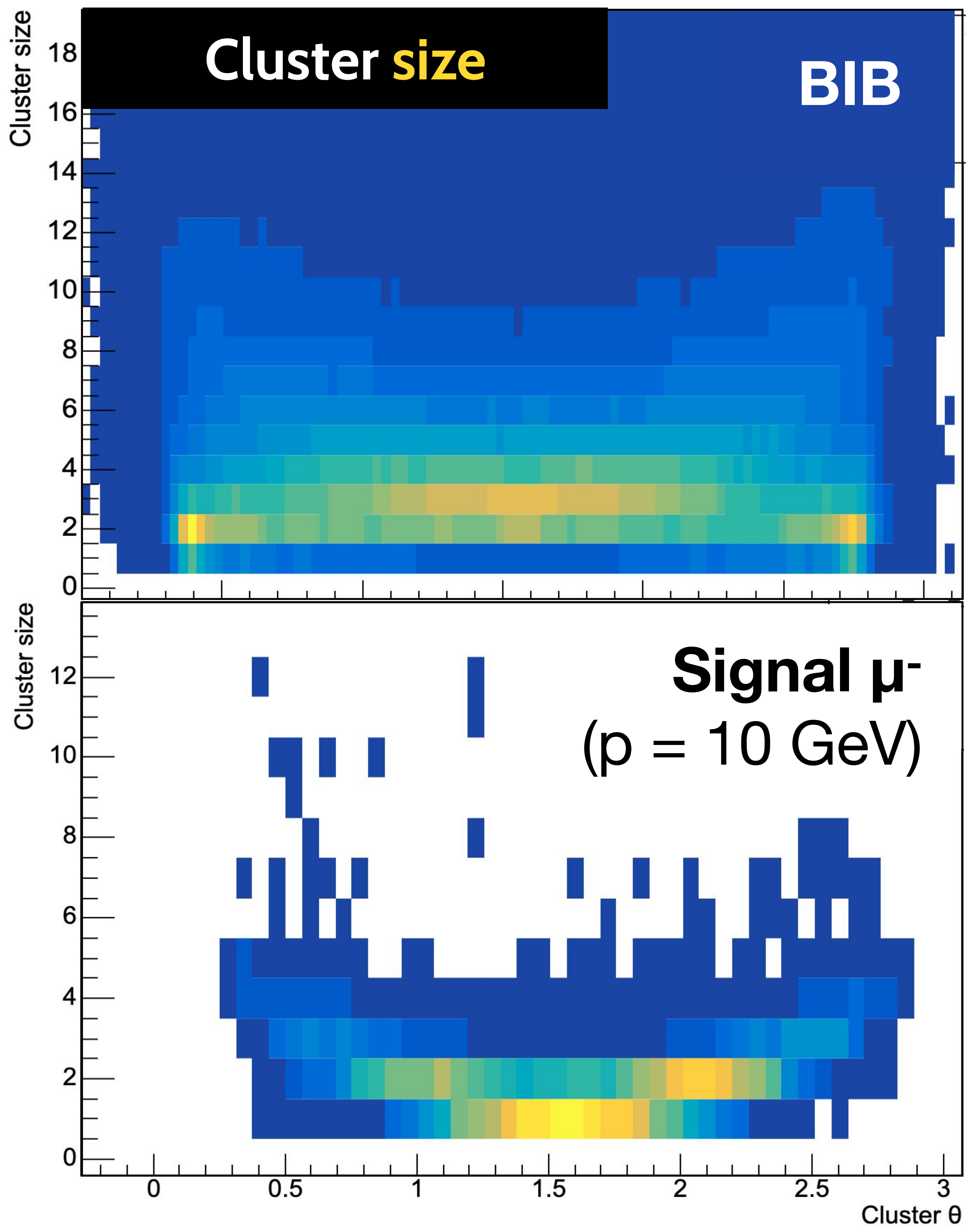


Material budget of the detector itself is relevant for the final detector-occupancy estimates

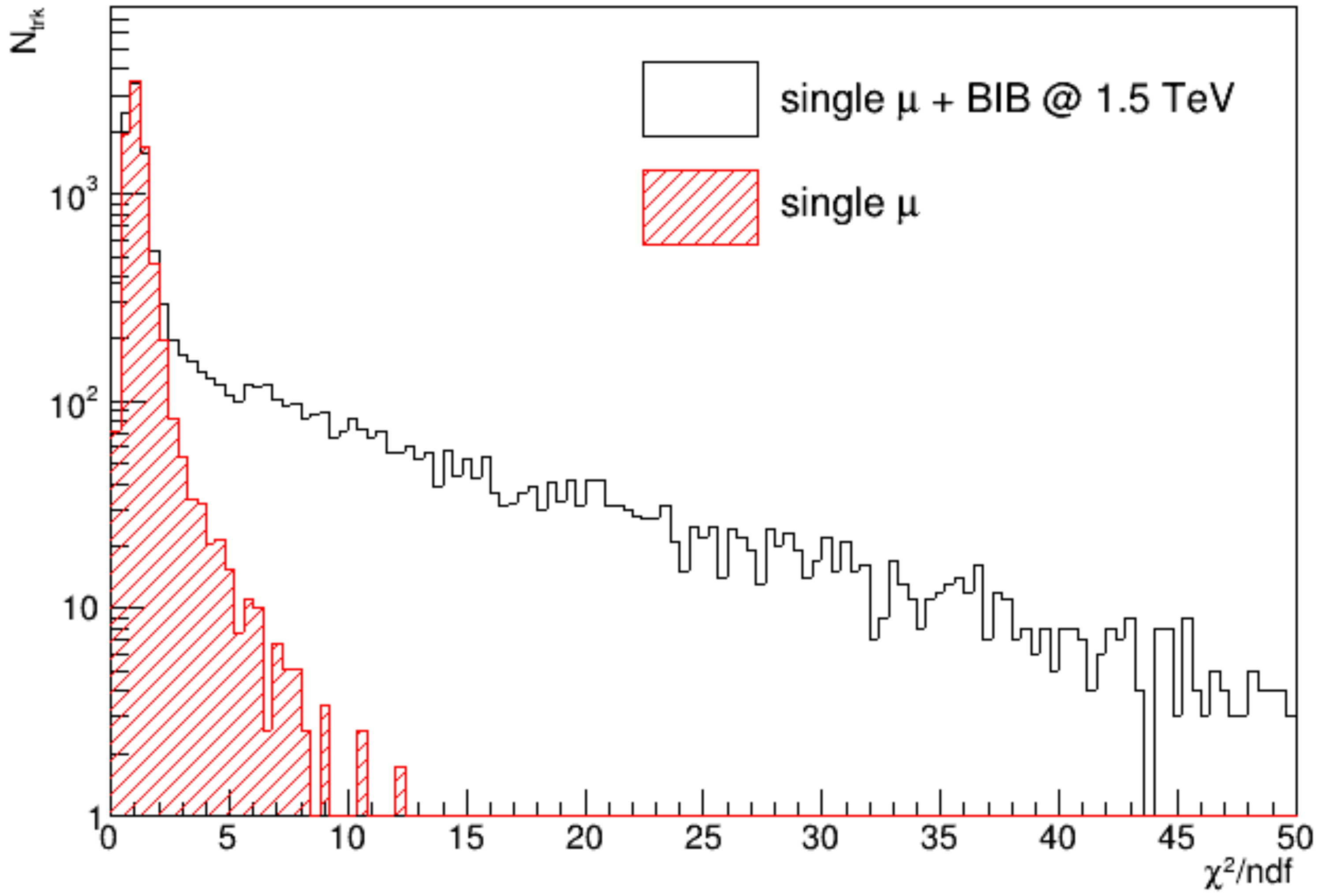
Higher-level BIB-mitigation strategies are less sensitive to the MDI design



Double layers
for BIB rejection
using stub filtering



Pixel clusters
from BIB particles
are typically larger



Track χ^2
to reject fake tracks

Tracks from BIB particles are not reconstructable

Inclusion in MDI workflow

The most detailed geometry is implemented in DD4hep → GEANT4 ▶

BIB samples generated for the detector simulation cannot include the detector itself to avoid double-counting of the materials

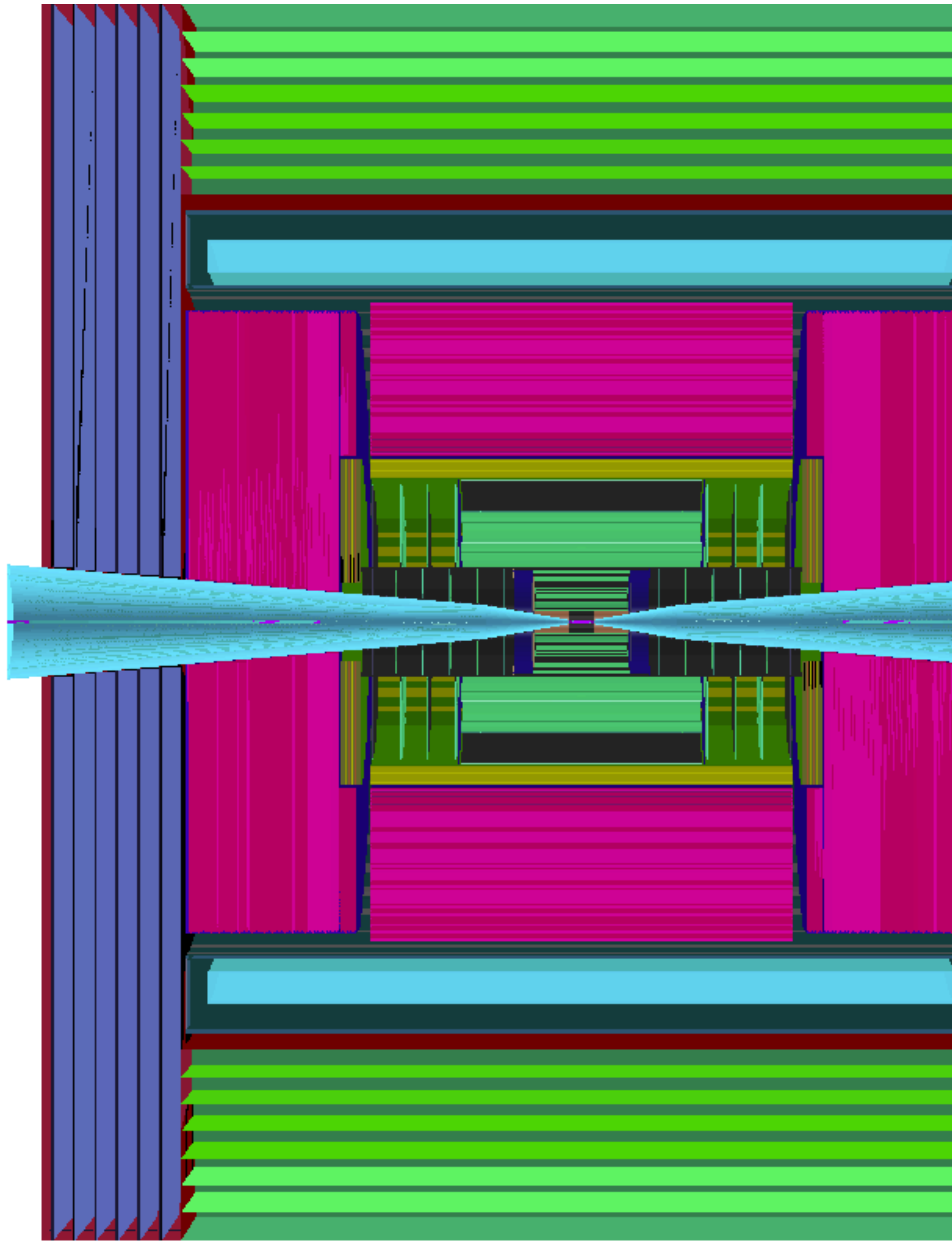
But for MDI optimisation + radiation maps it is useful to have a rough material distribution of the detector directly in FLUKA

↳ dimensions + materials of each subdetector provided in text format and added to the FLUKA geometry by hand

Several critical scoring surfaces are defined directly in FLUKA
to count BIB particles in more relevant places, not just on the MDI surface

↳ e.g. to provide a fast estimate of the effect on Vertex Detector

Yet any MDI candidate must be used in the detector simulation to have a definitive answer about the actual level of BIB suppression



Technical remarks

Transition from the old MAP samples to the FLUKA-generated ones has several important aspects:

- **data format**
  FLUKA output stored in a custom binary format *(very compact)*
  ↪ converted to LCIO::MCParticles → directly compatible with our detector-simulation software
- **unit conventions**
  MARS15 vs FLUKA vs GEANT4 vs DD4hep all have their own conventions
  ↪ several iterations were needed to ensure consistency between all data formats
- **storage and book-keeping**
  BIB samples relevant for detector simulation are now stored in the dedicated EOS space

By now we can run a full simulation cycle from initial muons in the lattice to a reconstructed event in just a few steps

The high priority tasks for now:

- **implementation of sampling for large statistics, preserving the relevant correlations**
- **streamlining the detector simulation step for a faster feedback loop**