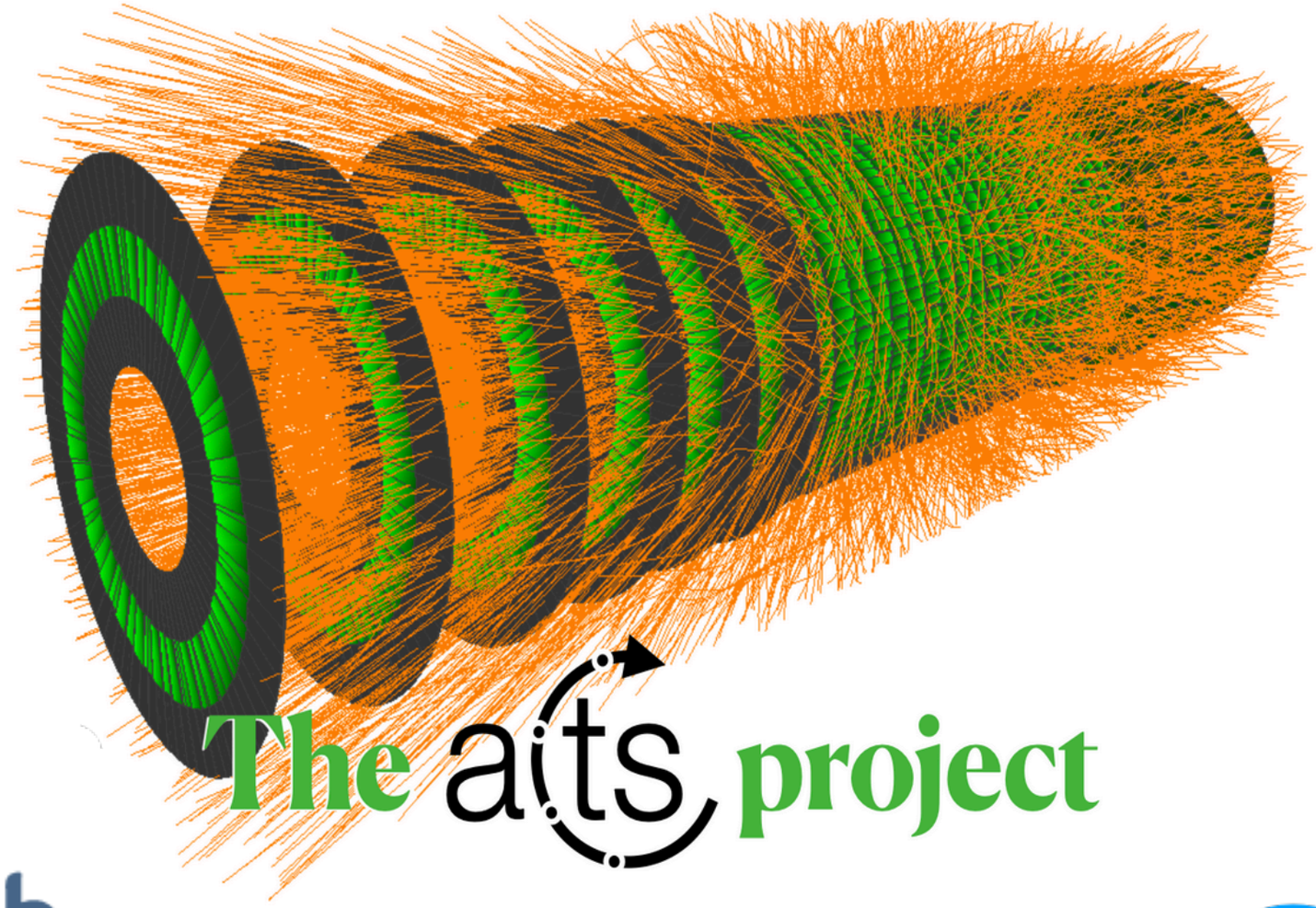
ACTS : A Common tracking software

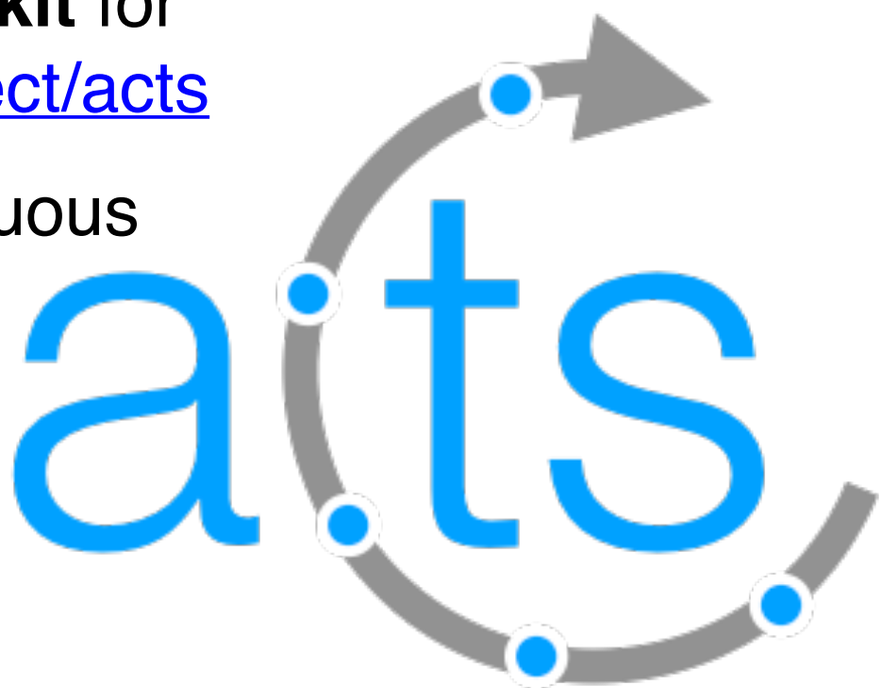


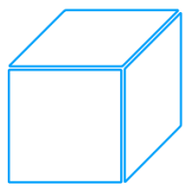
The acts project

Corentin Allaire
on behalf of the
ACTS team

Acts: A Common Tracking Software

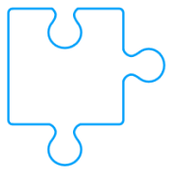
- Open source and **experiment-independent toolkit** for track reconstruction: https://github.com/acts-project/acts
- Developed with **modern C++**: unit testing, continuous integration…
- Minimal dependency for ease of building
- **Community** project: used by many **experiments** and **R&D** projects





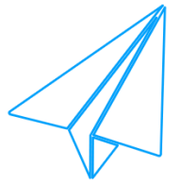
Geometry/Detector*
(Surface based geometry)

Plugins to DD4hep, TGeo, etc.

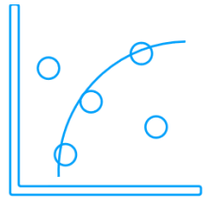


Event Data Model
target track reconstruction

backend separation
with different I/O models

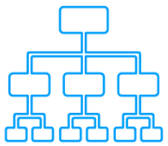


Propagation
parameter + covariance
transport through
magnetic field



Seeding
Seed finding with
Triplet seeder,
OrthogonalSeedFinder

Track Fitting
parameter estimation
with Kalman Filter,
GSF, GX2F**



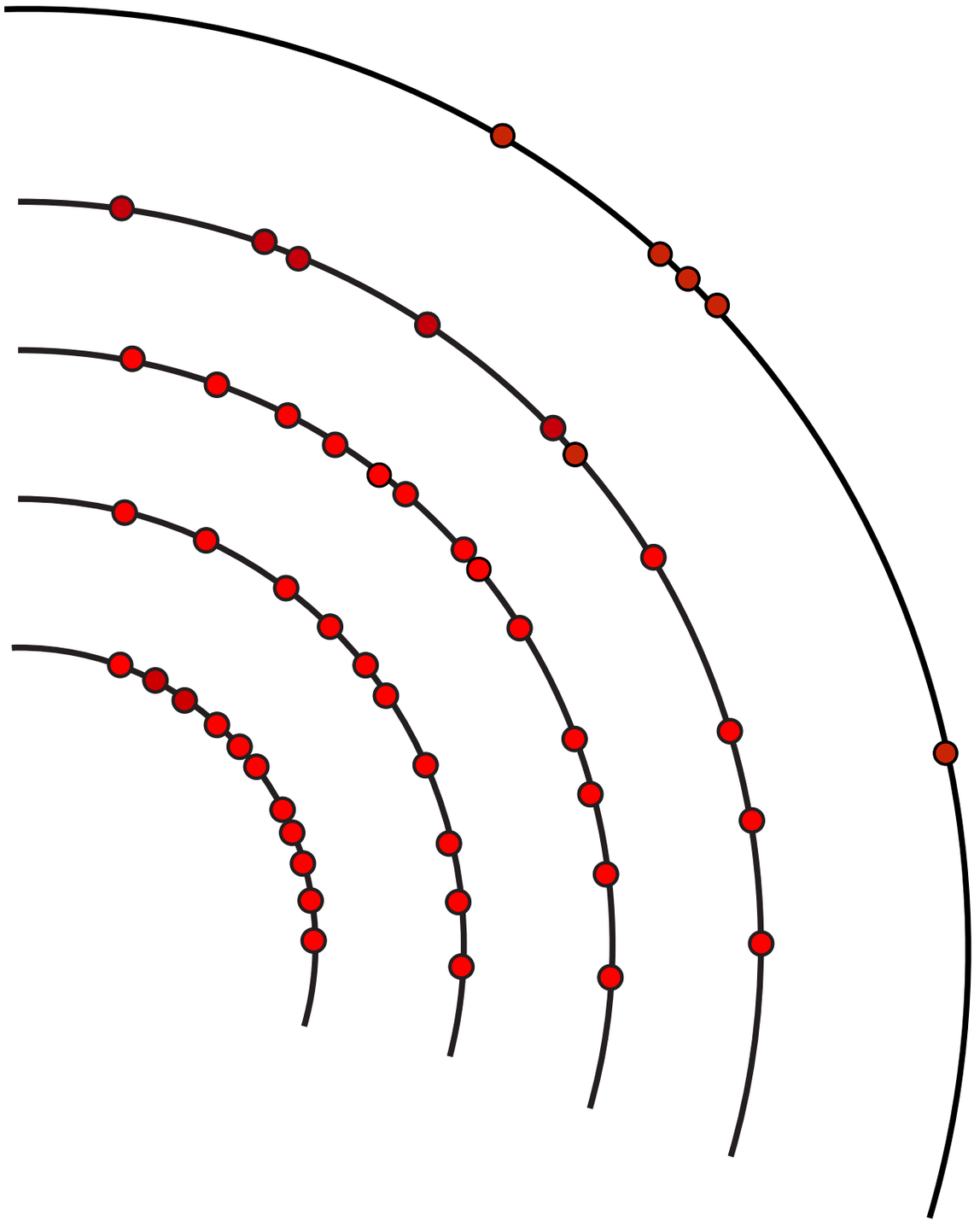
Combinatorial track finding
Combinatorial Kalman Filter
for track finding

Vertex finding + fitting
Iterative, multi variant
primary vertex finders
and fitters

Detector alignment
KF based alignment
functionality

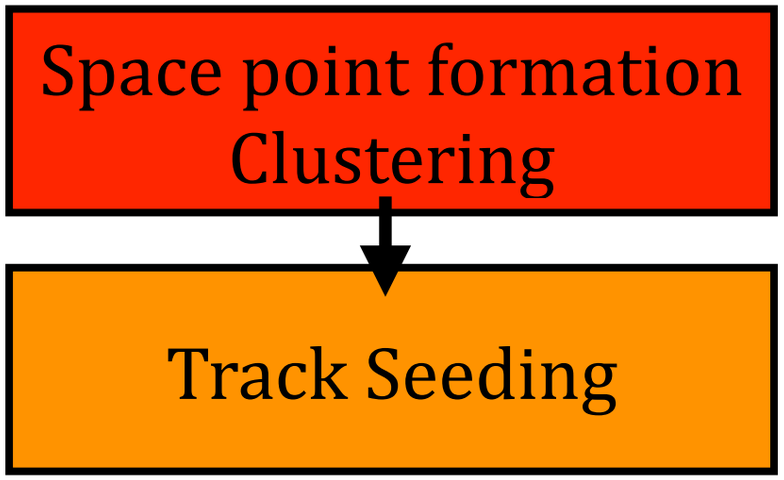
Track Reconstruction

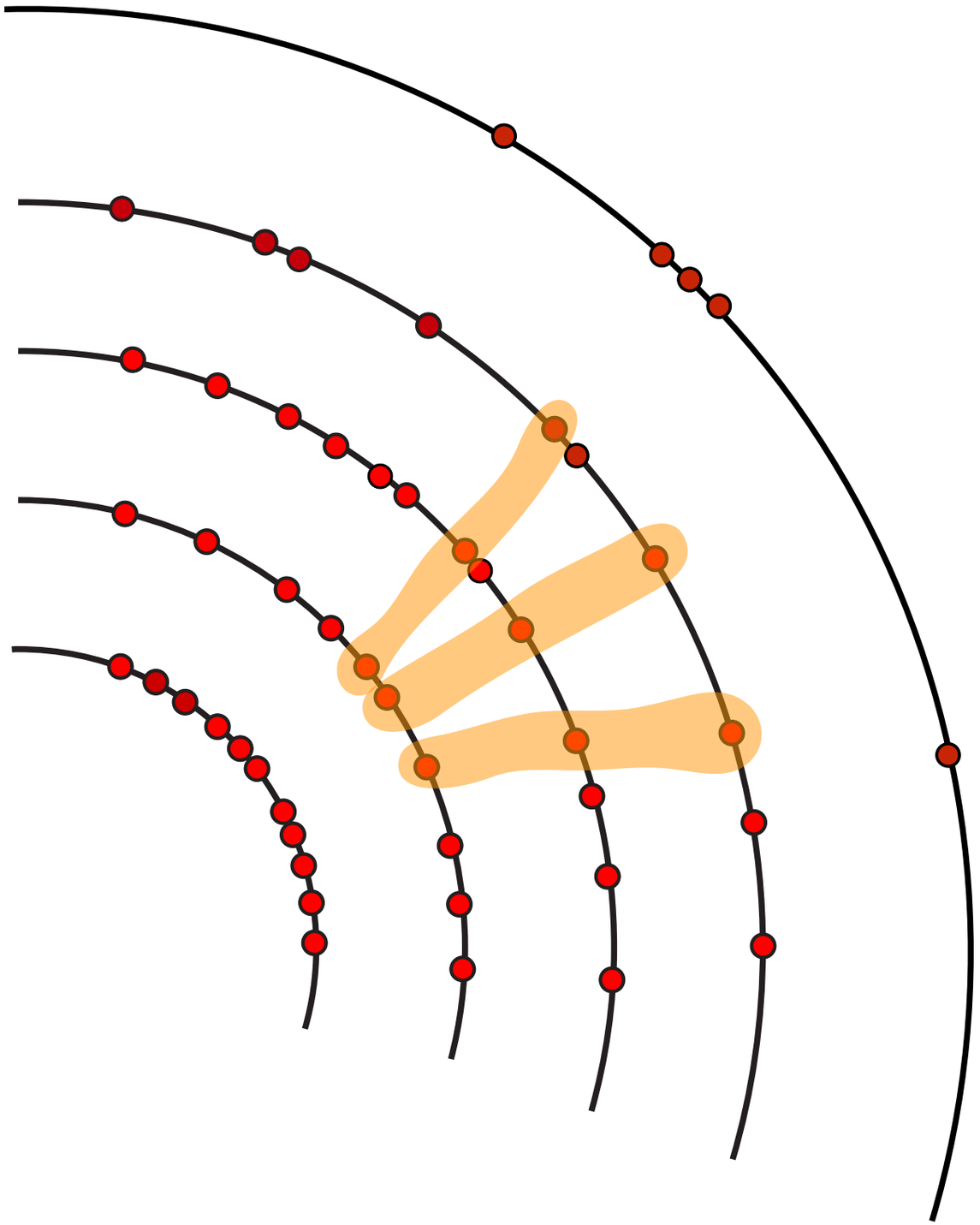
Space point formation
Clustering



- Hits in the detector are collected to create measurement points

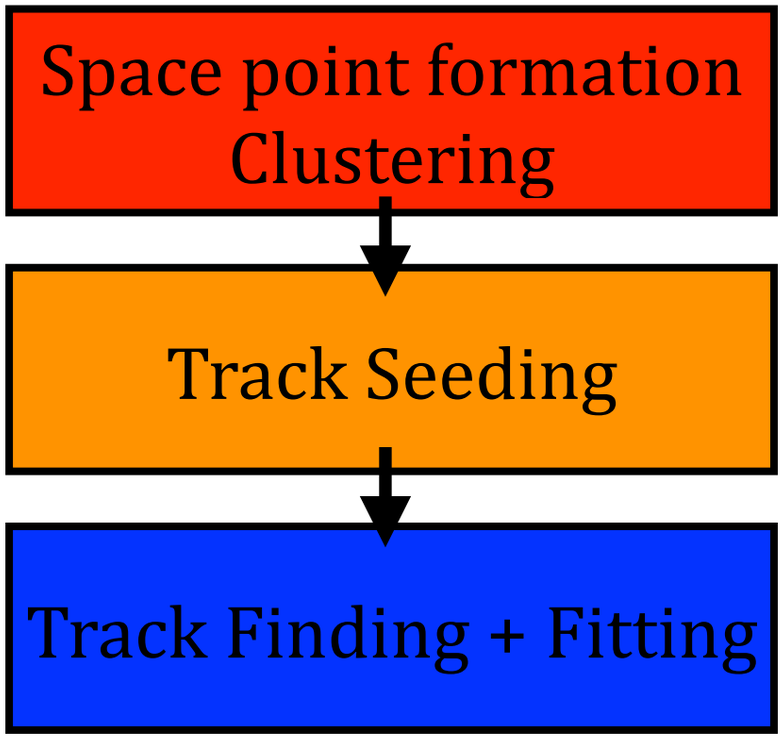
Track Reconstruction

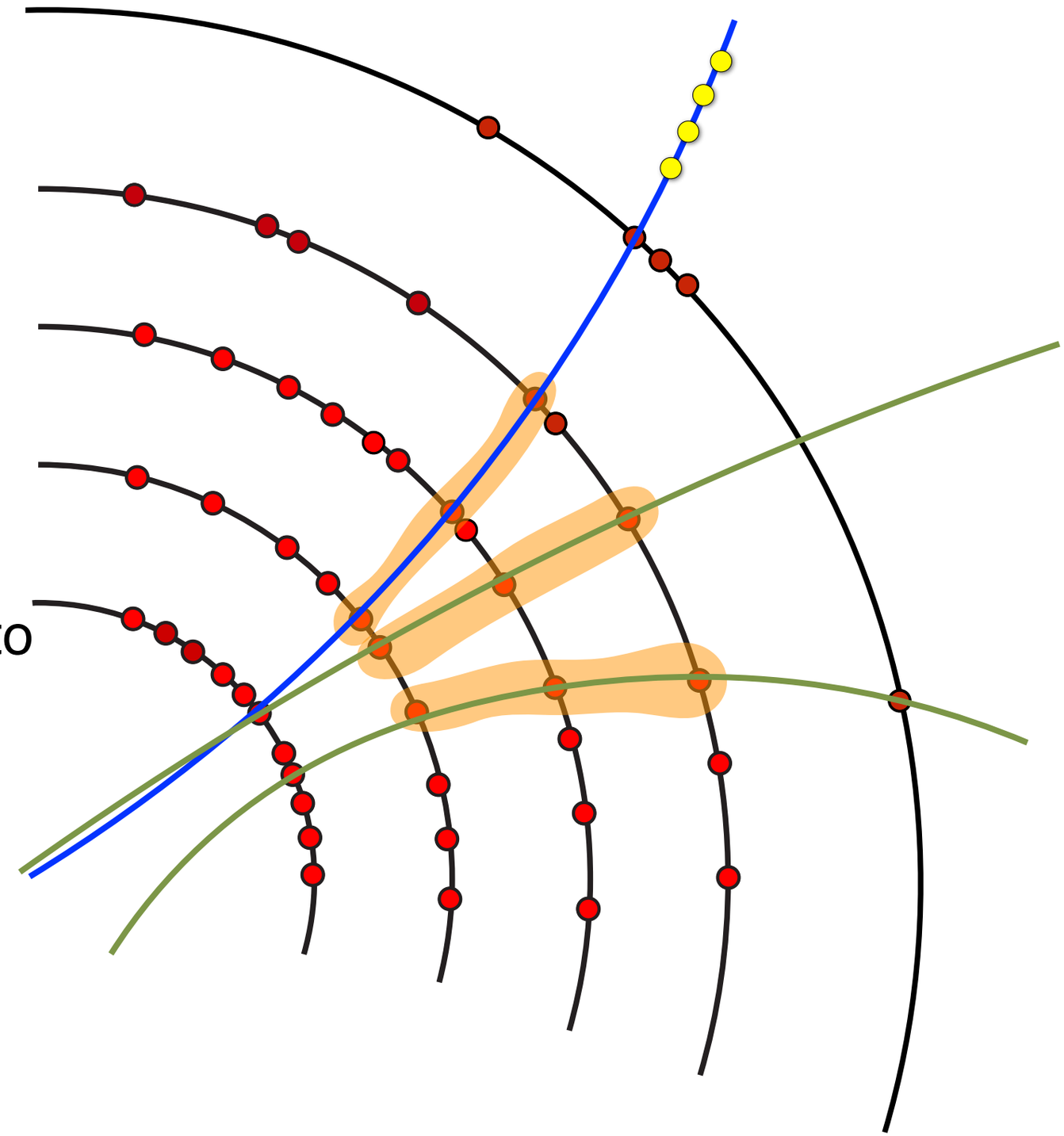




- **Seeds**: groups of **three measurements** compatible with track's hypothesis
- Extended to create the track candidates

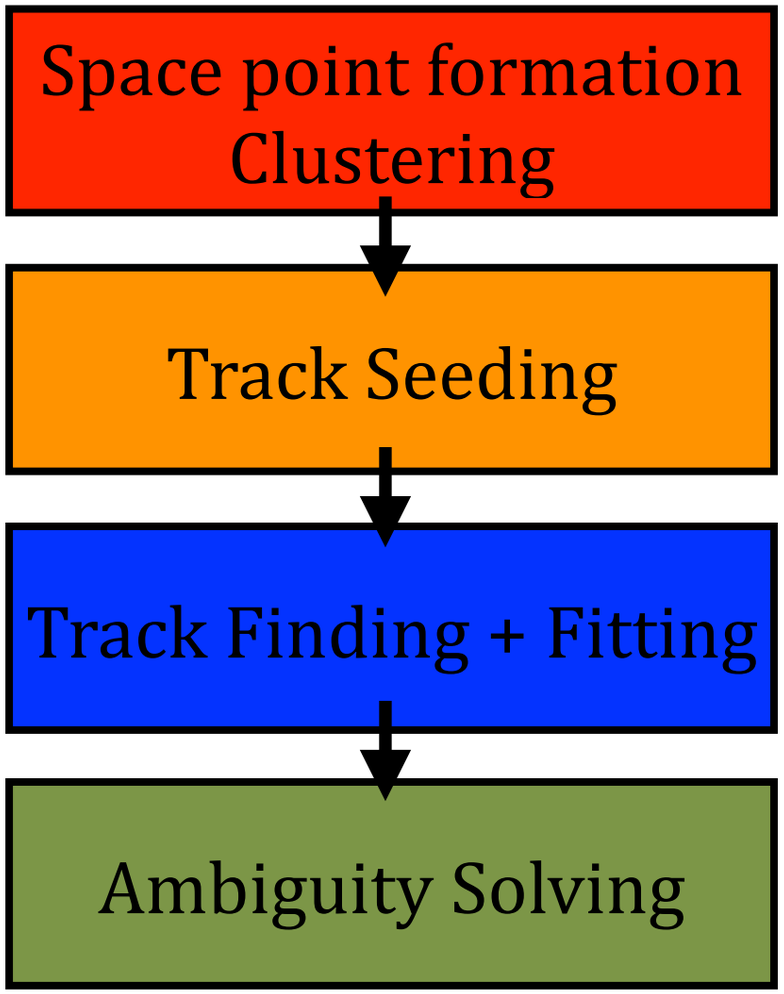
Track Reconstruction



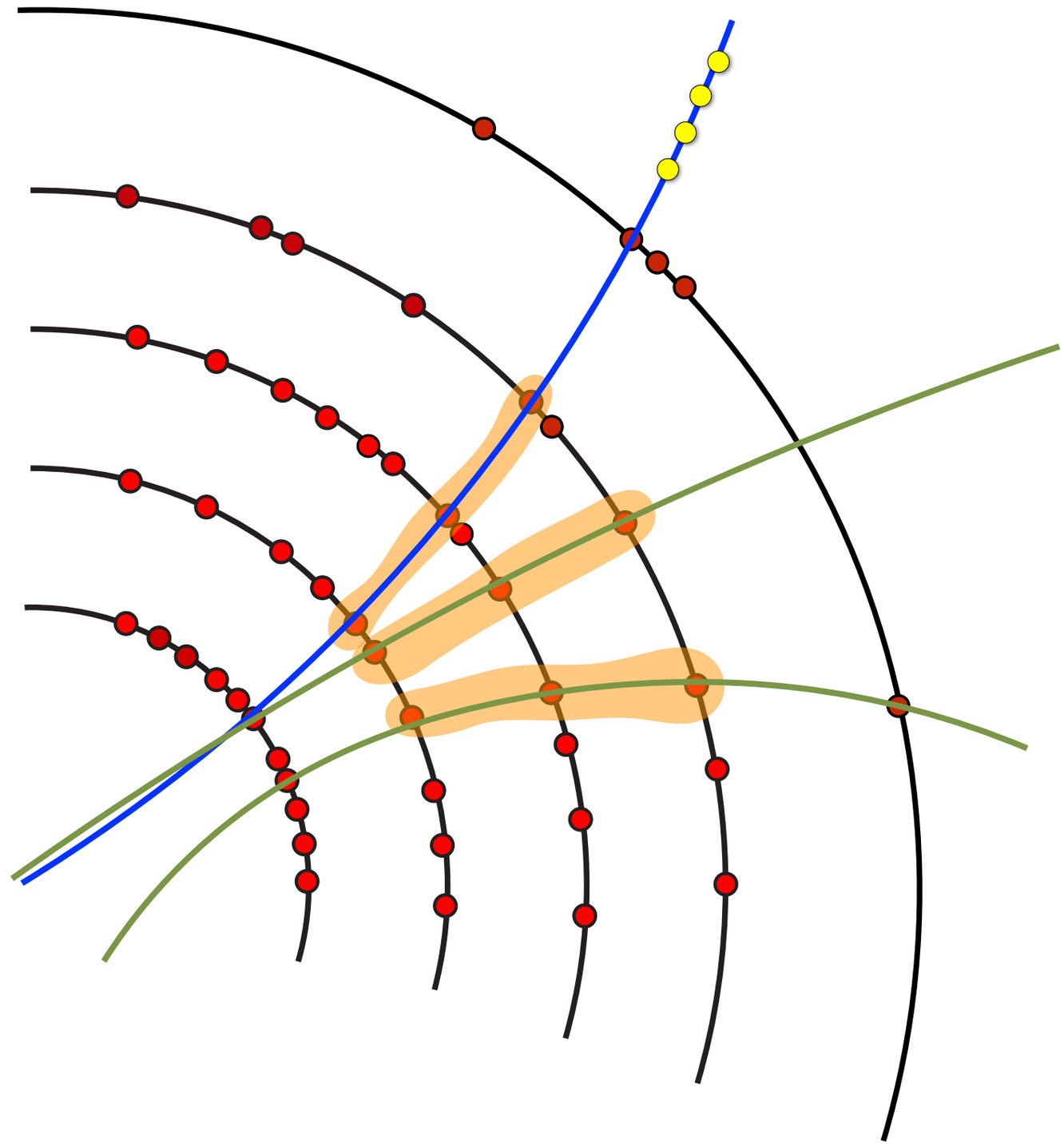


- Combinatorial Kalman filter used to build track candidates
- **Track Candidate** = Seed + compatible measurements
- More than one **track candidate** per seed if multiple paths are possible
- **Tracks** are fitted on the fly

Track Reconstruction

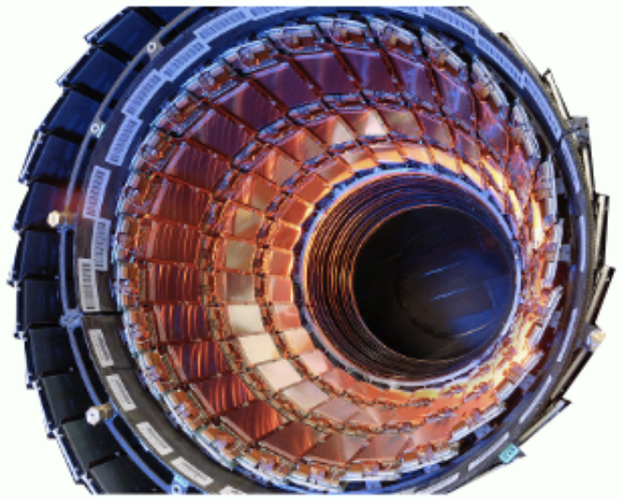


- Score associated with each track candidate
- **Resolve ambiguity**: reconstruct tracks in descending order of a track score

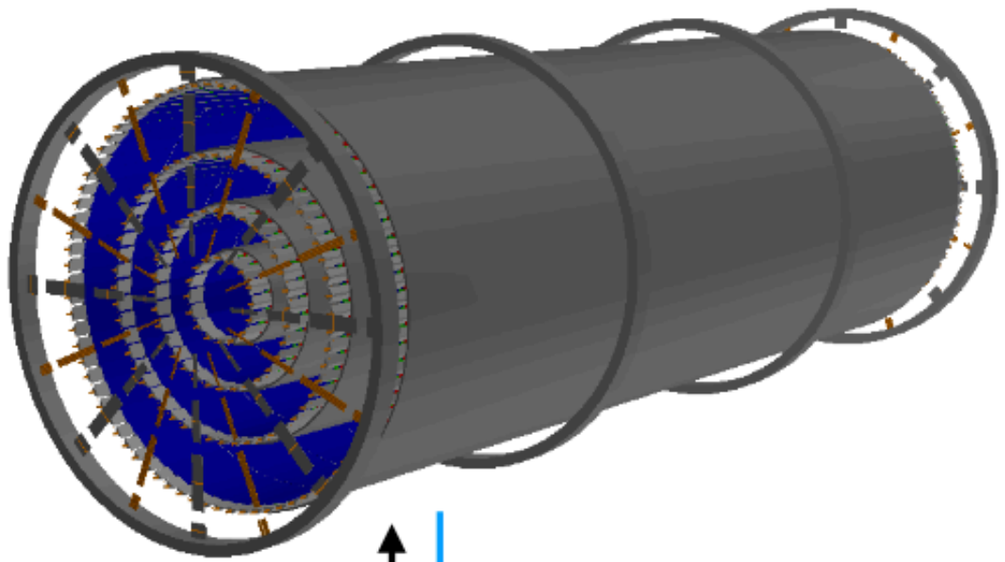


Geometry and detector

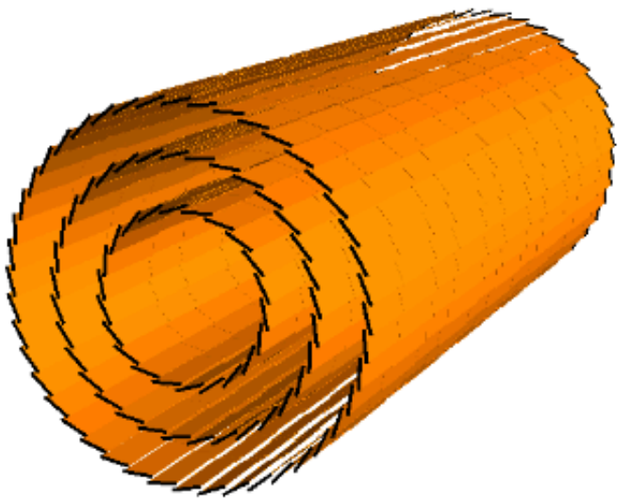
Reality



Full simulation



Reconstruction



Relatively detailed description
using a 3D geometry modeller
(DD4hep, TGeo, GeoModel)
usually interfaced to Geant4

Geometry building in ACTS:

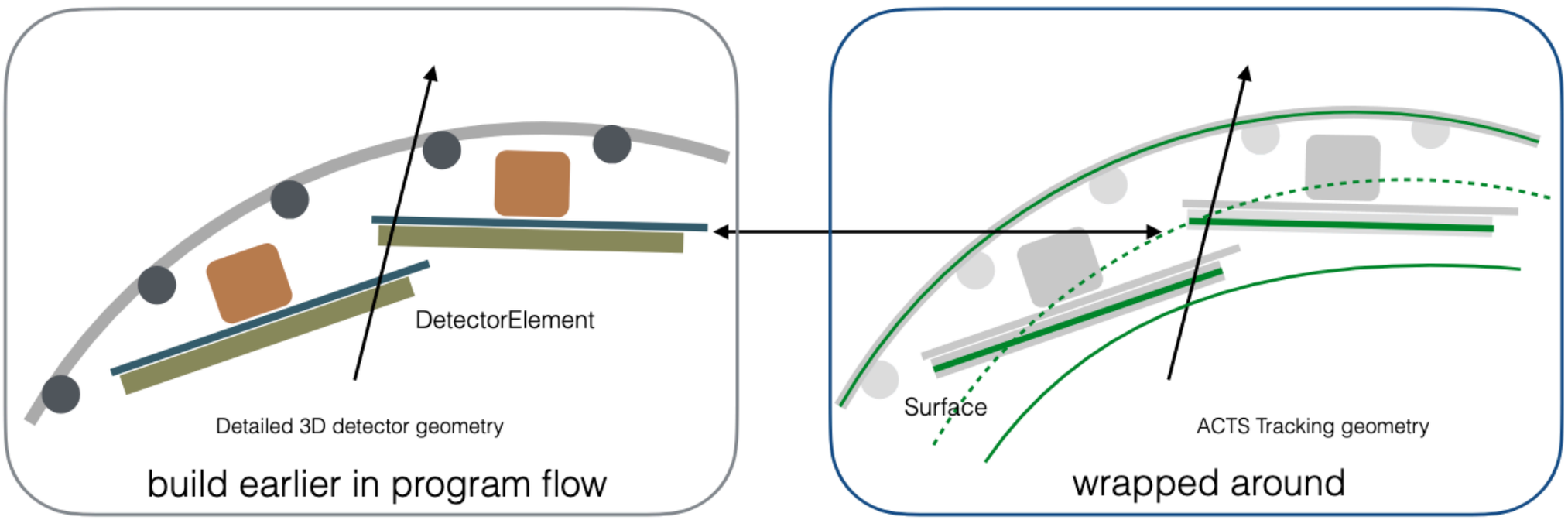
- representing sensitive surfaces
- simplifying material

- Simplified geometry needed to capture material effect
- Built on top of the full detector simulation

Geometry and detector

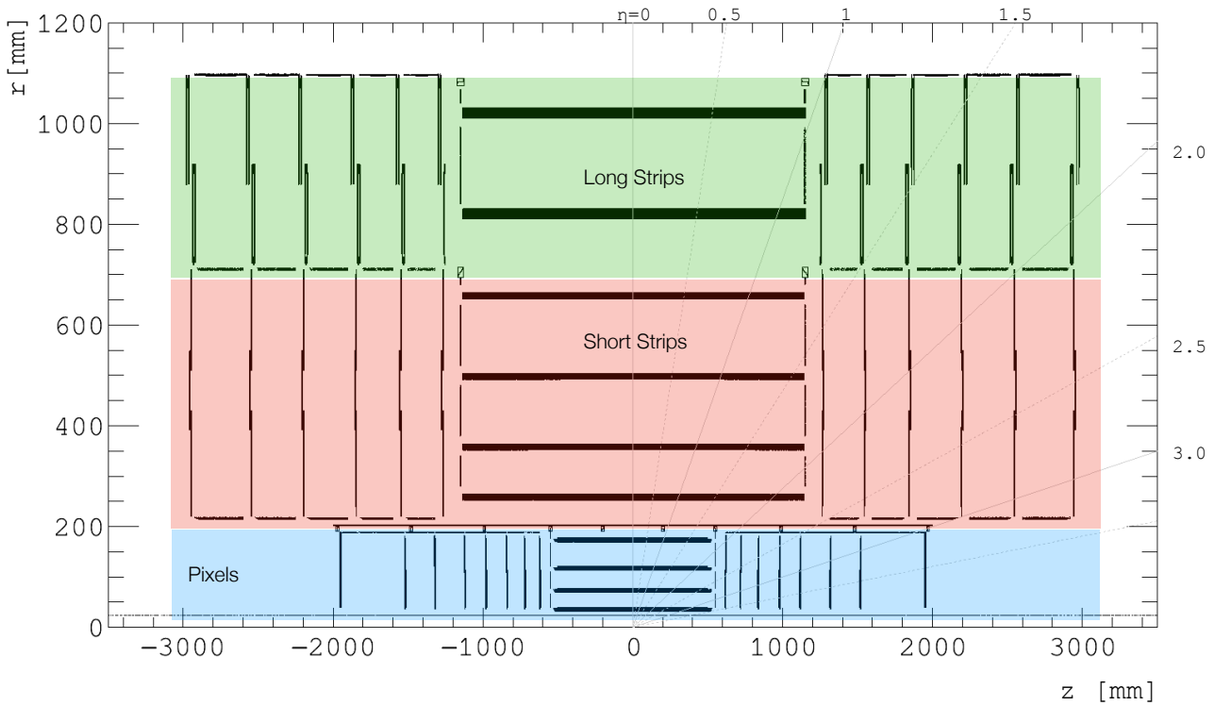
Detailed geometry

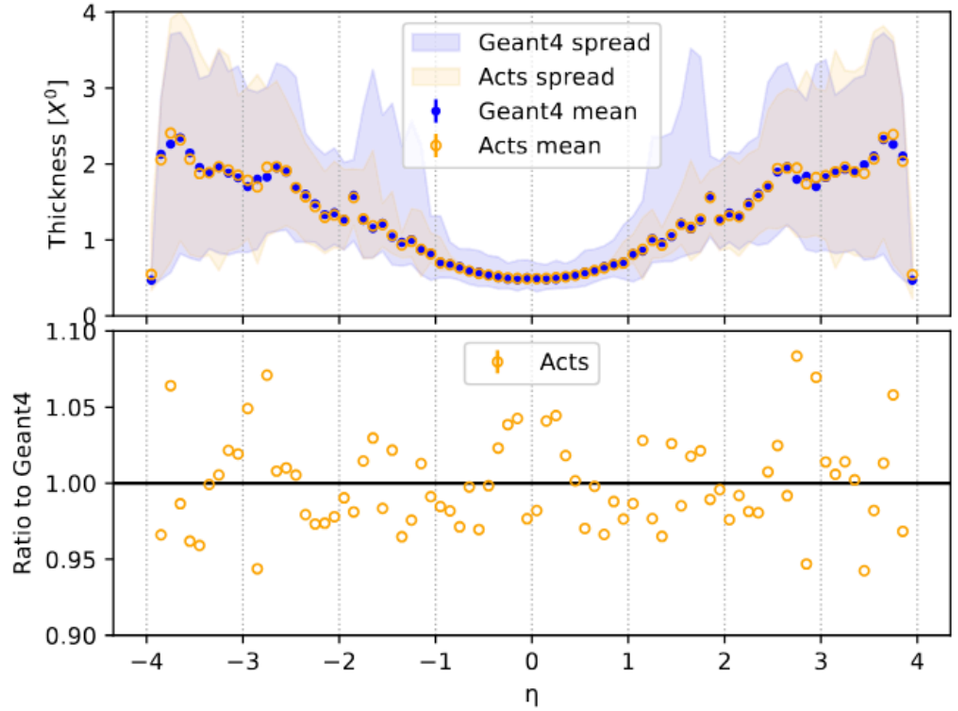
Acts surface based geometry



- Detector implementation done through **DD4Hep**, **TGeo** or **GeoModel**
- Automatic **translation algorithm:** go from those to a simplified « tracking optimised » geometry

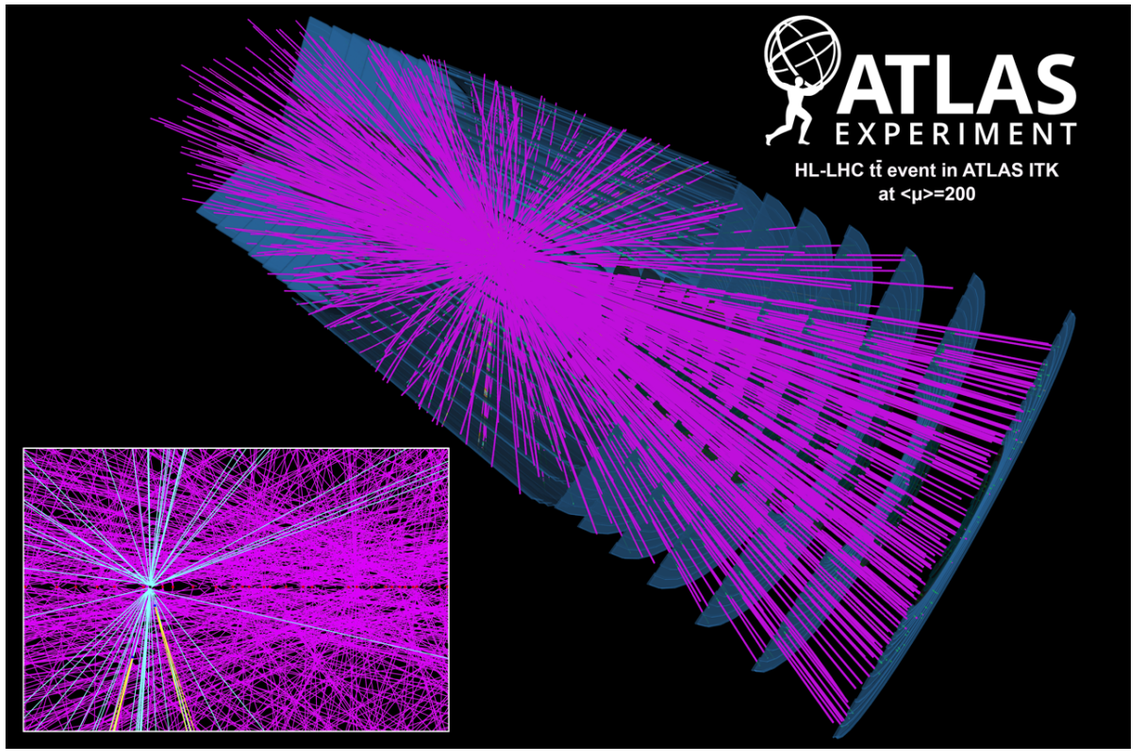
Geometry and detector

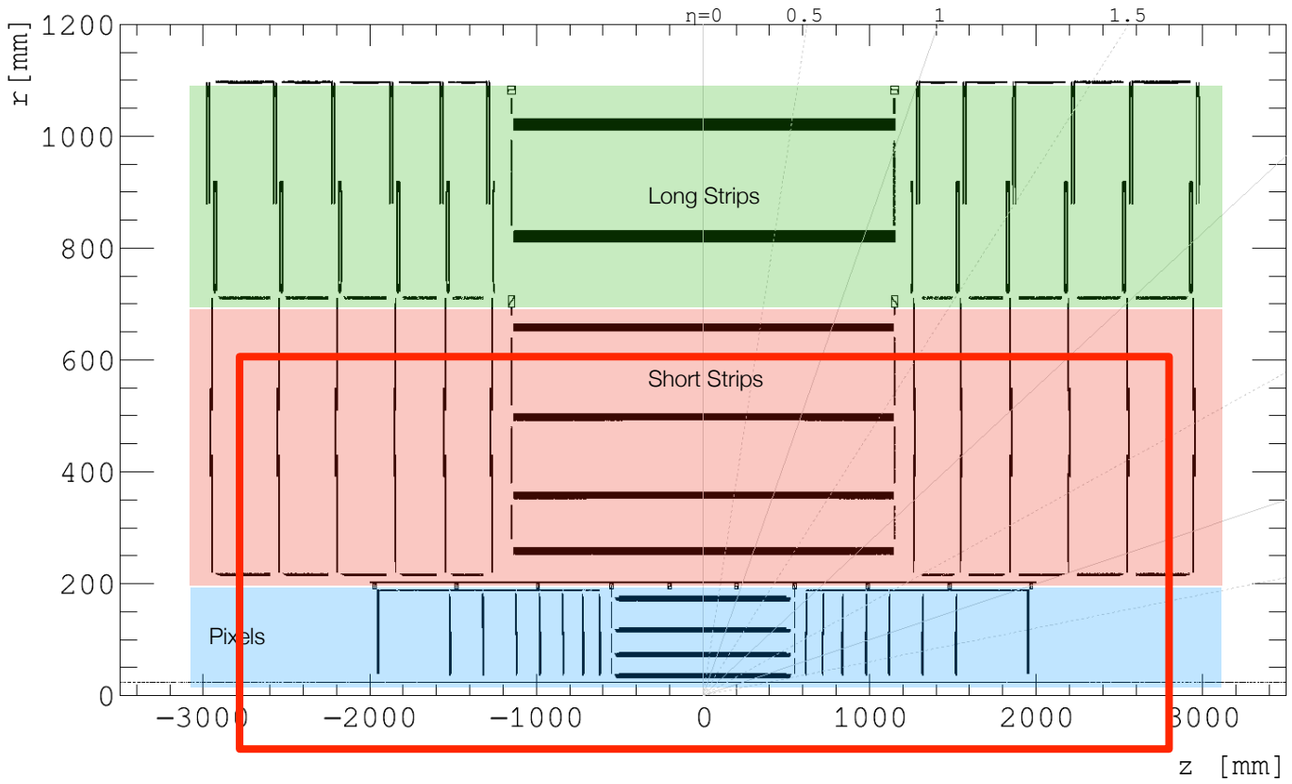




- Tomography of the detector using Geant4 to extract material position
- **Material mapping** module: projection on pre-selected surfaces
- **Material interaction** module: account for particle/matter interaction during particle propagation

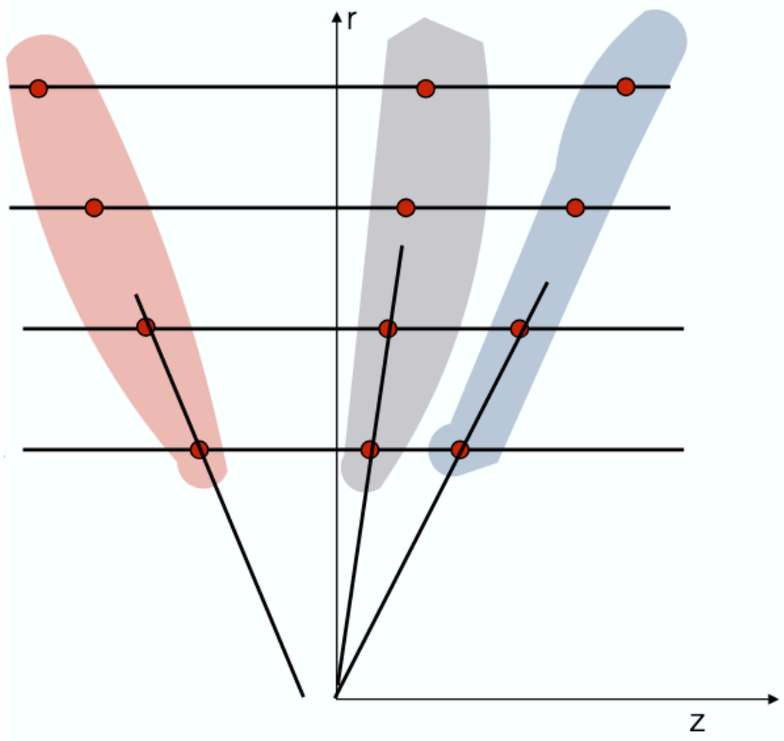
Track reconstruction: Seeding

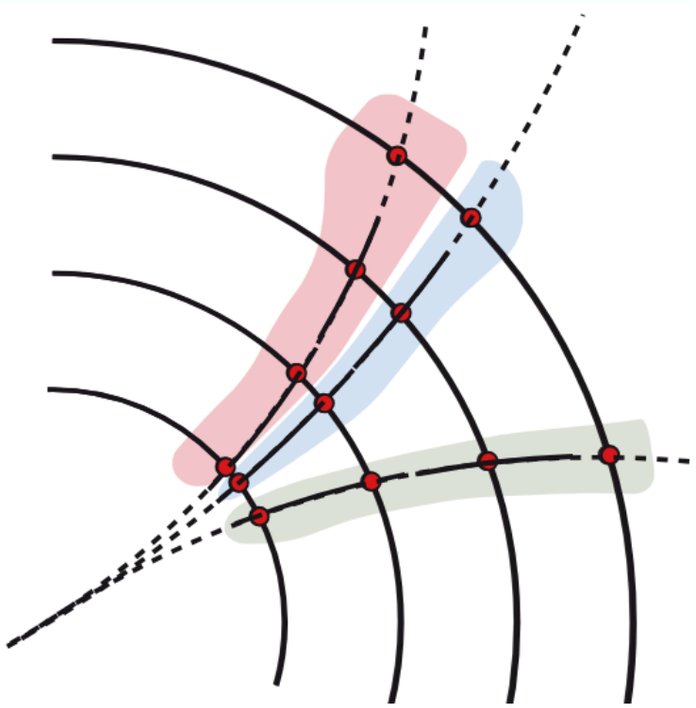


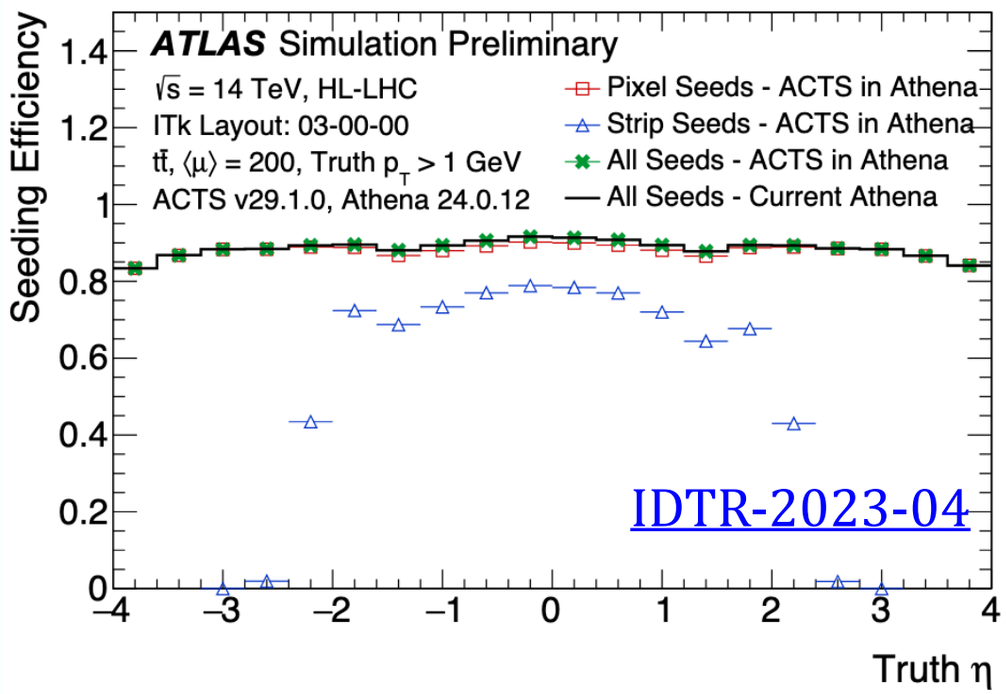


- Combinatoire grows quadratically with the number of hits
- We limit ourselves to the centre of the detector
- Two main approaches in ACTS

Seeding: triplet finding





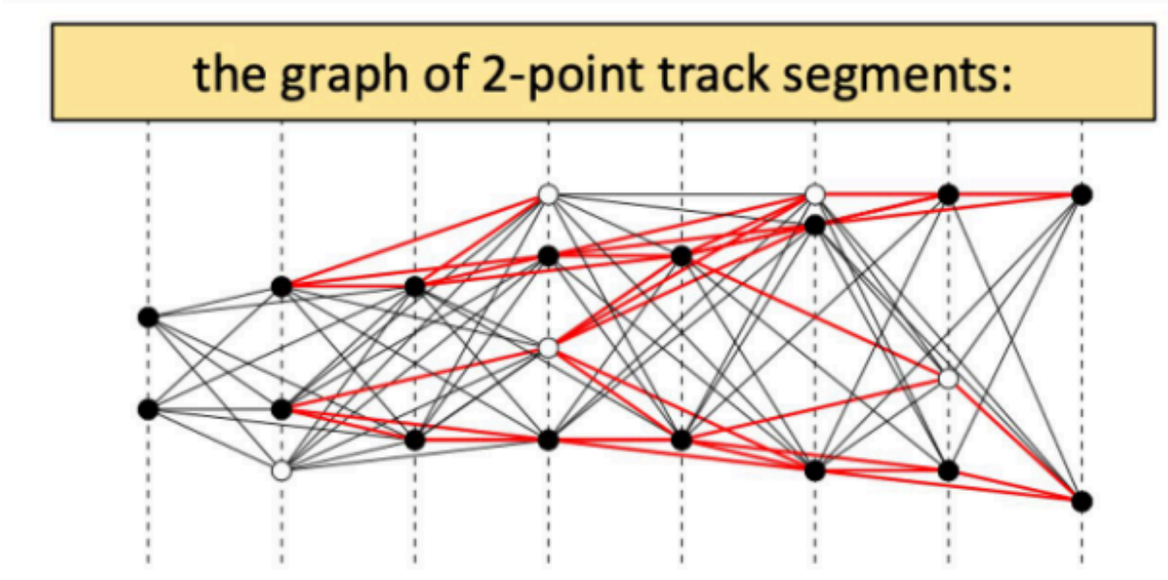


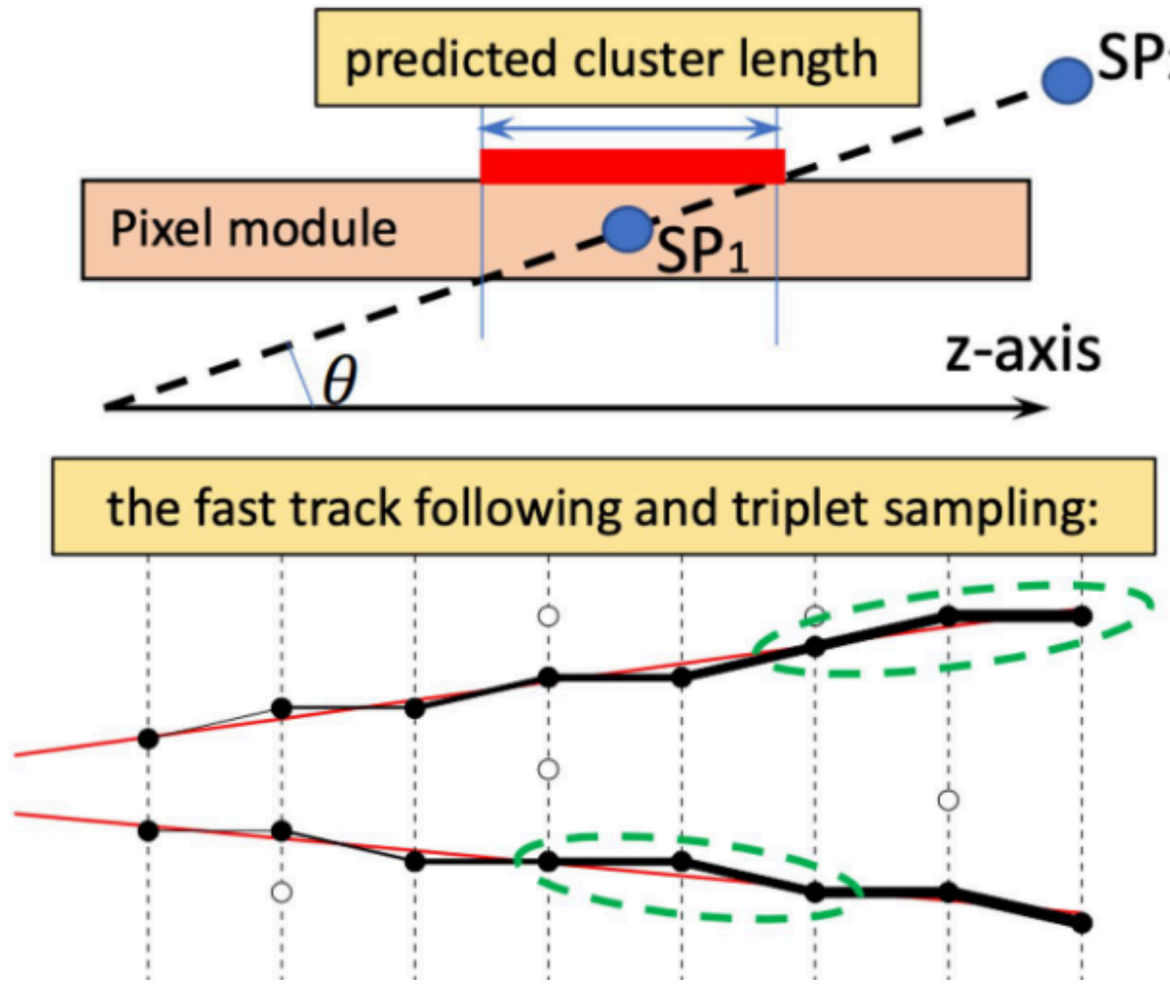
- Loops through all the central hits
- Build doublets with the previous and following layer
- Use doublets to build triplets compatible with the particle hypothesis
- 3 particles, 4 hits each ➡ 12 seeds; Need a filtering step

Seeding: GBTS (Graph Based Track Seeder)

- Create a graph of doublets
- Reduce the number of doublets with an ML classifier
- Use a Connected Component Analysis to find the seed

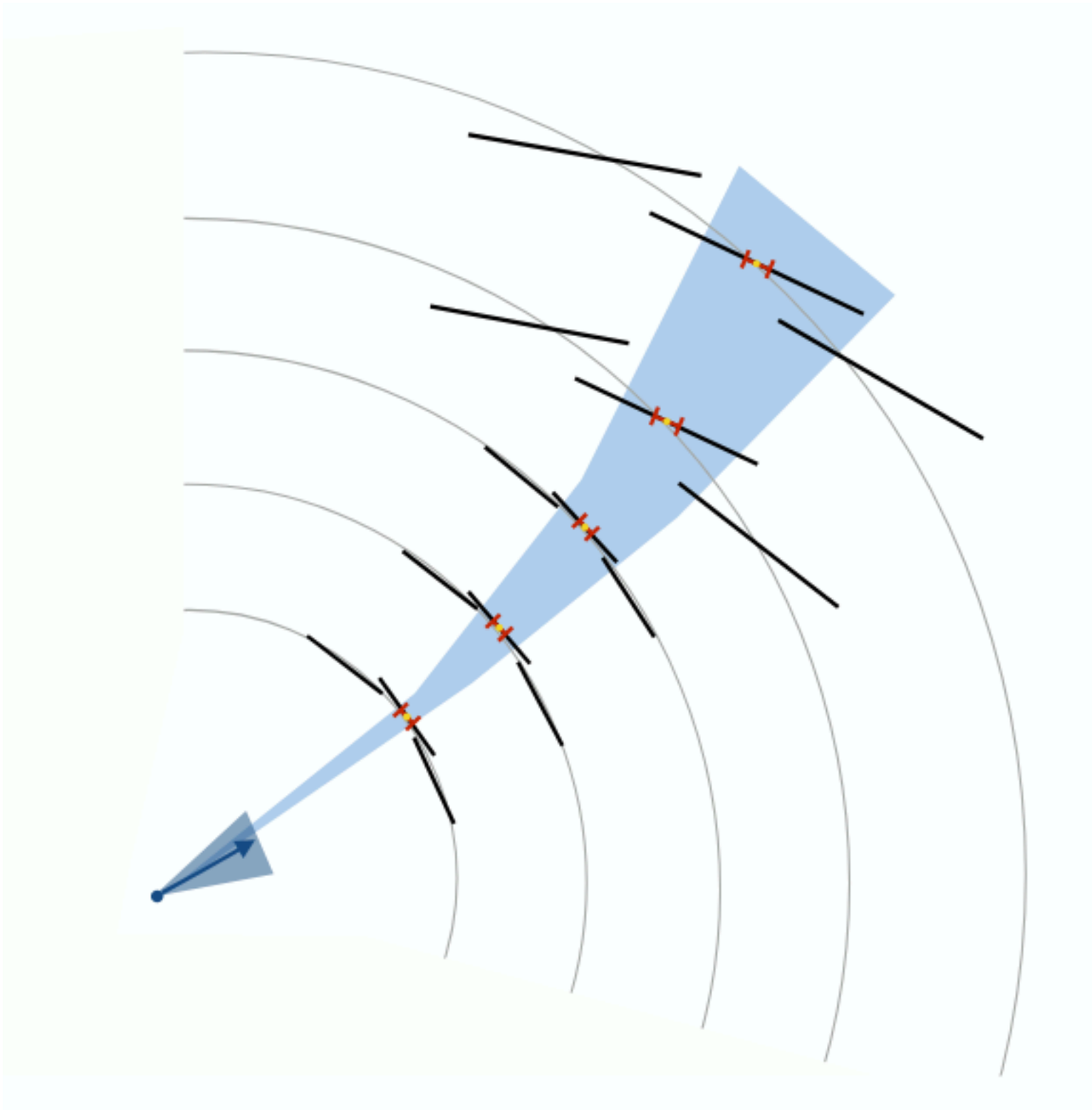
- Originally worked on in the context of ATLAS run3 High Level Trigger (HLT)
- Now integrated back in ACTS





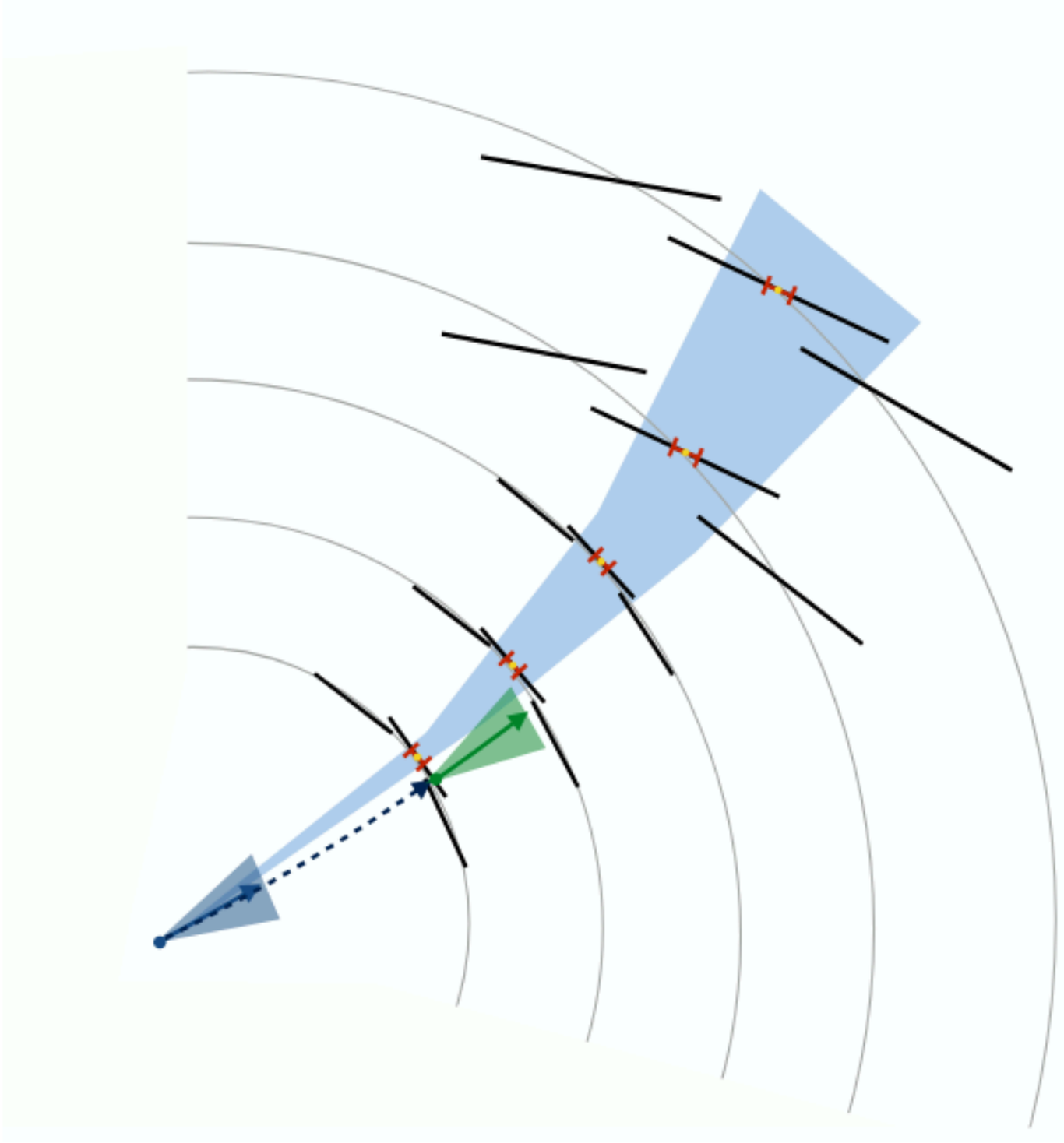
Track reconstruction: Kalman Filter

- Start from the seed direction



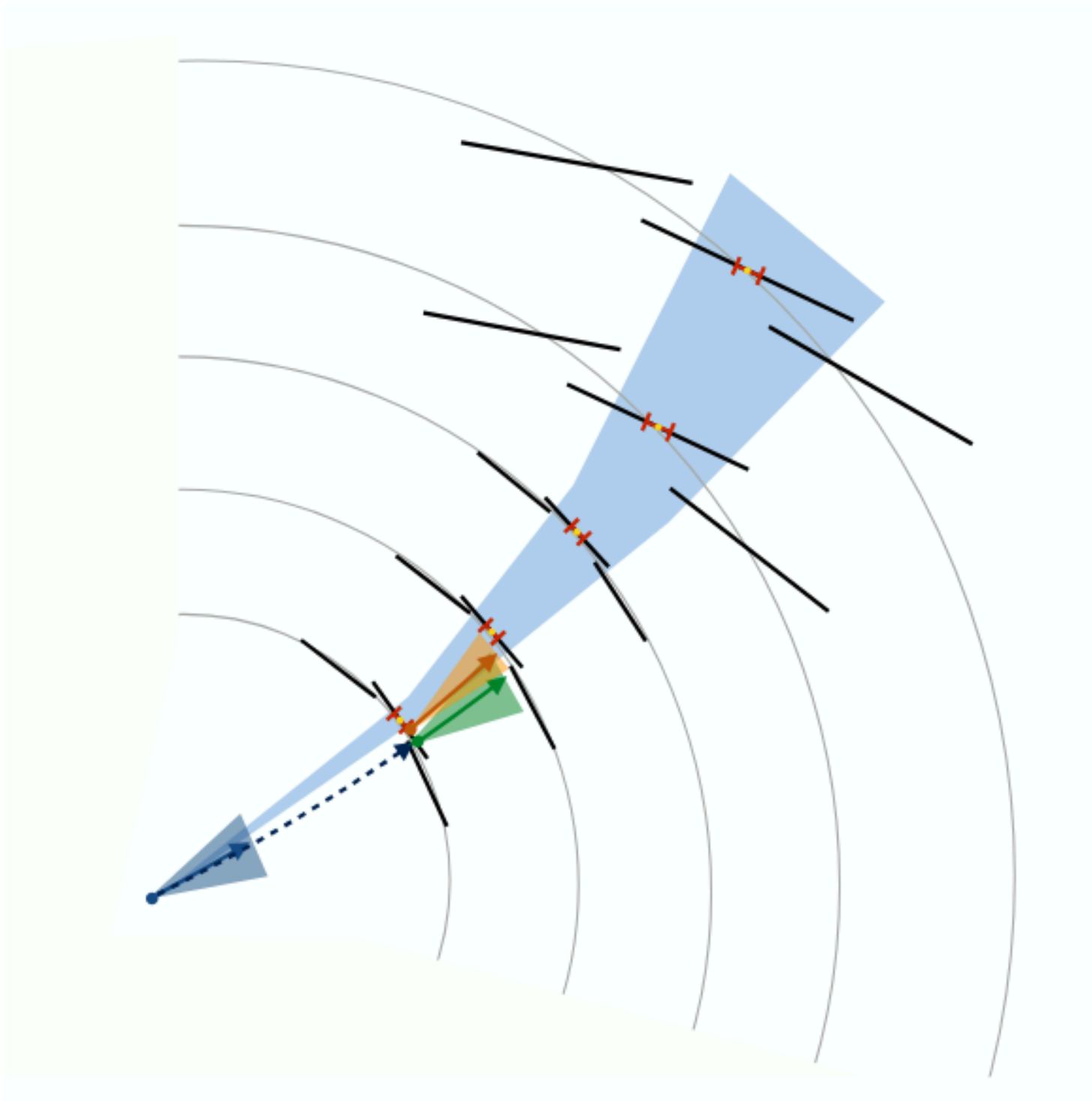
Track reconstruction: Kalman Filter

- Start from the seed direction
- Extrapolation to the next layer



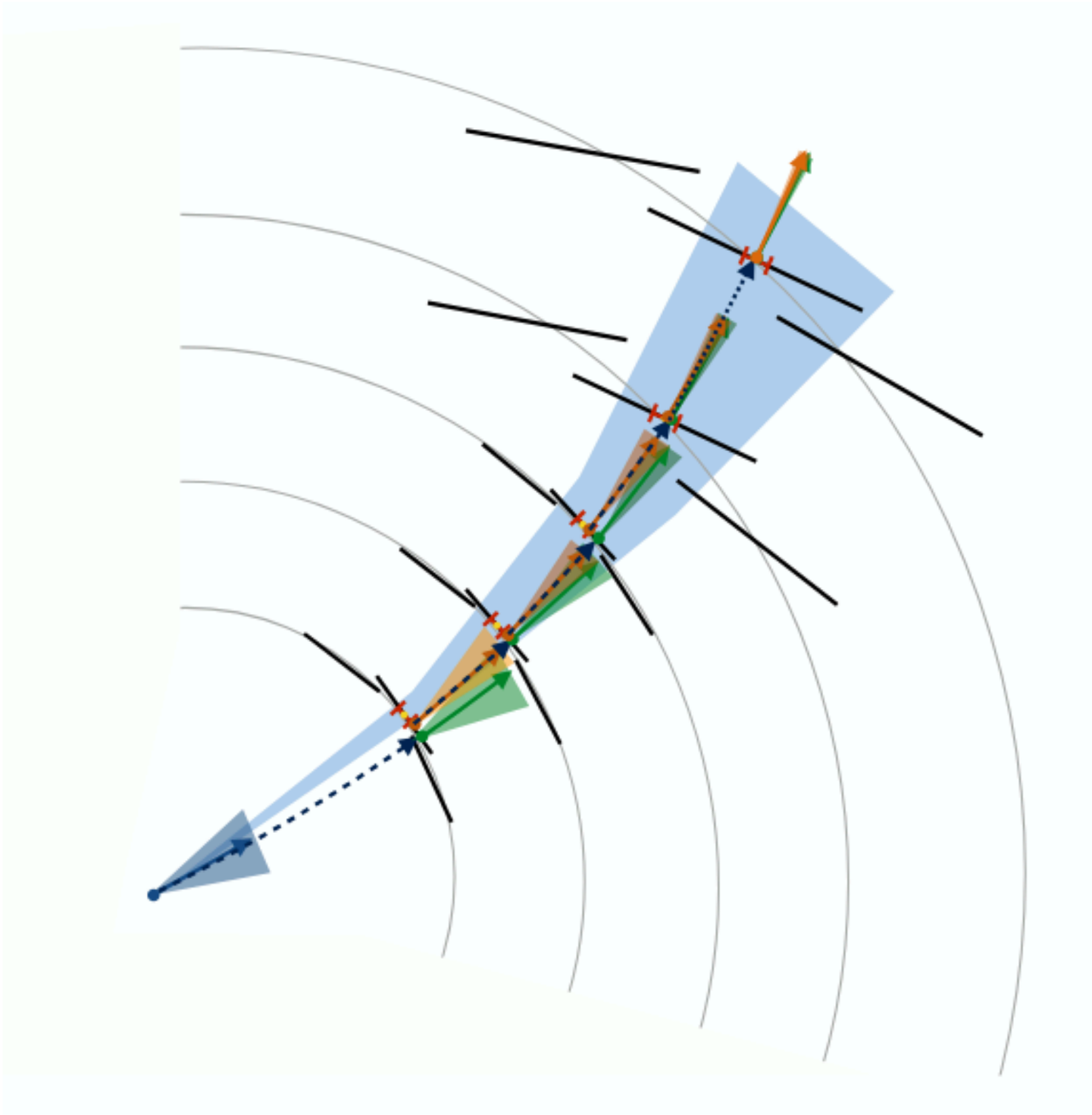
Track reconstruction: Kalman Filter

- Start from the seed direction
- Extrapolation to the next layer
- Correction based on the layer hit



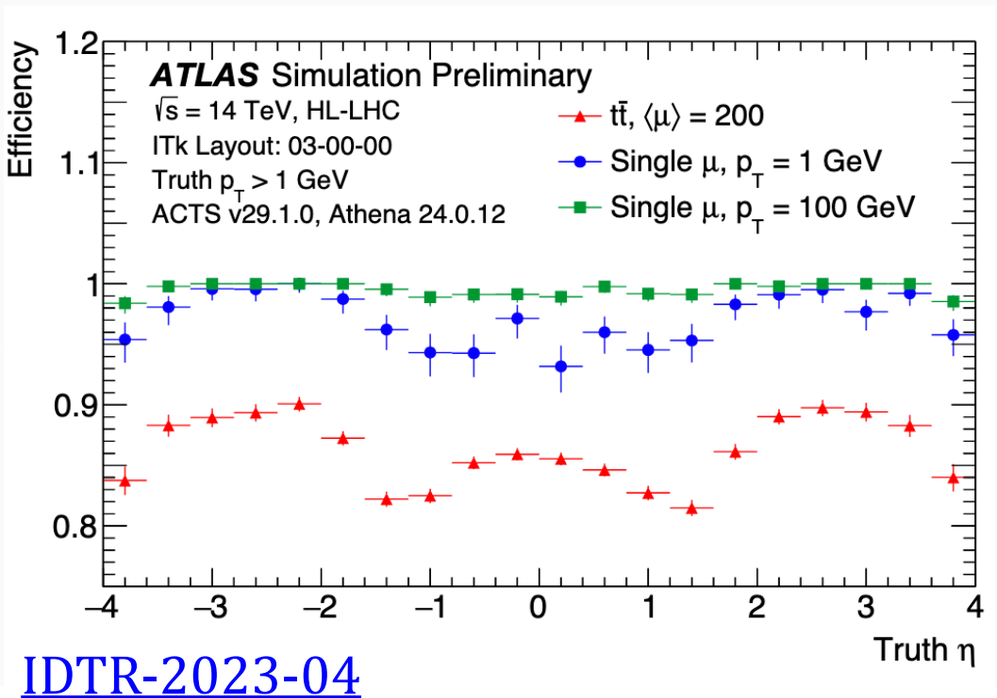
Track reconstruction: Kalman Filter

- Start from the seed direction
- Extrapolation to the next layer
- Correction based on the layer hit
- Iterate through the detector

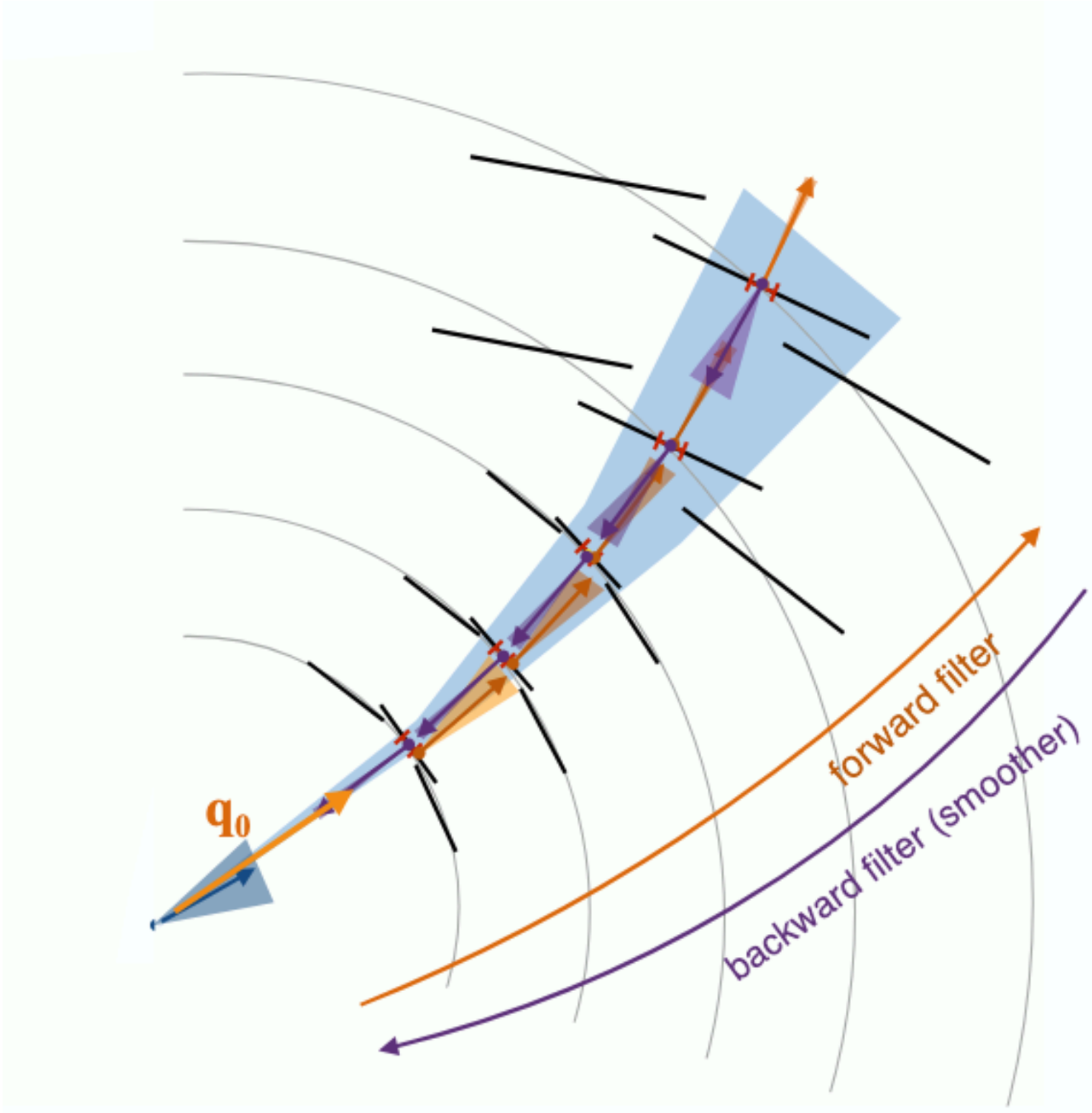


Track reconstruction: Kalman Filter

- Start from the seed direction
- Extrapolation to the next layer
- Correction based on the layer hit
- Iterate through the detector
- Fit of the trajectory

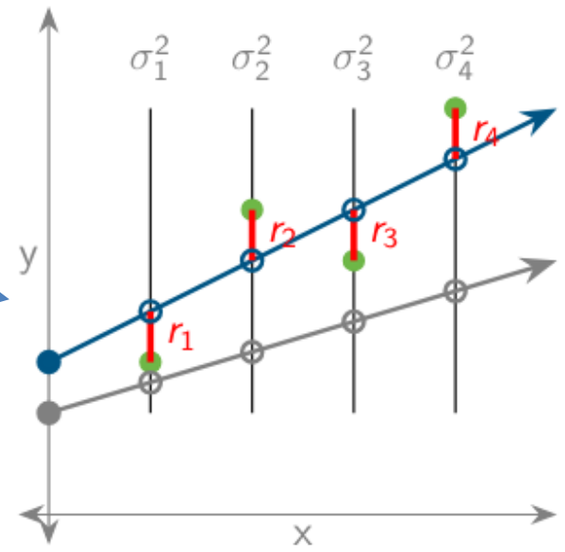


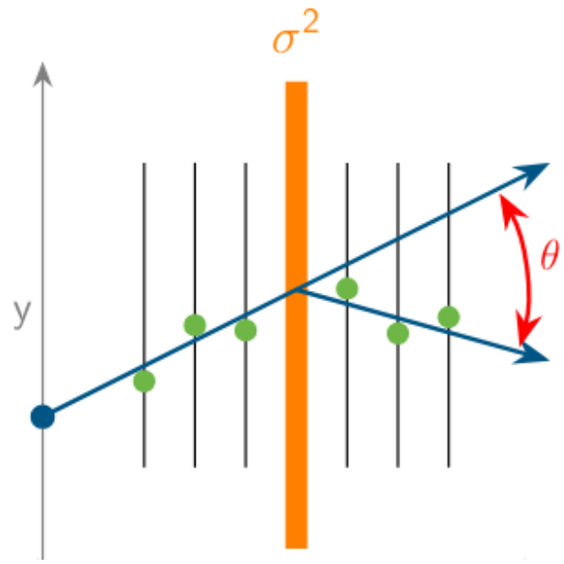
[IDTR-2023-04]

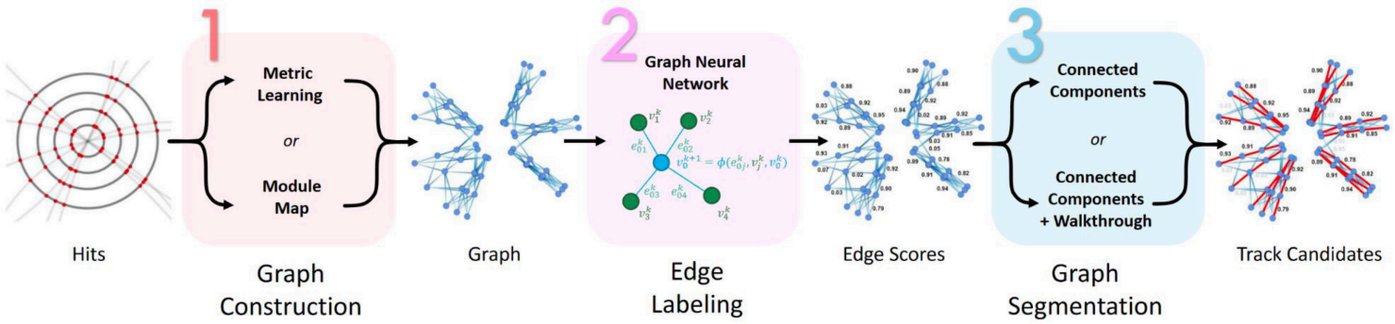


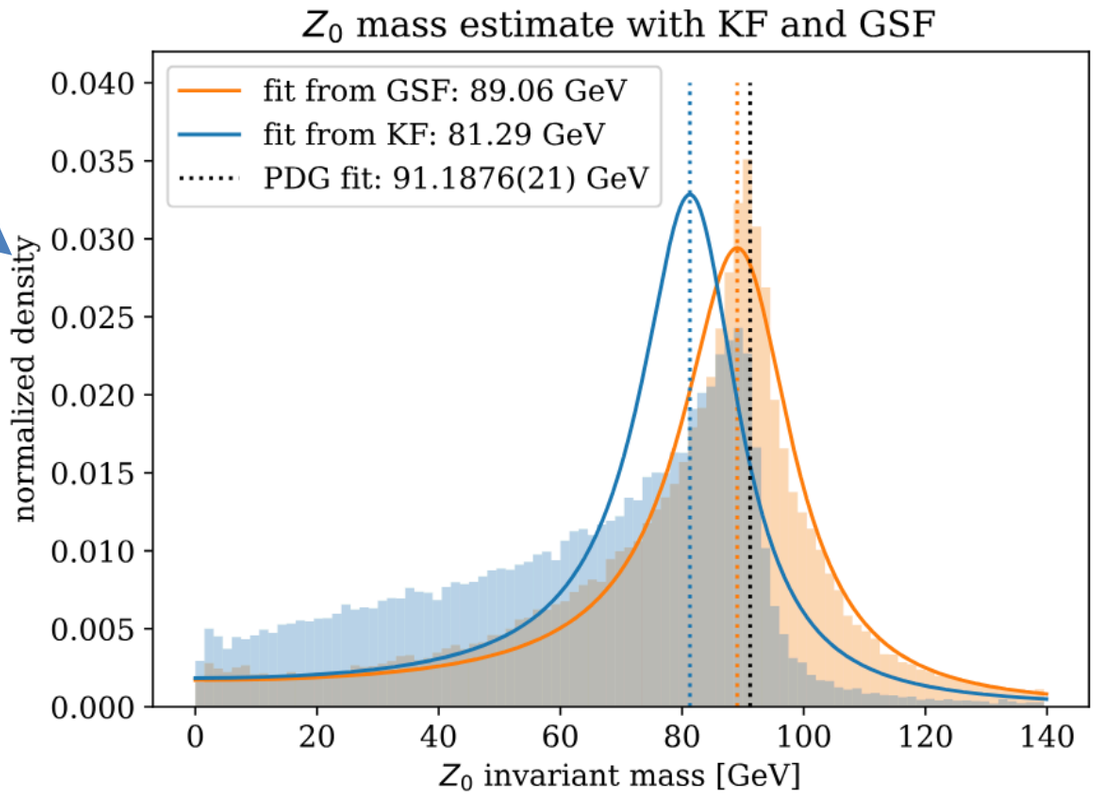
Track reconstruction: Kalman Filter

- Other approaches available:
  - χ^2 base track finding
  - Gaussian Sum Filter for electrons fitting
  - Graph Neural Network-based track finding



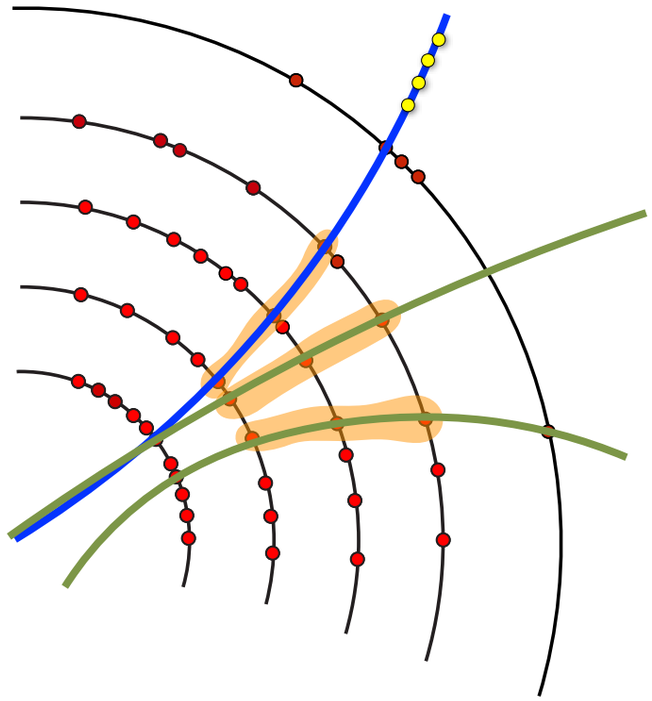






Track reconstruction: Ambiguity resolution

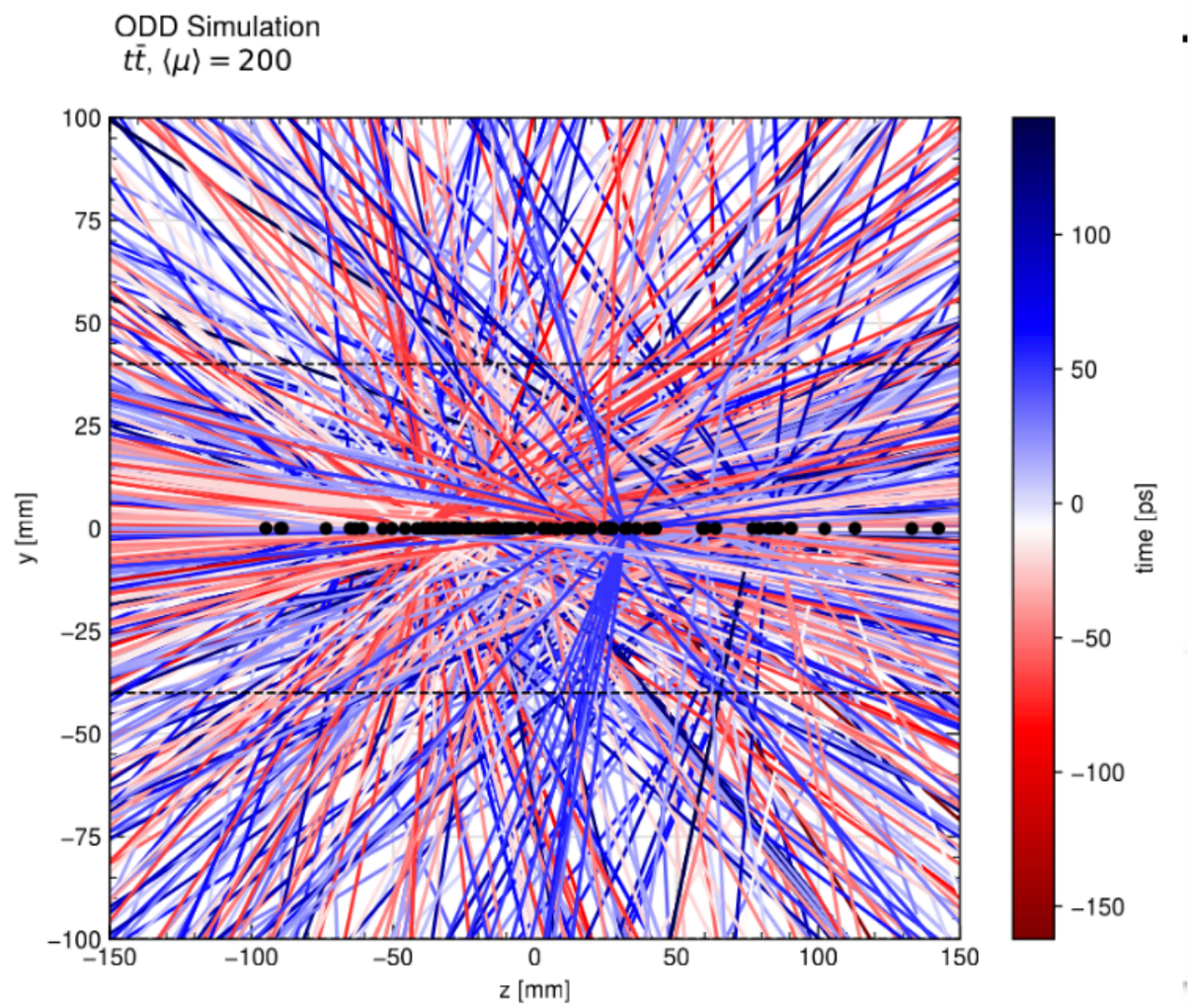
- Remove fake and duplicated tracks
- Three implementations available:



- **Greedy** :
  - Iterate through the track by decreasing track quality
  - Remove tracks that share too many hits
- **Score-based**:
  - Score tracks based on parameters and detector region
  - Hits get associated with higher score tracks
  - Remove low hit track
- **DL-based**:
  - Cluster nearby tracks with DBScan
  - Ranking-based scoring in each cluster
  - Keep the best track per cluster

Vertexing

- Find back the interaction point from reconstructed tracks
- Already implemented and used in ATLAS run3
- Fully functional with timing information

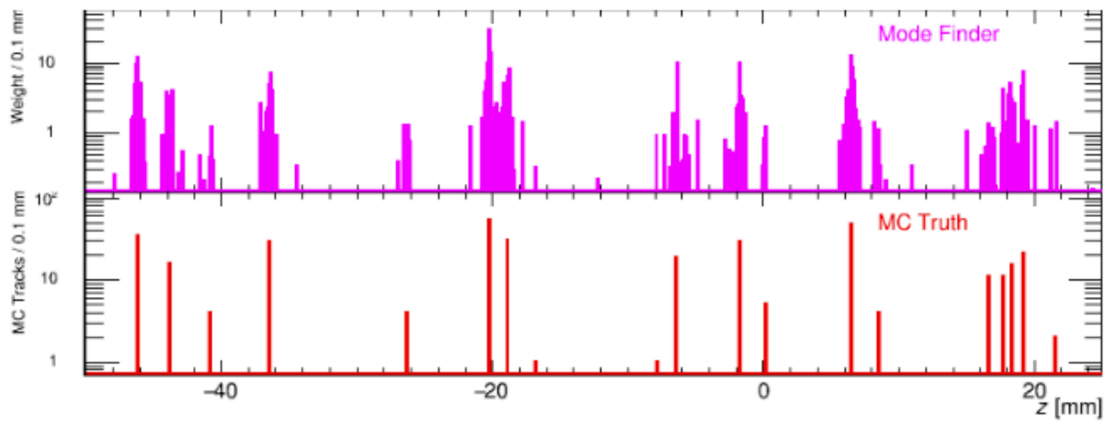


Vertexing

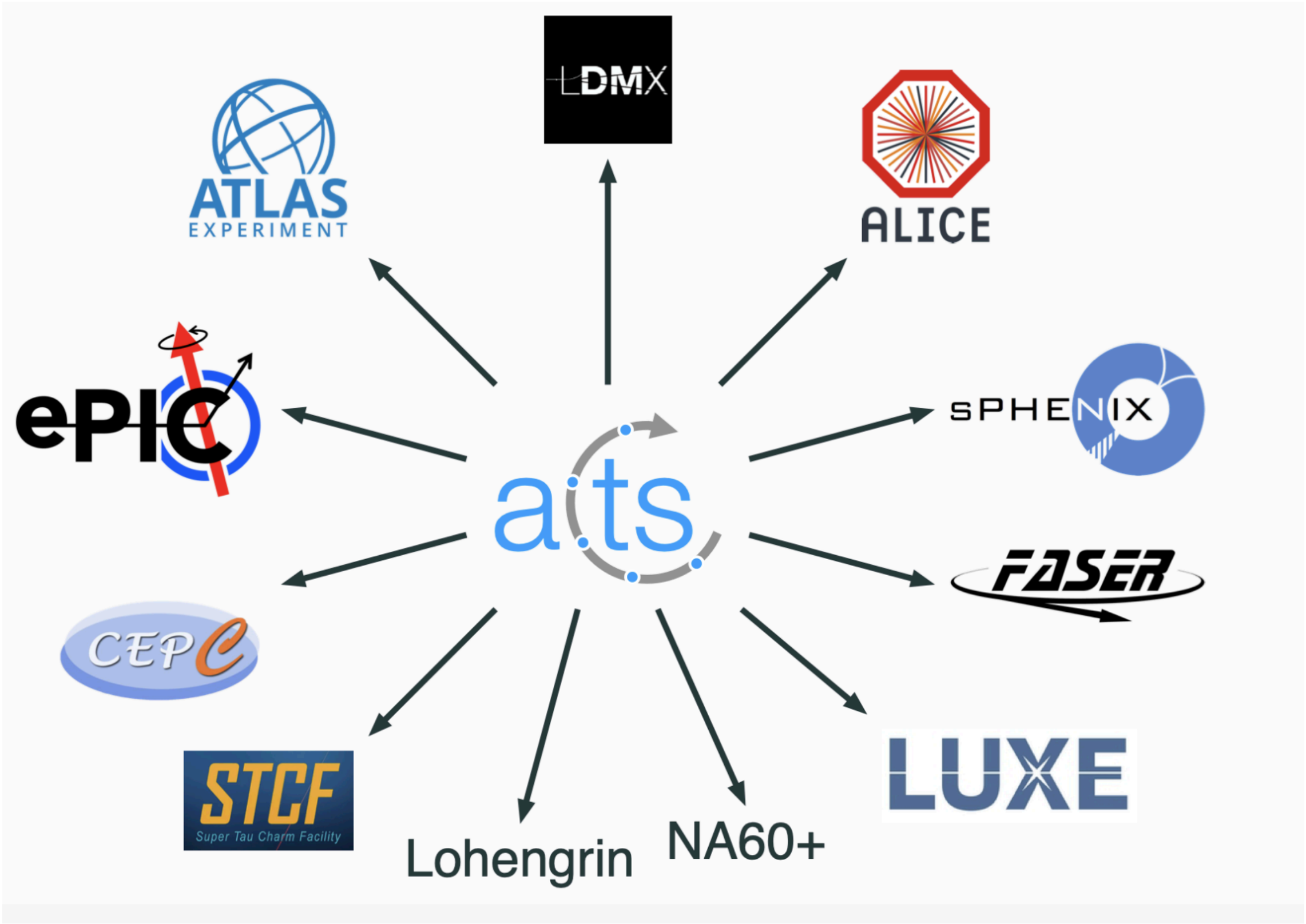
- Find back the interaction point from reconstructed tracks
- Already implemented and used in ATLAS run3
- Fully functional with timing information
- Two implementation:

- **Iterative Vertex Finding (IVF)** :
  - Reconstruct vertices seed based on an analysis of Z0 density
  - Vertices are fitted based on the nearby track (with outliers being progressively removed)

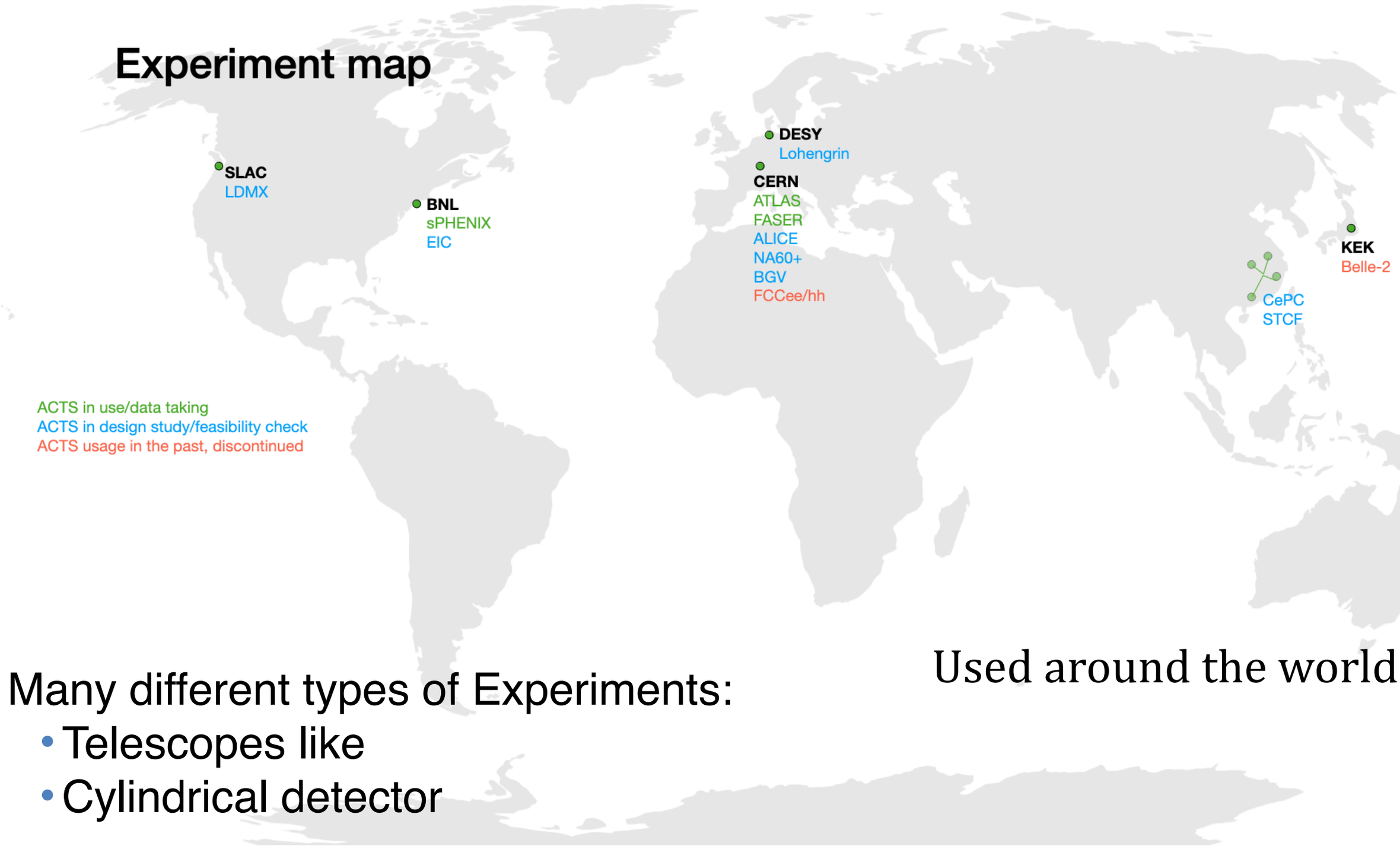
- **Adaptive Multi Vertex Finding (AMVF):**
  - At the start of the fit, tracks can belong to multiple vertices (but at the end, only 1 vertex per track)
  - More performant in high PU environments



The ACTS Community: Users



The ACTS Community: Users

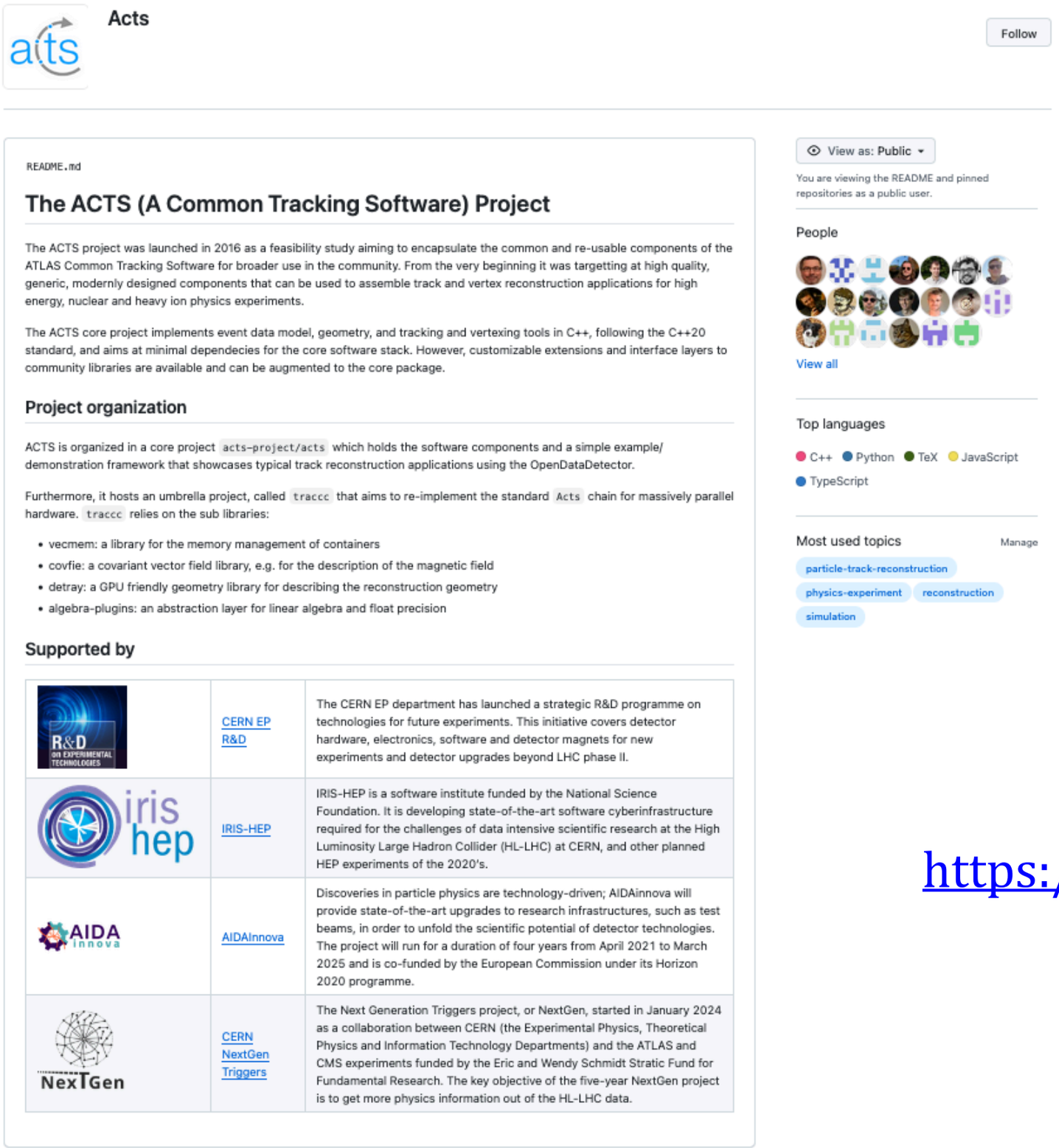


Used around the world !

Many different types of Experiments:

- Telescopes like
- Cylindrical detector

The ACTS Community: Collaborating



Acts

Follow

README.md

The ACTS (A Common Tracking Software) Project

The ACTS project was launched in 2016 as a feasibility study aiming to encapsulate the common and re-usable components of the ATLAS Common Tracking Software for broader use in the community. From the very beginning it was targetting at high quality, generic, modernly designed components that can be used to assemble track and vertex reconstruction applications for high energy, nuclear and heavy ion physics experiments.

The ACTS core project implements event data model, geometry, and tracking and vertexing tools in C++, following the C++20 standard, and aims at minimal dependecies for the core software stack. However, customizable extensions and interface layers to community libraries are available and can be augmented to the core package.

Project organization

ACTS is organized in a core project `acts-project/acts` which holds the software components and a simple example/demonstration framework that showcases typical track reconstruction applications using the OpenDataDetector.

Furthermore, it hosts an umbrella project, called `traccc` that aims to re-implement the standard `Acts` chain for massively parallel hardware. `traccc` relies on the sub libraries:

- vecmem: a library for the memory management of containers
- covfie: a covariant vector field library, e.g. for the description of the magnetic field
- detray: a GPU friendly geometry library for describing the reconstruction geometry
- algebra-plugins: an abstraction layer for linear algebra and float precision

Supported by

R&D on Experimental Technologies	CERN EP R&D	The CERN EP department has launched a strategic R&D programme on technologies for future experiments. This initiative covers detector hardware, electronics, software and detector magnets for new experiments and detector upgrades beyond LHC phase II.
iris hep	IRIS-HEP	IRIS-HEP is a software institute funded by the National Science Foundation. It is developing state-of-the-art software cyberinfrastructure required for the challenges of data intensive scientific research at the High Luminosity Large Hadron Collider (HL-LHC) at CERN, and other planned HEP experiments of the 2020's.
AIDA innova	AIDAInnova	Discoveries in particle physics are technology-driven; AIDAinnova will provide state-of-the-art upgrades to research infrastructures, such as test beams, in order to unfold the scientific potential of detector technologies. The project will run for a duration of four years from April 2021 to March 2025 and is co-funded by the European Commission under its Horizon 2020 programme.
NexTGen	CERN NextGen Triggers	The Next Generation Triggers project, or NextGen, started in January 2024 as a collaboration between CERN (the Experimental Physics, Theoretical Physics and Information Technology Departments) and the ATLAS and CMS experiments funded by the Eric and Wendy Schmidt Stratic Fund for Fundamental Research. The key objective of the five-year NextGen project is to get more physics information out of the HL-LHC data.

View as: Public

You are viewing the README and pinned repositories as a public user.

People

View all

Top languages

C++ Python TeX JavaScript TypeScript

Most used topics Manage

particle-track-reconstruction physics-experiment reconstruction simulation

- Issues and PR are welcome
- Mattermost chat for discussion with the core developers
- Meeting every Tuesday 17h00

https://github.com/acts-project/acts

The ACTS Community: Collaborating

```
▸ Run pre-commit run --all-files --show-diff-on-failure
clang-format.....................................................Passed
Trim Trailing Whitespace.........................................Passed
Fix End of Files.................................................Passed
Check Yaml.......................................................Passed
Check for added large files......................................Passed
black-jupyter....................................................Passed
gersemi..........................................................Passed
codespell........................................................Passed
license..........................................................Passed
include_guards...................................................Passed
pragma_once......................................................Passed
type_t...........................................................Passed
boost_test.......................................................Passed
cmake_options....................................................Passed
Leftover conflict markers........................................Passed
math_macros......................................................Passed
codegen_dependencies.............................................Passed
```

- Rules in place to facilitate collaboration
- Extensive use of CI to ensure quality and style

- Performance Monitoring
- Regression test
- Test of the integration within the ATLAS framework

github-actions bot commented 3 days ago • edited

📊: Physics performance monitoring for `21f5b8a`

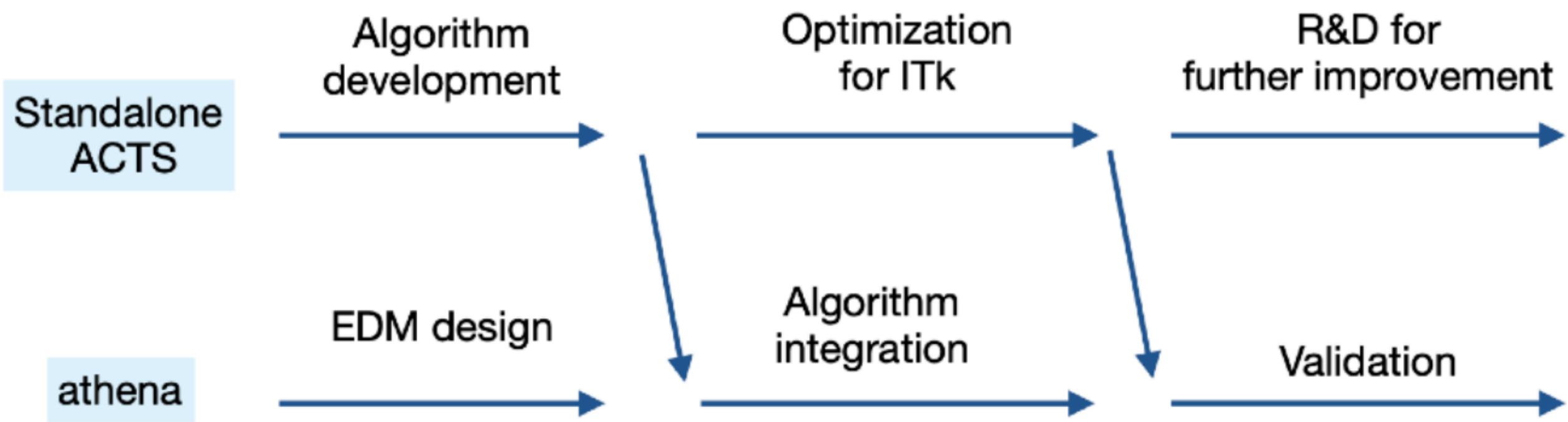
Full contents

physmon summary

- ✅ Particles fatras
- ✅ Particles geant4
- ✅ Particles ttbar
- ✅ Vertices ttbar
- ✅ Truth tracking (KF)
- ✅ Truth tracking (GSF)
- ✅ Truth tracking (GX2F)

Athena integration

- ACTS used a **sandbox** for R&D:
  - Used to test and develop new algorithms
  - Some Athena algorithms are also being reimplemented in ACTS
  - Periodically release to Athena
- A lot of work is necessary to connect the ACTS Event Data Model (EDM) to the ATLAS one (for downstream users)



Athena integration

- Conversion between the ACTS's EDM and ATHENA's EDM implemented
- Some Athena-specific concepts still missing, but work is ongoing

Internal ACTS EDM

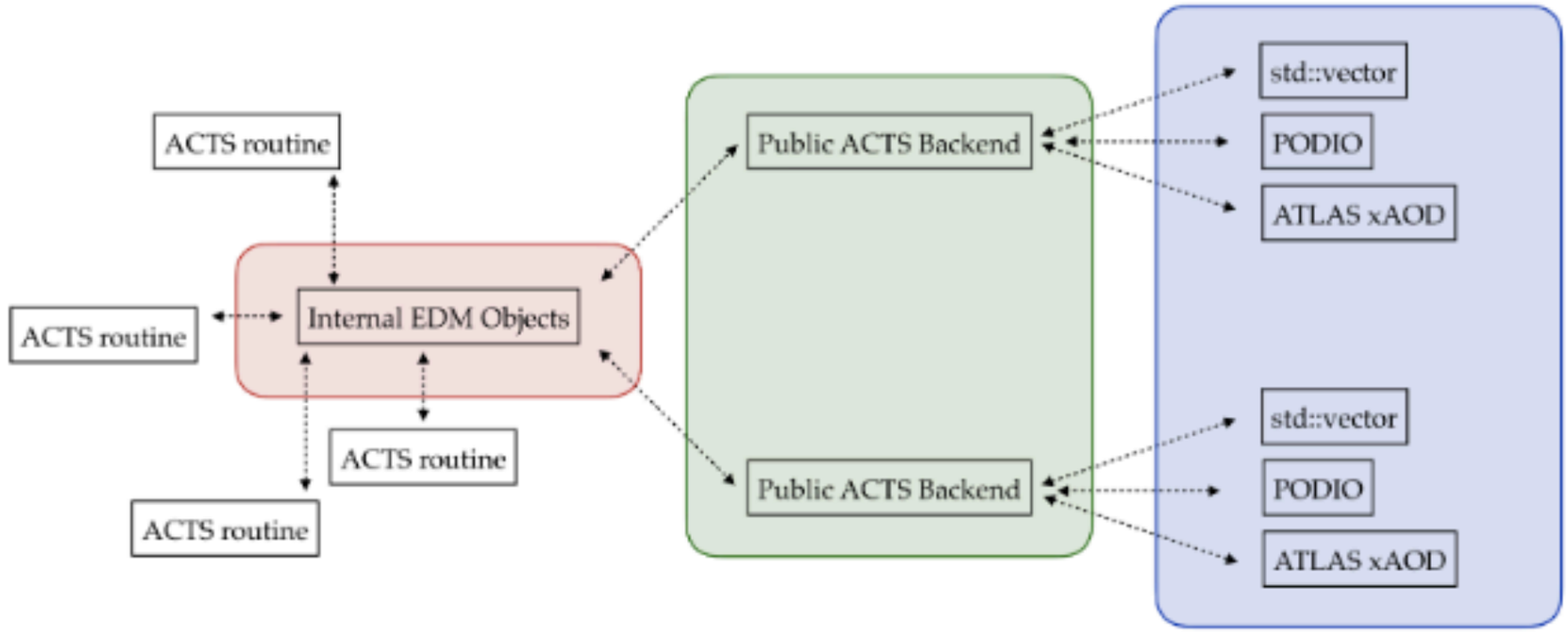
- Data objects to pass around between different parts of the ACTS library
- Library-specific, tightly coupled to the algorithm

Public ACTS EDM

- Data objects clients directly interact with
- Should be experiment agnostic
- Extensible by experiment, easy integration

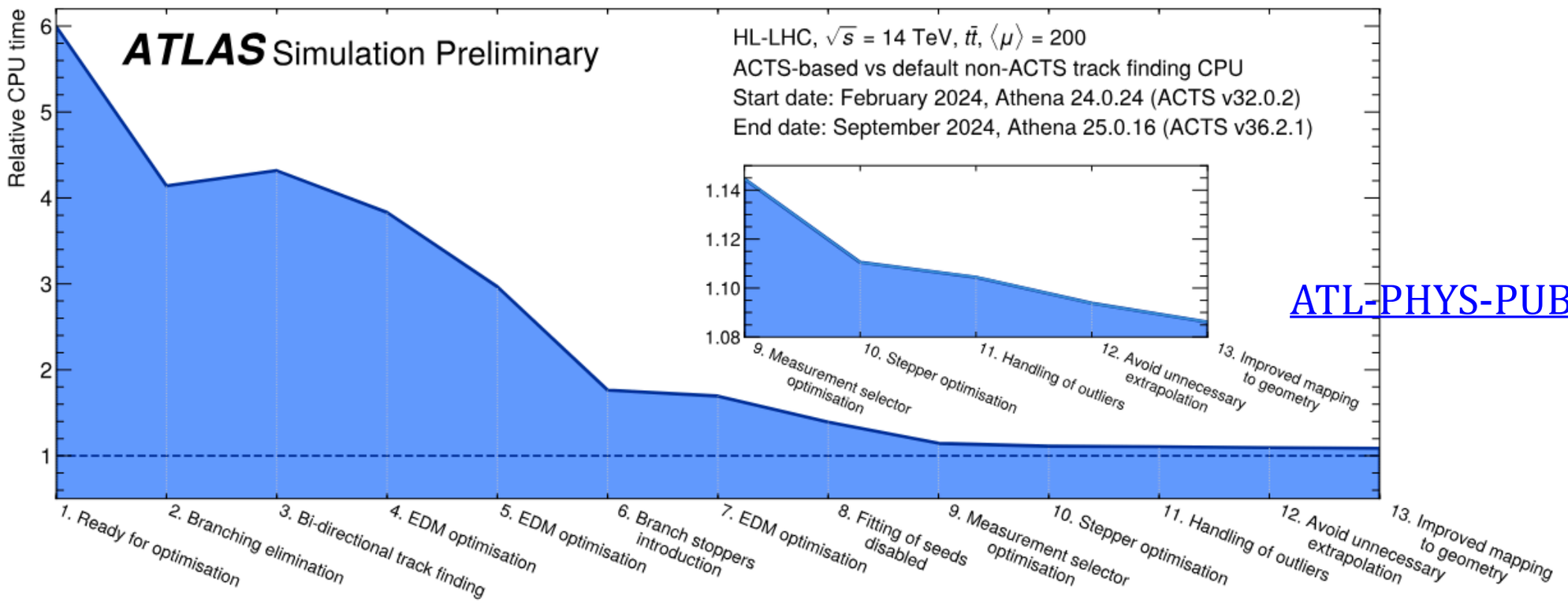
Experiment ACTS EDM

- Data objects experiments use, can save to file



Athena integration

- The entire ACTS **tracking chain** is now available in Athena
- Performance compatible with the Athena one
- A lot of work has been performed to **match the speed** Athena speed (currently 8% slower)



ATL-PHYS-PUB-2024-017

Conclusion

- ACTS: charge particle **tracking toolbox**
- Implement all the needed components for tracking
- Offer an excellent **testing environment** for new tracking algorithms

- Implemented by many **current and future experiments**
- Full integration in **ATLAS** is ongoing

Backup