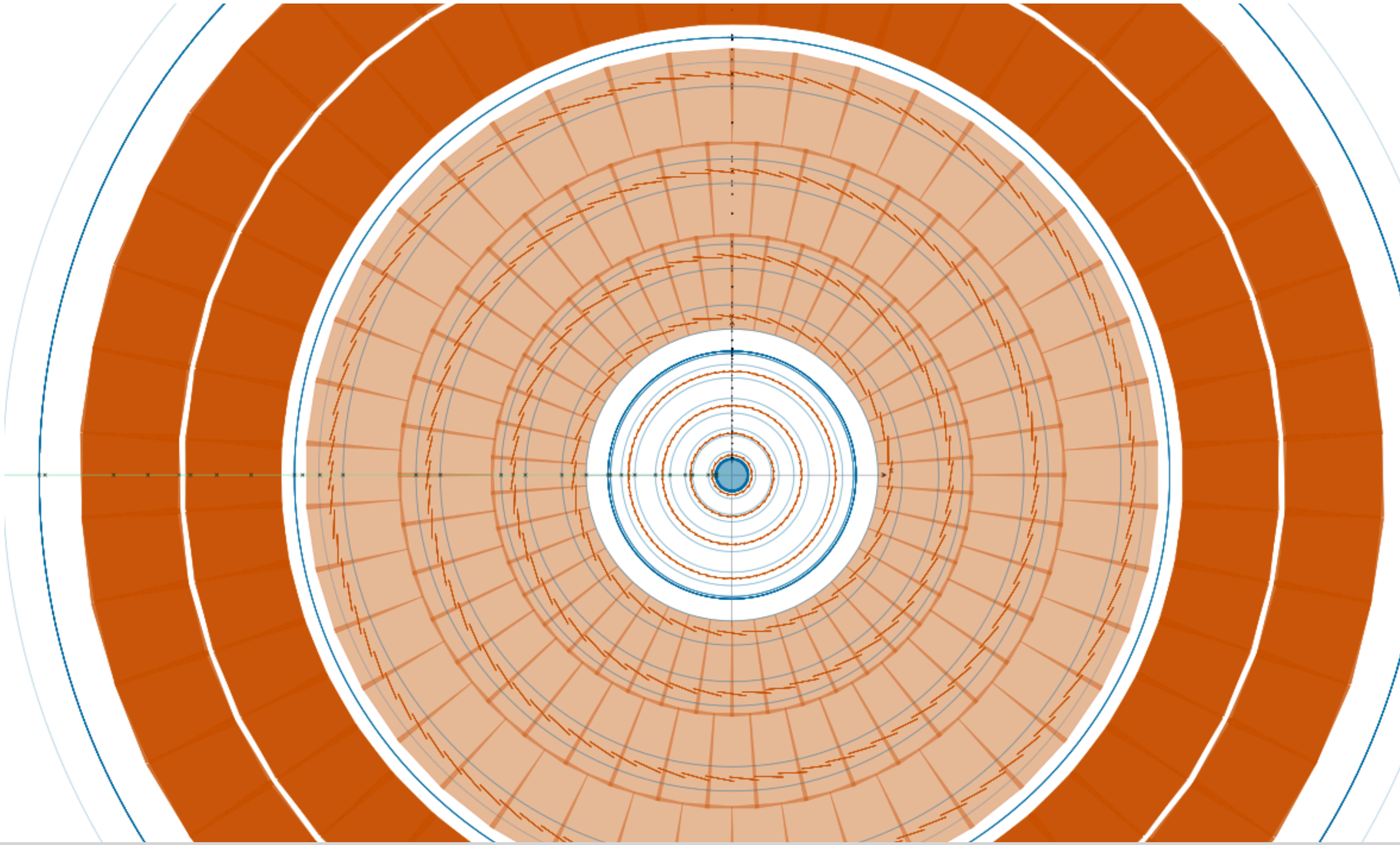


ACTS project: Status and R&D

A. Salzburger (CERN)

From where we come

- The acts project was started roughly 6 years ago

- Project Mission Statement:
  - Creating a toolbox of re-usable tools for experiments
    - detector agnostic top level tools
    - specification possibility for dedicated detectors/experiment
    - component library design

- Facilitate algorithmic and technology research
  - Allow easy extensibility
  - ML / Accelerator integration (two R&D lines on acceleration / ML algorithms)

Where we stand now

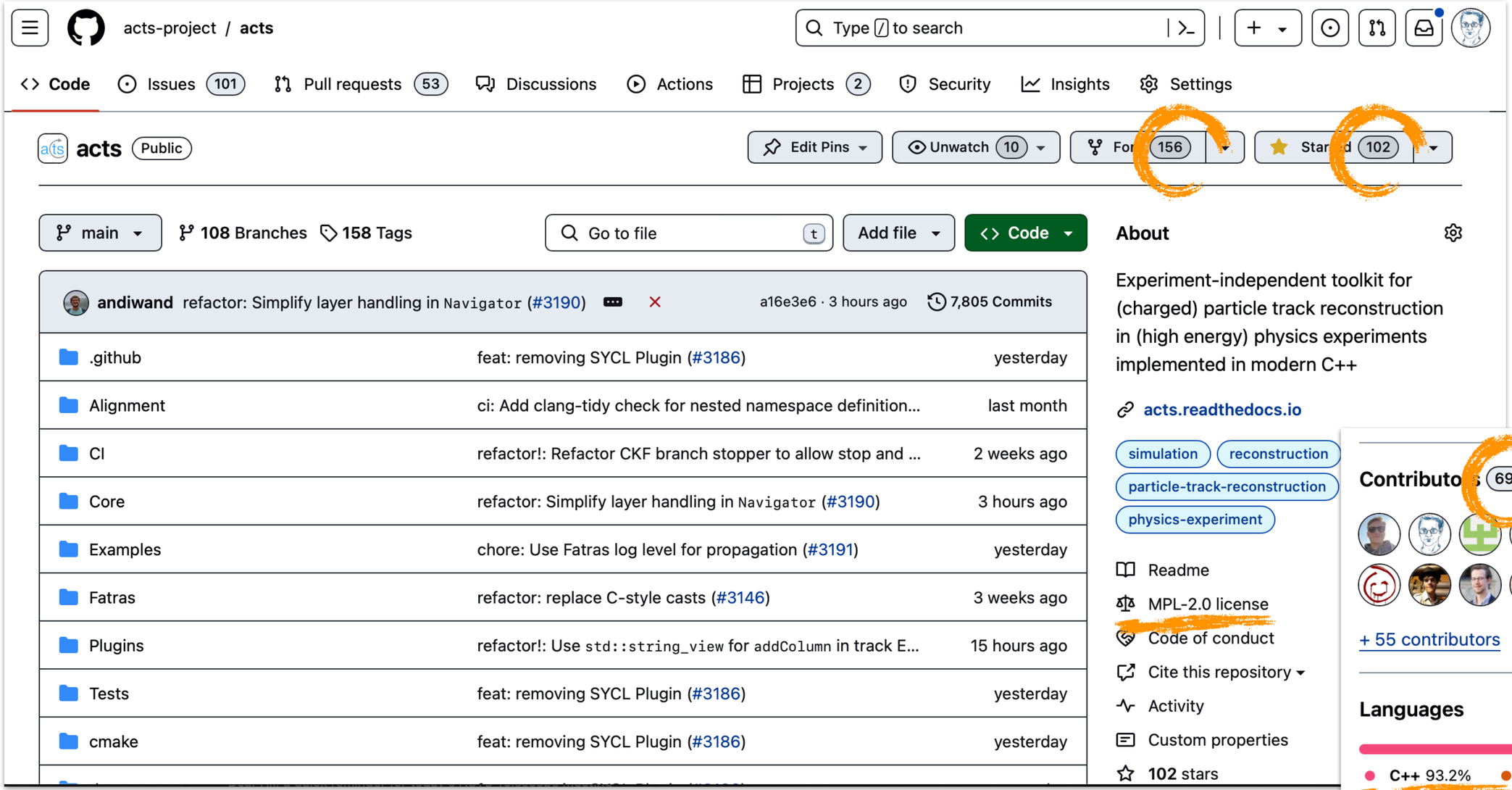
- Establish a feature rich toolbox
  - C++17 standard (preparing move to C++20)
  - Minimal dependencies (CMake, Eigen)
  - Plugins to enhance functionality
  - Enables parallel processing

- The acts project has an increased several base, e.g.

acts-telescope@cern.ch

LUXE Lohengrin

Where we stand now



Core concepts: multi threading and contextuatlity

Built-in parallelisation support

```
namespace Acts {
  class Module {
    /// @param gctx the geometry context (e.g. alignment)
    /// @param input the input data
    OutputData geometricOperation(const GeometryContext& gctx,const InputData& input) const;

  };
}
```


Allows parallel execution of this operation (without explicit technology binding, such as `tbb`) within and across events, nested **`State`** structs are used for necessary caching operations

```
namespace Acts {
  class Module {
    /// Nested State struct
    struct State { ActsScalar cache = 0.;};
    /// @param state is a cache for this operation
    /// @param input the input data
    OutputData operationWithCache(State& state, const InputData& input) const;

  };
}
```

Core concepts: multi threading and contextuatlity

Built-in parallelisation support and contextuality

```
namespace Acts {
   /// @param gctx the geometry context (e.g. alignment)
   /// @param input the input data
   OutputData geometricOperation(const GeometryContext& gctx,const InputData& input) const;

  };
}
```

```
using GeometryContext = std::any;
```

ACTS allows you to pack your own contextual data into the context objects (geometry, magnetic, field) and will carry it through the code base (untouched)

```
auto Experiment::applyCorrection(const GeometryContext& gctx, const InputData& input) const {

   const Experiment::Payload& payload = std::any_cast<const Experiment::Payload&>(gctx);
}
```

Core concepts: data driven, configuration & options

Design convention for data driven design, configuration and option

```
namespace Acts {
  /// doxygen documentation
  class Module {
   /// @struct Config for this module,
    struct Config {
       ActsScalar globalParameter; ///< configure this module
   };

   /// @struct Options for this module, changeable on call
    struct Options {
       ActsScalar callParameter; ///< how the horse feels today
   };

   /// @param cfg the configuration struct for this module
   Module(const Config& cfg) : m_config(cfg){};

   /// @param input the input data
   OutputData operation(const InputData& input, const Options& opt) const;

  };
}
```

Core concepts: configuration binding

Simple Config structs on ACTS side

```
namespace Acts {
  /// doxygen documentation
  class WorkHorse {
   /// @struct Config for To
    struct Config {
       ActsScalar coatColor; ///< configure the coat color
       ActsScalar maxPath;    ///< set the max path this horse can run
    };
  };
}
```

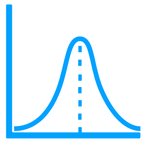
Connection to experiment framework, e.g. Gaudi/Athena

```
/// feed from Framework into ACTS configuration
declareProperty("CoatColor", m_cfg.coatColor);
declareProperty("MaxPath",   m_cfg.maxPath);
```

Toolbox: track reconstruction building blocks

Geometry & Material

Event Data Model



Track Finding & Fitters

Vertexing

R&D lines (ML, GPU)

Geometry - Concepts

- ACTS creates a reconstruction view of the detailed geometry
  - Plugin mechanism ensures compatibility with many geometry sources
  - Context mechanism ensures MT ready contextual geometry

Detailed geometry model,
e.g. DD4hep, TGeo, GeoModel, etc.

ACTS geometry model
with builtin navigation

Geometry R&D (1)

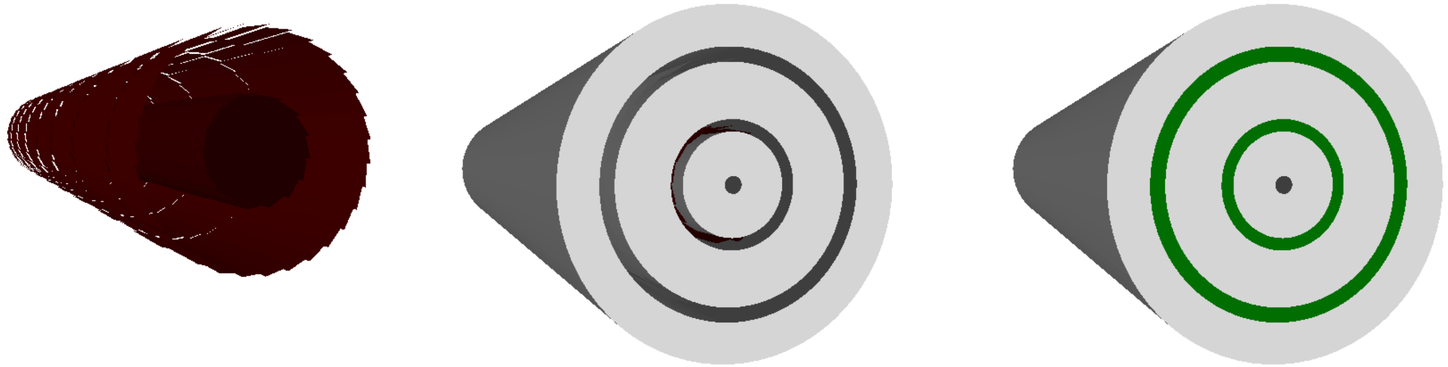
- Geometry model of ACTS stems from ATLAS `Trk::TrackingGeometry`
  - Conceptual building blocks

`TrackingVolume`

`Layer`

`Surface`

Quite some overlap between those



  - `detray` GPU R&D geometry: re-implemented w/o layer concept
    - huge simplification in navigation code
    - can we do this also for ACTS/Core ?

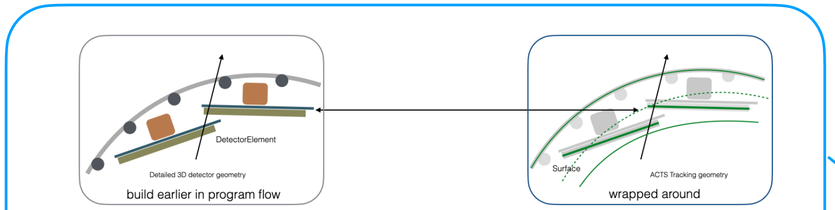
Geometry R&D (2) - Experimental

- `Experimental::Detector` Geometry model of ACTS

Gen1 geometry type	Gen2 geometry type	
`Acts::Surface`	`Acts::Surface`	Surface objects are unchanged, allows client code to be untouched
`Acts::Layer`		Layer objects do not exist anymore, they are represented by volumes
`Acts::TrackingVolume`	`Acts::Experimental::DetectorVolume`	Double serving of volumes as containers or navigation volumes omitted
`Acts::BoundarySurfaceT<Acts::TrackingVolume>`	`Acts::Experimental::Portal`	Portal objects are not templated anymore, they are holder classes of surfaces and volume switches
`Acts::TrackingGeometry`	`Acts::Experimental::Detector`	Portal objects the top level entry point that will guide into the root volumes

Geometry R&D (3) - Blueprint

- New type of geometry building using `Experimental::Blueprint`



Detailed geometry model,
e.g. DD4hep, TGeo, GeoModel, etc.

ACTS geometry model
with built-in navigation

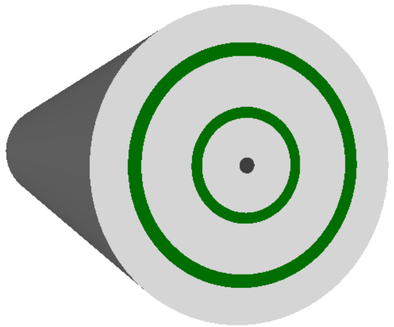
Translation of objects from geometry model,
e.g. DD4hep

from one source, but not necessarily

Logic of how to build/group
e.g. DD4hep

Detector Blueprint

Instruction set how to build the detector



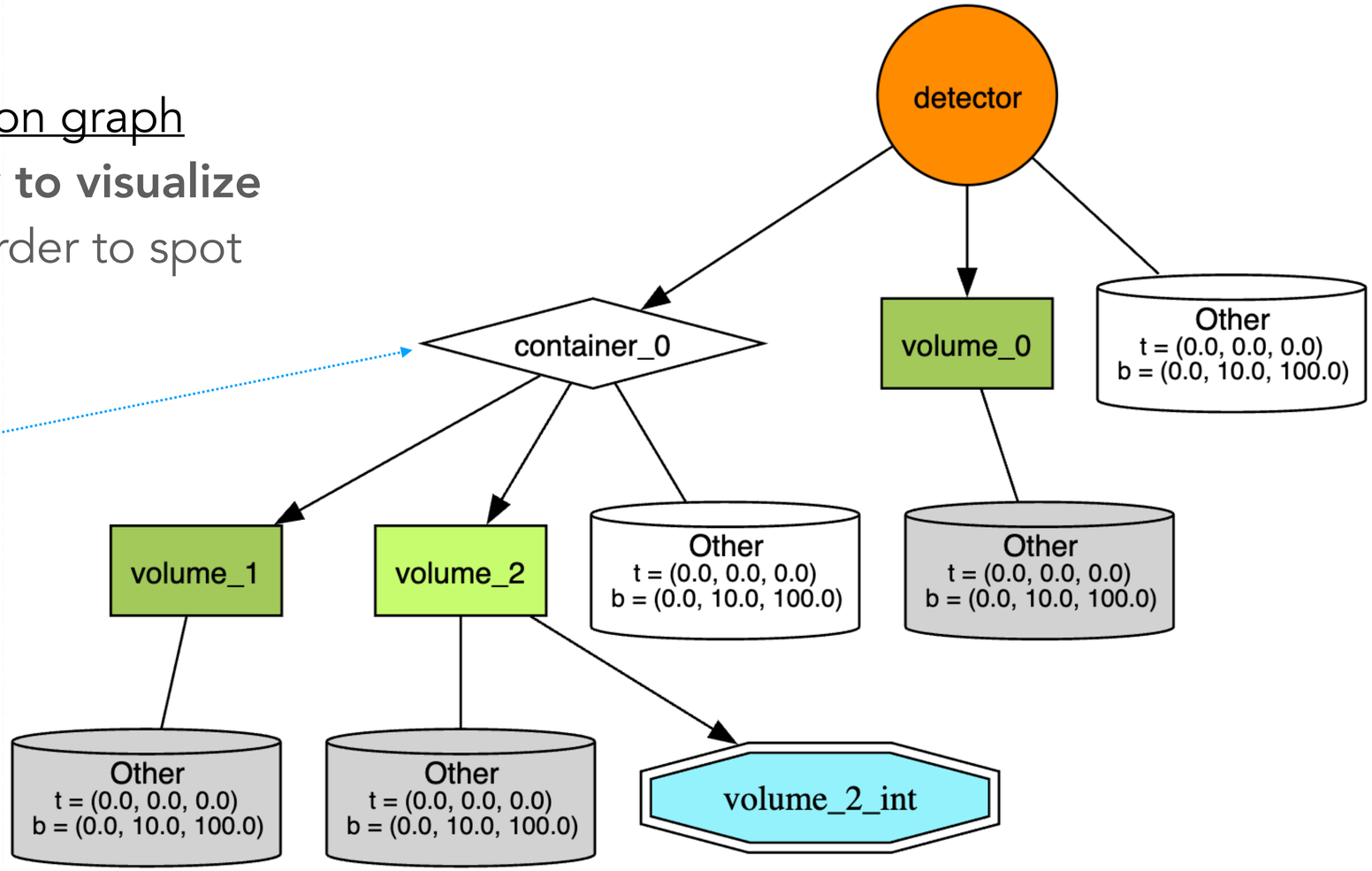
Geometry R&D (4) - Blueprint

- New type of geometry building using `Experimental::Blueprint`

Blueprint is an instruction graph

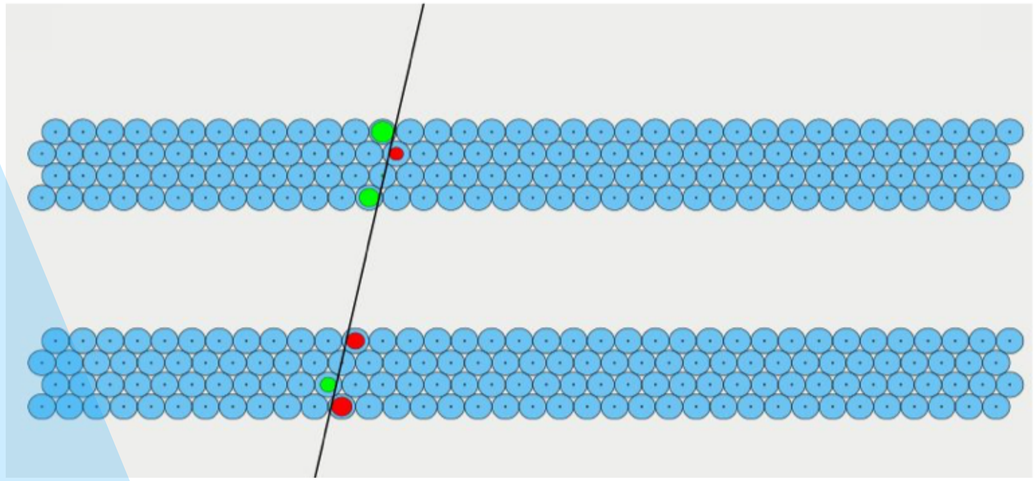
- **Added functionality to visualize** before building, in order to spot problems

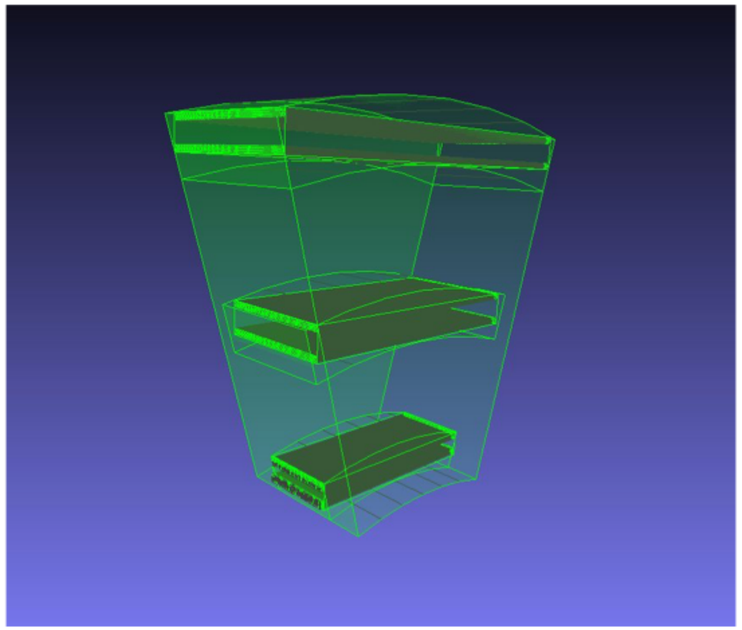
non-coloured nodes
are virtual containers



Geometry R&D (5) - drift detectors

- In Gen2 geometry, navigation is outsourced to `Delegates`
  - allows for client-specified navigation
  - helped developing first prototypes for (ATLAS) Muon System

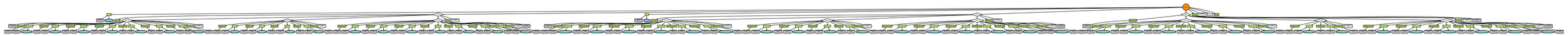




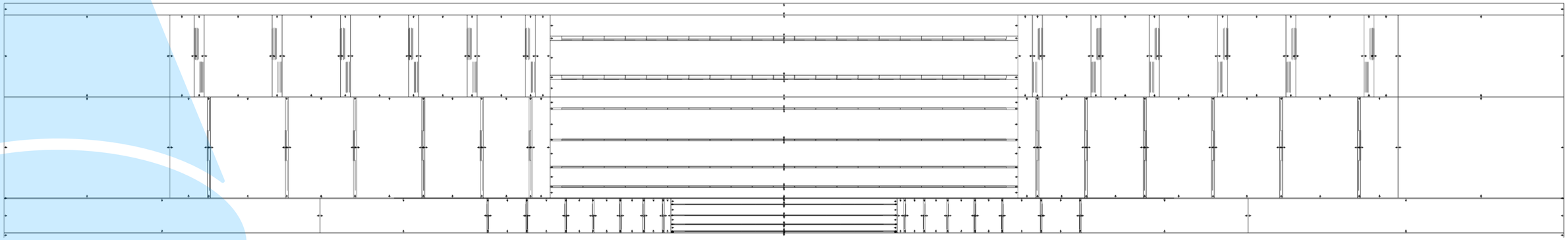
Mock up muon sector spectrometer .
Every detector volume holds the navigation delegate

Geometry R&D (5) - Blueprint on ODD

ODD building blueprint from DD4hep:



Resulting ODD detector



Geometry (6) - Quo vadis ?

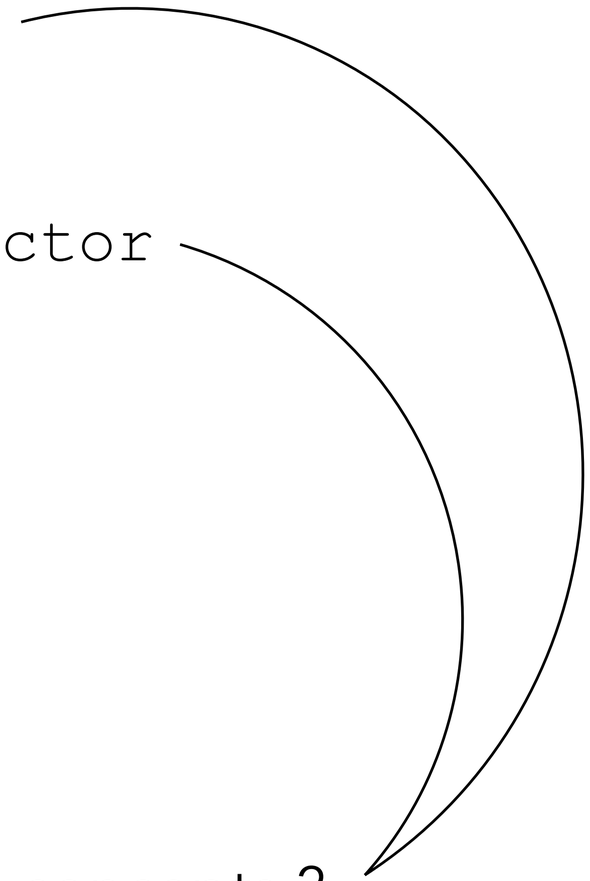
- **Gen1** geometry: `Acts::TrackingGeometry`

  `Well established, baseline`

- **Gen2** geometry: `Acts::Experimental::Detector`

  `Blueprint`
  `Layer-less`

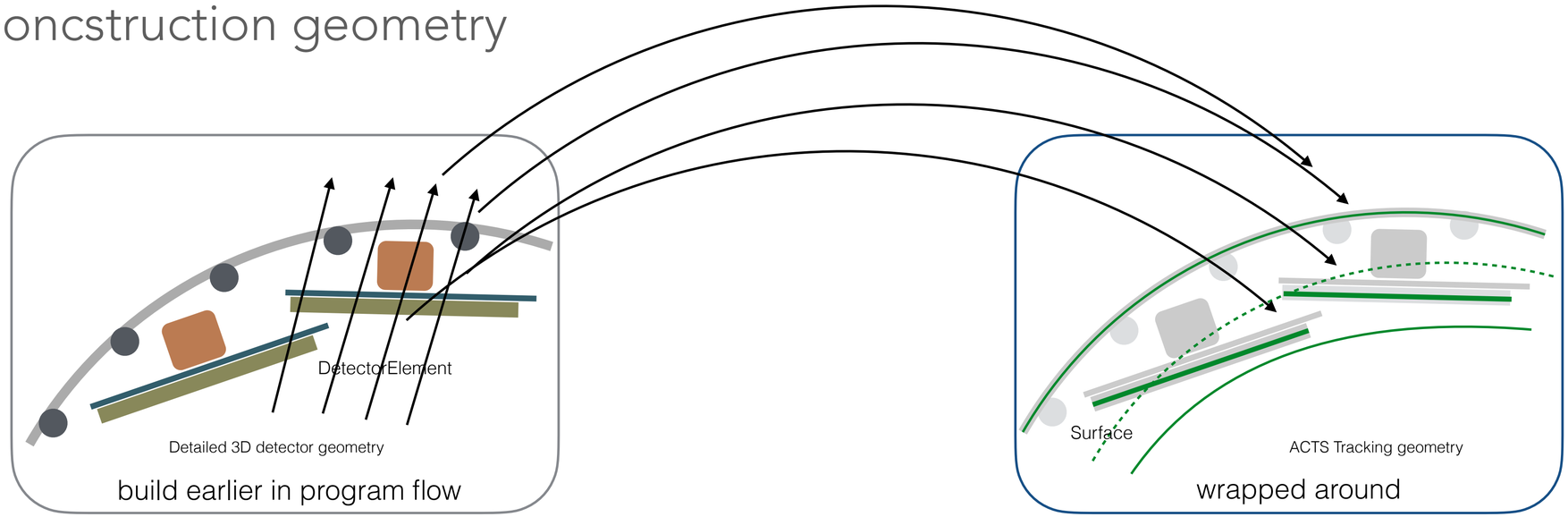
  `Navigation delegates`



- **Gen3** geometry:
  - adiabatic merge of those two concepts ?
  - morph of Gen2 into full functionality of Gen1 ?

Material

- ACTS ships with a material mapping module
  - allows to transcribe the full Geant4 geometry and map it onto the simplified reoncstruction geometry

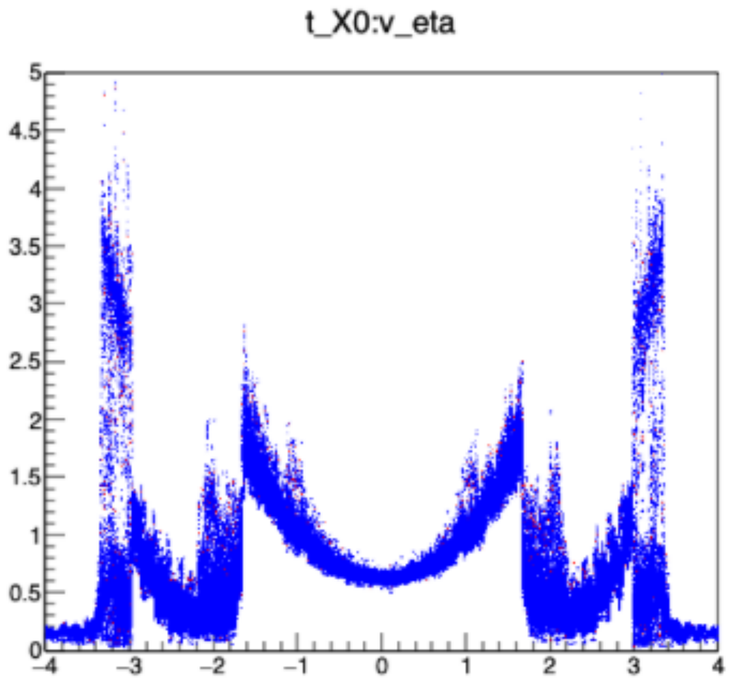


Detailed geometry model,
e.g. DD4hep, TGeo, GeoModel, etc.

ACTS geometry model
with builtin navigation

Material R&D

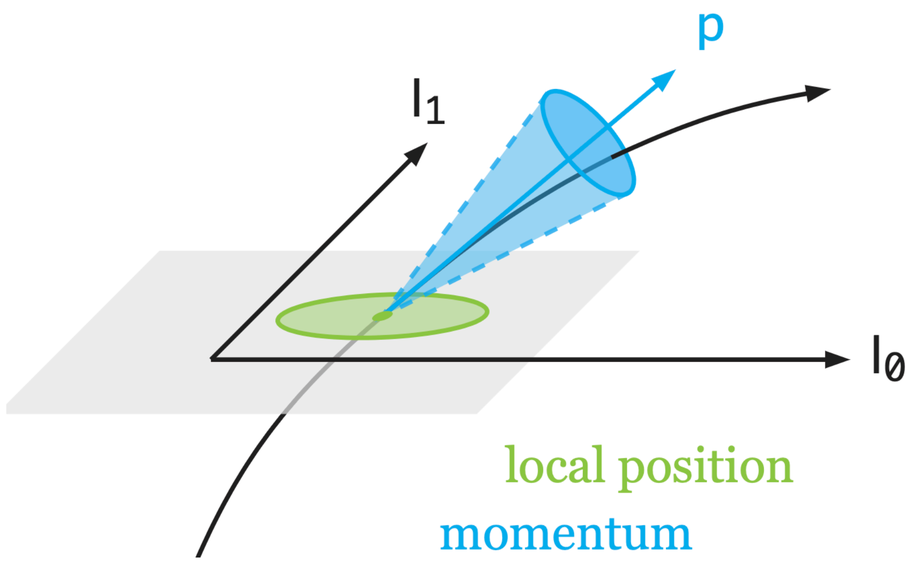
- New Grid based material classes introduced
  - Including a k-means compression algorithm
- Material mapping/validation without & with propagation/navigation
  - This is to allow for material mapping/ validation with optionally bypassing the propagator infrastructure
  - Support for Gen1/Gen2 geometry model
- Move most material mapping/validation into Core
  - Allow for more seemingness integration into SW stack

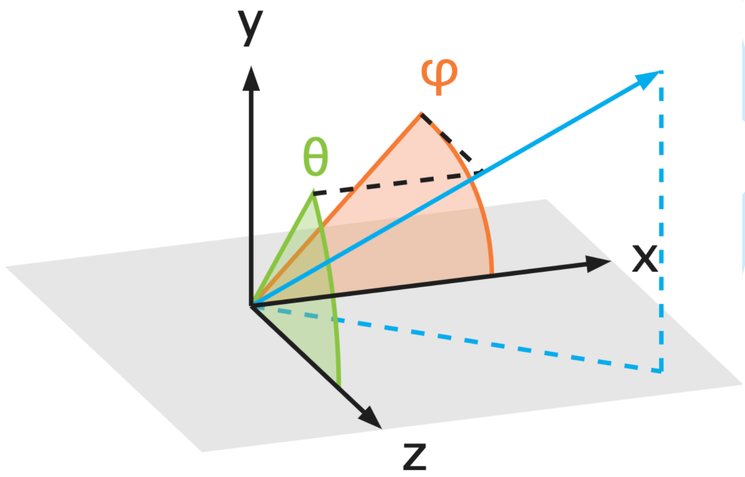


Event Data Model (1)

(Bound) track parameterisation is defined:

local coordinates of the surface + global momentum





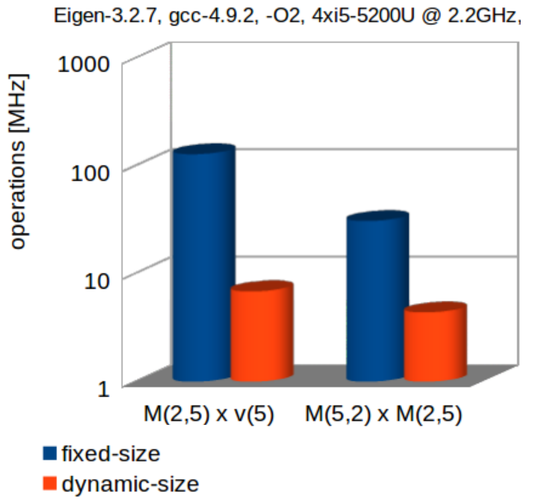
$$\vec{x} = (l_0, l_1, \phi, \theta, q/p, t)^T$$

Event Data Model (2)

Parameter	l_0	l_1	phi	theta	q/p	t
Bound track parameters						
Pixel measurement						
Pixel measurement with time						
Strip measurement (along local x)						
Strip measurement (along local y)						
Drift time/circle measurement						
Track segment (straight line)						
…						

Measurements can be represented as subsets of the full bound parameter space.

This is done at compile time to increase computing performance.

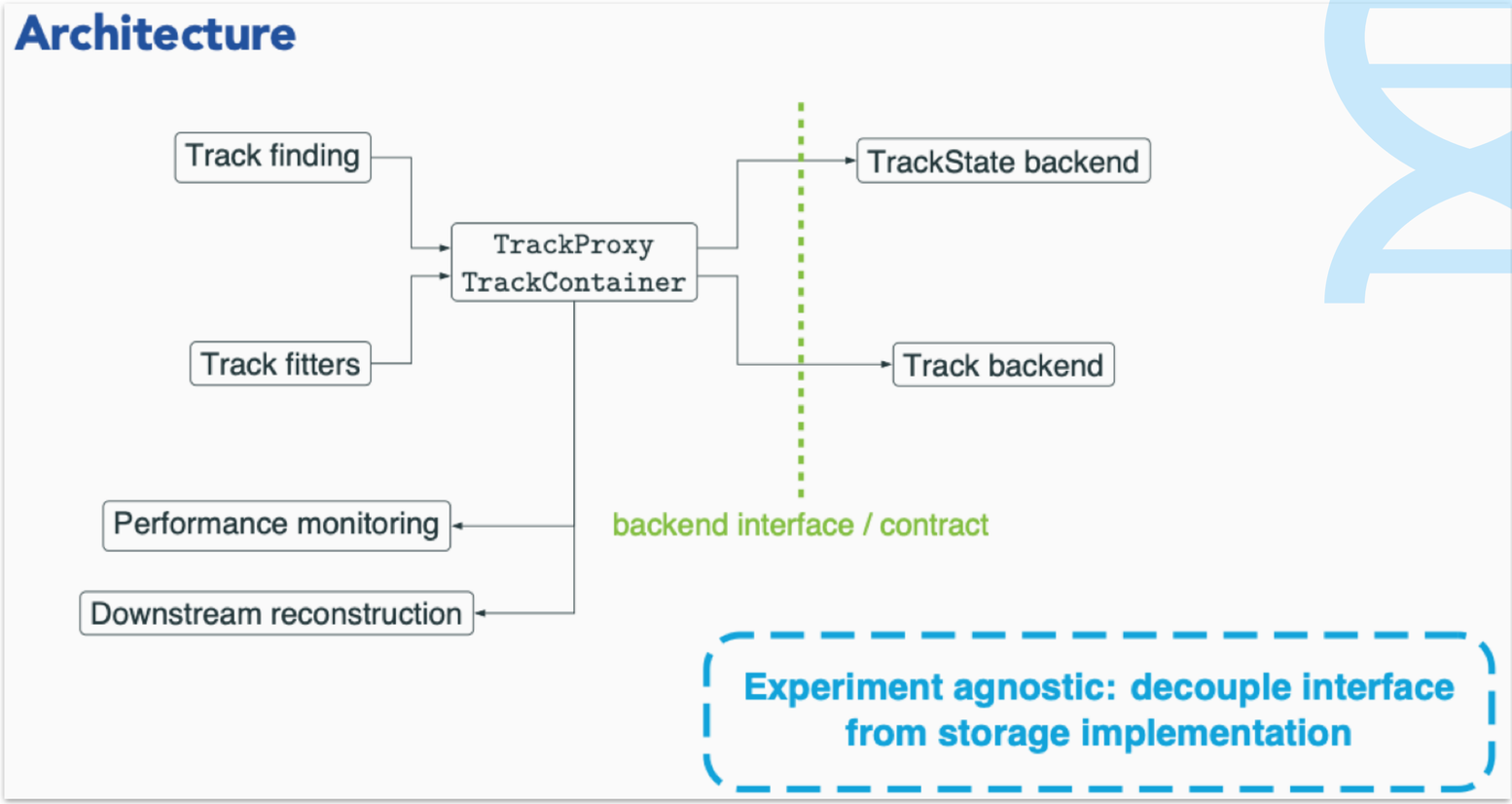


Event Data Model (3)

- MultiTrajectory with frontend/backend split

ACTS has an internal EDM optimised for track reconstruction.

- recent work to separate transient model from I/O backend
- demonstrator with PODIO established
- Non-optimised EDM4Hep version also available

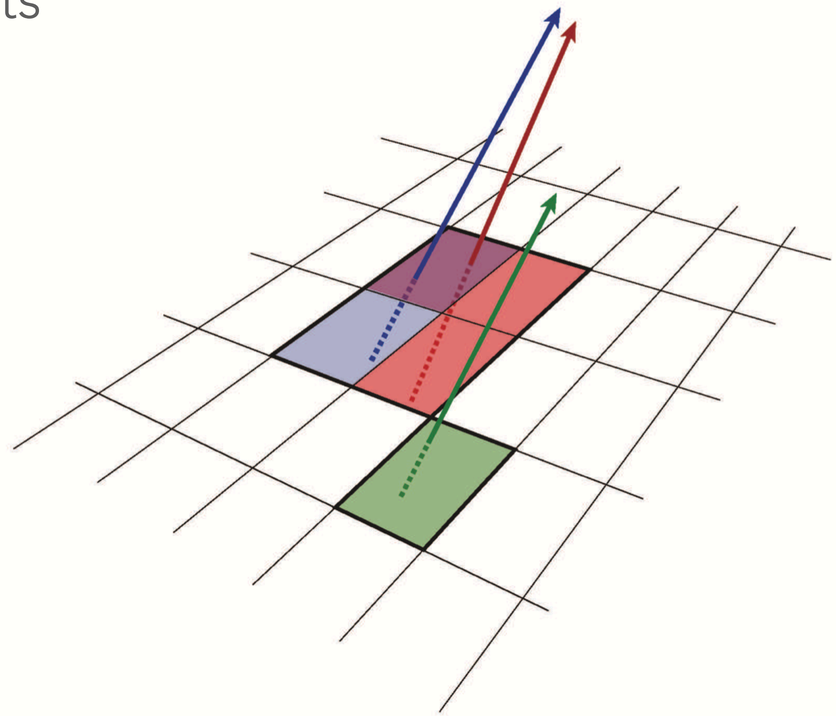


[Paul Gessinger-Befurt, CHEP2023, Parallel talk]

Fitters (1) - Kalman Filter

- Kalman Filter implementation very matured
  - Designed as a plugin into the Propagation engine
  - shows nice performance on Geant4 simulated results

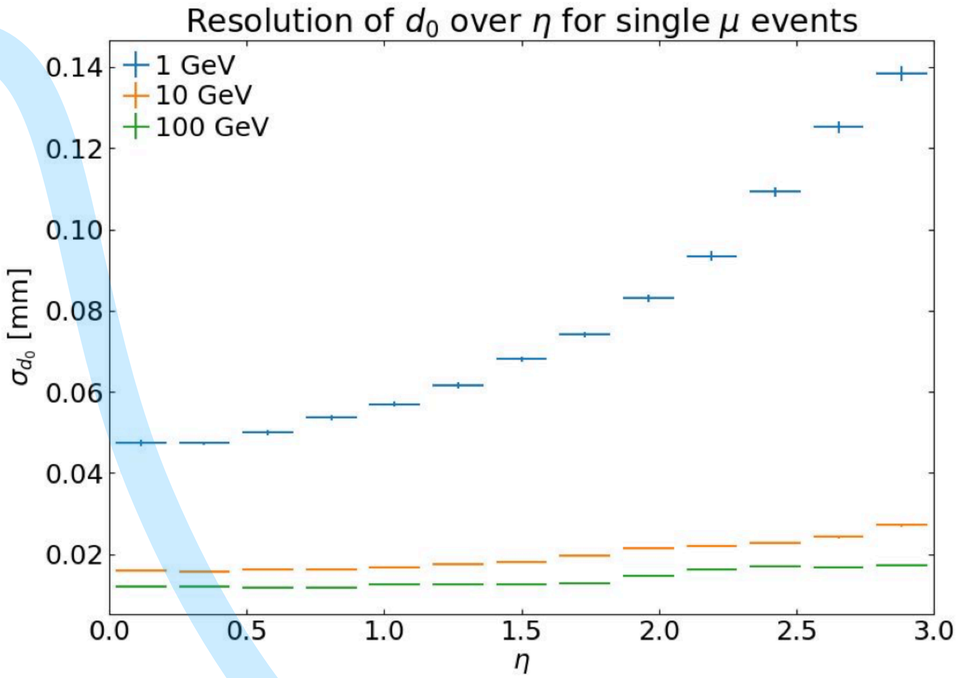
- Calibrator
  - Allows to do on the fly measurement creation
    - Helps to get ultimate resolution for a detector
    - Can help to resolve ambiguities
    - A way to start with a misaligned detector

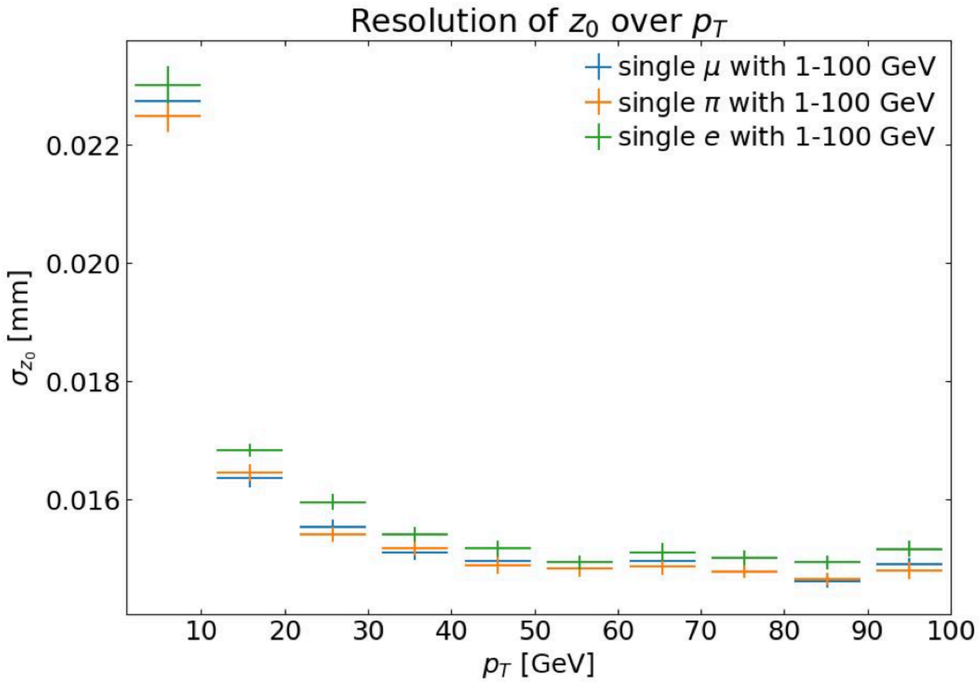


Example, tracking in dense environment

Kalman Filter (2)

- Extremely high level of accuracy control
  - Given by stringent mathematical validation
  - Detailed material description

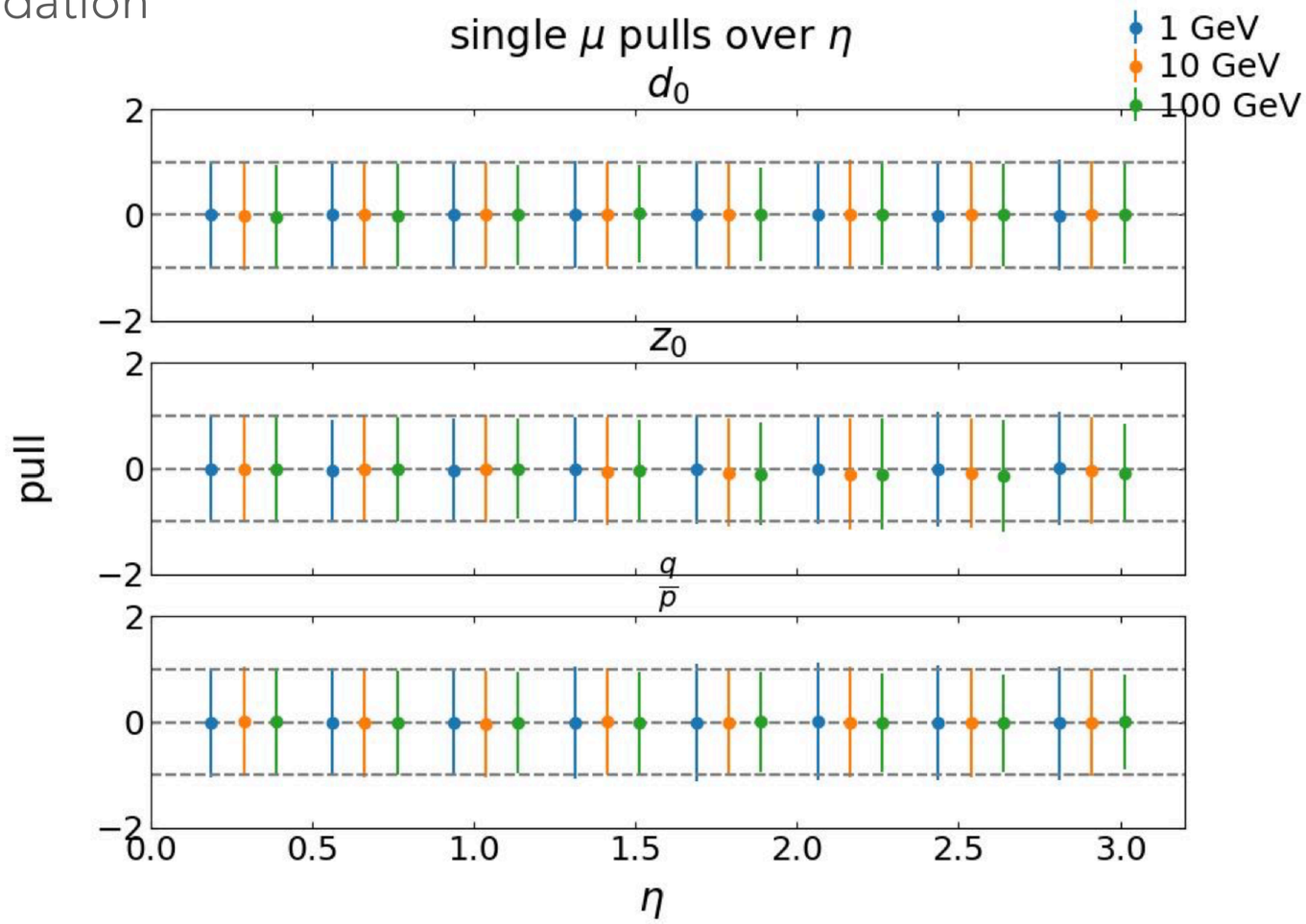




Kalman Filter (3)

- Extremely high level of accuracy control
  - Given by stringent mathematical validation
  - Detailed material description

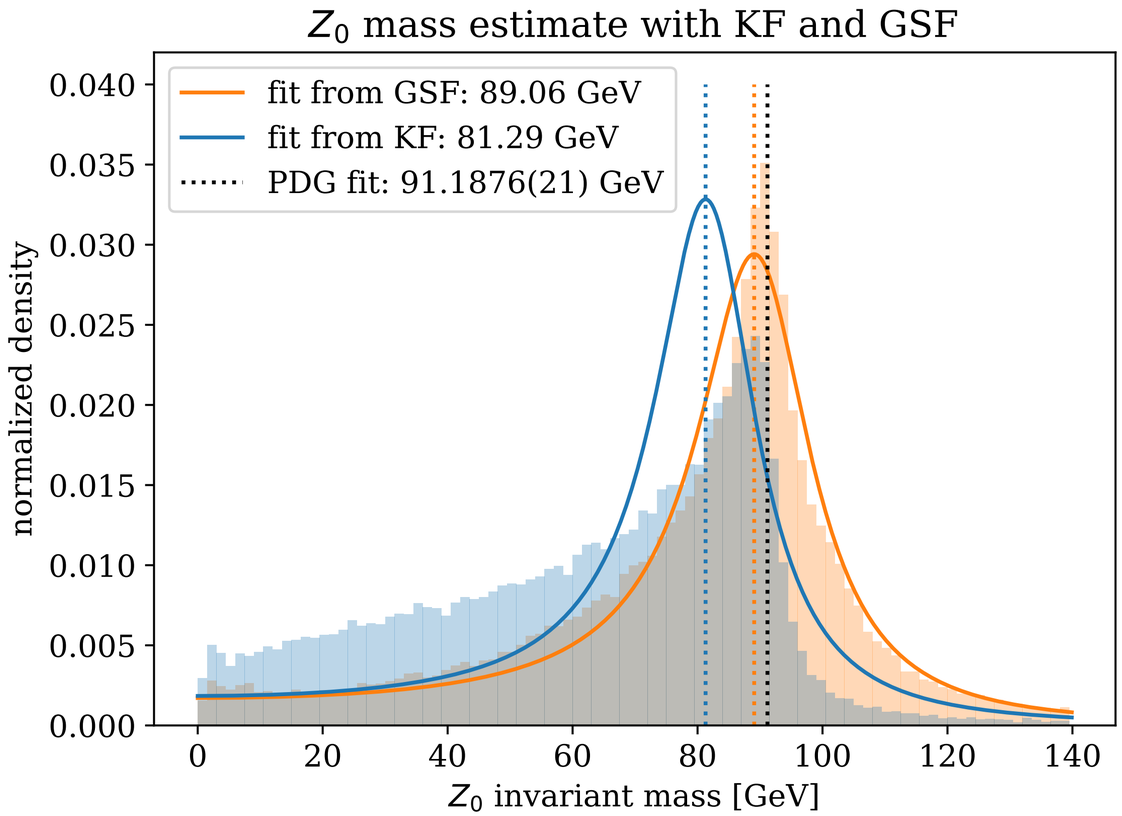
- Example on OpenDataDetector (Geant4 simulation)



Fitters (2) - Gaussian Sum Filter

- Gaussian Sum Filter has been validated on Open Data Detector
  - shows nice performance on Geant4 simulated results
  - Is designed as a re-fitter, i.e. after electron pattern recognition

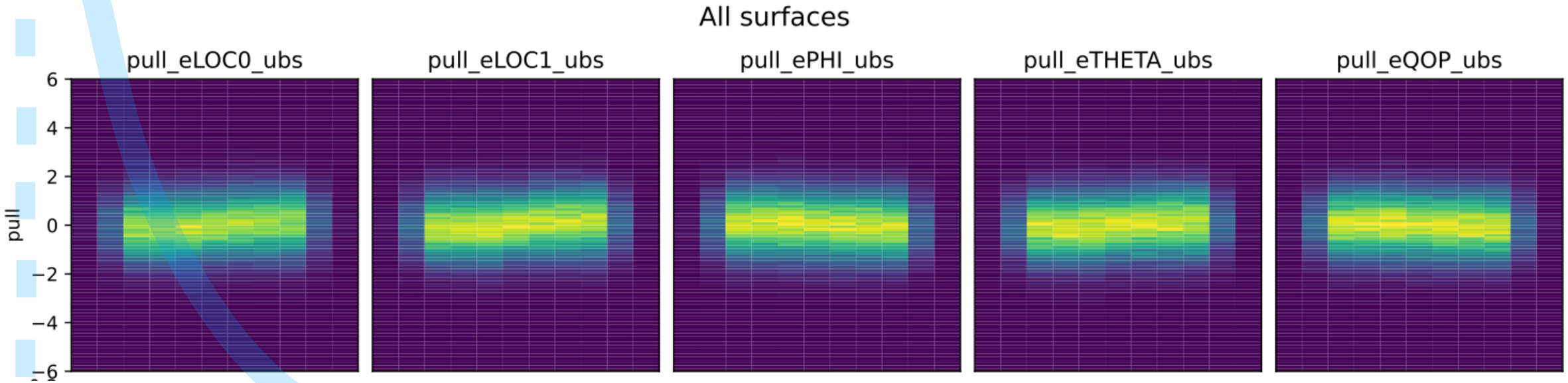
- Electron pattern recognition not yet implemented
  - start with concept from ATLAS to enlarge window if electron hypothesis is triggered …



Fitters (3) - Global Minimisation

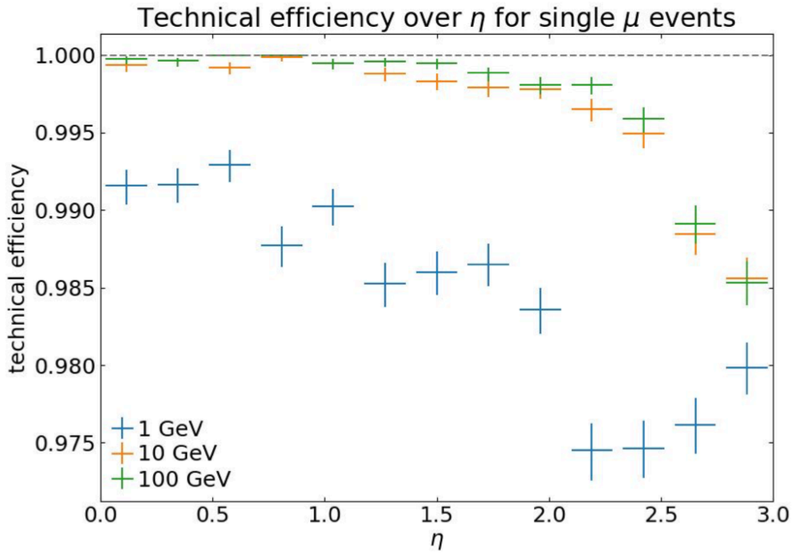
- Global chi2 fitter progress
  - First pipe-line on OpenDataDetector implemented

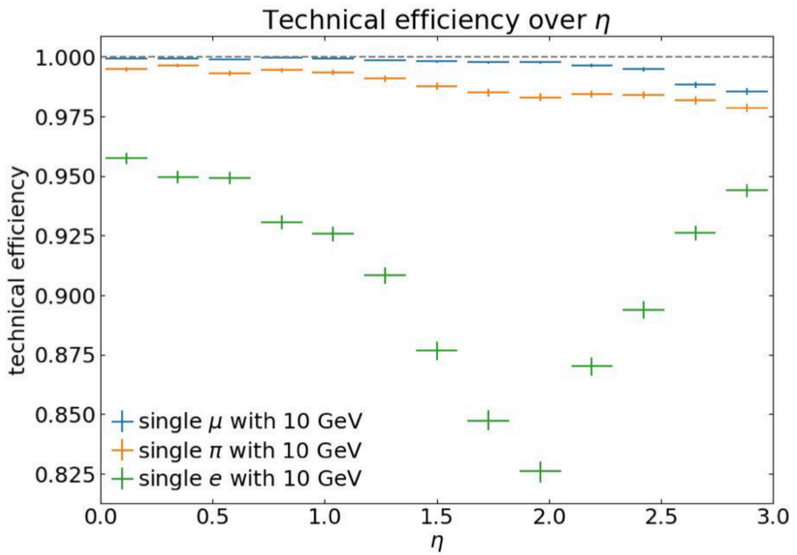
- Material effect integration not yet implemented
  - Exists in a python based prototype



Combinatorial Kalman Filter (1)

- Track finding implementation using a Combinatorial Kalman Filter (CKF)
  - Achieves almost perfect technical efficiency for muons
- Runs on top of different seeding strategies
  - Triplet seeding
  - Orthogonal seed finder
  - GNN
  - New seeding for telescope detectors

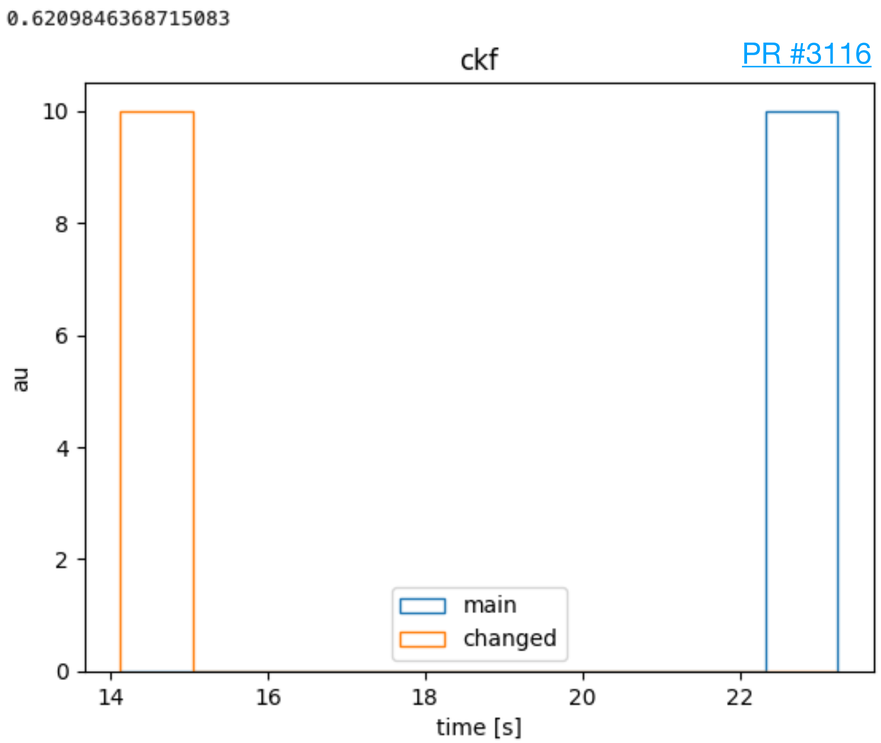




Combinatorial Kalman Filter (2)

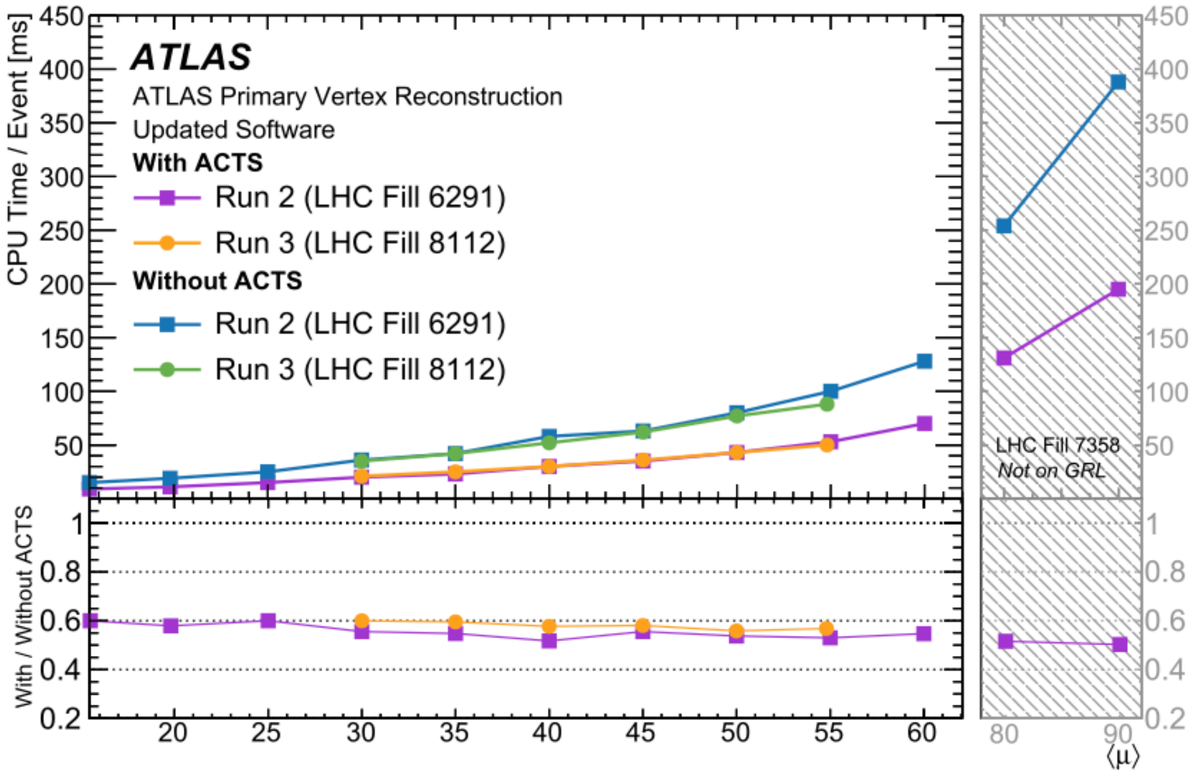
- Speed performance optimisation
  - Work on a new stepper has started (based on Symbolic math transcription)
- Combinatorial Kalman filter updates
  - Improved branch stopping logic introduced
  - Smoothing separated from forward filtering
  - New, alternative CKF with external propagator steering in development

ttbar @ <μ> ~200



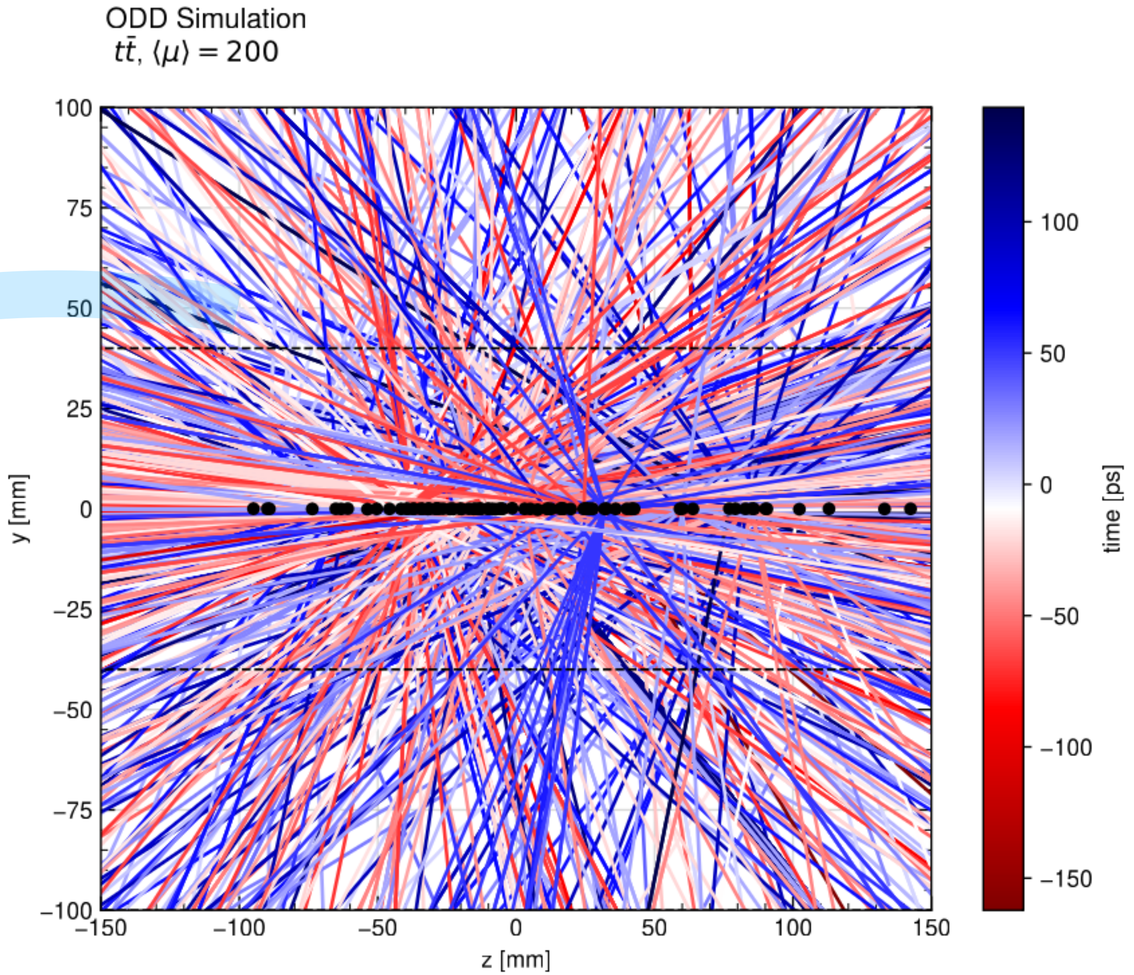
Vertex reconstruction

- ACTS implements a fully fledged primary vertex reconstruction suite
  - Iterative Finder + Billoir Fitter
  - MultiAdaptiveVertexFinder + Fitter
    - Optimised for very high track and vertex multiplicities (HL-LHC)
  - Was the first module to be deployed in ATLAS from ACTS
    - Huge speed update with identical results



Vertex reconstruction R&D: fully time-aware

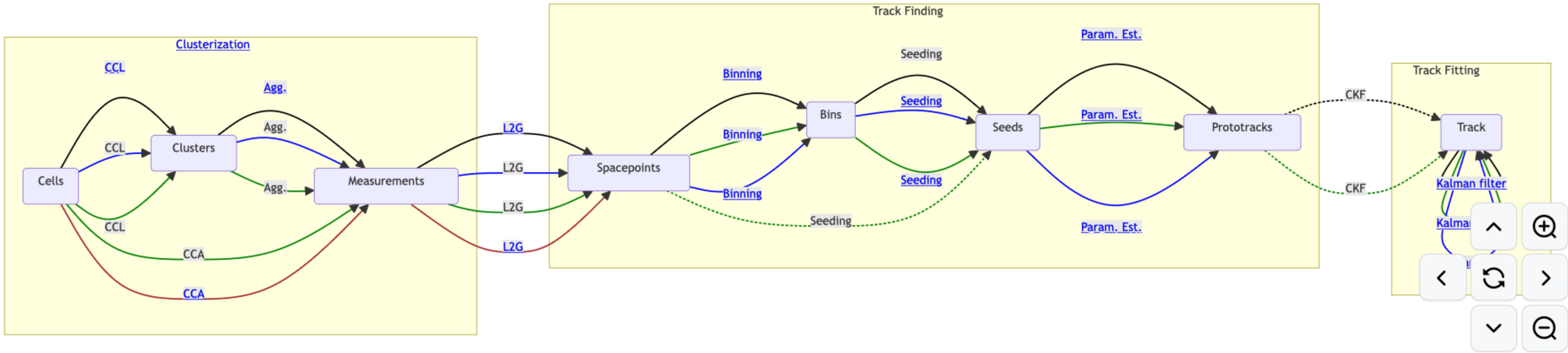
- Introduction of time in all components of vertex reconstruction
  - full exercise on OpenDataDetector in progress



R&D line: parallelisation

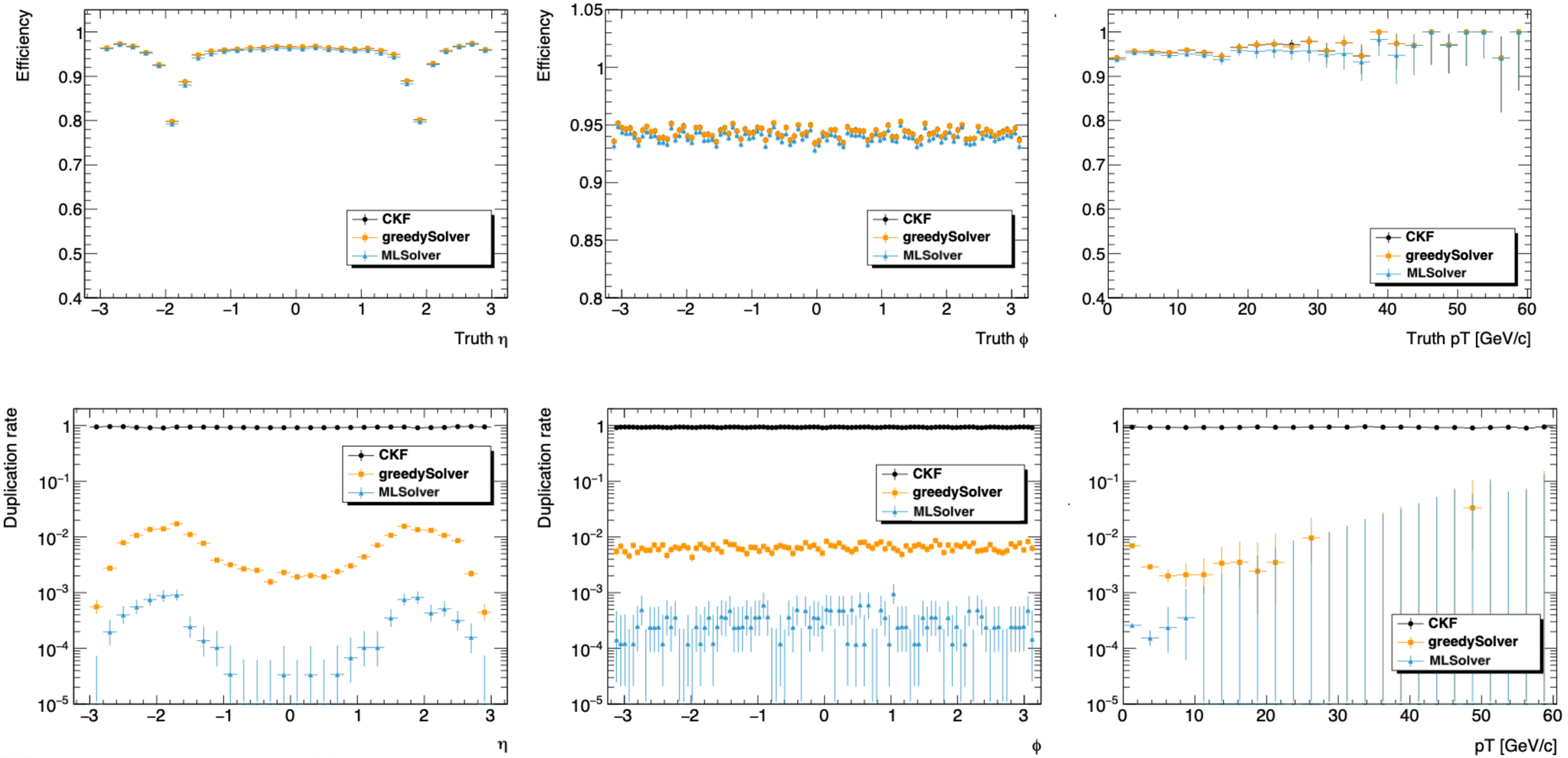
- First chain runs on OpenDataDetector in stand-alone
  - Performance (physics/computing) evaluation to start

- Integration of ‘traccc’ suite as Plugins started (talk by Beomki Yeo, tomorrow)
  - Aim is to be able to evoke a traccc reconstruction chain from ACTS



R&D line: machine learning

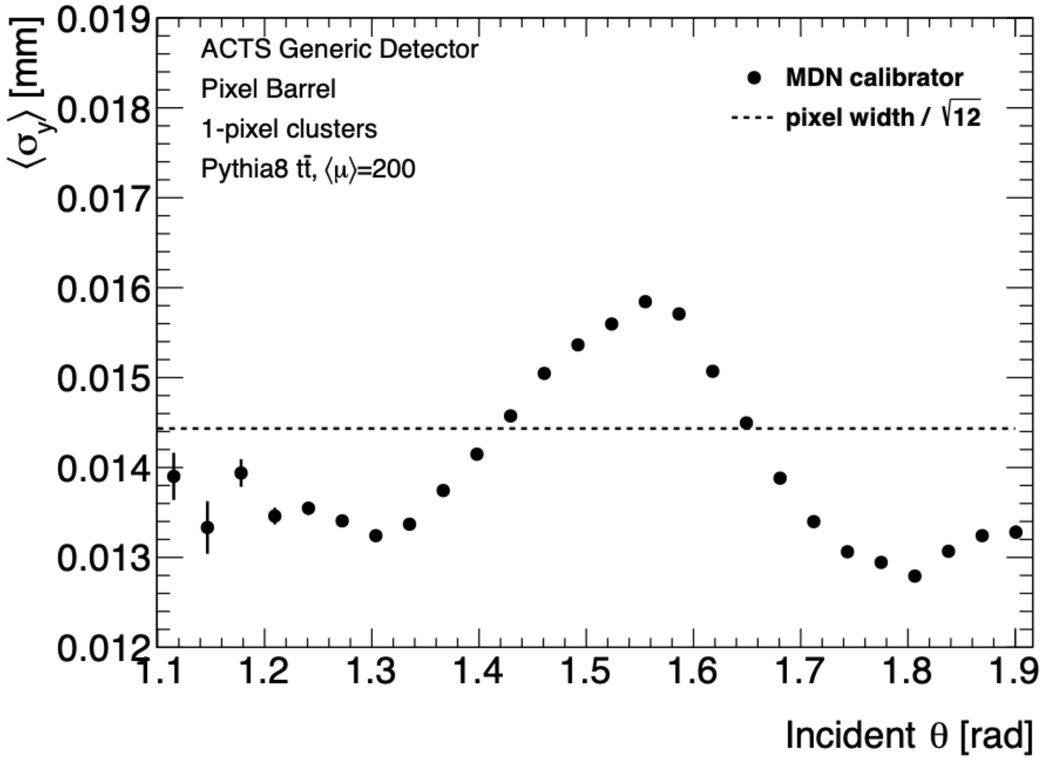
- ML based ambiguity solver

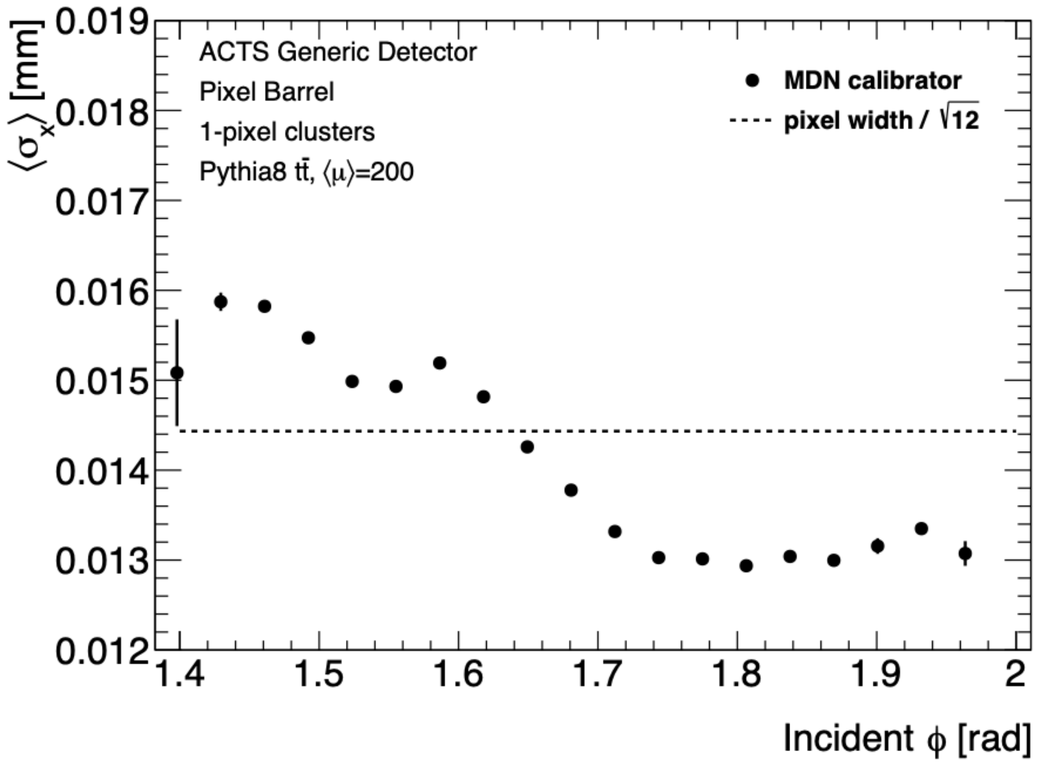


[C. Allaire, EF Tracking meeting, Jan 16, 2024]

R&D line: machine learning

- NN based cluster position / calibration
  - NN based clusterization available



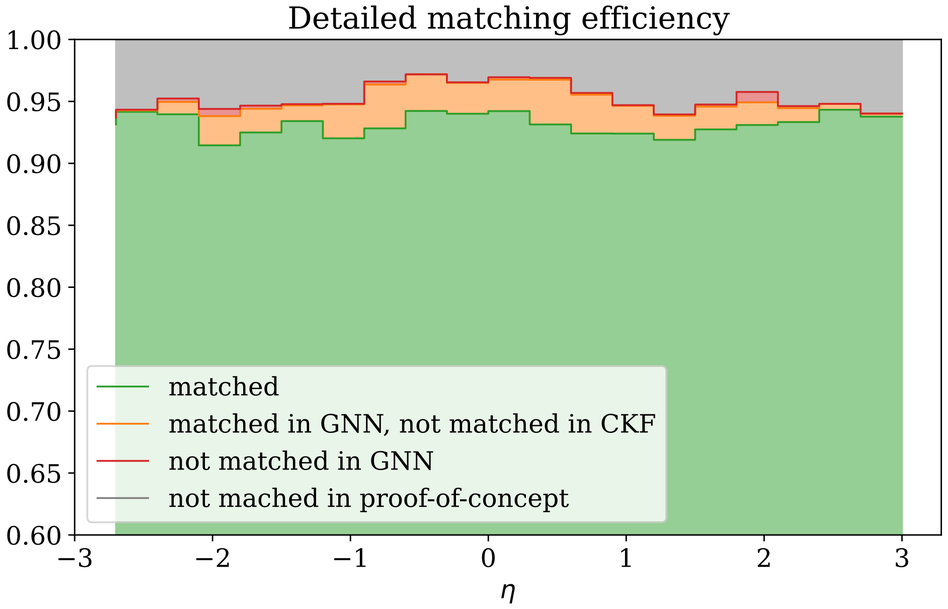


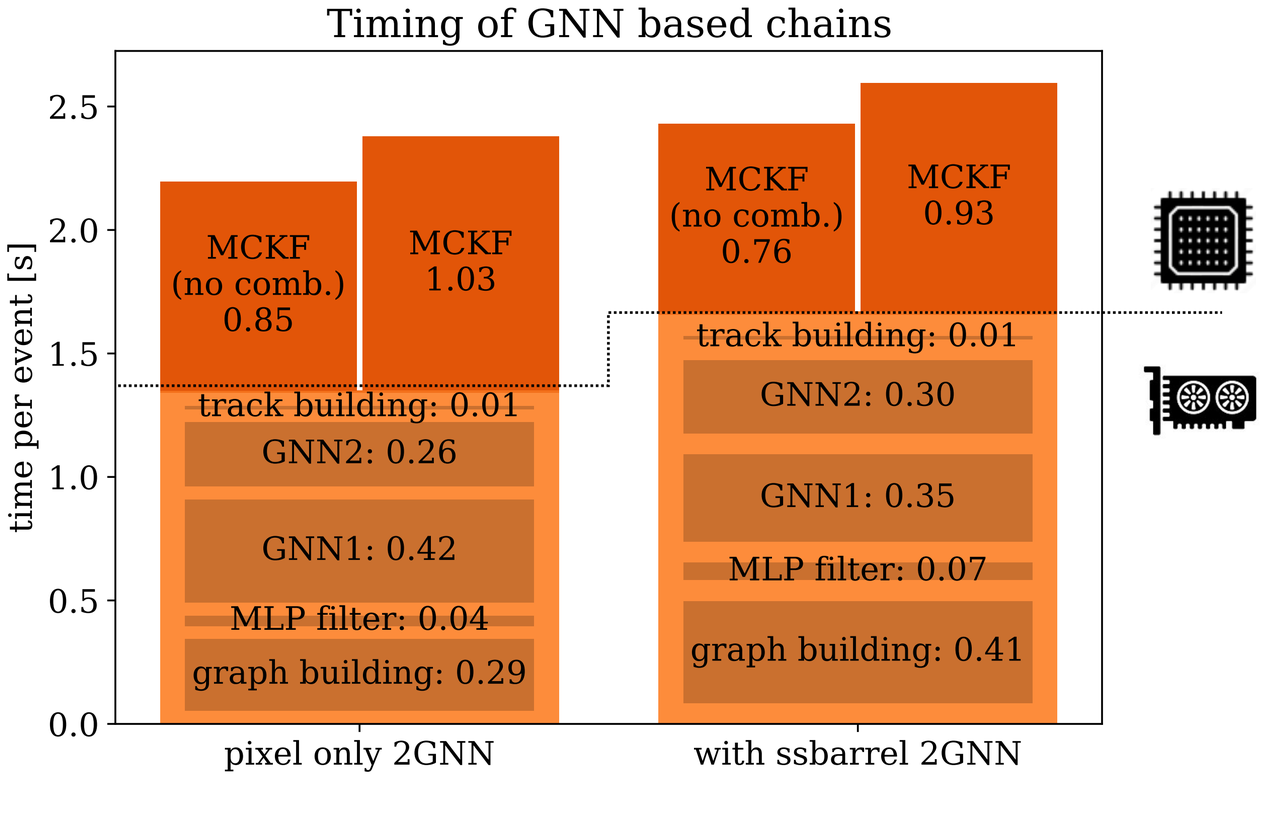
[L-G Gagnon, CTD 2023]

R&D line: machine learning

- Graph network based pattern recognition with CKF on top

Preliminary results show that the CKF can be restricted to a branch number = 1
if first two (short) strip layers are also included (taking best hit only)





C C C

Collaborating

ACTS is Open Source and invites contributions, corrections, interactions

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

Clone:

https://github.com/<username>/acts

Develop & Make a PR

Development, Exchange with Experts, Collaboration, Code review, CI testing

Make an Issue:

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

Ask on mattermost:

https://mattermost.web.cern.ch/acts/channels/town-square

Discuss at the open develops meeting

https://indico.cern.ch/category/7968/

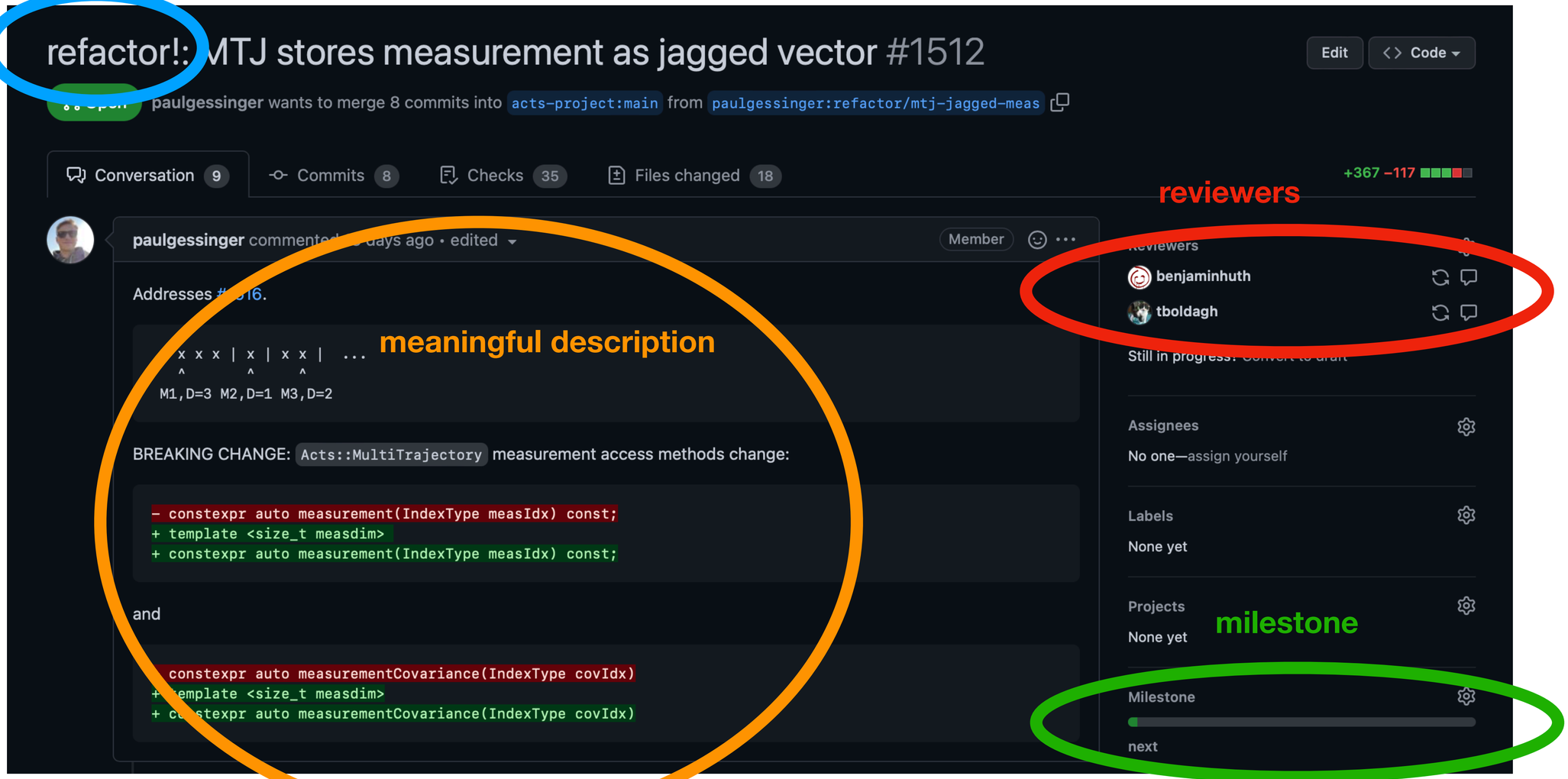
Tuesday 17:00, CE(S)T

new, periodic Asia-friendly slot 9:00 CE(S)T ~ 1/month

Contributing

Pull requests come with a template that guides through a proper submission

semantic naming: feat, doc, refactor, fix



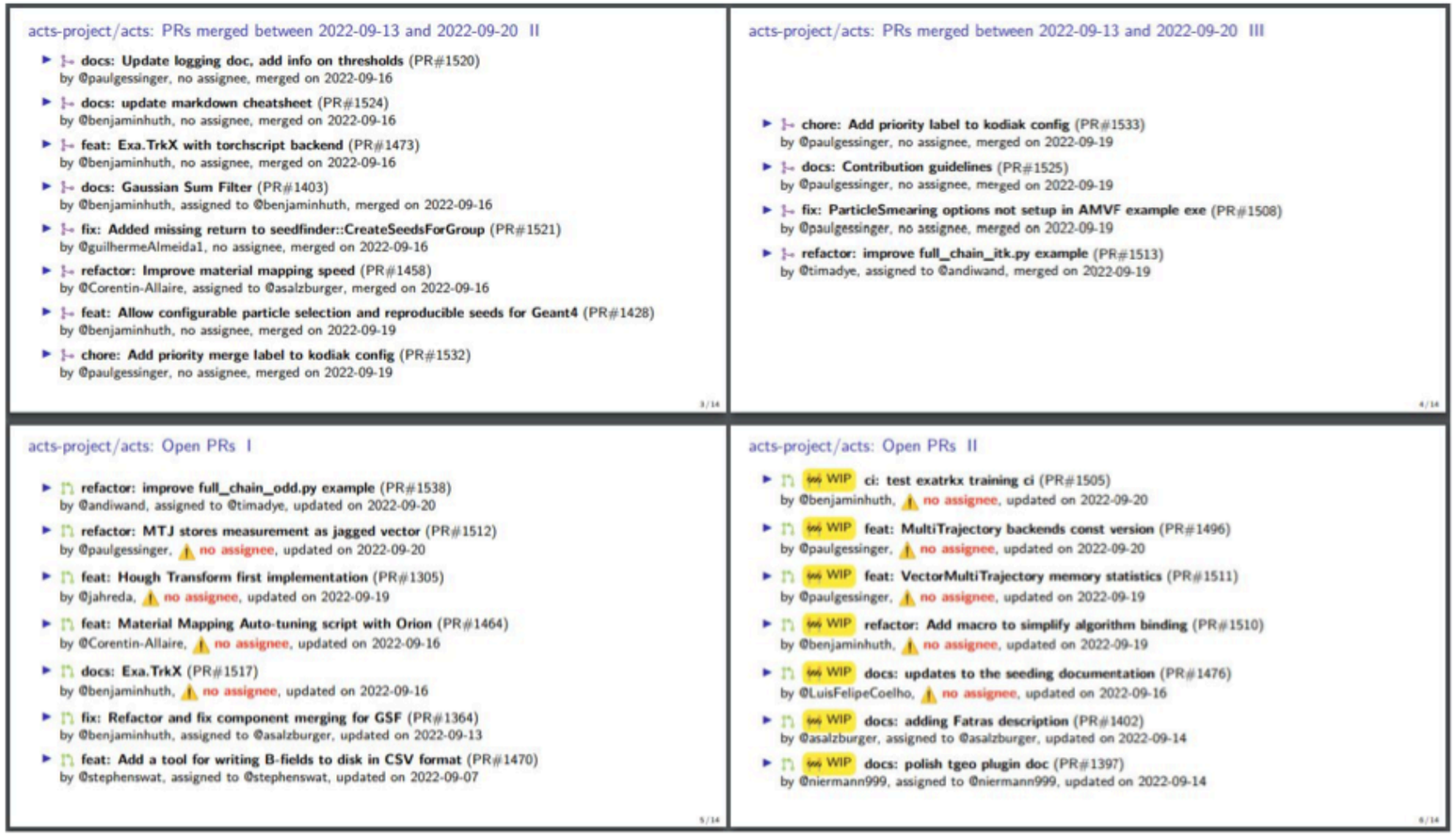
Community

Community-Supported Components: Acts

Weekly dev meeting with involvement of users at multiple experiments

Status of work visibility through presentations

Example of agile in community software



acts-project/acts: PRs merged between 2022-09-13 and 2022-09-20 II

- **docs: Update logging doc, add info on thresholds** (PR#1520)
  by @paulgessinger, no assignee, merged on 2022-09-16
- **docs: update markdown cheatsheet** (PR#1524)
  by @benjaminhuth, no assignee, merged on 2022-09-16
- **feat: Exa.TrkX with torchscript backend** (PR#1473)
  by @benjaminhuth, no assignee, merged on 2022-09-16
- **docs: Gaussian Sum Filter** (PR#1403)
  by @benjaminhuth, assigned to @benjaminhuth, merged on 2022-09-16
- **fix: Added missing return to seedfinder::CreateSeedsForGroup** (PR#1521)
  by @guilhermeAlmeida1, no assignee, merged on 2022-09-16
- **refactor: Improve material mapping speed** (PR#1458)
  by @Corentin-Allaire, assigned to @asalzburger, merged on 2022-09-16
- **feat: Allow configurable particle selection and reproducible seeds for Geant4** (PR#1428)
  by @benjaminhuth, no assignee, merged on 2022-09-19
- **chore: Add priority merge label to kodiak config** (PR#1532)
  by @paulgessinger, no assignee, merged on 2022-09-19

3/14

acts-project/acts: PRs merged between 2022-09-13 and 2022-09-20 III

- **chore: Add priority label to kodiak config** (PR#1533)
  by @paulgessinger, no assignee, merged on 2022-09-19
- **docs: Contribution guidelines** (PR#1525)
  by @paulgessinger, no assignee, merged on 2022-09-19
- **fix: ParticleSmearing options not setup in AMVF example exe** (PR#1508)
  by @paulgessinger, no assignee, merged on 2022-09-19
- **refactor: improve full_chain_itk.py example** (PR#1513)
  by @timadye, assigned to @andiwand, merged on 2022-09-19

4/14

acts-project/acts: Open PRs I

- **refactor: improve full_chain_odd.py example** (PR#1538)
  by @andiwand, assigned to @timadye, updated on 2022-09-20
- **refactor: MTJ stores measurement as jagged vector** (PR#1512)
  by @paulgessinger, ⚠ no assignee, updated on 2022-09-20
- **feat: Hough Transform first implementation** (PR#1305)
  by @jahreda, ⚠ no assignee, updated on 2022-09-19
- **feat: Material Mapping Auto-tuning script with Orion** (PR#1464)
  by @Corentin-Allaire, ⚠ no assignee, updated on 2022-09-16
- **docs: Exa.TrkX** (PR#1517)
  by @benjaminhuth, ⚠ no assignee, updated on 2022-09-16
- **fix: Refactor and fix component merging for GSF** (PR#1364)
  by @benjaminhuth, assigned to @asalzburger, updated on 2022-09-13
- **feat: Add a tool for writing B-fields to disk in CSV format** (PR#1470)
  by @stephenswat, assigned to @stephenswat, updated on 2022-09-07

5/14

acts-project/acts: Open PRs II

- WIP **ci: test exatrkx training ci** (PR#1505)
  by @benjaminhuth, ⚠ no assignee, updated on 2022-09-20
- WIP **feat: MultiTrajectory backends const version** (PR#1496)
  by @paulgessinger, ⚠ no assignee, updated on 2022-09-20
- WIP **feat: VectorMultiTrajectory memory statistics** (PR#1511)
  by @paulgessinger, ⚠ no assignee, updated on 2022-09-19
- WIP **refactor: Add macro to simplify algorithm binding** (PR#1510)
  by @benjaminhuth, ⚠ no assignee, updated on 2022-09-19
- WIP **docs: updates to the seeding documentation** (PR#1476)
  by @LuisFelipeCoelho, ⚠ no assignee, updated on 2022-09-16
- WIP **docs: adding Fatras description** (PR#1402)
  by @asalzburger, assigned to @asalzburger, updated on 2022-09-14
- WIP **docs: polish tgeo plugin doc** (PR#1397)
  by @niermann999, assigned to @niermann999, updated on 2022-09-14

6/14

[Wouter Deconinck, CHEP2023, Plenary Contribution]

Final remarks

- The ACTS project has grown immensely during the last years
  - a very feature rich toolbox that is still enlarging
  - increased focus on consolidation & performance tuning has started

- Collaboration is invited
  - This is an open source project where we want to serve many clients
    - Have the resources to optimise known algorithms and concepts
    - Free resources to do innovative R&D

Links

acts-developers@cern.ch
acts-users@cern.ch
acts-parallelization@cern.ch
acts-machineleaning@cern.ch
acts-telescope@cern.ch

Email lists

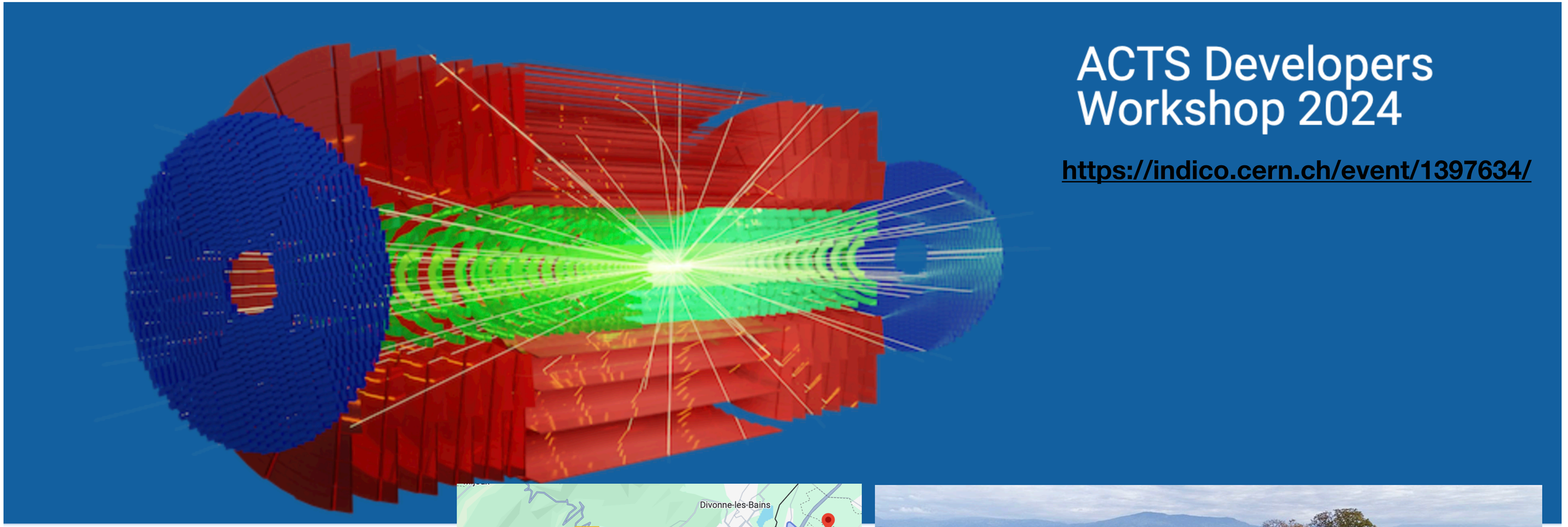
Code base for acts-project, R&D lines, spin offs

Communication channel

Online documentation (built from latest snapshot)

[CSBS, ACTS]

Write-up documentation



ACTS Developers Workshop 2024

https://indico.cern.ch/event/1397634/

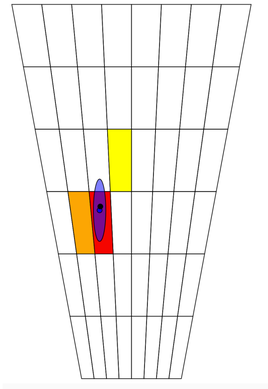
18–21 Nov 2024
Chateau de Bossey
Europe/Zurich timezone

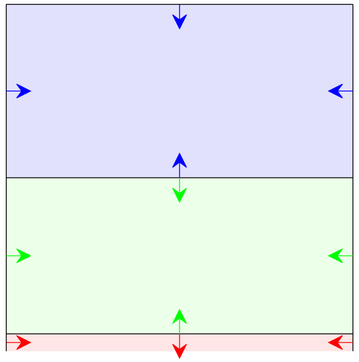
Spin-offs

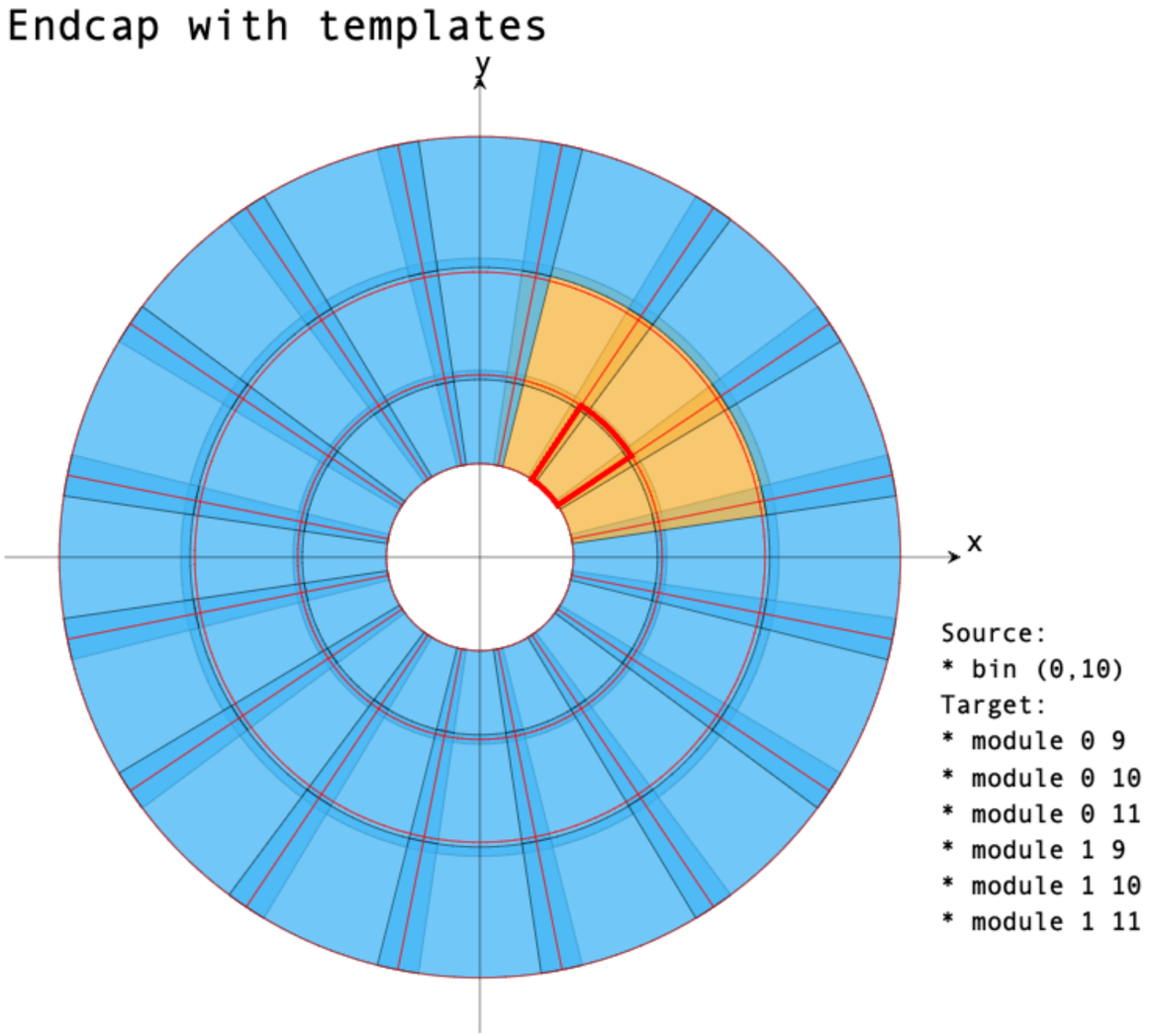
Plotting: actsvg

2D plotting library dedicated for tracking

- No dependencies, C++ header only, no ACTS dependency
  - ACTS and detray translate into `actsvg::meta` objects
- Plot geometry & geometric relations (on mouse over effects for debugging)
- Plot clusters & cluster information







actsvg is a generic plotting library.

Community: Open Data Detector & key4hep

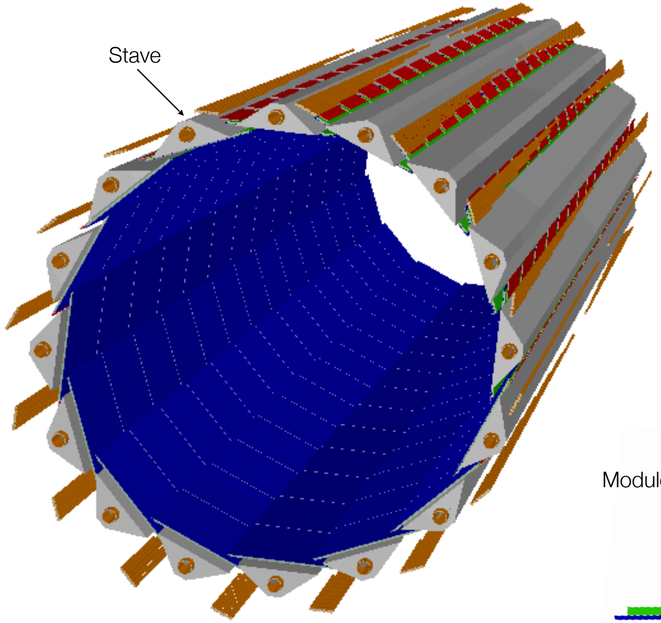
Evolution of TrackML detector

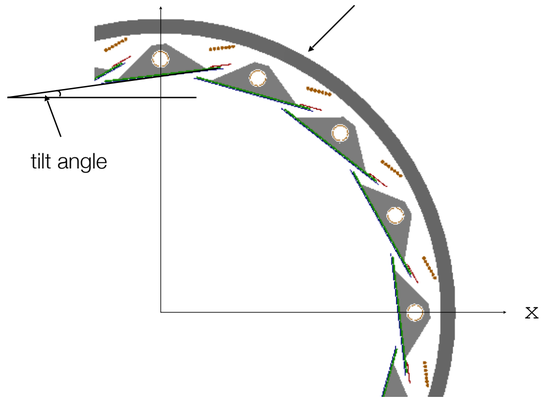
- Re-implemented in DD4Hep to enable full/fast simulation
- Quasi-realistic feedback to allow real-life scenario testing of algorithms
- Supports TrackML output format through ACTS binding (work ongoing to also support edm4hep)

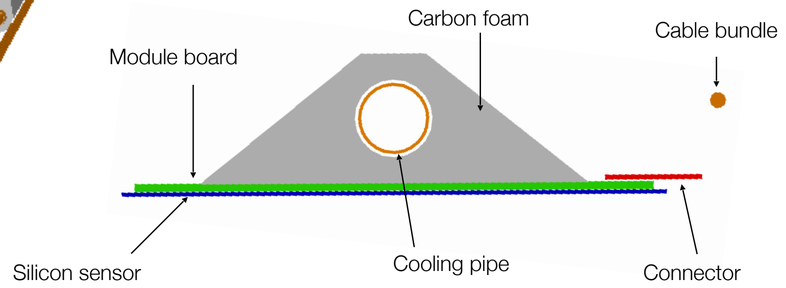
ACTS integration into key4hep SW stack

- Codename: acts4hep
- Summer student project to make a ACTS Gaudi based demonstrator

[AS, CHEP2023 Parallel talk]







Ongoing activity to include first Calorimeter description, MS to follow.