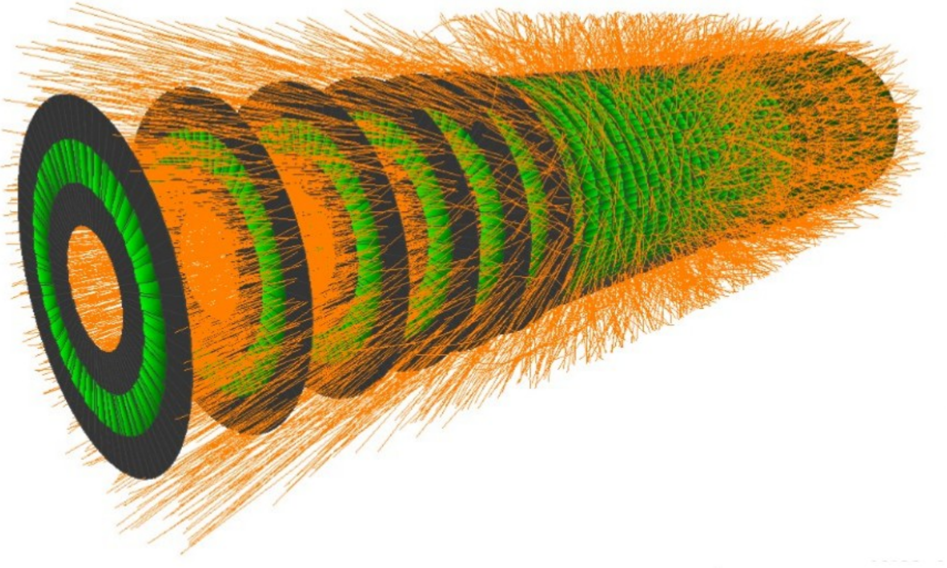
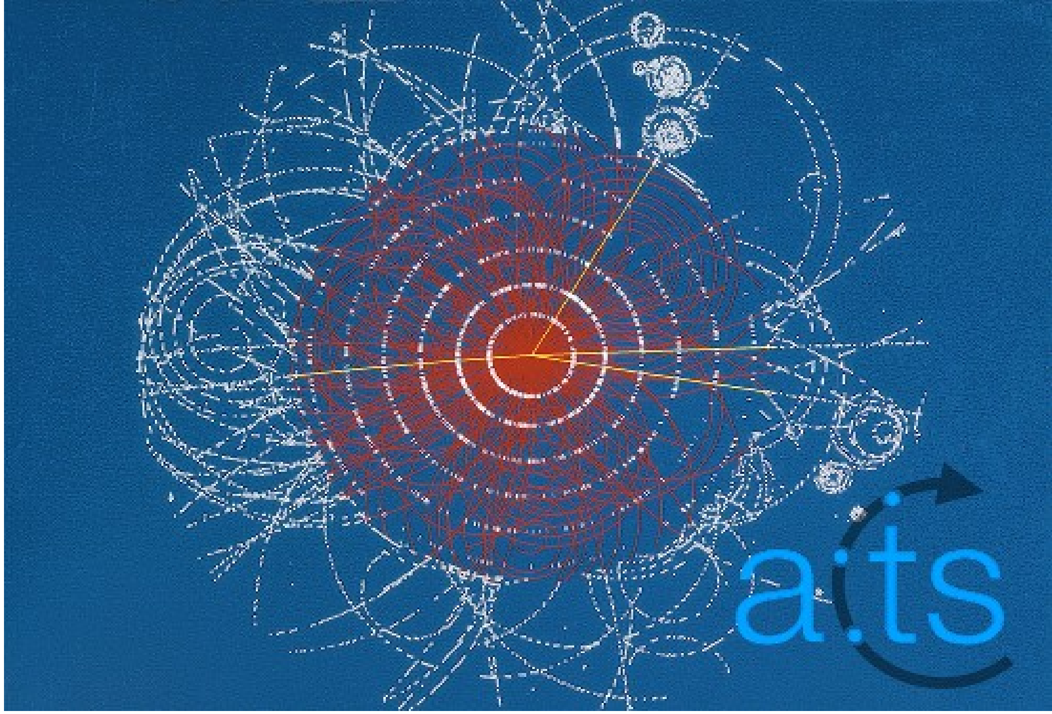




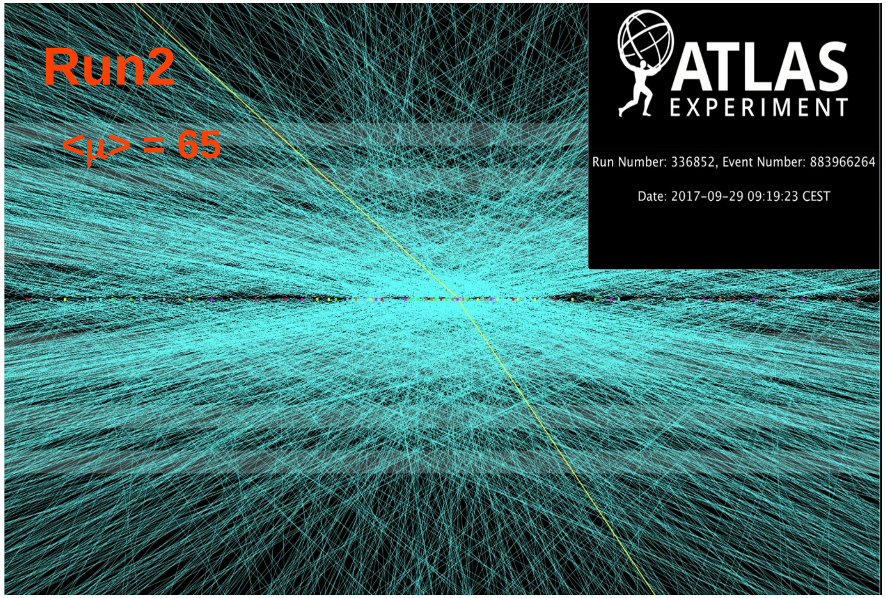
ACTS Project Status

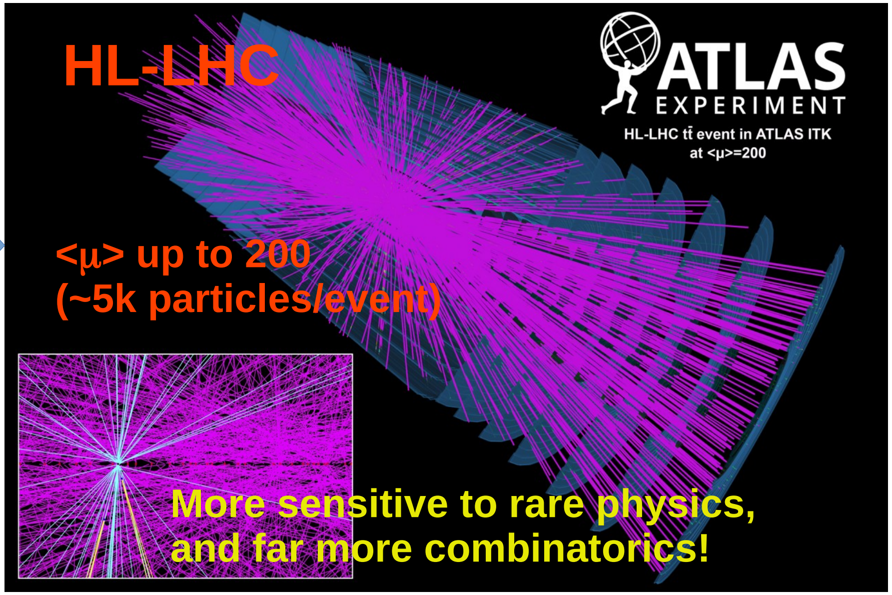
Xiaocong Ai for the ACTS developers

ACTS Workshop 2020, May 25, 2020

The tracking challenge

- Much increased combinatorics with high pileup at future hadron colliders
  - Great strain on CPU
  - Increased track reconstruction time
- Accurate, efficient and fast tracking software is needed to achieve physics goals
  - **How to exploit fast tracking techniques, parallelism and acceleration?**

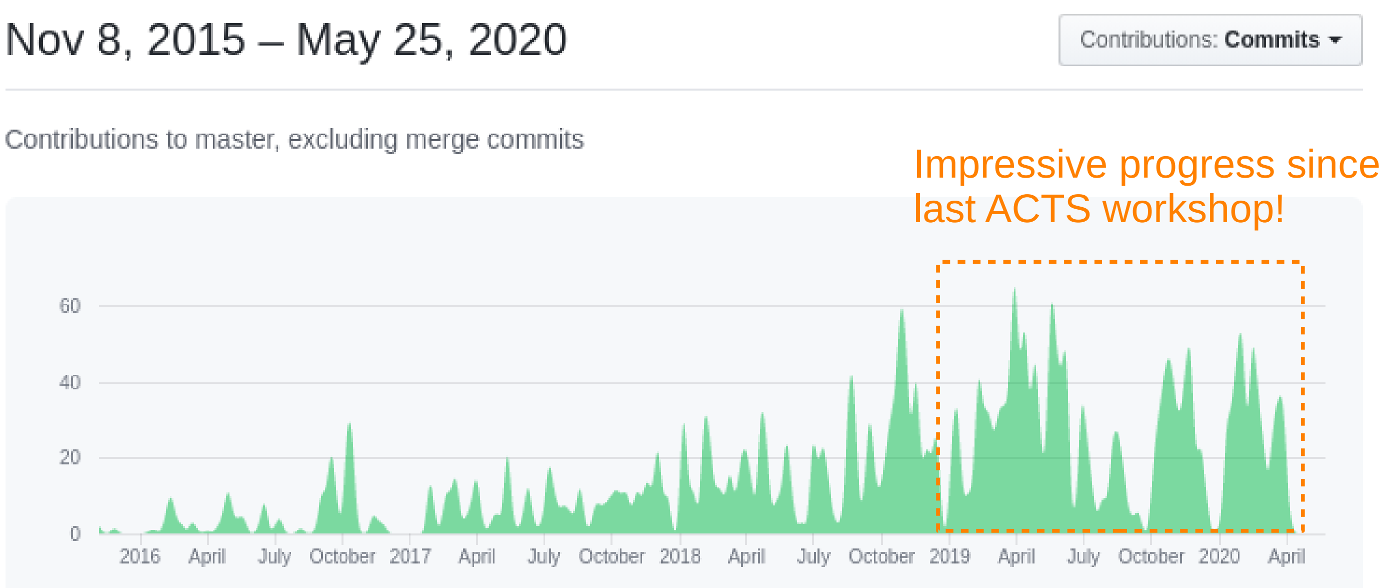




ACTS goals

- Prepare an experiment-independent tracking toolkit for future detectors, e.g. ATLAS at HL-LHC, based on ATLAS tracking experience
  - ATLAS tracking software is well tested but thread-unsafe and difficult to maintenable
- Provide an open-source R&D platform:
  - New tracking algorithms (see M. Kiehn's talk)
  - Hardware architectures (see G. Mania's talk)

Migration to github last month allowing contribution from people without CERN account



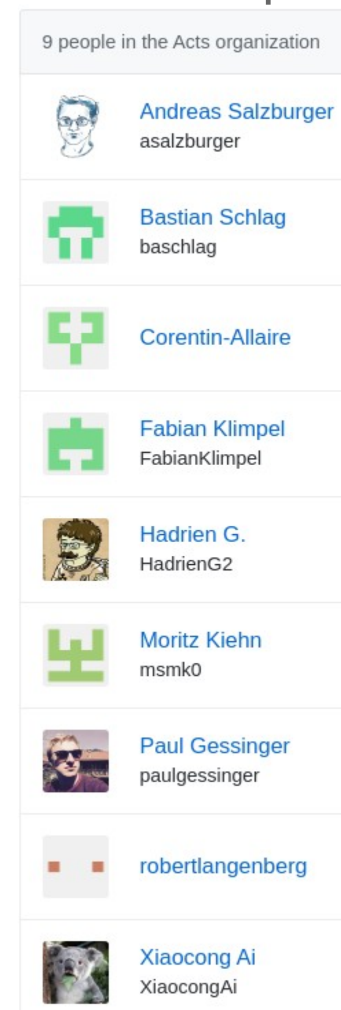
Jun 2016 - **v0.01.00**

Jan 2019 - **v0.08.00**

May 2020- **v0.25.00**

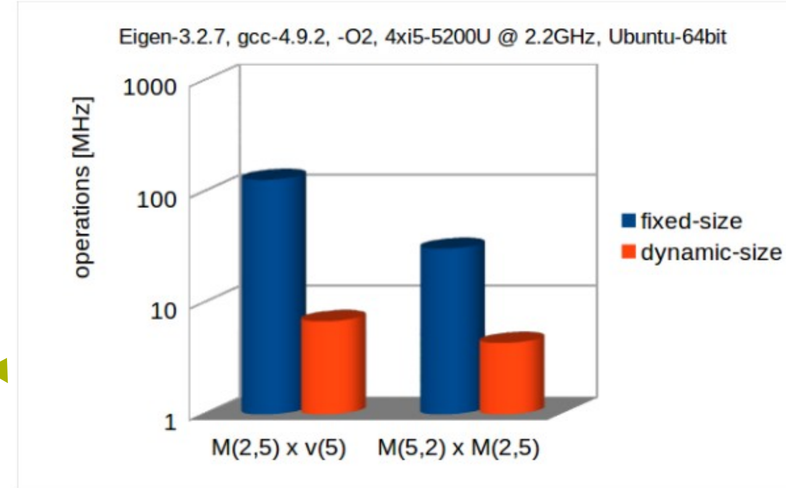
https://github.com/acts-project/acts

An incomplete list of ACTS developers



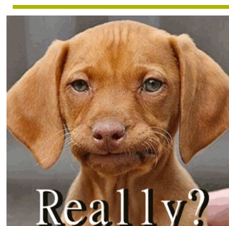
ACTS design

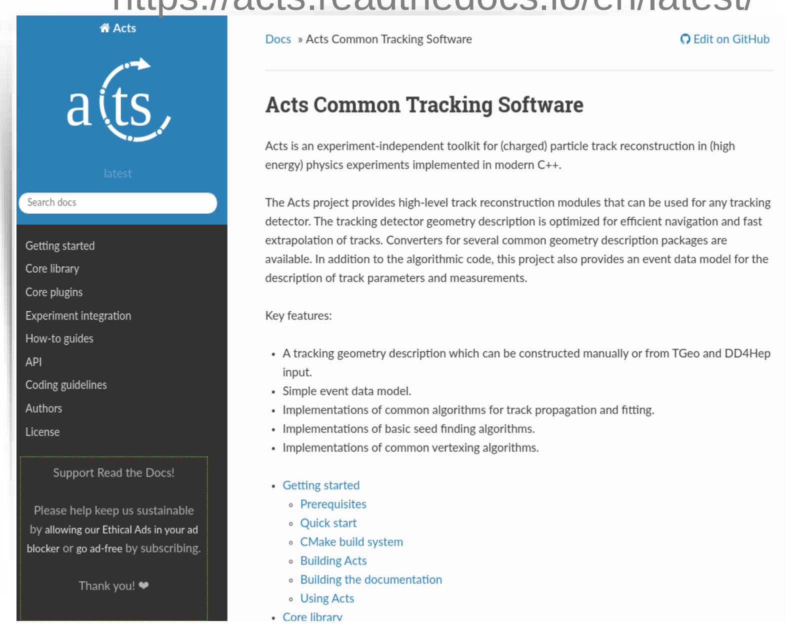
- Modern C++ 17 concepts
- Highly-templated design to avoid virtual lookup
  - Detector and magnetic field agnostic
- Strict thread-safety to facilitate concurrency
  - Const-correctness, stateless tools
- Efficient memory allocation and access
  - Eigen-based Event Data Model (EDM)
- Supports for contextual condition data
  - Geometry/Calibration/Magnetic field
- Rigorous unit tests
- Highly configurable for usability
- Minimal dependencies
- Well-documented




```
size_t algorithmNumber;                ///< Unique algorithm identifier
size_t eventNumber;                    ///< Unique event identifier
WhiteBoard& eventStore;                ///< Per-event data store
Acts::GeometryContext geoContext;      ///< Per-event geometry context
Acts::MagneticFieldContext
    magFieldContext;                   ///< Per-event magnetic Field context
Acts::CalibrationContext calibContext; ///< Per-event calbiration context
```

https://acts.readthedocs.io/en/latest/





ACTS components and functionalities

Infrastructure consolidation and new features since last ACTS workshop

acts-core

- CI
- Core
- Examples
- Fatras
- Plugins
- Tests

Core:

- EventData
- Fitter (see X. Ai's tutorial on truth fitting)
- Geometry (see A. Salzburger's tutorial on geometry building)
- MagneticField
- Material
- Propagator (see A. Salzburger's tutorial on propagation)
- Seeding
- Surfaces
- TrackFinder (see X. Ai's tutorial on CKF)
- Utilities
- Vertexing (see B. Schlag's tuturial on Vertexing)
- Visualization

acts-fatras

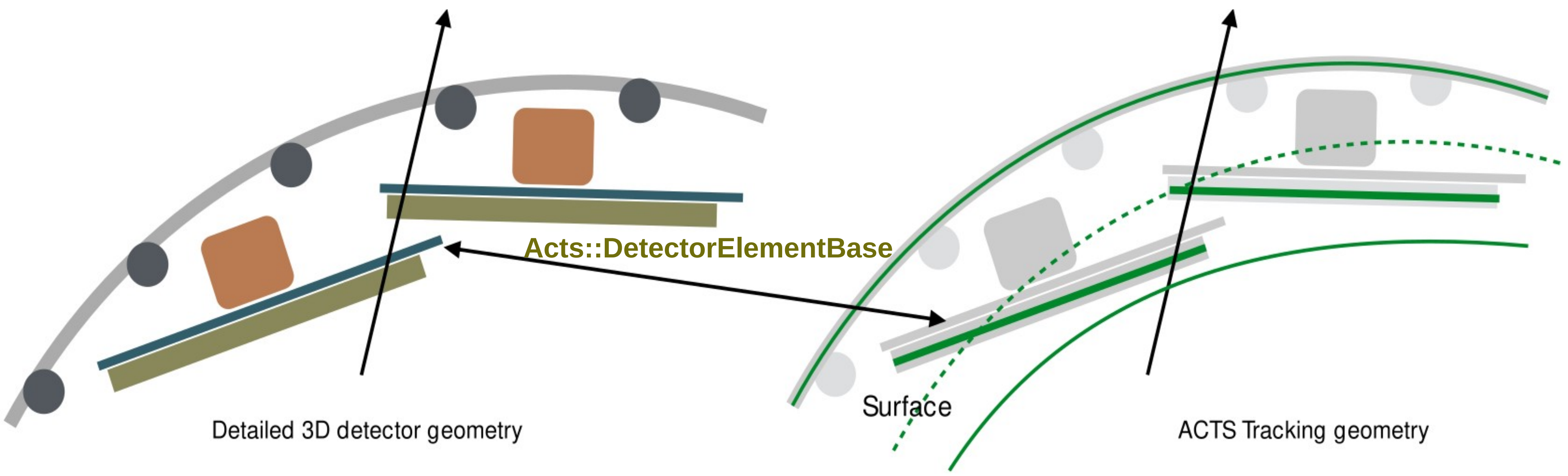
- Fast simulation engine (see M. Kiehn's tutorial)
- Integrated into acts-core

acts-framework

- A light-weight Gaudi style test framework for event processing, integration and concurrency test
- Integration into acts-core to ease maintenance and performance monitoring

Geometry

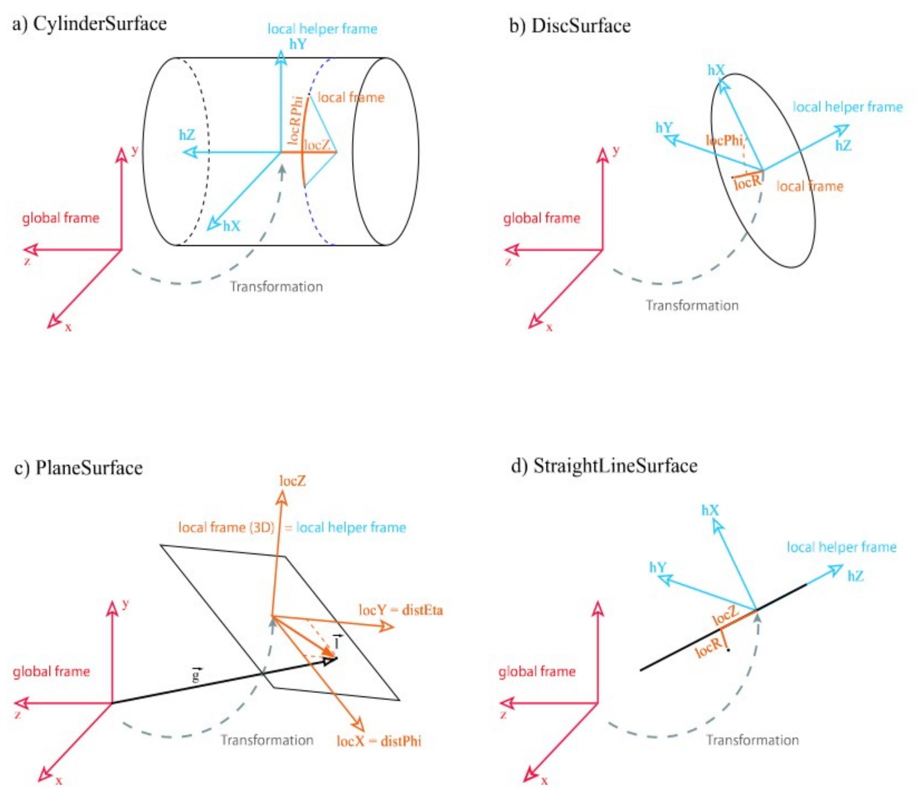
- To reduce CPU consumption and navigation speed-up, tracking geometry (i.e. geometry used for track reconstruction) is simplified from full simulation geometry
  - Binding via Acts::DetectorElementBase which can be converted from other detector element representation via geometry plugins:
    - DD4hep, TGeo, GeoModel Plugin
- Implemented HEP detector geometry
  - Silicon, Calorimeter, MuonSpectrometer



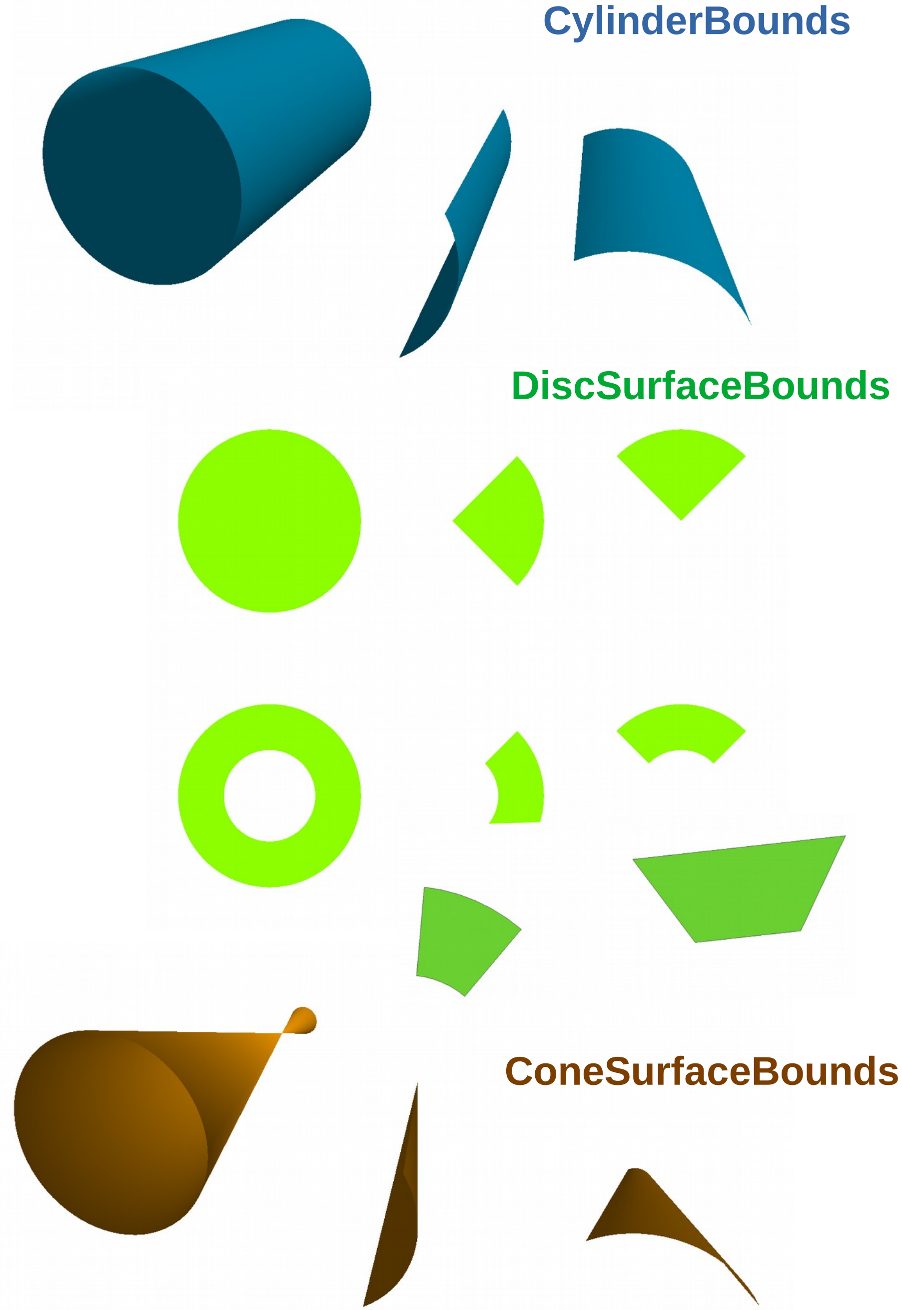
The Surface class

- Acts::Surface is the key component of tracking geometry
  - Largely transcribed from ATLAS SW
- All surface types have a polygonal representation allowing for triangular mesh of all surfaces (could further speed up navigation)

Surface types in ATLAS SW

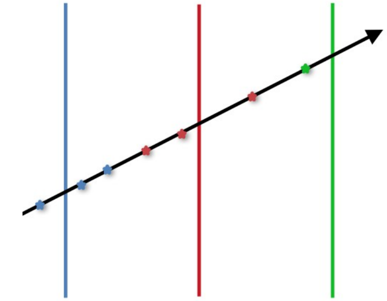


ACTS Surface bound examples

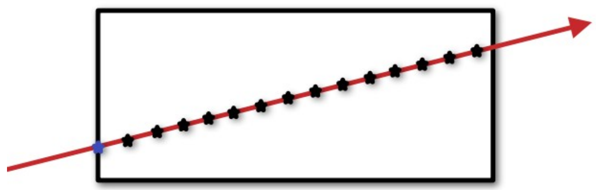


Material description

- Material effects need to be considered in tracking
- Material mapping tools (see C. Allaire's slides) are available to map Gean4-based full detector material (recorded using Geantino scan) onto surfaces or volumes:
  - Discrete binned surfaces (for e.g. Silicon detector)
  - 3D volume grid points (for e.g. Calorimeter)

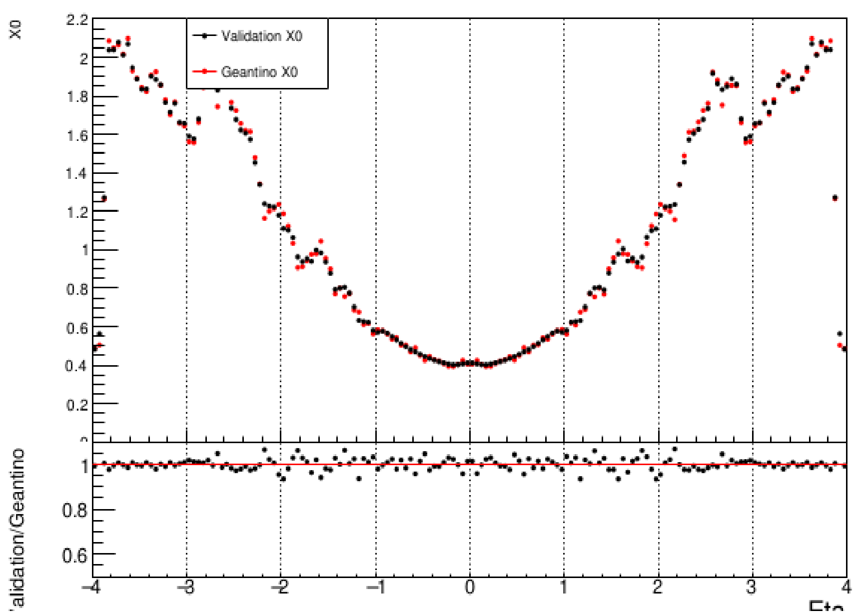


Surface Material Mapping

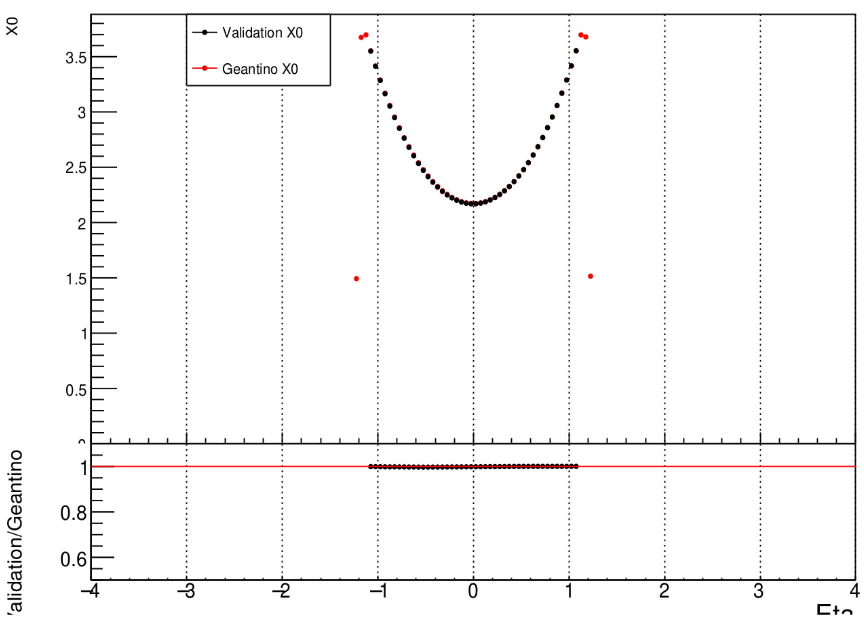


Volume Material Mapping

X0 ratio Validation/Geantino vs η for ITk

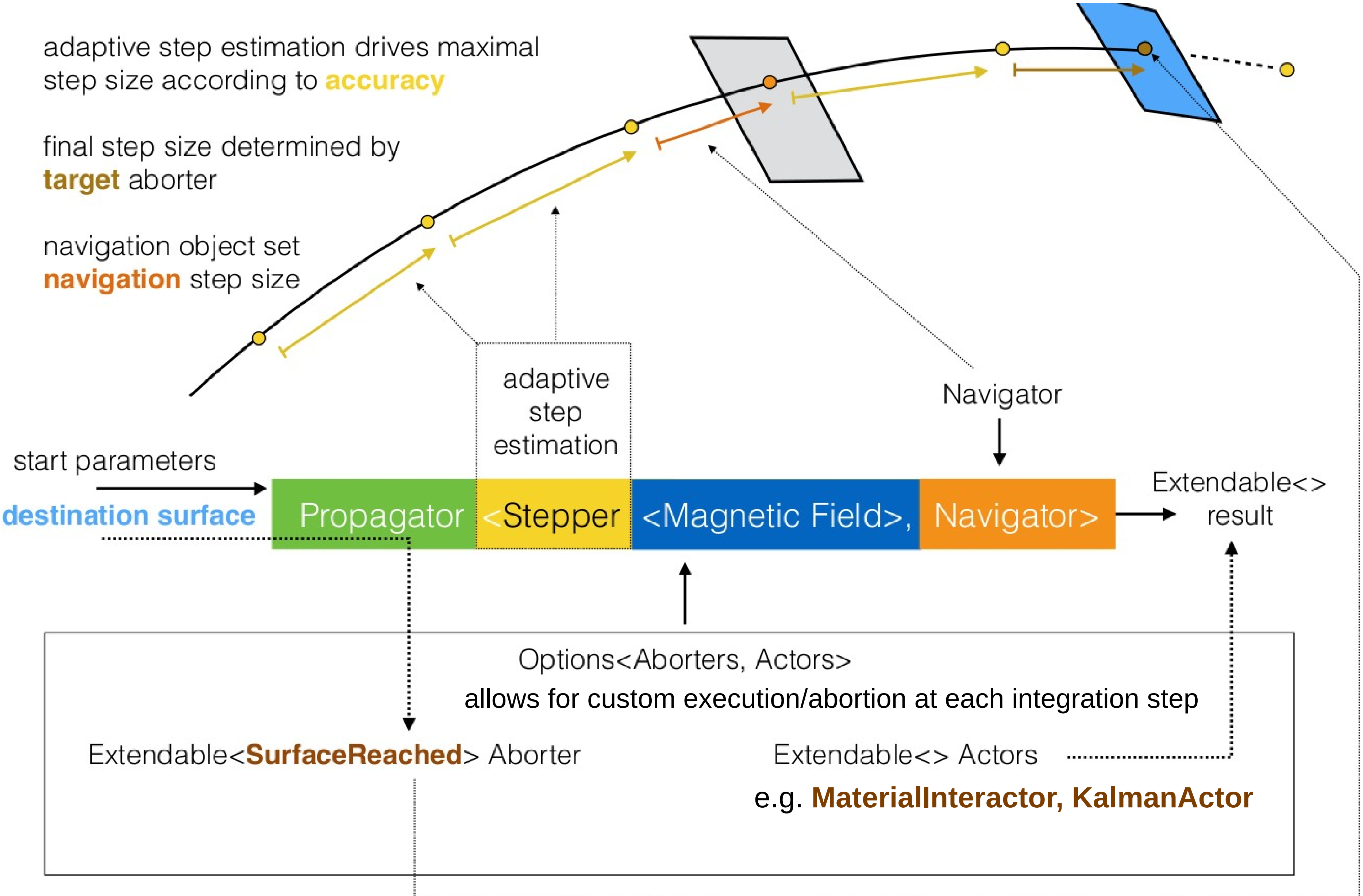


X0 ratio Validation/Geantino vs h for a dummy Calorimeter



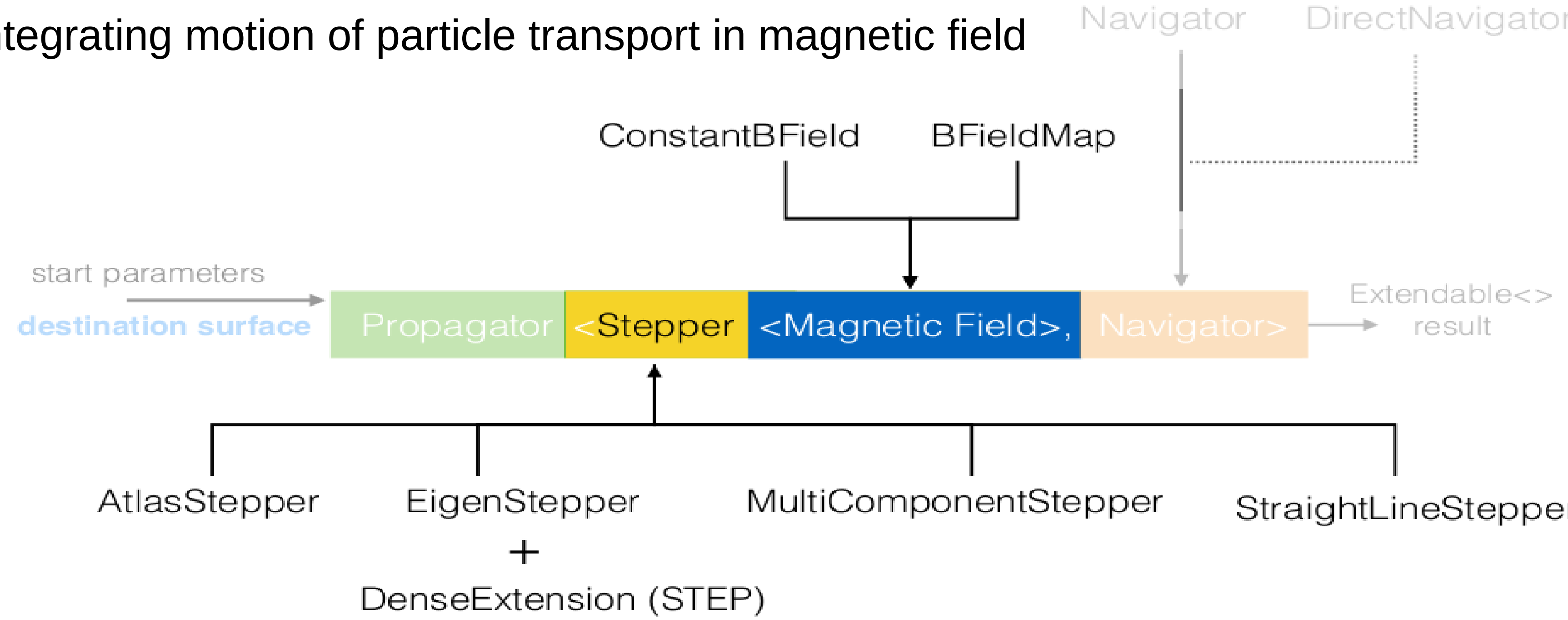
Propagator interface

Integrating particle transport and geometry navigation
Highly-templated design emphasizing on speed and customizability



Track parameter propagation

Integrating motion of particle transport in magnetic field



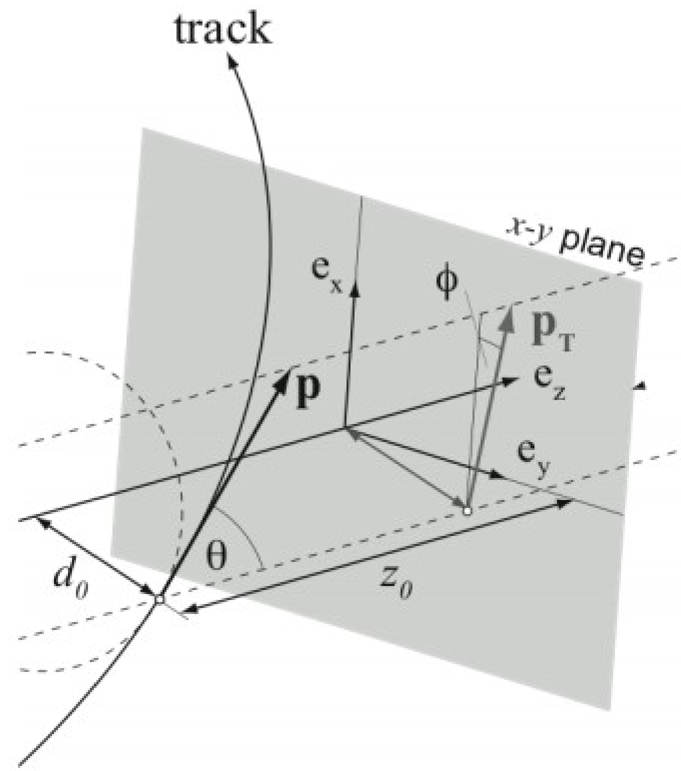
- EigenStepper as the primary integrator: ATLAS adaptive Runge-Kutta stepper rewritten using Eigen
- Dense Environment Extension for transport in dense volumes, e.g. calorimeter
- Timing information is included in integration (x, y, z, **t**, T_x, T_y, T_z, q/p) to allow for time measurement
- Supports free parameters and covariance representation without binding to surface

Track Parameter and Measurement EDM

- Coefficients and covariance are described using fixed-size Eigen::Matrix

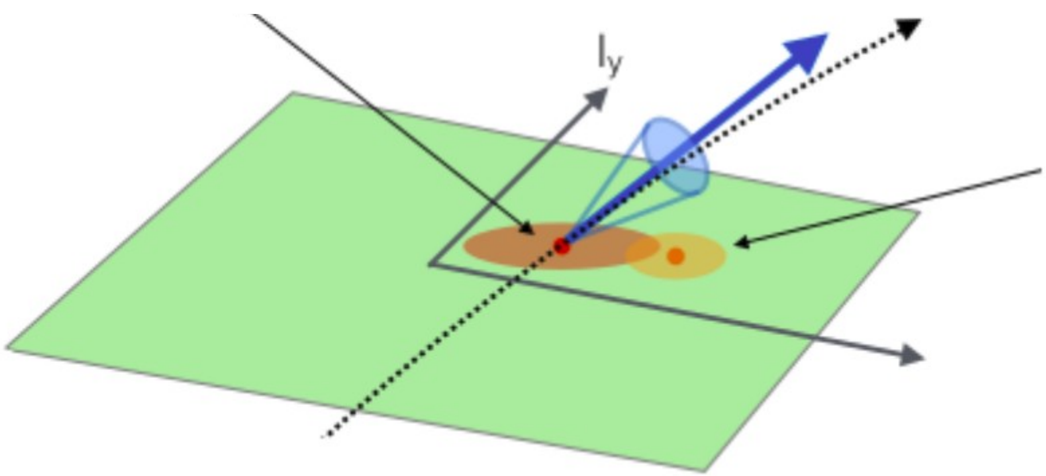
TrackParameters: (l_0, l_1, ϕ, θ, q/p, t)

- l_0, l_1: Coordinate in local surface frame (Bound Parameters) or curvilinear frame (Curvilinear Parameters)
  - Meaning of l_0, l_1 varies depending on surface type
- p, ϕ, θ: Momentum and direction
- q: Charge
- t: Per-track timing info



e.g. perigee track parameters at perigee surface
$l_0 = d_0$, $l_1 = z_0$

Supports multi-component track parameters representation



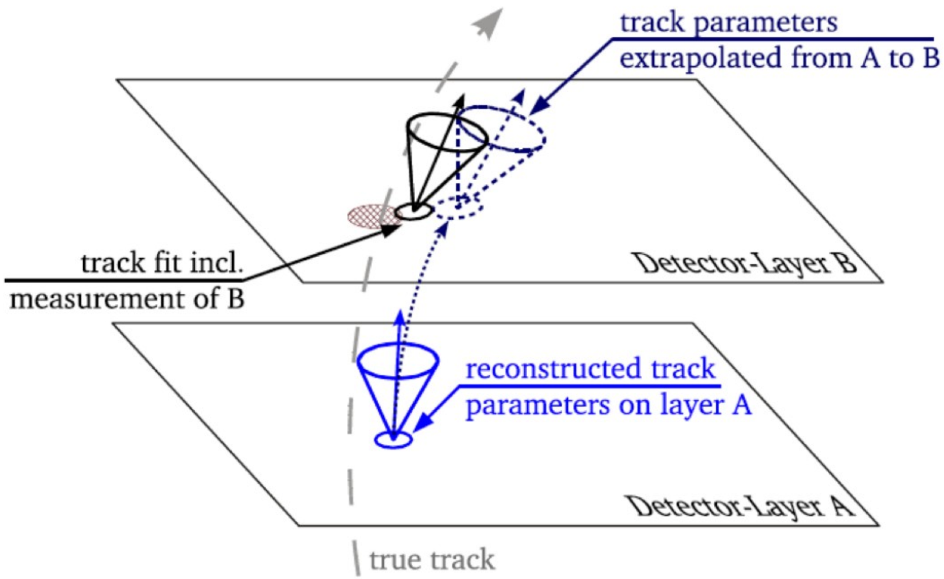
Measurement:

- Contains a SourceLink to original detector measurement
- Uses std::variant to as a wrapper of heterogeneous measurement (1D, 2D, …)

Track State EDM

- TrackState EDM is designed based on concept of KalmanFilter
  - Includes parametric and measured part

```
/// The parameter part
/// This is all the information that concerns the
/// the track parameterisation and the jacobian
/// It is enough to to run the track smoothing
struct {
  /// The predicted state
  std::optional<Parameters> predicted{std::nullopt};
  /// The filtered state
  std::optional<Parameters> filtered{std::nullopt};
  /// The smoothed state
  std::optional<Parameters> smoothed{std::nullopt};
  /// The transport jacobian matrix
  std::optional<Jacobian> jacobian{std::nullopt};
  /// The path length along the track - will help sorting
  double pathLength = 0.;
  /// chisquare
  double chi2 = 0;
} parameter;
```




```
/// @brief Nested measurement part
/// This is the uncalibrated and calibrated measurement
/// (in case the latter is different)
struct {
  /// The optional (uncalibrated) measurement
  std::optional<SourceLink> uncalibrated{std::nullopt};
  /// The optional calibrabed measurement
  std::optional<FittableMeasurement<SourceLink>> calibrated{std::nullopt};
} measurement;
```

Via Calibrator during fitting

Track EDM

- Eigen::Array based track EDM (Acts::MultiTrajectory), i.e. container of track states on trajectories
  - Provides read-write views into separate storage of parameter coefficients and covariance

```
using Coefficients = Eigen::Matrix<Scalar, Size, 1, Flags>;
using Covariance = Eigen::Matrix<Scalar, Size, Size, Flags>;
using CoefficientsMap = Eigen::Map<ConstIf<Coefficients, ReadOnlyMaps>>;
using CovarianceMap = Eigen::Map<ConstIf<Covariance, ReadOnlyMaps>>;
```

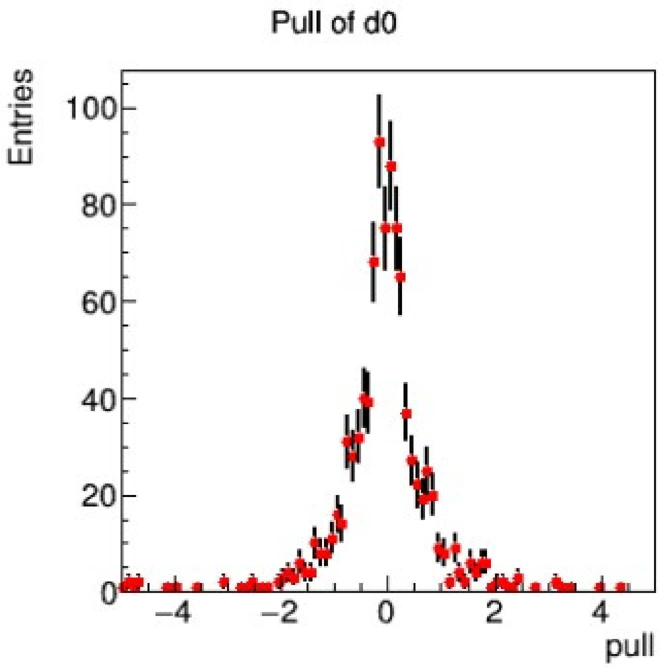
- Keeps track of storage index
  - Allows for branching of tracks (multi-trajectories case) via parent relationship
  - Avoids storage duplication for shared measurements and parameters

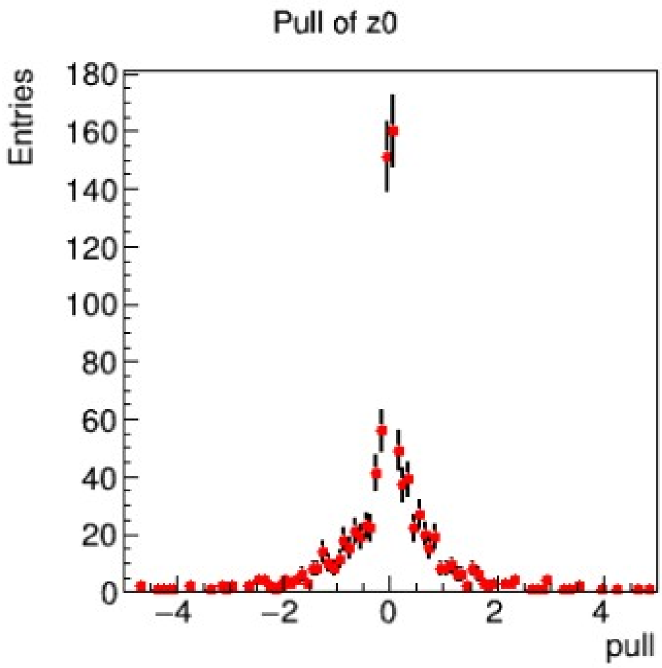
```
struct IndexData {
  using IndexType = uint16_t;

  static constexpr IndexType kInvalid = UINT16_MAX;

  IndexType irefsurface = kInvalid;
  IndexType iprevious = kInvalid;
  IndexType ipredicted = kInvalid;
  IndexType ifiltered = kInvalid;
  IndexType ismoothed = kInvalid;
  IndexType ijacobian = kInvalid;
  IndexType iprojector = kInvalid;

  double chi2;
  double pathLength;
  TrackStateType typeFlags;

  IndexType iuncalibrated = kInvalid;
  IndexType icalibrated = kInvalid;
  IndexType icalibratedsourcelink = kInvalid;
  IndexType measdim = 0;
};
```

Track fitting

- KalmanFilter (KF) is used as an Actor in propagator
- Supports hole search and outlier rejection during the fitting
- Supports two different approaches for smoothing
  - Using 'smoothing-matrix' formalism based on Jacobians in forward filtering
  - Run an additional Kalman filtering in backward direction
- Gaussian Sum Filter as non-gaussian extension of KF is available

Perigee track parameter resolution validation TrackML detector, ATLAS B field





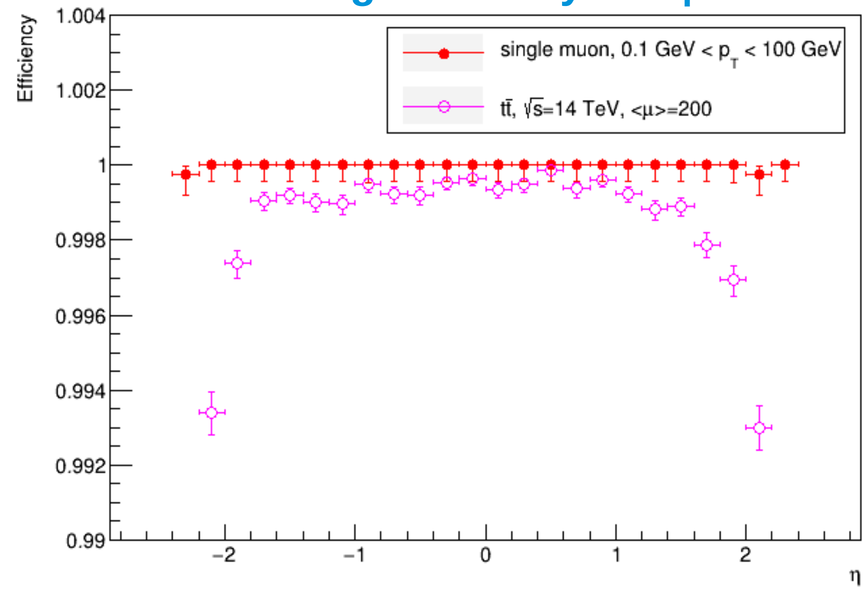
Little material for TrackML detector, hence no great justification for gaussian pulls

KalmanFitter performance

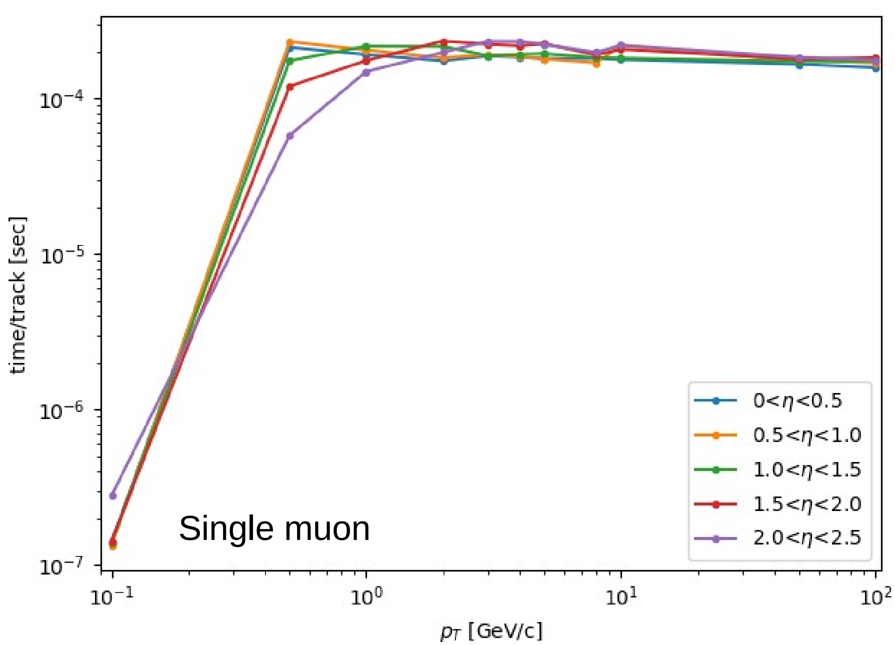
TrackML detector, ATLAS B field

- Validated with p_T down to 100 MeV
- 100% fitting efficiency
  - Defined as $\frac{N_{fit\ succeeds}}{N_{truth}}$

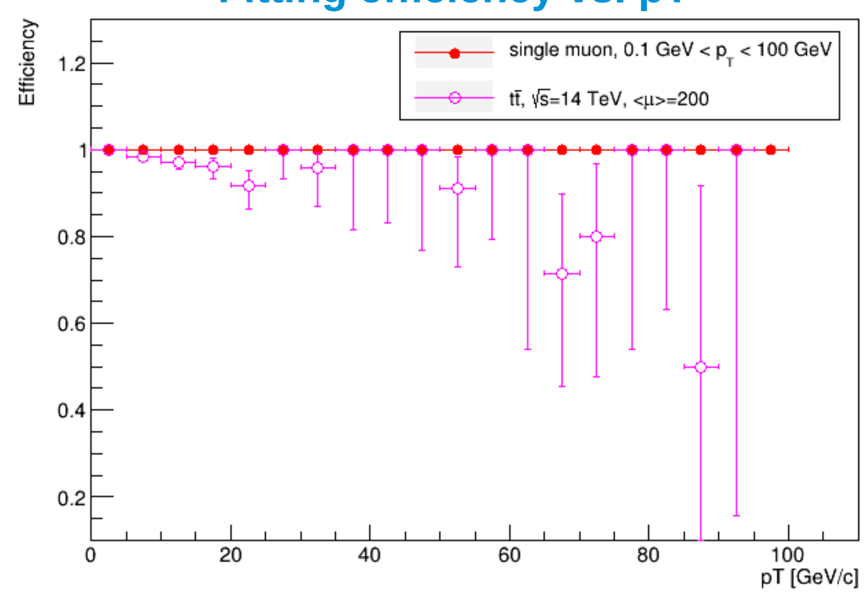
Fitting efficiency vs. η



Single track fitting time vs. pT

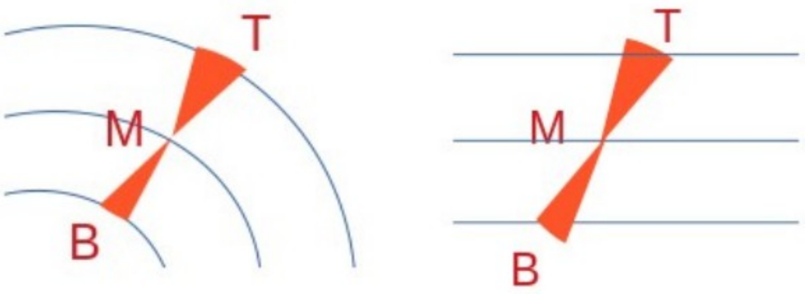


Fitting efficiency vs. pT

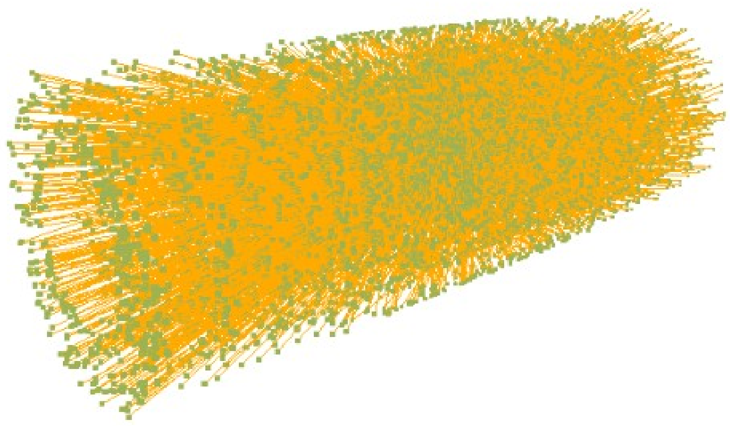


Track finding

- A combinatorial seed finder for track seeding
  - Fine-grained parallelism (independent search of Top and Bottom SpacePoint for Middle SP)



- The Combinatorial Kalman Filter (CKF) for track following
  - Simultaneous tracking fitting and finding (no refitting is needed)
  - Allows track branching if more than one compatible measurement found on a surface
    - Supports user-defined measurement search and branching strategy
      - Default selection criteria is based on Kalman filtering χ^2
  - Allows stopping of bad quality branch



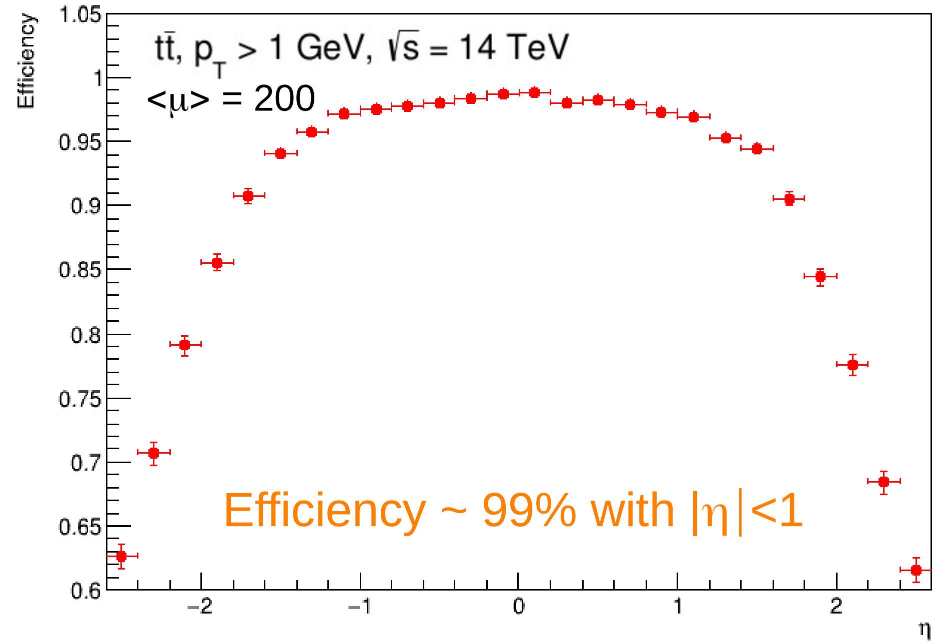
CKF results for ttbar events with μ = 200 (~7k particles, ~80k hits)

TrackML detector, ATLAS B field

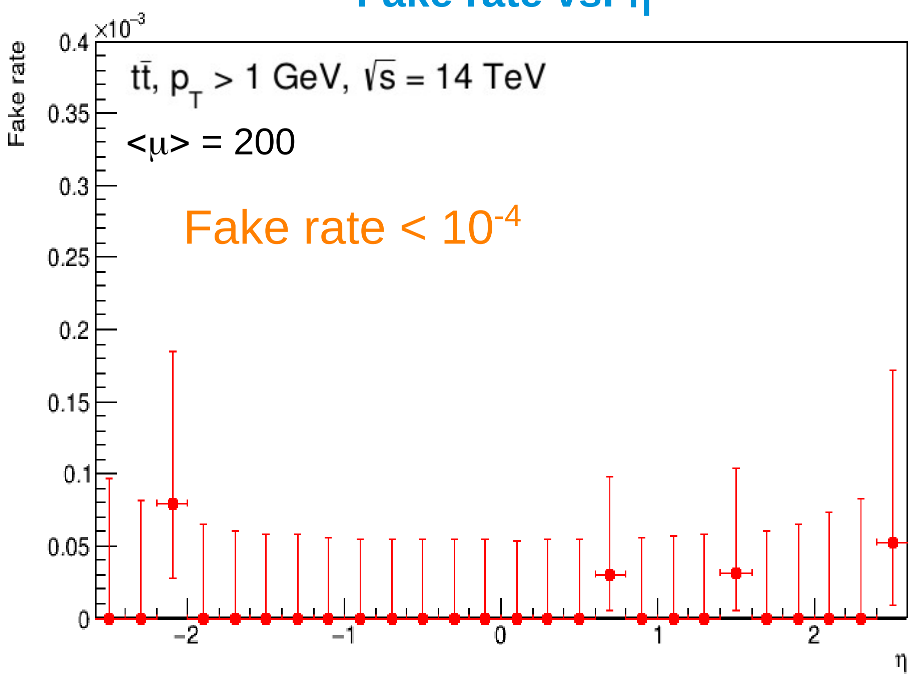
CKF performance

- All hits from truth particles with p_T>100 MeV are considered
  - Track finding efficiency: $\frac{N_{reco}(selected, matched)}{N_{truth}(selected)}$
  - Fake rate: $\frac{N_{reco}(selected, unmatched)}{N_{reco}(selected)}$
  - Duplication rate: $\frac{N_{reco}(selected, matched, duplicated)}{N_{reco}(selected, matched)}$
    - ➔ Reco-truth matching: $\frac{N_{hits}(Majority)}{N_{hits}(Total)} > 0.5$
    - ➔ Simple track selection: n_{Hits}>=9

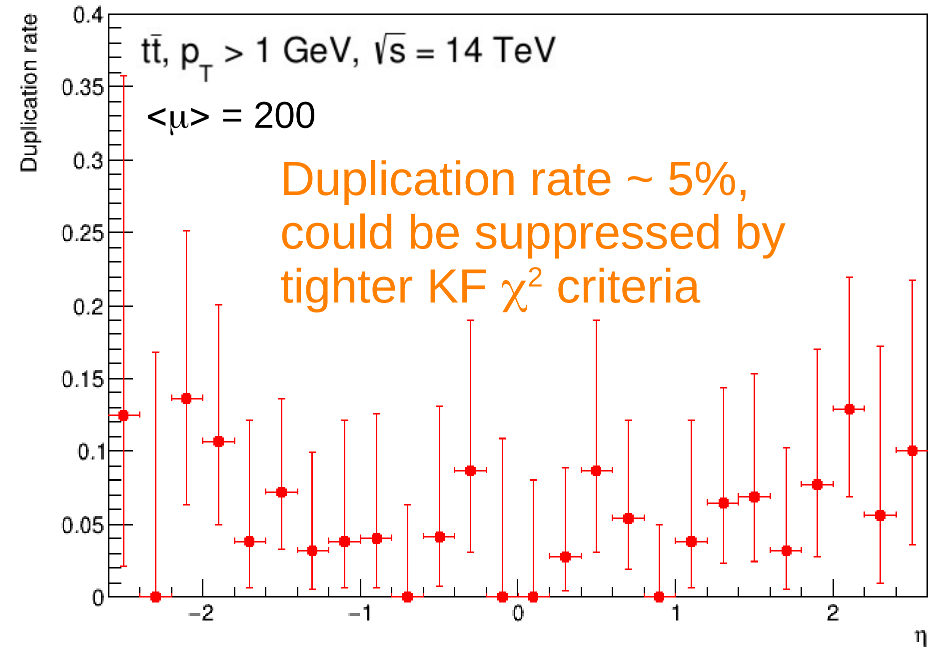
Efficiency vs. η



Fake rate vs. η



Duplication rate vs. η



Vertex finding/fitting

- Various vertexing tools have been transcribed from ATLAS vertexing algorithms with performance well validated against ATLAS SW
  - It might be interesting to explore new techniques
- Two approaches:
  - Iterative fitting-after-finding
    - **Iterative Vertex Finder (IVF)** (used at ATLAS Run-2)
  - Finding-through-fitting
    - **Adaptive Multi-Vertex Finder (AMVF)** (to be used at ATLAS Run-3)

Portable tools used in IVF and AMVF

- Seed finder:
  - Z-Scan Seed Finder
  - Gaussian Track Density Vertex Finder
  - Gaussian Grid Track Density Vertex Finder

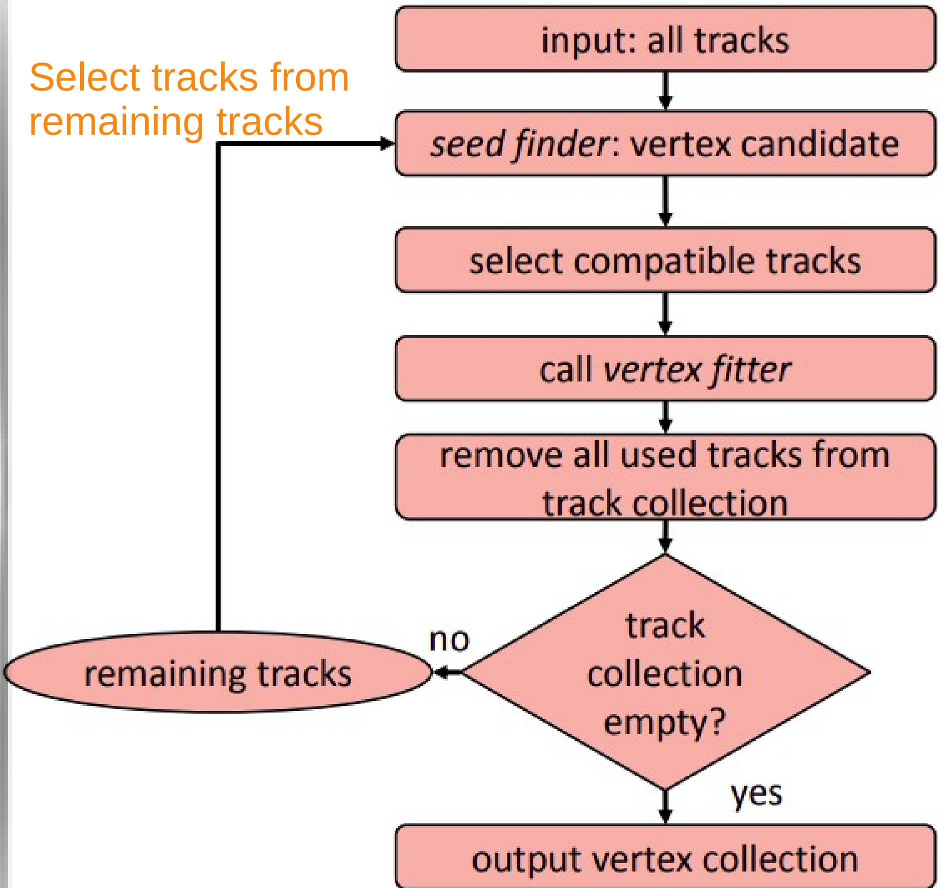
- Vertex fitter
  - Full-Billoir Vertex Fitter
  - Adaptive Multi-Vertex Fitter

- Utilities: track selection, track linearizer, impact point estimator, deterministic annealing tool etc.

Iterative Vertex Finder (IVF)

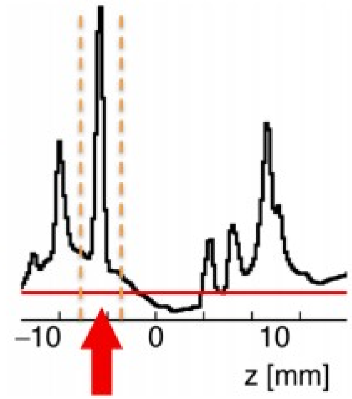
(see B.Schlag's slides)

Select tracks from remaining tracks



ZScanSeedFinder:

- find mode value of all z_0 values
- vertex candidate at position (z_0, 0, 0)

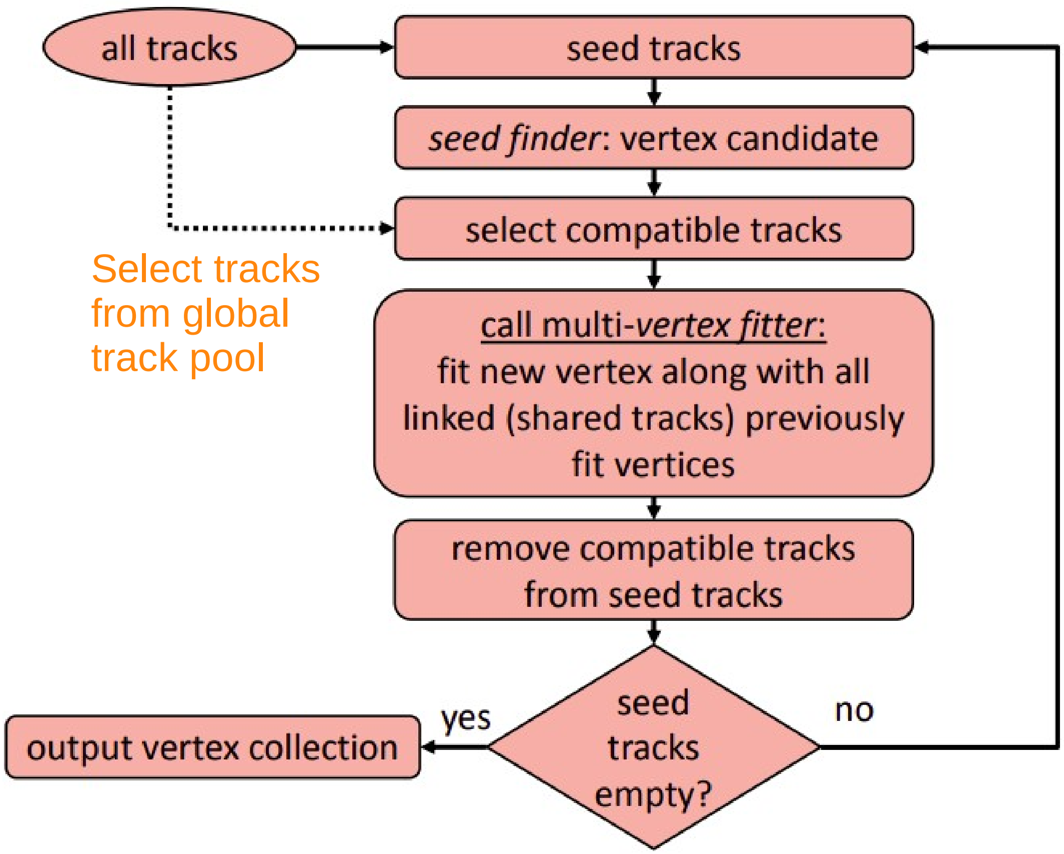


Iterative fitting-after-finding approach:

- iteratively find vertex and fit with compatible tracks
- single track always associated to at most one vertex
- tracks removed from pool after fitting

Adaptive Multi-Vertex Finder (AMVF)

(see B.Schlag's slides)



Gaussian Track Density Seed Finder:

- model each track as 2-dim Gaussian distribution in d_0-z_0-plane around (d_0, z_0)
- find z value of highest track density along z-axis

Adaptive Multi-Vertex Fit:

- weighted adaptive Kalman filter using deterministic annealing scheme
- subject to beamspot and seed constraint
- Simultaneous refit of all vertices connected through a chain of vertices and tracks, with weights:

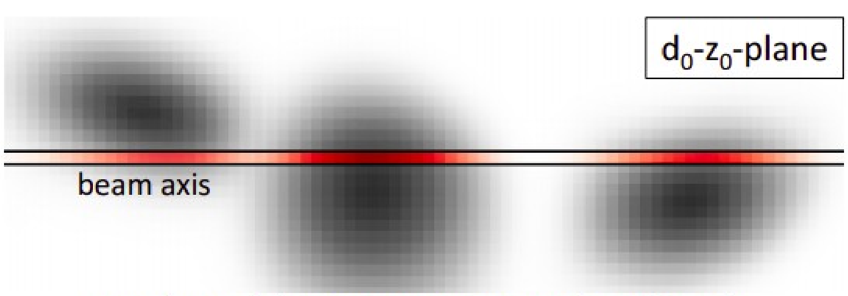
$$\omega_i(\chi_i^2, T) = \frac{e^{-\frac{1}{2}\chi_i^2/T}}{\sum_j e^{-\frac{1}{2}\chi_j^2/T} + e^{-\frac{1}{2}\chi_0^2/T}}$$

tracks can have weights to multiple vertices

→ *Finding-through-fitting approach*

Gaussian Grid Track Density Vertex Seed Finder:

- Model track as 2-dim Gaussian density grid in d_0-z_0-plane
- Interested only in density distribution along beam axis:
  → calculate only track contribution along beam axis (red)
- Superimpose all tracks and find maximum along beam axis



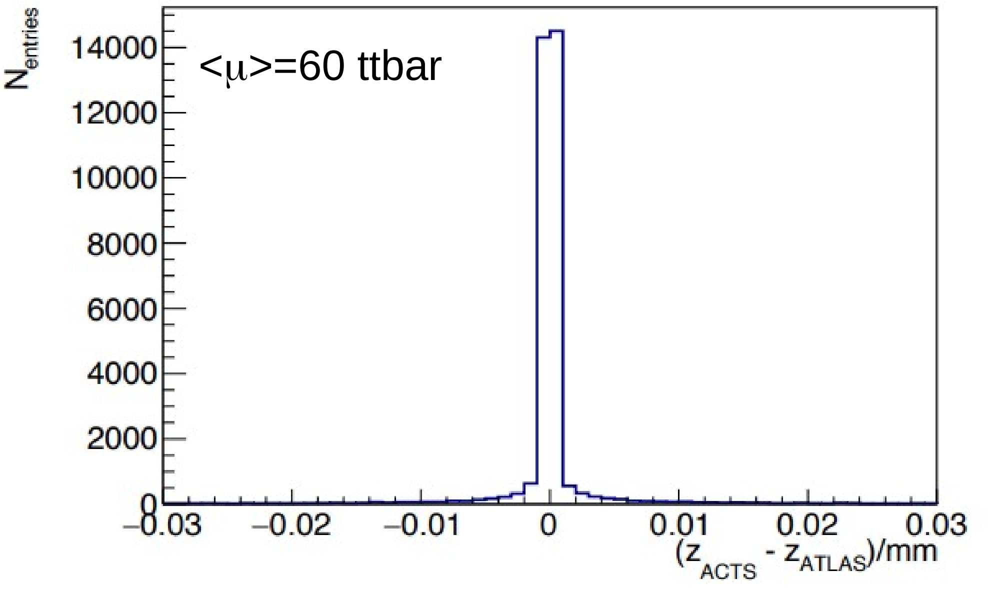
Example: Track density representations of 3 single tracks

Vertexing performance

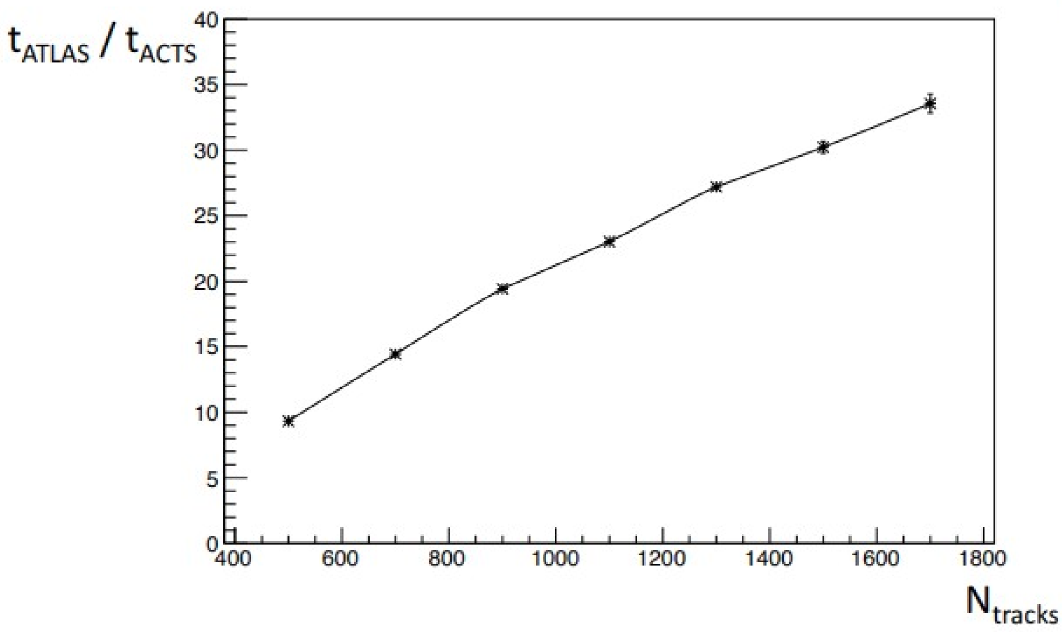
- Vertex position resolution agrees with ATLAS results on mircometer level
- Significant speed-up w.r.t. to ATLAS algorithm

(see B.Schlag's slides)

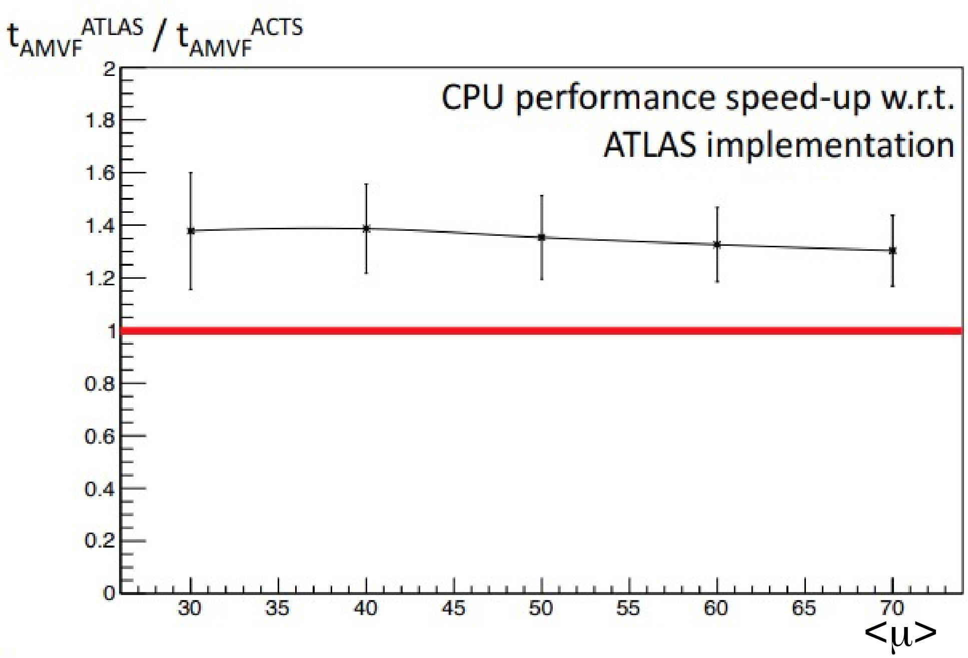
AMVF Vertex z positon resolution



Gaussian Grid Track Density Vertex Finder timing performance



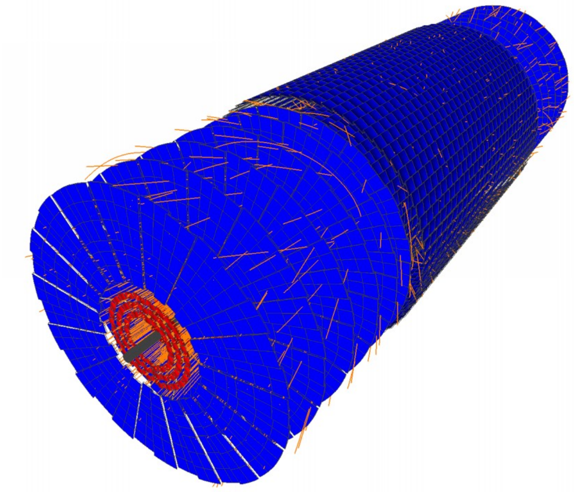
AMVF timing performance



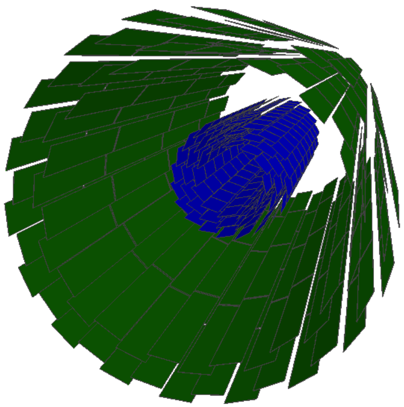
Application to (experiment) detectors

- Detector geometry implemented:
  - TrackML Detector
  - Open Data Detector
  - ATLAS ID+Calo, ATLAS ITK (see C. Allaire's talk)
  - FASER Silicon (see K. Li's talk)
  - CEPC Silicon+TPC (see J. Zhang's talk)
  - sPHENIX Silicon + TPC (see J. Osborn's talk)
  - Belle-II Silicon (see R. Farkas's talk)
  - FCC-hh
- On-going/planned implementation:
  - ATLAS Muon System
  - Belle-II Drift Chamber

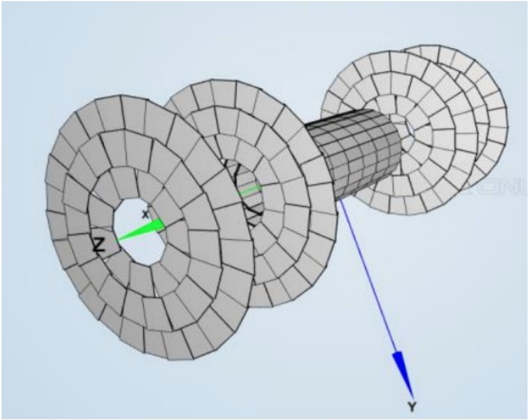
ATLAS ITk



sPHENIX Silicon

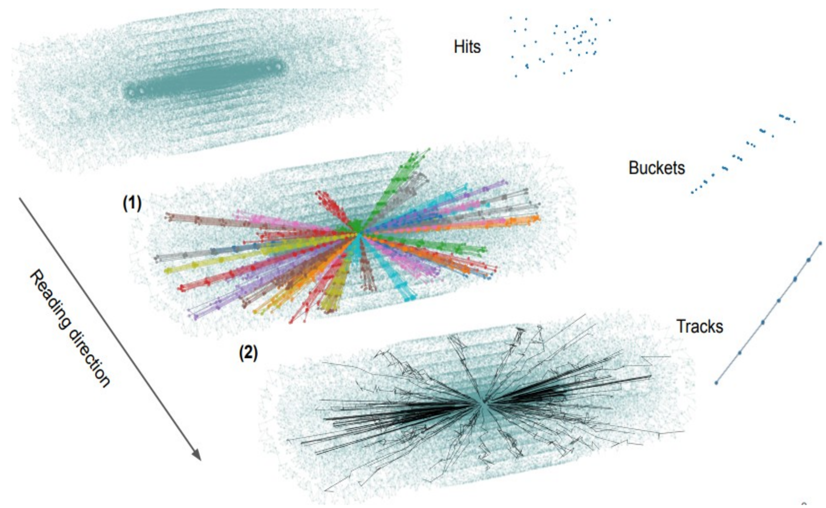


CEPC VTX

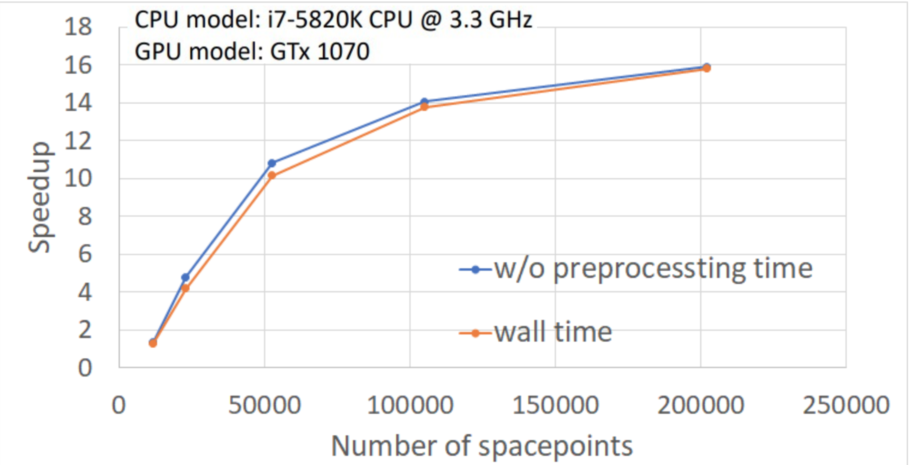


R&D

- Provides support for new tracking techniques R&D (see M. Kiehn's talk)
  - Similarity Hashing and learning
  - Hep.TrkX & Exa.TrkX project
- Parallelism and acceleration facilitated by hardware architecture (see G. Mania's talk)
  - Intra-event parallelism
  - GPUs-accelerated tracking



ACTS seedfinder with CUDA (@NVIDIA GTX 1070)



B. Yeo, C. Legett et. al

Summary

- ACTS has matured a lot as a tracking toolkit over the past year
  - Consolidation of tracking infrastructure, e.g. geometry, propagator, EDM
  - Implementation of new tracking features, e.g. KalmanFilter, CKF, IVF, AMVF
- ACTS is an active R&D platform for new tracking techniques (ML) and hardware architectures
- Growing interest in experiment application& contribution
  - ATLAS ID+Calo, ATLAS ITK, FASER, CEPC, sPHENIX, BELLE-II
- Project mission:
  - Facilitate detector application& software integration, e.g. ATLAS at LHC Run3
  - Provide full tracking solution for future detectors, e.g. ATLAS at LHC Run4

ACTS members have grown from ~27 to 50 in ~ 1 year !

Europe
- CERN (*)
- Université de Genève (*)
- JGU Mainz (*)
- Universität Bonn (*, *)
- Technische Universität München (*, *)
- DESY (*)
- CNRS (*, *)

North America
- UC Berkeley (*)
- LBNL (*, *)
- Stanford Univeristy (*)
- BNL (*)
- University of Washington (*)

Asia
- IHEP, CAS (*)

ATLAS LHCb CEPC BELLE-II
sPHENIX EIC FASER

Welcome to the discussion sessions for mid/long-term Core and R&D development plans!

backup

ACTS members/developers

Andreas Salzburger @asalzbur · acts
Given access 4 years ago

Noemi Calace @ncalace · acts
Given access 4 years ago

Valentin Volkl @vavolkl · acts
Given access 4 years ago

Benedikt Hegner @hegner
Given access 4 years ago

David Rousseau @droussea
Given access 4 years ago

Markus Elsing @elsing
Given access 4 years ago

Paolo Calafiura @calaf
Given access 4 years ago

ATS Jenkins @atsjenkins
Given access 4 years ago

Moritz Kiehn @msmk · acts
Given access 3 years ago

Hadrien Benjamin Grasland @hgraslar · acts
Given access 3 years ago

Stewart Martin-Haugh @smh · acts
Given access 3 years ago

Karolos Potamianos @karolos · acts
Given access 3 years ago

Edward Moyse @emoyse · acts
Given access 3 years ago

Nicholas Styles @nstyles · acts
Given access 3 years ago

Dmitry Emeliyanov @demelian · acts
Given access 3 years ago

Sarka Todorova @nova · acts
Given access 3 years ago

Wolfgang Liebig @liebig · acts
Given access 3 years ago

Shaun Roe @sroe · acts
Given access 3 years ago

Vincenzo Innocente @innocent · acts
Given access 3 years ago

Frank-Dieter Gaede @fgaede · acts
Given access 3 years ago

David Chamont @chamont · acts
Given access 2 years ago

Tobias Golling @golling · acts
Given access 2 years ago

Robert Johannes Langenberg @rlangenl
Given access 2 years ago

Felice Pantaleo @fpantale · acts
Given access 2 years ago

Marco Rovere @rovere · acts
Given access 2 years ago

Paul Gessinger-Befurt @pagessin
Given access 2 years ago

Julia Hrdinka @jhrdinka Blocked
Given access 2 years ago

Fabian Klimpel @fklimpel
Given access 2 years ago

John Smith @jrsmith Blocked · acts
Given access 1 year ago

Shih-Chieh Hsu @schsu · acts
Given access 1 year ago

Heather Gray @hgray · acts
Given access 1 year ago

Lauren Alexandra Tompkins @tompkins · acts
Given access 1 year ago

Jin Zhang @jinz · acts
Given access 1 year ago

Xiaocong Ai @xai 2FA · acts
Given access 1 year ago

Simone Pagan Griso @spagan · acts
Given access 1 year ago

Gang Zhang @gang · acts
Given access 1 year ago

Tim Adye @adye · acts
Given access 10 months ago

Ke Li @keli · acts
Given access 8 months ago

Jessica Leveque @leveque · acts
Given access 6 months ago

Corentin Allaire @corentin · acts
Given access 5 months ago

Georgiana Mania @gmania · acts
Given access 4 months ago

Laurent Roger Igor Basara @lbasara · acts
Given access 4 months ago

Stanislava Sevova @ssevova
Given access 4 months ago

Florian Urs Bernlochner @fbernloc · acts
Given access 4 months ago

Peter Alan Steinberg @steinber · acts
Given access 3 months ago

Vincent Pascuzzi @vpascuzz
Given access 2 months ago

Bastian Schlag @bschlag
Given access 2 months ago

Charles Leggett @leggett · acts
Given access 1 month ago

Tomohiro Yamazaki @toyamaza · acts
Given access 1 month ago

Ralf Farkas @rafarkas · acts
Given access 1 week ago

50 ACTS Members (~1 year ago: 27)

Experiments:

ATLAS

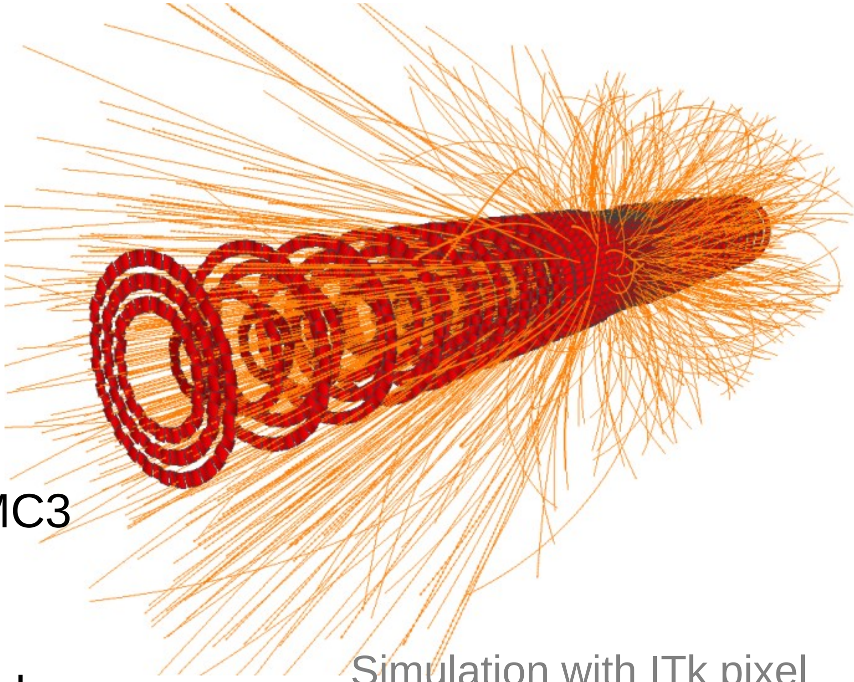
LHCb

CEPC

FASER

Belle-II

Fast Simulation

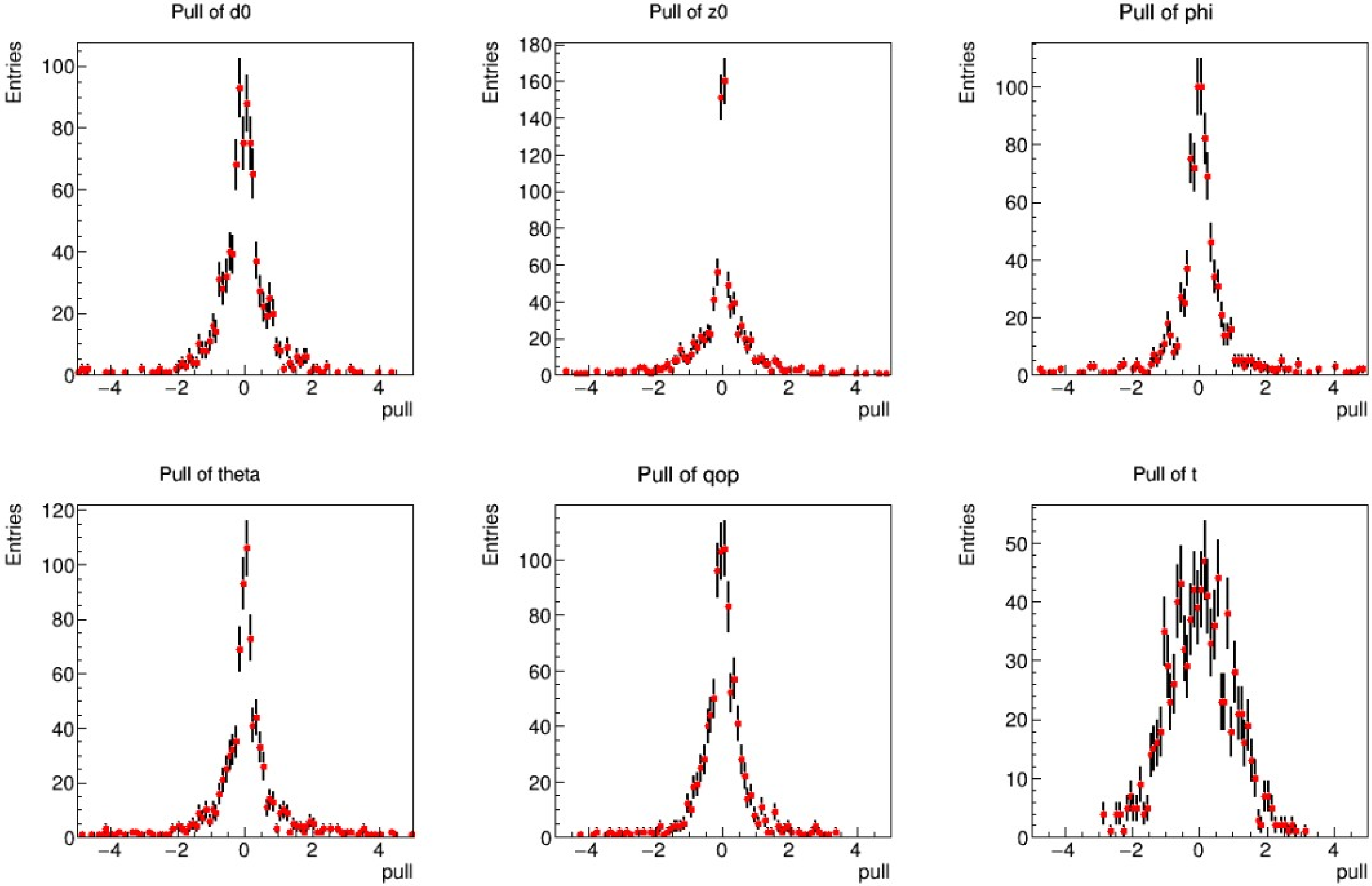


Simulation with ITk pixel

- Event Data Model
  - Concrete particle and hit type
  - Flat, sorted data container for particle and hit
- Event generator
  - Particle Gun and interface to Pythia8 and HepMC3
- Detector material effects modeling
  - Energy loss and multiple scattering are validated
  - Hadronic interaction is currently reparameterised
  - Foreseen use of Geant4 for particle decay
  - Photon Conversion and positron annihilation are missing
- Detector response emulation (i.e. digitization)
  - Including pseudo-realistic clustering model (without clustering merging yet)
- Work-in-progress to use Json-based geometry/segmentation/material information at fast simulation chain

KF fitting resolution

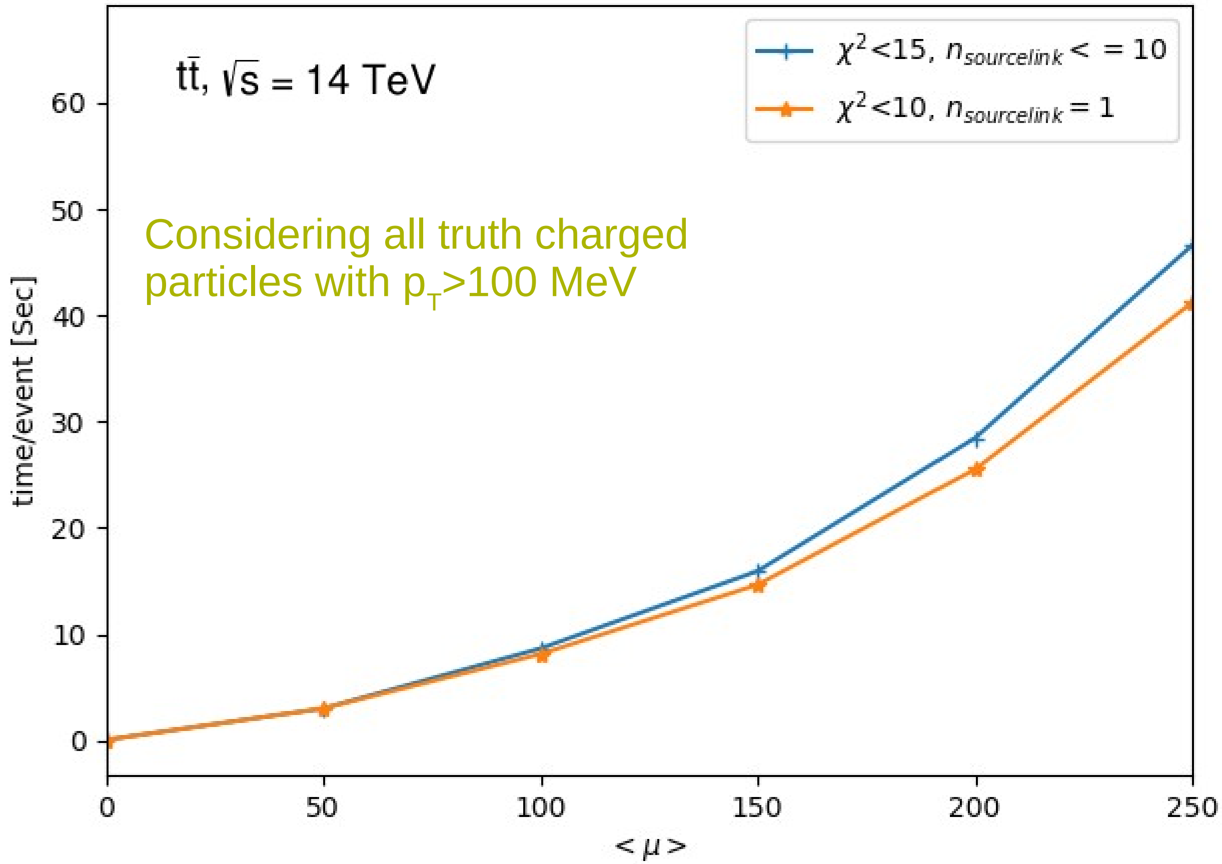
- Single muon, $0.1<p_T<100$ GeV, $|\eta|<2.4$
- TrackML detector, ATLAS B field



CKF timing test

CKF time/event vs. <μ>

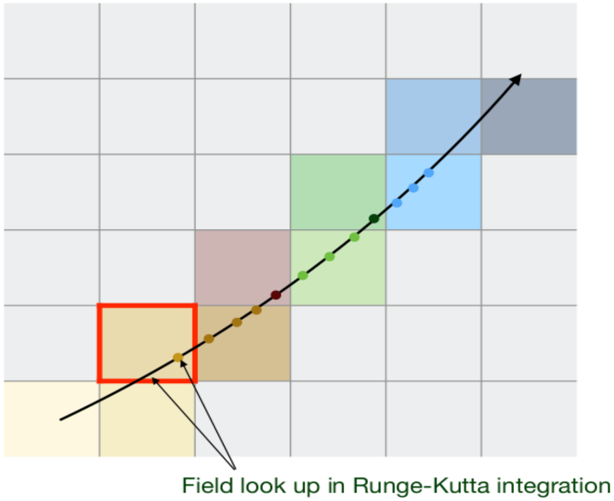
TrackML detector, ATLAS B fileld



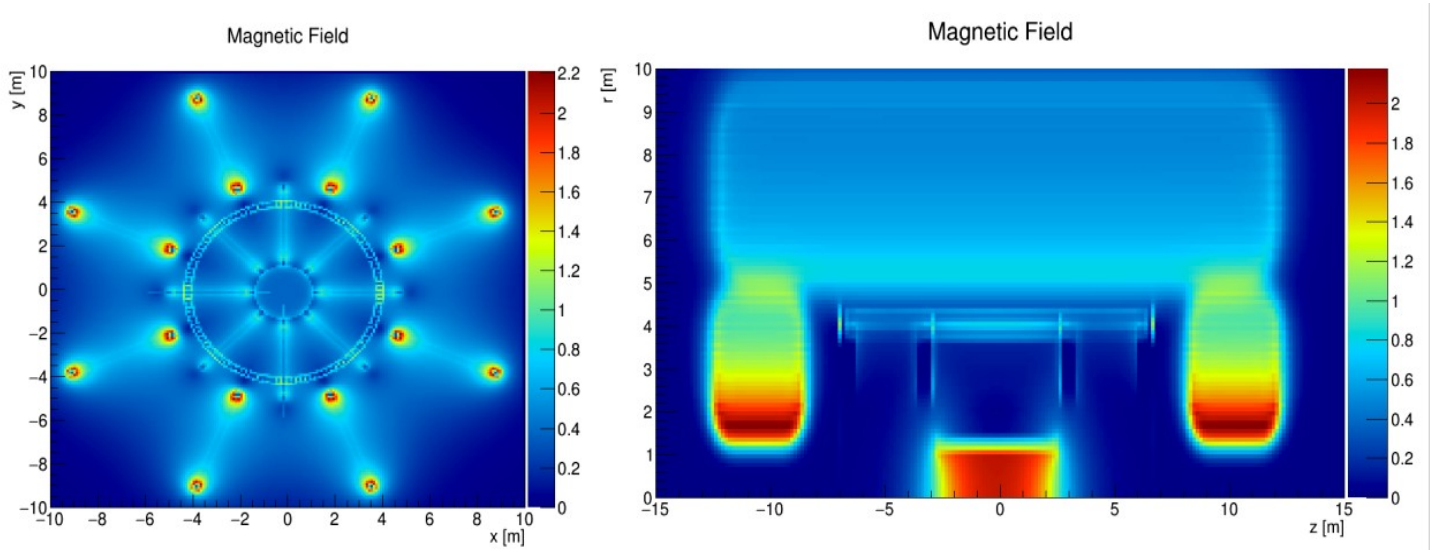
Each filtering step needs to loop over all the source links on the surface for the source link selection, Could be speed-up by fast source link selection

Magnetic field

- Simulation and reconstruction needs extensive lookup of magnetic field
- Cache of field value could make the access less expensive
  - Facilitate repetitive access to similar locations
- In ACTS, the cache is passed between magnetic field service and client via client function argument
  - Cache is thread-local thus thread-safe



ATLAS Magnetic field in ACTS



Contextual alignment and calibration

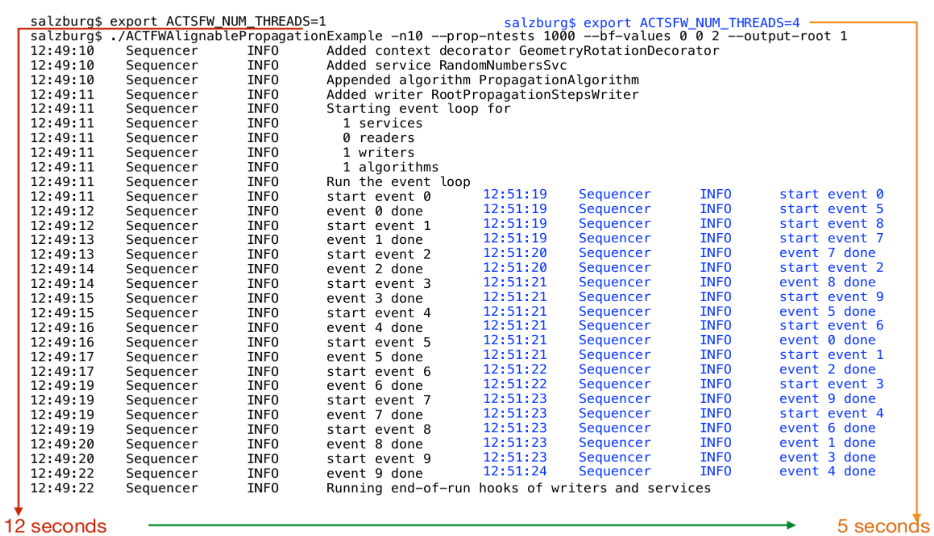
- An AlgorithmContext object is used to support on-the-fly event-dependent changes of alignment/ calibration/magnetic field

```
size_t                 algorithmNumber;  ///< Unique algorithm identifier
size_t                 eventNumber;      ///< Unique event identifier
WhiteBoard&            eventStore;       ///< Per-event data store
Acts::GeometryContext  geoContext;       ///< Per-event geometry context
Acts::MagneticFieldContext
                          magFieldContext;  ///< Per-event magnetic Field context
Acts::CalibrationContext calibContext;     ///< Per-event calbiration context
```

- Concept of contextual alignment and calibration has been validated

Propagation tests with contextual alignment

(Different alignment every single event, $n_{threads} = 4$)

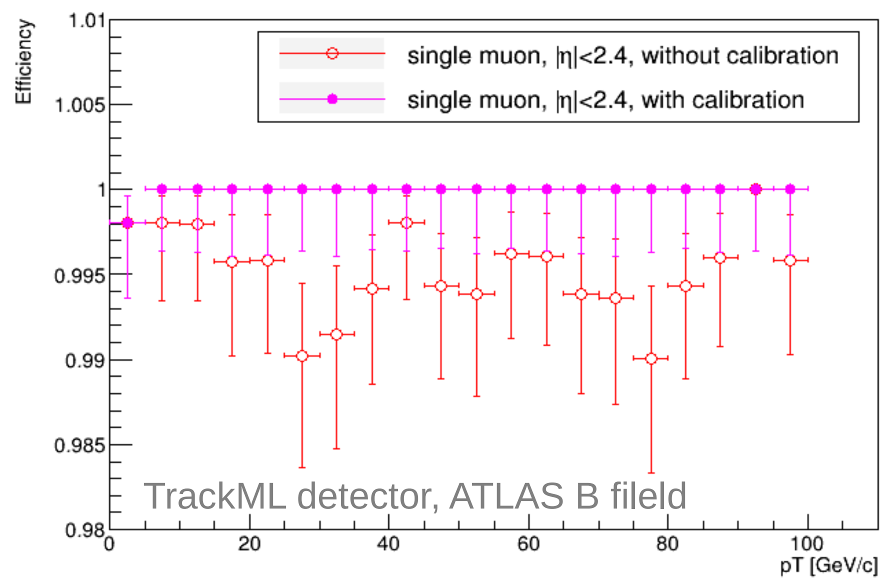


```
salzburg$ export ACTSFW_NUM_THREADS=1
salzburg$ ./ACTFWAlignablePropagationExample -n10 --prop-ntests 1000 --bf-values 0 0 2 --output-root 1
12:49:10    Sequencer    INFO    Added context decorator GeometryRotationDecorator
12:49:10    Sequencer    INFO    Added service RandomNumbersSvc
12:49:10    Sequencer    INFO    Appended algorithm PropagationAlgorithm
12:49:11    Sequencer    INFO    Added writer RootPropagationStepsWriter
12:49:11    Sequencer    INFO    Starting event loop for
12:49:11    Sequencer    INFO      1 services
12:49:11    Sequencer    INFO      0 readers
12:49:11    Sequencer    INFO      1 writers
12:49:11    Sequencer    INFO      1 algorithms
12:49:11    Sequencer    INFO    Run the event loop
12:49:11    Sequencer    INFO    start event 0
12:49:12    Sequencer    INFO    event 0 done
12:49:12    Sequencer    INFO    start event 1
12:49:13    Sequencer    INFO    event 1 done
12:49:13    Sequencer    INFO    start event 2
12:49:14    Sequencer    INFO    event 2 done
12:49:14    Sequencer    INFO    start event 3
12:49:15    Sequencer    INFO    event 3 done
12:49:15    Sequencer    INFO    start event 4
12:49:16    Sequencer    INFO    event 4 done
12:49:16    Sequencer    INFO    start event 5
12:49:17    Sequencer    INFO    event 5 done
12:49:17    Sequencer    INFO    start event 6
12:49:19    Sequencer    INFO    event 6 done
12:49:19    Sequencer    INFO    start event 7
12:49:19    Sequencer    INFO    event 7 done
12:49:19    Sequencer    INFO    start event 8
12:49:20    Sequencer    INFO    event 8 done
12:49:20    Sequencer    INFO    start event 9
12:49:22    Sequencer    INFO    event 9 done
12:49:22    Sequencer    INFO    Running end-of-run hooks of writers and services
```

```
salzburg$ export ACTSFW_NUM_THREADS=4
12:51:19    Sequencer    INFO    start event 0
12:51:19    Sequencer    INFO    start event 5
12:51:19    Sequencer    INFO    start event 8
12:51:19    Sequencer    INFO    start event 7
12:51:20    Sequencer    INFO    event 7 done
12:51:20    Sequencer    INFO    start event 2
12:51:21    Sequencer    INFO    event 8 done
12:51:21    Sequencer    INFO    start event 9
12:51:21    Sequencer    INFO    event 5 done
12:51:21    Sequencer    INFO    start event 6
12:51:21    Sequencer    INFO    event 0 done
12:51:21    Sequencer    INFO    start event 1
12:51:22    Sequencer    INFO    event 2 done
12:51:22    Sequencer    INFO    start event 3
12:51:23    Sequencer    INFO    event 9 done
12:51:23    Sequencer    INFO    start event 4
12:51:23    Sequencer    INFO    event 6 done
12:51:23    Sequencer    INFO    event 1 done
12:51:23    Sequencer    INFO    event 3 done
12:51:24    Sequencer    INFO    event 4 done
```


Track fitting test with contextual calibration

(Different calibration every 10 events, $n_{threads}=8$)



The **detector**

Defined a Phase-2 like detector

- full silicon detector with realistic resolution, material budget, magnetic field
- composed as Pixel, short strip, long strip
- restricted to size of tracking volume to $|\eta| < 3$