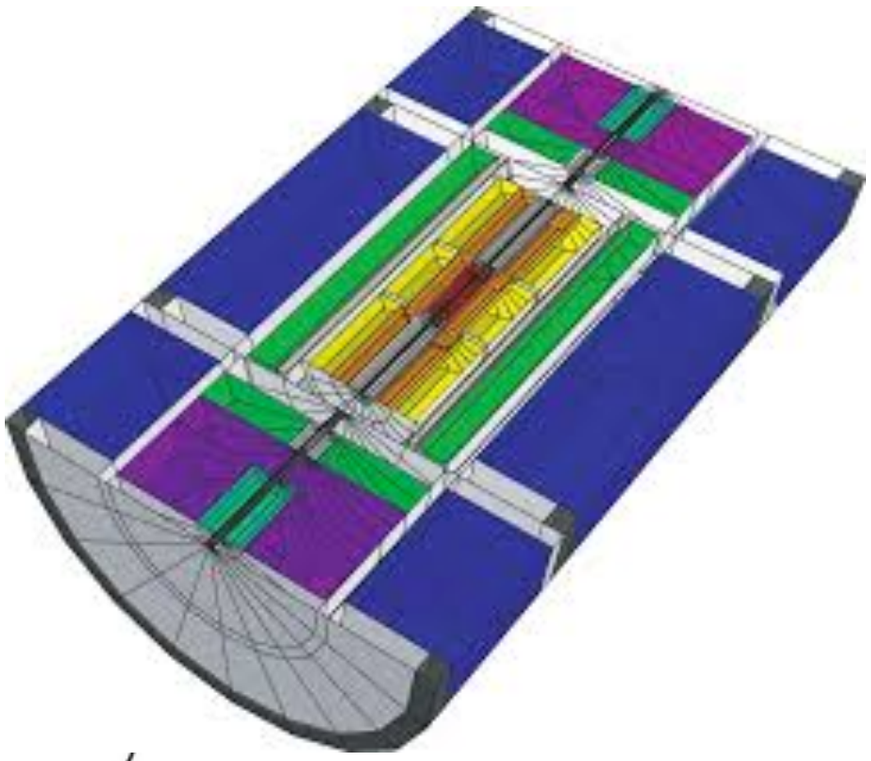


A(TLAS) Common Tracking Software

A. Salzburger (CERN) - for the ACTS developers

Tracking in Run 1 & 2

- Inner Tracker reconstruction became CPU dominating factor in Run-1 of event reconstruction
  - combinatorial scaling with pile-up as main problem

- CPU needs for pile-up scenarios for Run-1 / Run-2 could be met by code optimisation
  - Both ATLAS, CMS achieved similar gains in code clean-up

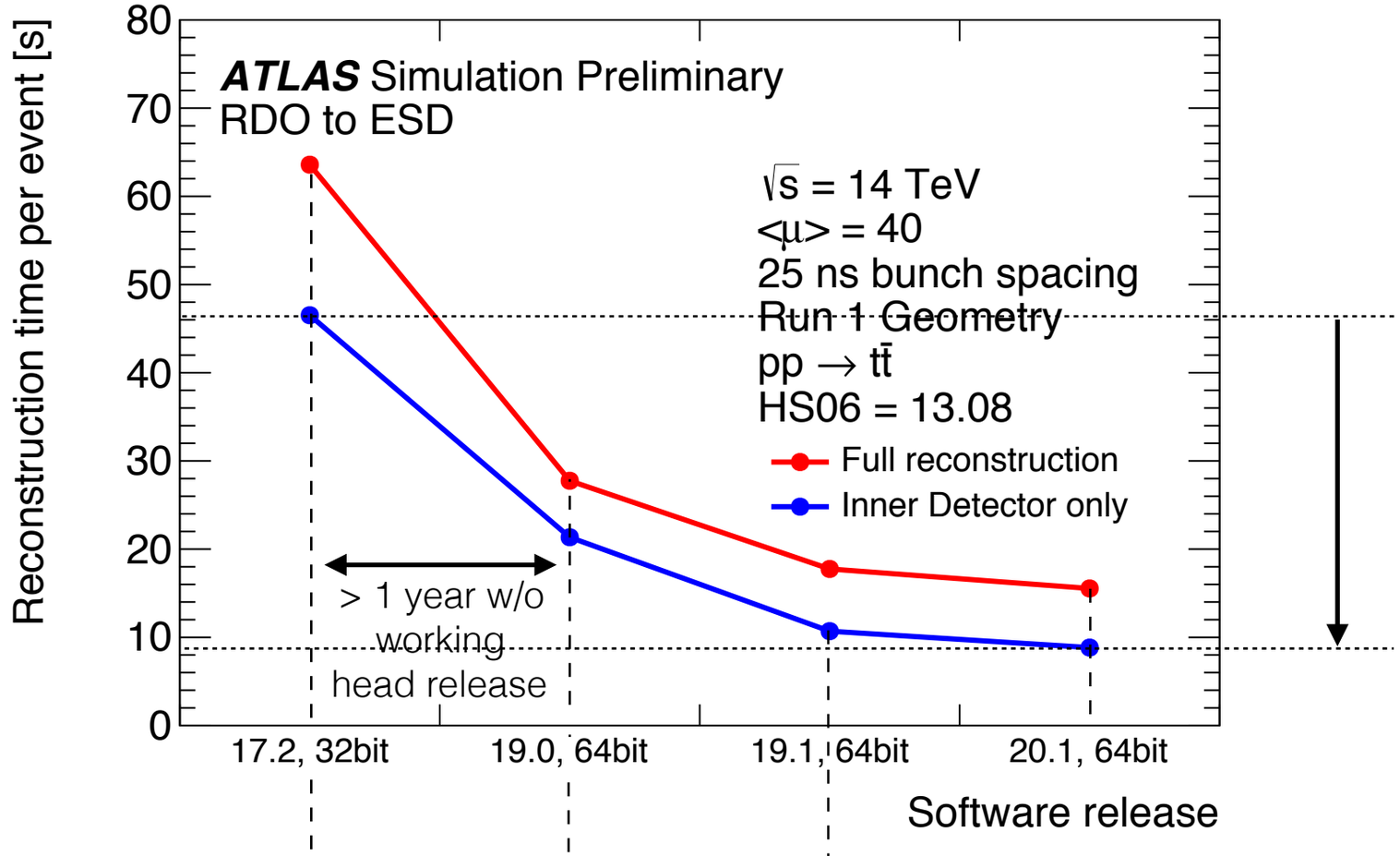
- Phase-2 studies show that current software is not suitable for production setup at pile-up of ~200
  - end of Moore's law does not help …

- R&D for track reconstruction needed but often difficult
  - high entry hurdles for newcomers
  - not really example packages around

Run-2 optimisation

very similar gain achieved in CMS

- CPU performance driven SW campaign to optimise the ID tracking



> 5 CPU gain,
no loss of physics performance

A. Salzburger, CHEP2015

LHC Run-1 | **LS 1** | **LHC Run-2**

- Eigen/algebra tests
- integration
- mag field
- pattern updates
- TIDE changes

Tracking SW workshop "Run-2 planning" Nov 2012

Tracking SW workshop "LS 1 Mid-term" Oct 2013

March/April 2015 "Run-2 release frozen"

A lot of activity & needs in Tracking community

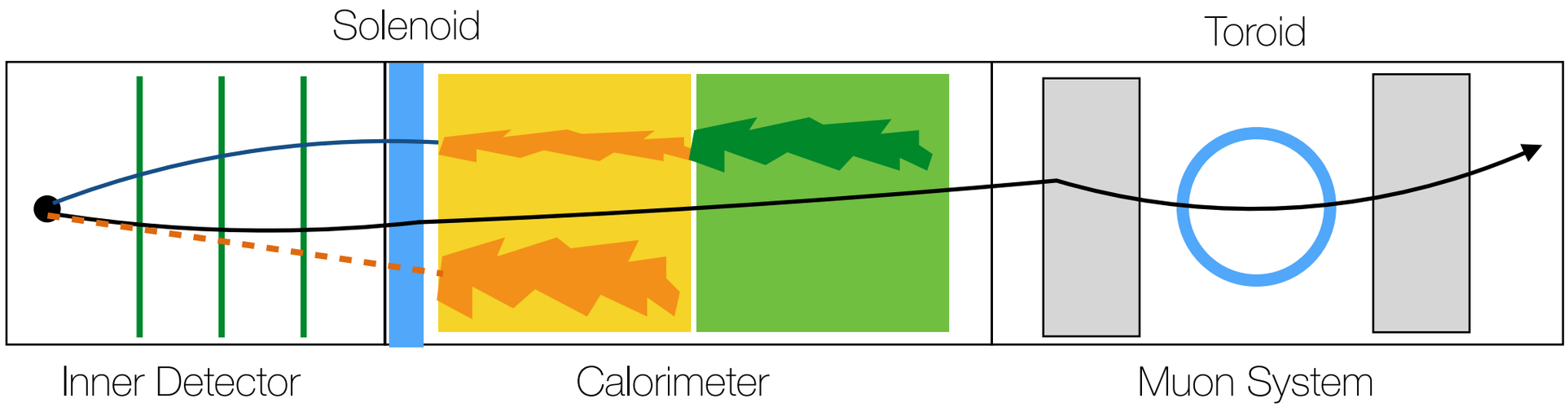
- High activity in the era of Machine Learning
  - LHC experiments use ML in track reconstruction mainly for classification
  - current ML Tracking challenge aims to open this to pattern recognition in a broader sense

- HL-LHC as first real stress-test for extreme luminosity reconstruction
  - Upgrade TDRs to be prepared in the coming years
  - need thorough simulation and reconstruction studies

- New detector concepts and milestone studies ahead
  - FCC design study (due in 2018)

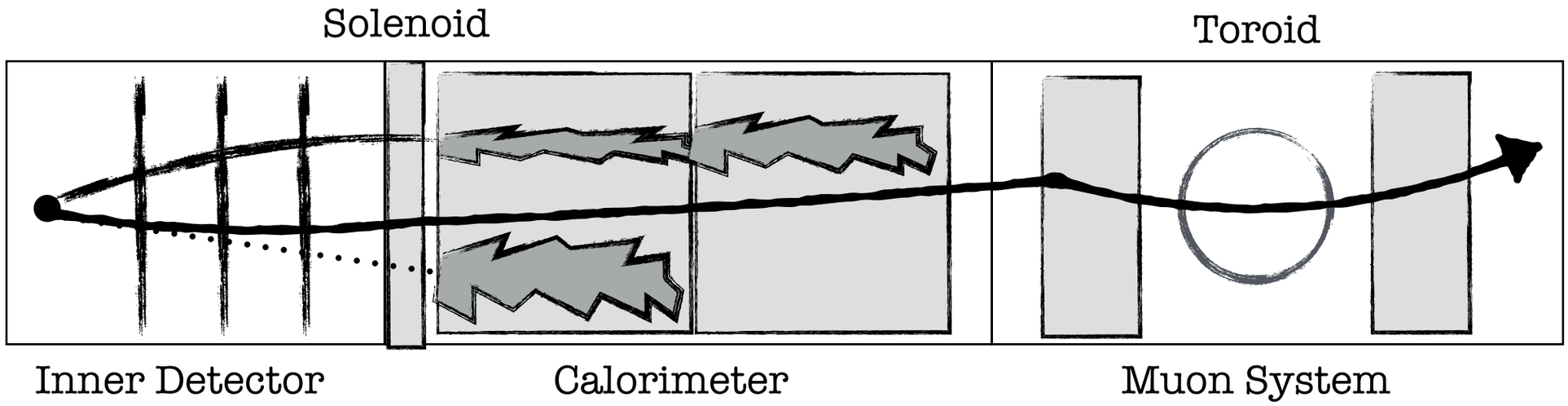
- R&D needed while community is not very large
  - idea to evolve some of the LHC SW into a Tracking toolkit

The ATLAS Tracking History



- A simplified “Tracker” view of ATLAS
  - two precision tracking systems having
    very different magnetic field setups
    very different detecting technologies
    very different dimensions
  - some lump of material in between

The **FORTRAN** times



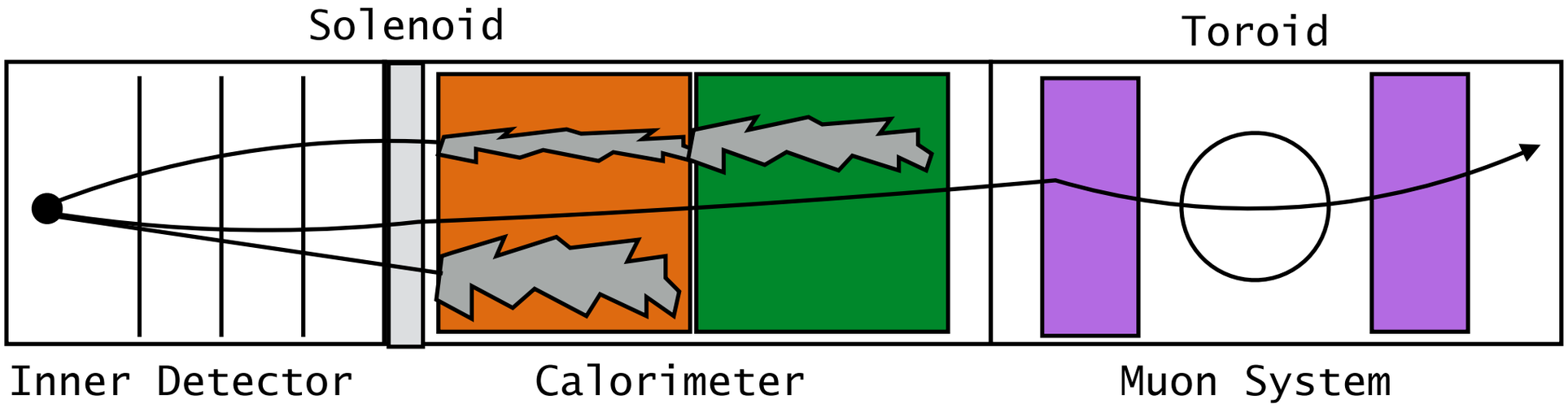
2 monolithic track reconstruction programs in FORTRAN:

- helical track model
- internal event data model
- from pattern recognition to track fitting within one program

something nasty you have to deal with

1 main track reconstruction programs in FORTRAN with a chi2 combination with the Inner Detector reconstruction, no combined fit

The **C** times / Early ATHENA days



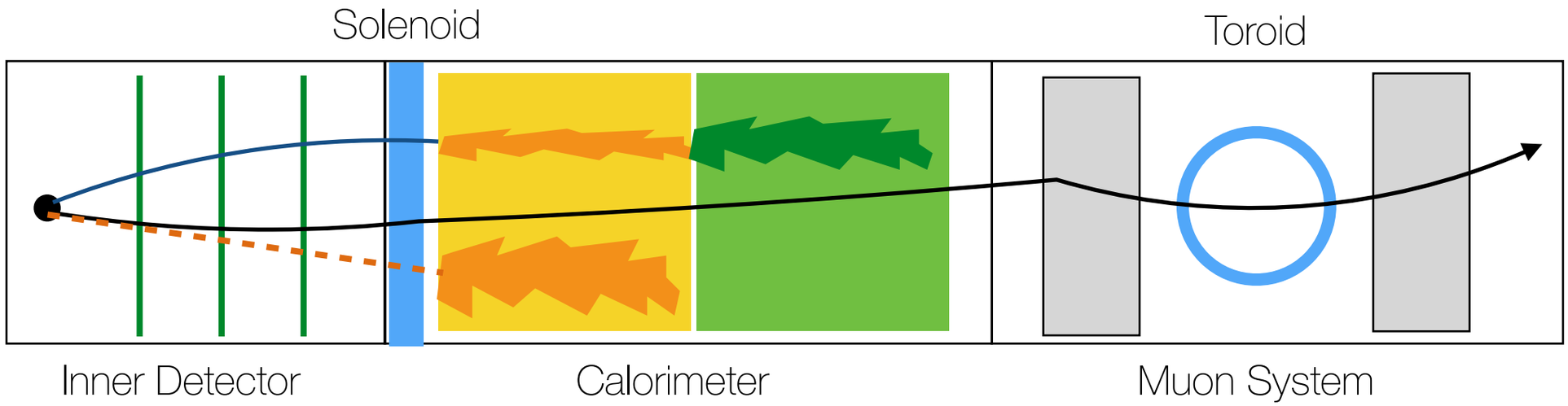
2 monolithic track reconstruction programs in FORTRAN with semicolons at the end
- helical track model
- internal event data model
- from pattern recognition to track fitting within one program

something nasty you have to deal with, 2 independent models to do so

2 main track reconstruction programs, one in FORTRAN, on C, with either a chi2 combination with the Inner Detector reconstruction, or a combined track fit

The RTF report / ATLAS Common Tracking

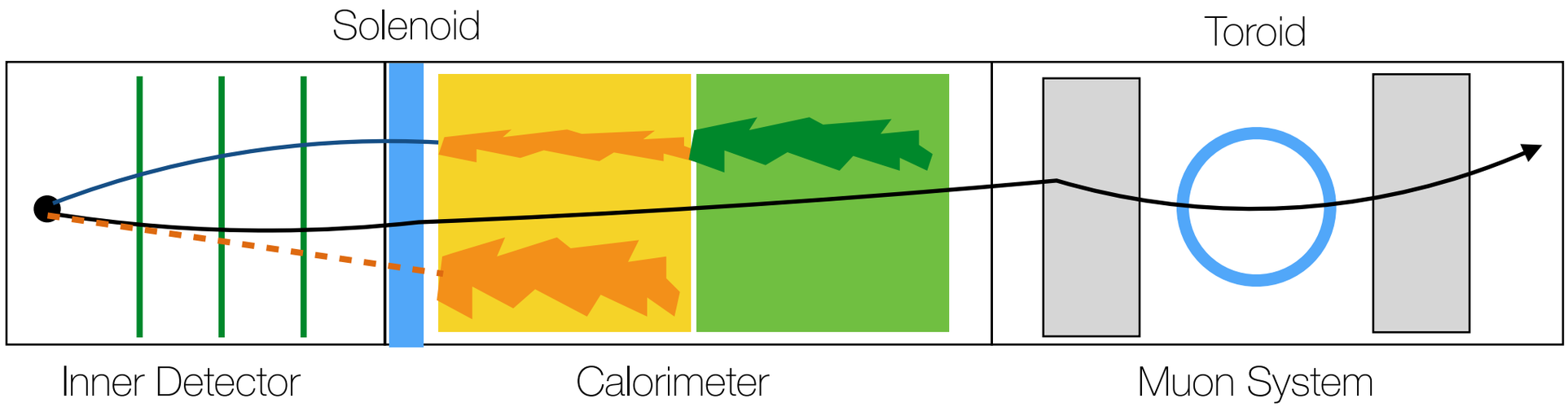
ATL-SOFT-2003-010



- ATLAS Reconstruction Task Force (RTF) review & recommendations
  - final report issued in 2003, recommendations to restructure ATLAS software
- A simplified “Tracker” view of ATLAS
  - two precision tracking systems having

very different magnetic field setups	→	field-agnostic parameterisation
very different detecting technologies	→	technology-agnostic high level tracking
very different dimensions	→	re-calibration on demand
- some lump of material in between	→	integration of calorimeter into tracking

Current structure - from reality to repository



Tracking					
Geometry (TrkDetDescr)	Event Data Model (TrkEvent)	A Extrapolation (TrkExtrapolation)	A Fitting (TrkFitter)	A Calibration, general (TrkTools)	A Alignment (TrkAlignment)

InnerDetector		
EDM (InDetRecEvent)	A Geometry (InDet TrackingGeometry)	A Tools (InDetRecTools)

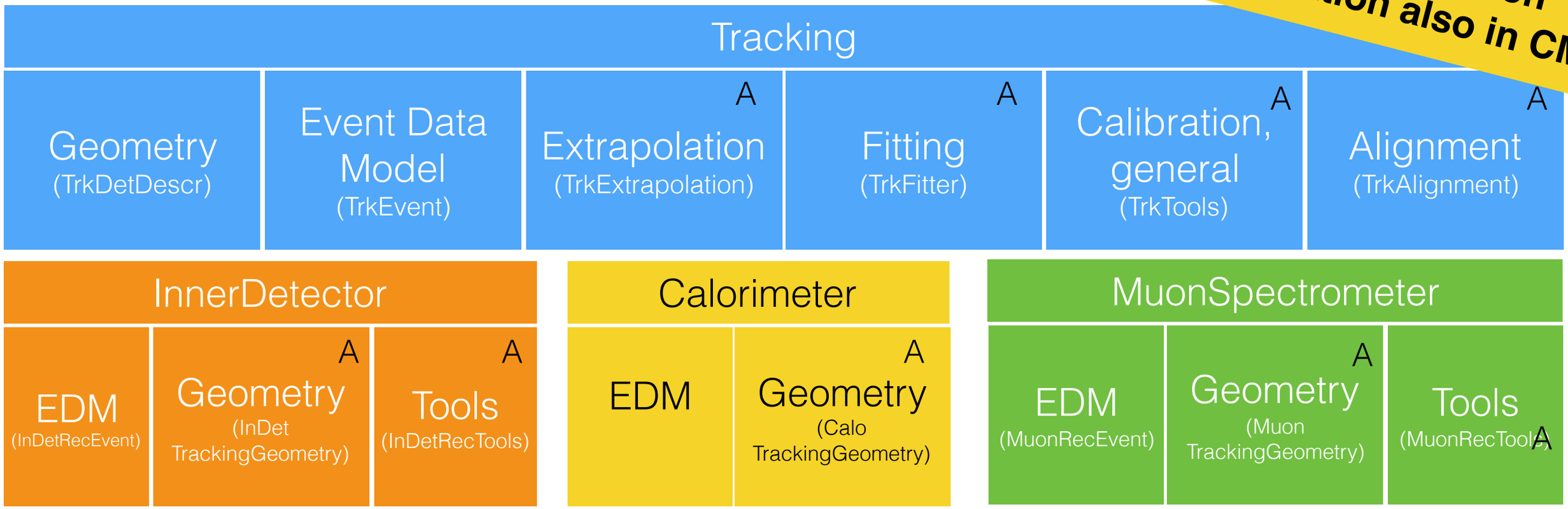
Calorimeter	
EDM	A Geometry (Calo TrackingGeometry)

MuonSpectrometer		
EDM (MuonRecEvent)	A Geometry (Muon TrackingGeometry)	A Tools (MuonRecTools)

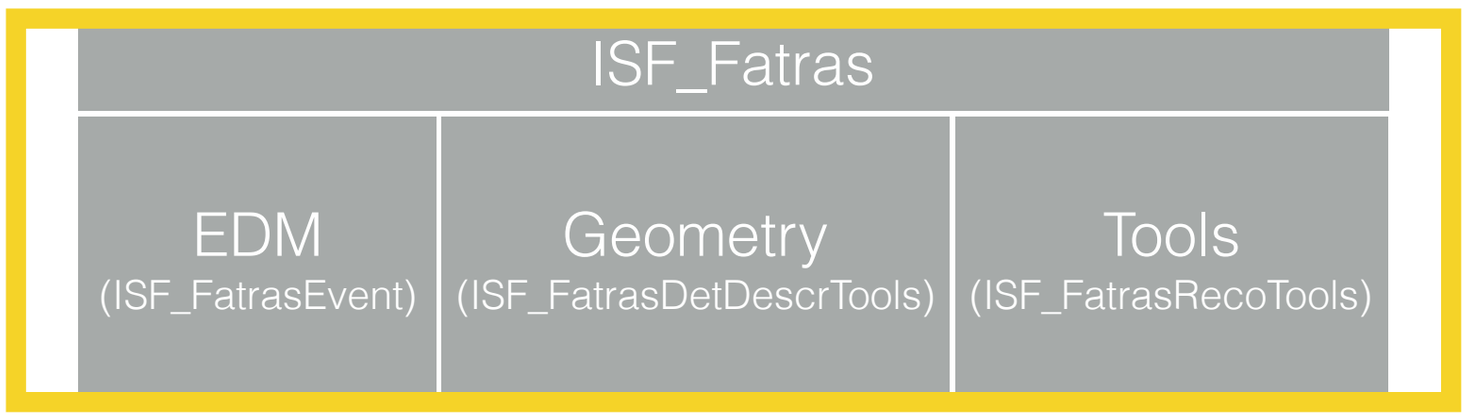
A … embedded in Gaudi/Athena structure with AlgTools/Algorithms/Services

Extending the ATLAS Tracking S

concept of using reconstruction geometry for fast simulation also in CMS



- Within Phase-2 upgrade we developed a fast detector prototyping
  - extended ATLAS tracking EDM/geometry with generic XML based builders
  - could easily run fast track simulation and refitting without actually building ATLAS (2014/15 in parallel a test study within FCC software context & DD4Hep binding)



ACTS - Why ?

- LHC detector software has really been stress-tested
  - and I think we learned a lot, and we start working on Upgrade/FCC
  - however, our concepts are sometime > 30 years old !

APPLICATION OF KALMAN FILTERING TO TRACK AND VERTEX FITTING

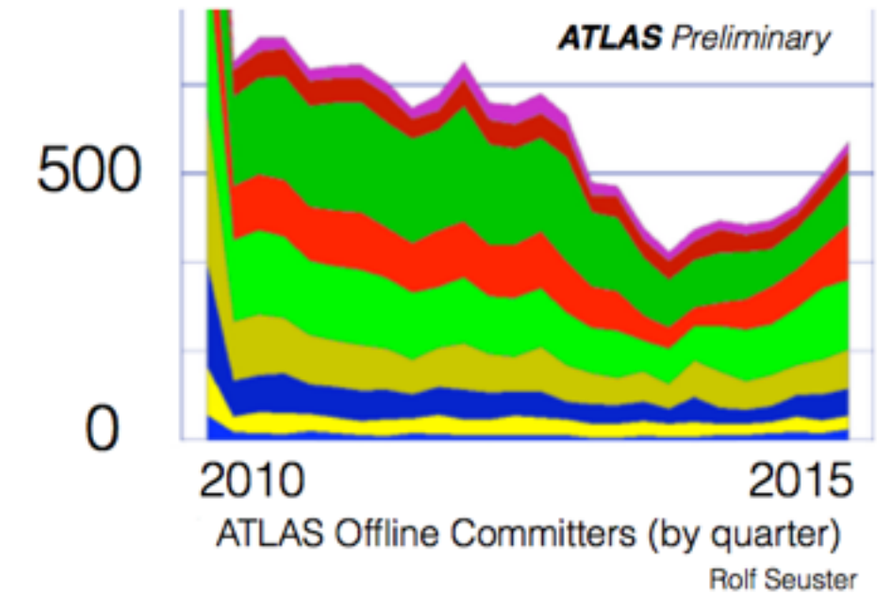
R. FRÜHWIRTH

Institut für Hochenergiephysik der Österreichischen Akademie der Wissenschaften, Vienna, Austria

Received 30 June 1987

- More importantly even

Algorithmic Code Evolution



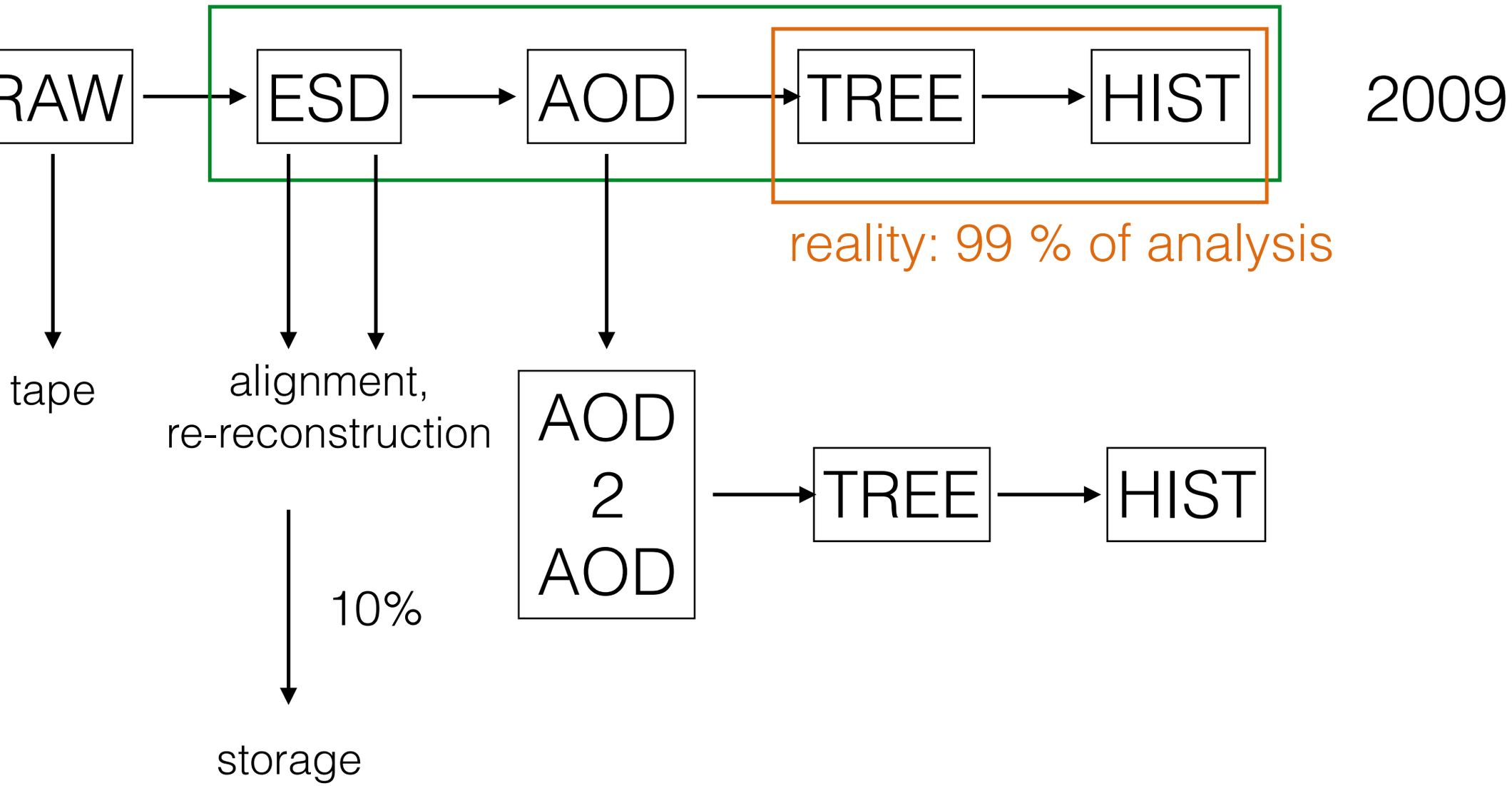
- Highest investment in algorithmic code — O(100M$) for LHC experiments
  - Vast majority of offline packages

[Graeme Stewart, Evolution of HEP SW, CHEP2015]

Software lessons from Run-1 - EDM (1)

- The reconstruction event data stays 99% internal
  - this was not what we planned for:

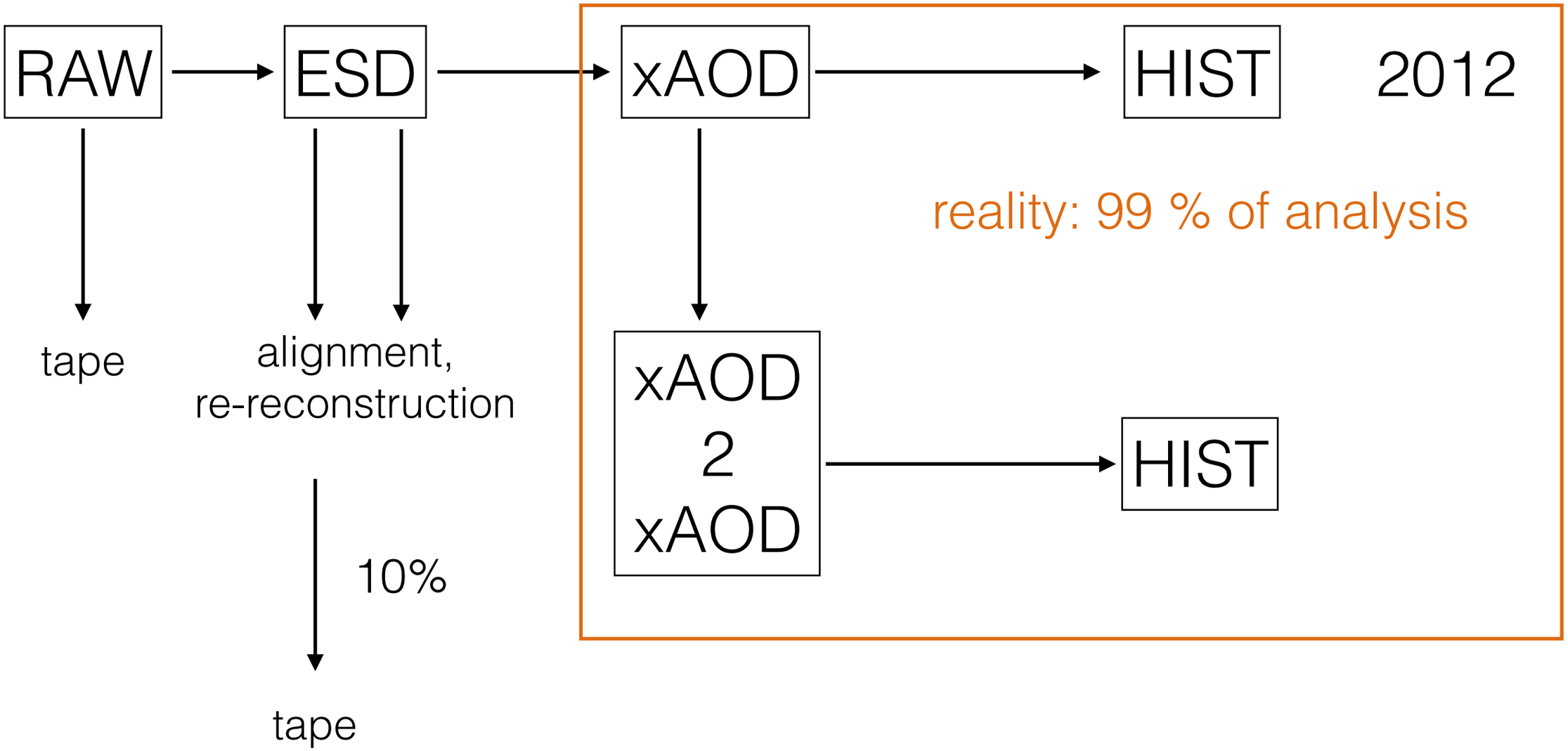
plan: available for analysis



this led to an evolution of the ATLAS EDM

Software lessons from Run-1 - EDM (2)

- Introduction of xAOD which is fully ROOT-readable



"life without ESD"

Summary of Lessons learned from Run-1 (ATLAS)

- Event Data Model (EDM) was too complicated and over designed
  - some compromises taken in order to guarantee usability for users that never existed
  - Tracking EDM users are experts per se: vertexing, alignment, particle flow
  - analyses use to overwhelming extend the particle view/representation
  - Tracking EDM was not suited for concurrency (e.g. no data locality, etc.)

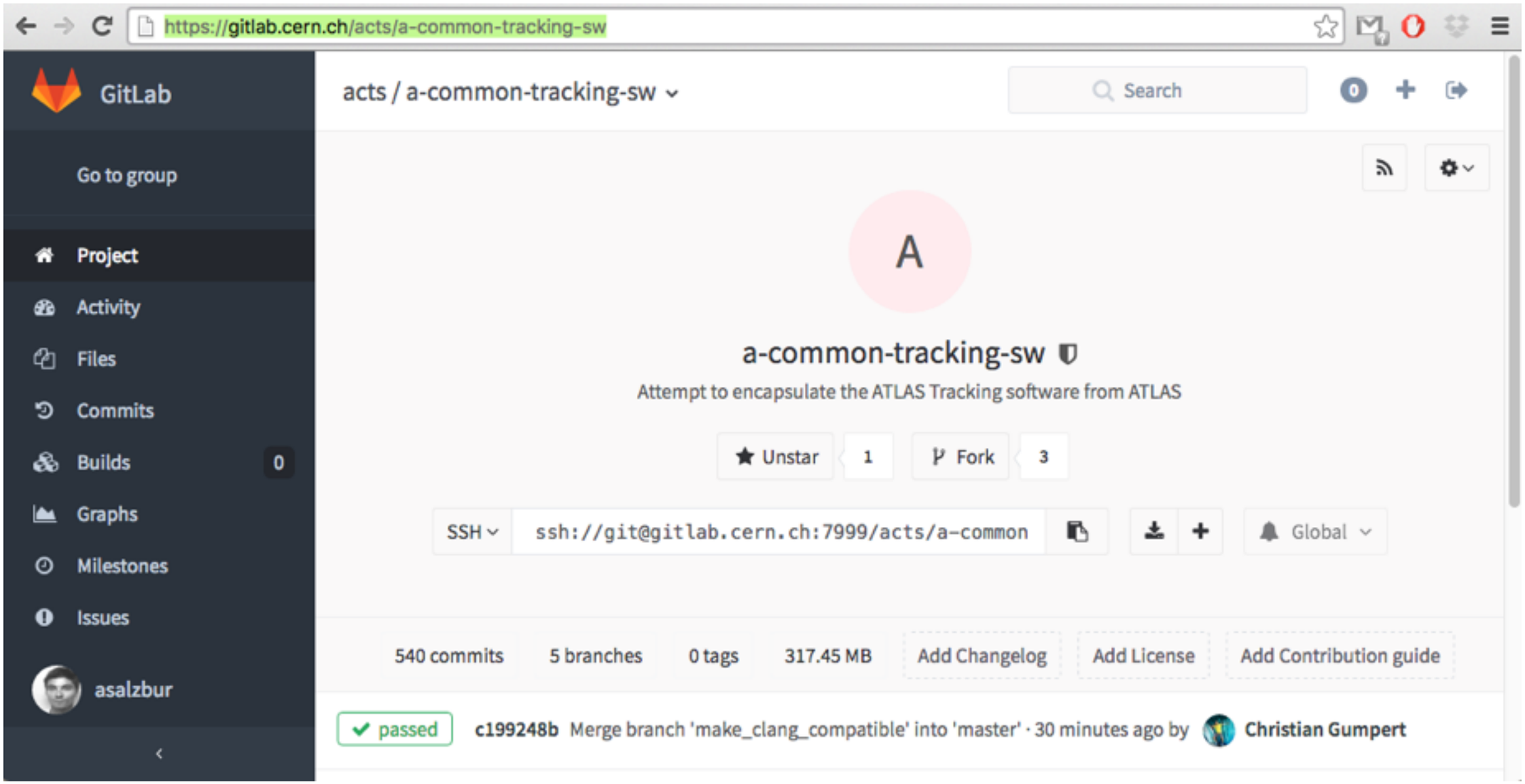
- Tracking code was highly non-thread aware
  - needs to be considered from the very design of it

- LS-1 SW campaign did help a lot to identify and mitigate hotspots
  - Will be difficult to do a similar gains with SW clean-up

- Need R&D in algorithms and approaches to support HL-LHC & FCC

ACTS - Core functionality

- ACTS core functionality now decoupled from any framework component:
  - gitlab project: https://gitlab.cern.ch/acts/a-common-tracking-sw



Core content

https://gitlab.cern.ch/acts/a-common-tracking-sw/tree/master/Core/include/ACTS

GitLab

Search

acts / a-common-tracking-sw · Files

Go to group

Project
Activity
Files
Commits
Builds 0
Graphs
Milestones
Issues
Merge Requests 0
Members
Labels
Wiki
Forks

asalzbur

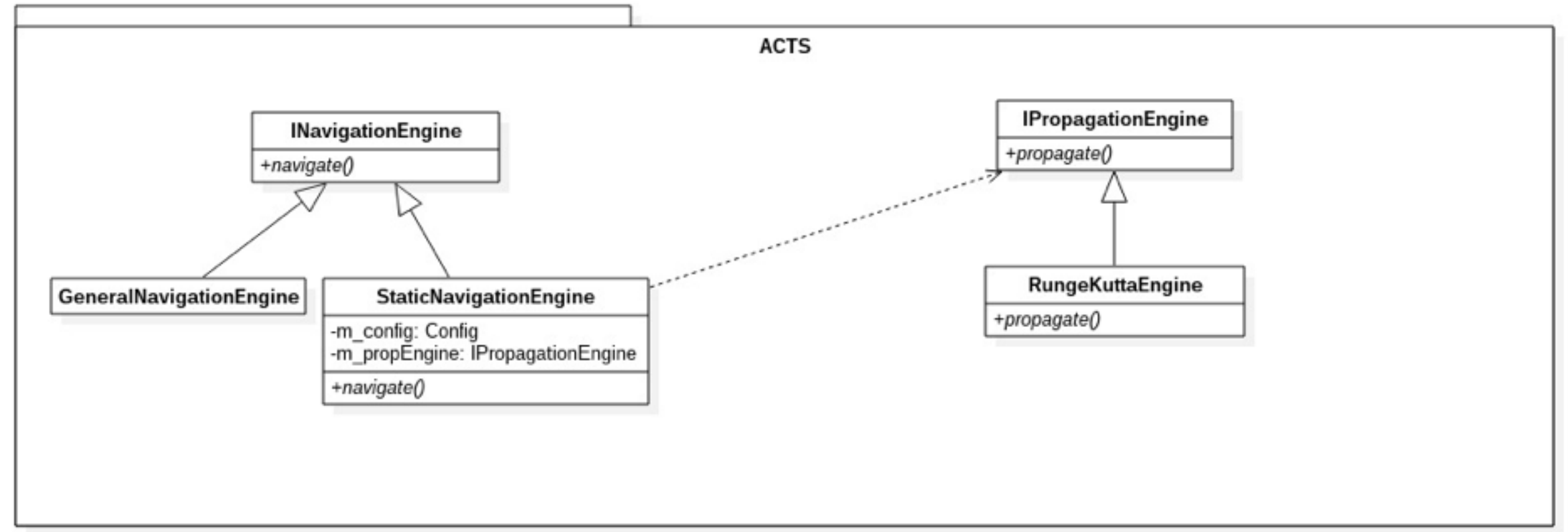
master | a-common-tracking-sw / Core / include / ACTS / | Find File | Download zip

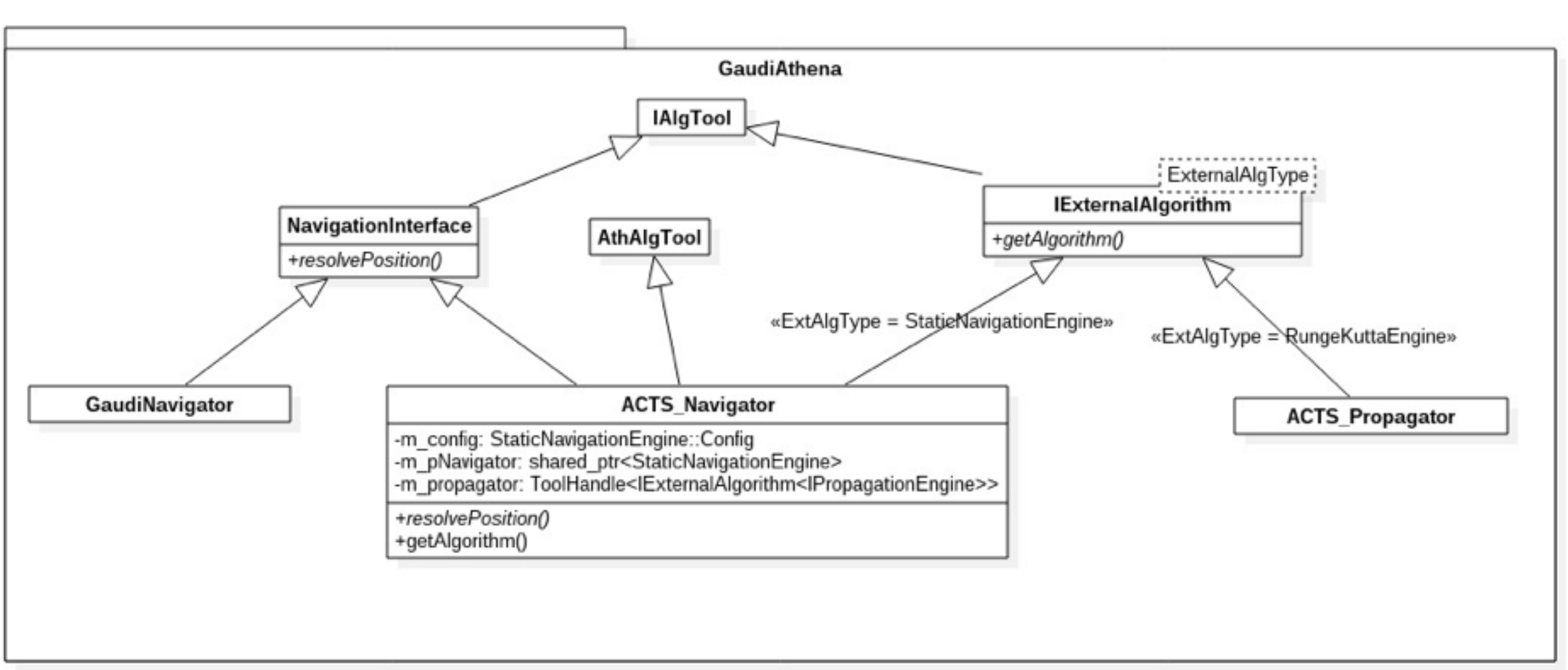
Name	Last Update	Last Commit > c199248b – Merge branch 'make_clang_compatible' into 'mast...	History
..			
Detector	about 4 hours ago	clang changes (not yet updated CMake file)	
EventData	about 4 hours ago	clang changes (not yet updated CMake file)	
Extrapolation	6 days ago	Merge remote branch 'origin/master' into ACTS-76_geometry_building_co…	
Layers	6 days ago	fix include path, remove duplicate includes	
MagneticField	6 days ago	fix include path, remove duplicate includes	
Material	6 days ago	fix include path, remove duplicate includes	
Surfaces	about 4 hours ago	clang changes (not yet updated CMake file)	
Tools	4 days ago	moved missing geometry tools to standalone	
Utilities	about an hour ago	Merge remote-tracking branch 'origin/master' into make_clang_compatible	
Volumes	6 days ago	fix include path, remove duplicate includes	

Framework decoupling & Gaudi/Athena support

- Every configurable code comes with a Class::Config definition
  - this can then be connected with the framework, e.g. **`declareProperty("UseLogInclude", cfg.log_include)`** in Gaudi/Athena

```
namespace Fatras {

  /** @class MultipleScatteringSamplerGaussianMixture
   *
   * @author Andreas Salzburger <Andreas.Salzburger@cern.ch>
   * @author Noemi Calace       <Noemi.Calace@cern.ch>
   * @author Artem Basalaev     <Artem.Baralaev@cern.ch>
   *
   */

class MultipleScatteringSamplerGaussianMixture : virtual public IMultipleScatteringSampler {

public:
    /** Config
    Configuration object for this MultipleScatteringSampler*/
    struct Config {
        std::shared_ptr<IRandomNumbers>  randomNumbers;   /** Random Generator service  */
        bool                                log_include;   /** boolean switch to include log term  */
        bool                       optGaussianMixtureG4;   /** modifies the Fruehwirth/Regler model to fit with G4 */

        Config() :
          randomNumbers(nullptr),
          log_include(true),
          optGaussianMixtureG4(true)
        {}
    };

    /** AlgTool like constructor */
    MultipleScatteringSamplerGaussianMixture(const Config& msConfig);
```


ACTS
INavigationEngine
+navigate()
IPropagationEngine
+propagate()
GeneralNavigationEngine
StaticNavigationEngine
-m_config: Config
-m_propEngine: IPropagationEngine
+navigate()
RungeKuttaEngine
+propagate()


GaudiAthena
IAlgTool
ExternalAlgType
IExternalAlgorithm
+getAlgorithm()
NavigationInterface
+resolvePosition()
AthAlgTool
«ExtAlgType = StaticNavigationEngine»
«ExtAlgType = RungeKuttaEngine»
GaudiNavigator
ACTS_Navigator
-m_config: StaticNavigationEngine::Config
-m_pNavigator: shared_ptr<StaticNavigationEngine>
-m_propagator: ToolHandle<IExternalAlgorithm<IPropagationEngine>>
+resolvePosition()
+getAlgorithm()
ACTS_Propagator

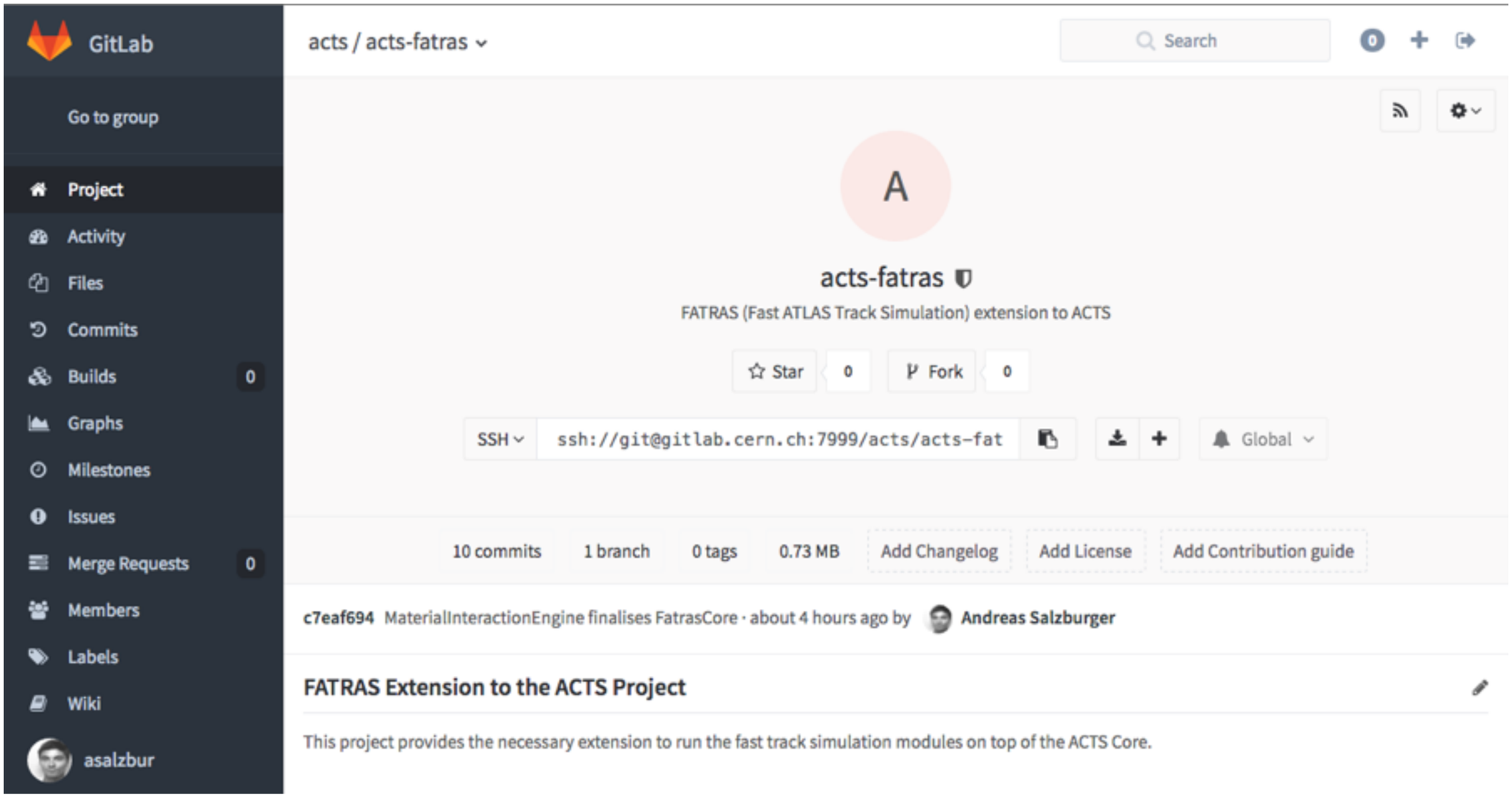
What ACTS does (not) provide

- ACTS is highly extendable/modifyable
  - (mis-)alignment infrastructure via the plugged-in underlying geometry
  - support for different fitters, extrapolators, etc.
  - support for different track parameterisation
    default can be overwritten $(l_0, l_1, \varphi, \theta, \frac{q}{p})$
  - measurement calibration & extension mechanism

- ACTS is not intended to replace core framework functionality
  - an underlying geometry model has to exist (and can be linked via a plug-in)
  - no ACTS persistency support
  - no ACTS event display (though we will write out some standard formats)

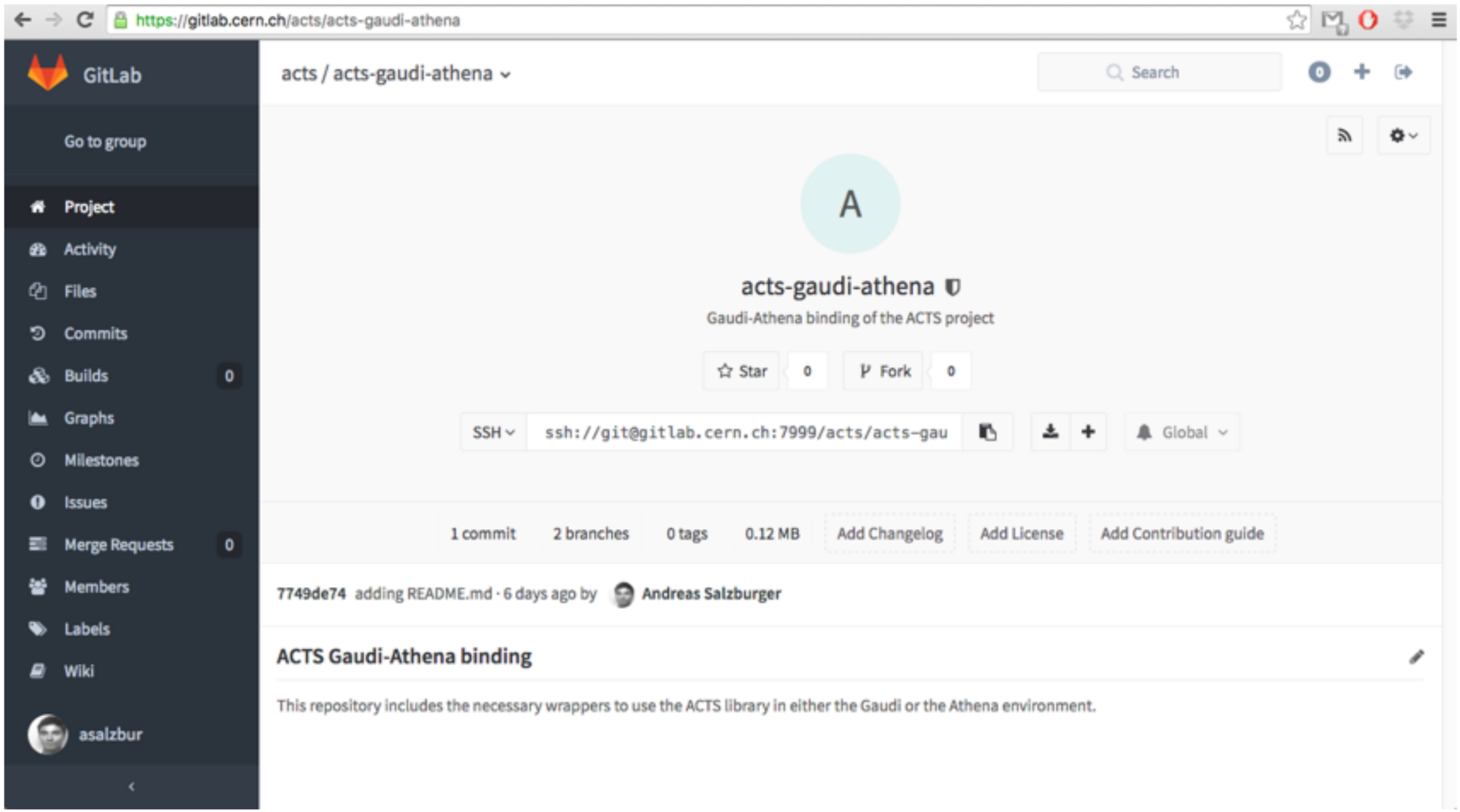
Simulation Extension

- Fast Track simulation extension to ACTS is in a separate repository
  - gitlab: https://gitlab.cern.ch/acts/acts-fatras

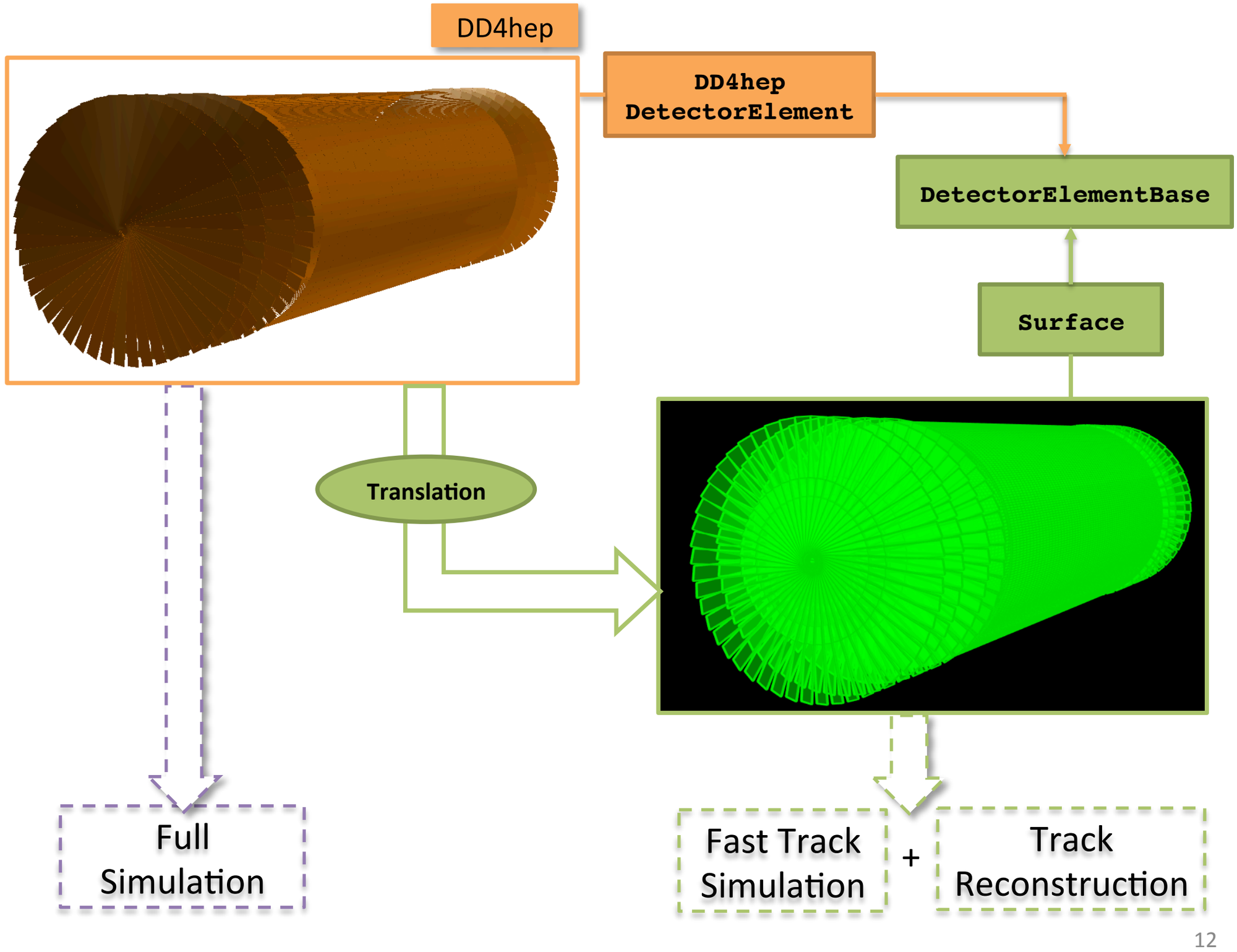


Gaudi-Athena Extension

- ACTS will have a native Gaudi-Athena binding
  - plan to use the ACTS as Core for the ATLAS tracking in Run 3
  - gitlab: https://gitlab.cern.ch/acts/acts-gaudi-athena



ACTS examples: FCC SW

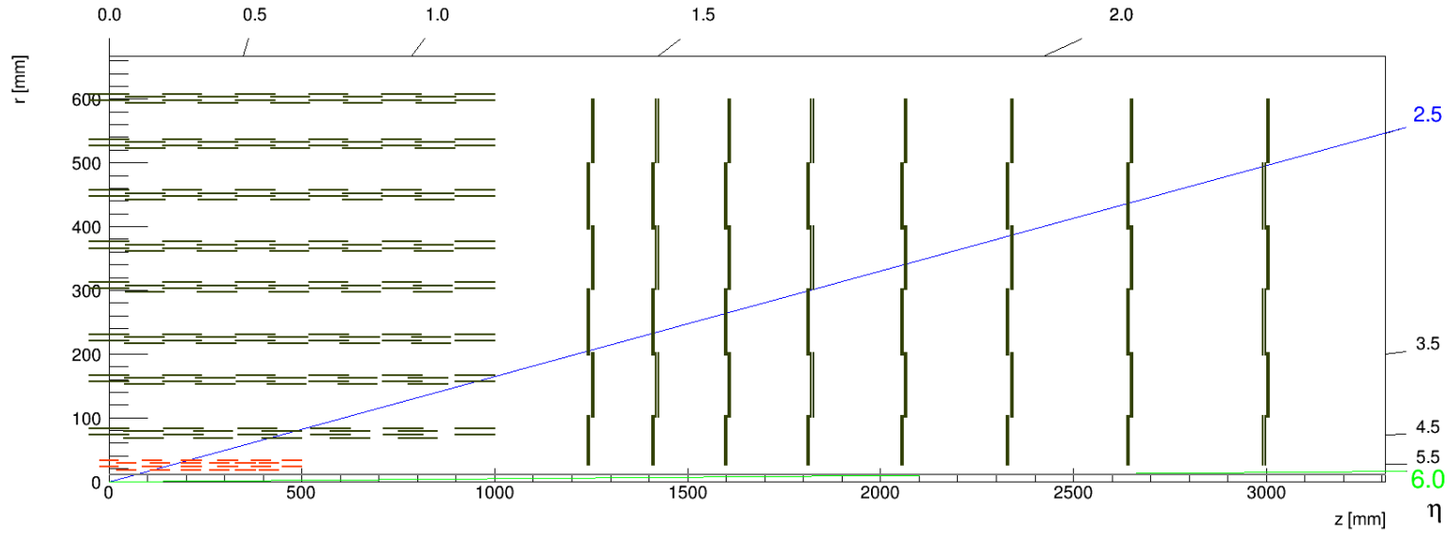


Julia Hrdinka, Tracking SW Developments, FCC Week Rome, 2016

FCC(-hh) Test Tracker

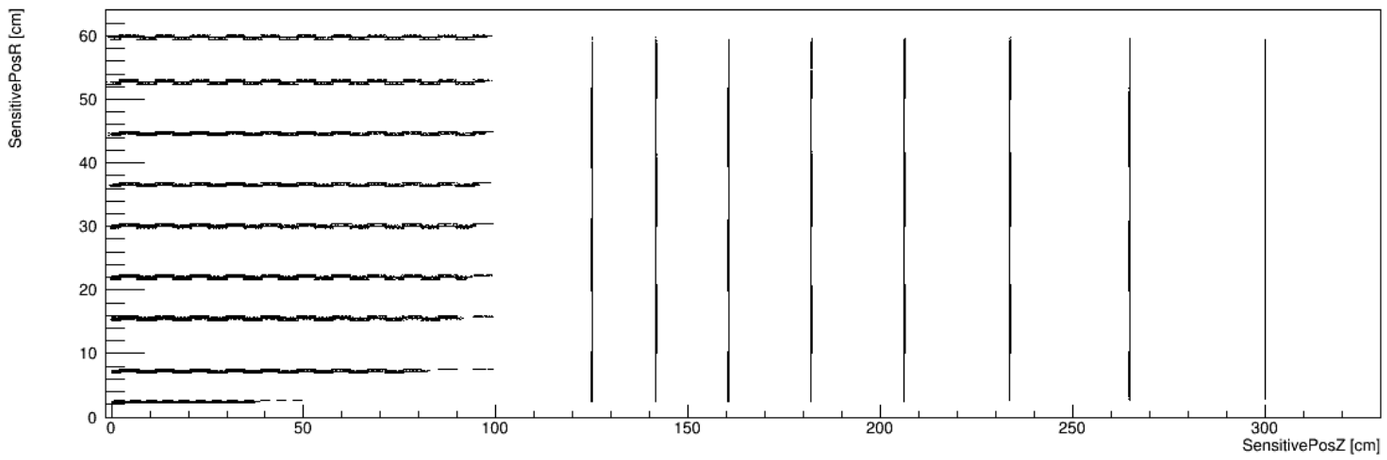
Geometry of FCC-hh tracker as created with tkLayout (a toolkit for ongoing tracker design studies of CMS phase 2 upgrade tracker, adapted to FCC-hh needs)

(-> Z.Drasal) https://indico.cern.ch/event/438866/contributions/1085165/



TKLayout

Translated into DD4hep and used in ACTS

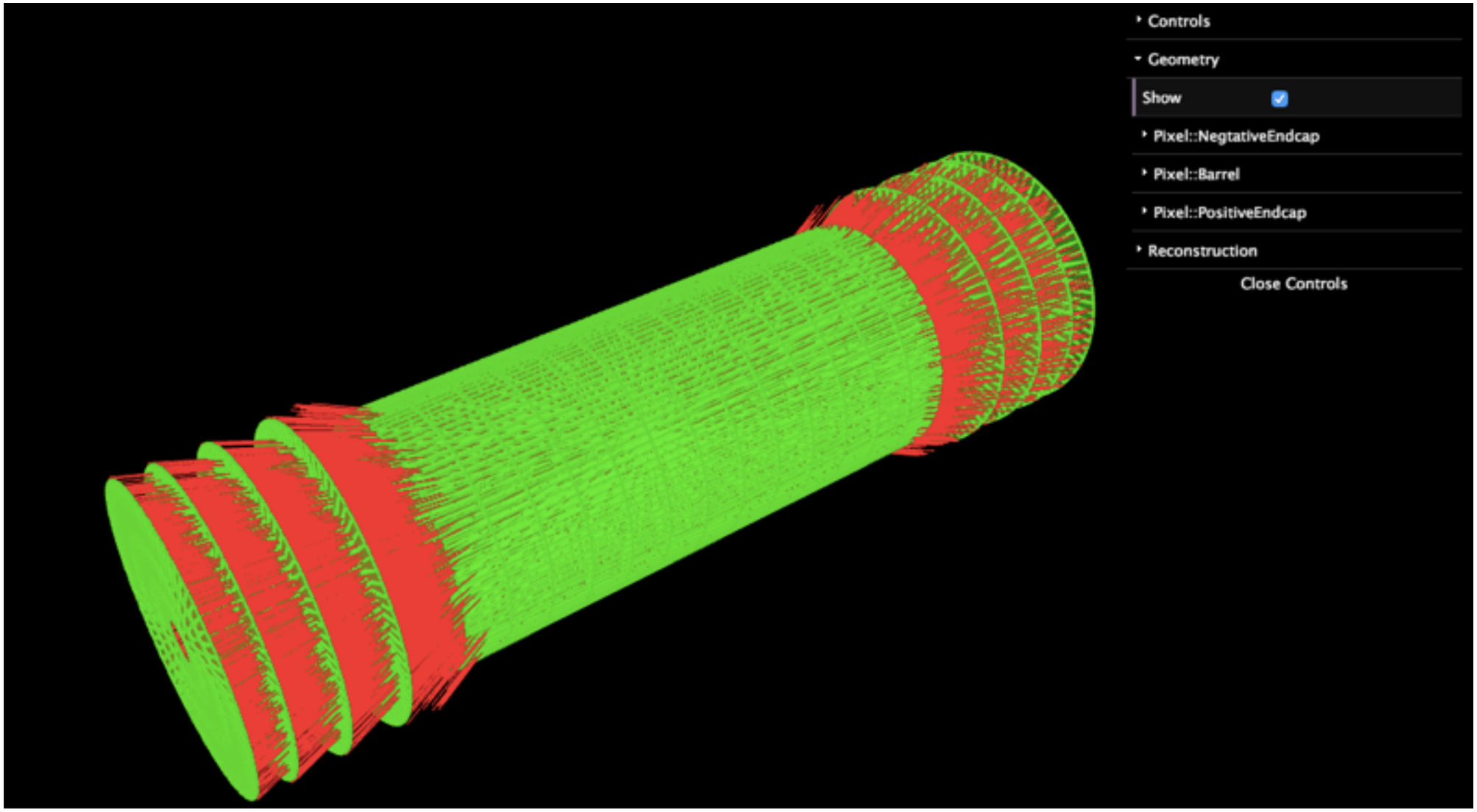


ACTS – Fast Simulation

Julia Hrdinka, Tracking SW Developments, FCC Week Rome, 2016

ACTS examples: Tracking ML

- Use of ACTS to produce first datasets with fast track simulation
  - defining a detector setup currently on the way



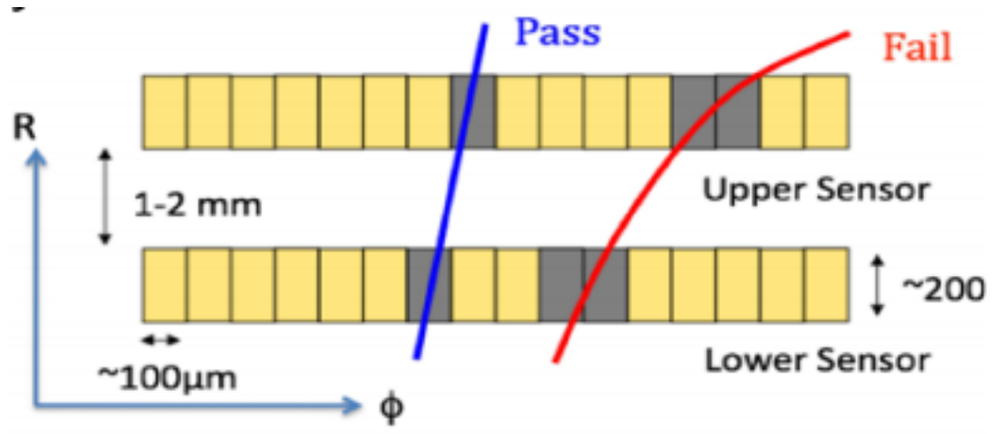
AS, Connecting the DOTS, Vienna, 2016

ACTS Examples: Tracking ML

- ATLAS & CMS use rather different concepts for Phase 2 tracker
  - LVL1 track triggering capability is essential part of CMS design

    tracklets used to mitigate fake hits (low momentum particle hits at a very early stage),

    tracklet information can be used in pattern recognition



correlate selected hits in two closely separated sensors to discriminate between high and low Pt tracks

  - ATLAS follows more space point design

    LVL1 track triggering via pattern banks (LVL1Track, FTK++), long clusters may be used as tracklets in very forward directly

- Needs discussion if we can create a hybrid detector setup for ML challenge
  - ACTS can simulate & handle both concepts

ACTS release plan/next steps

- Last two weeks spent on framework decoupling
  - message logging is one of the biggest hurdles to solve

- FCC-(hh) and Tracking ML demonstrator should be working soon again
  - need additional GaudiWrapper (FCC-hh)
  - could be done in a stand-alone executable (Tracking ML)

- Full session planned at the ATLAS TIM in Glasgow (Jun 6-10)
  - GaudiHive demonstrator
  - first CPU and memory profiling (and optimisations)

- More info
  - ACTS gitlab: **https://gitlab.cern.ch/groups/acts**
  - ACTS mailing list: **acts-developers@cern.ch**