



FCC-hh Tracking Milestones for 2016

projects & possible studies

Andreas Salzburger (CERN)



Project timescales



short time-scale
doable
(Rome-ready)

should be
doable
(Rome-possible)

not really doable
(Byzantine ?)



Part 0 - piece & paper studies



Idealistic detector resolution

- ▶ ATLAS/CMS/(and others) have analytic programs to evaluate the detector resolution

- encapsulate the formulas/code and estimate first resolution curves (feed from DD4Hep description)

- design a resolution database for parametric simulation

- investigate a further integration of this into either FCC/HSF software

- ▶ programs also allow to create crude material maps given some reasonable input

- create first material maps for calorimeter studies

- design an efficiency database for efficiency correction for parametric simulation

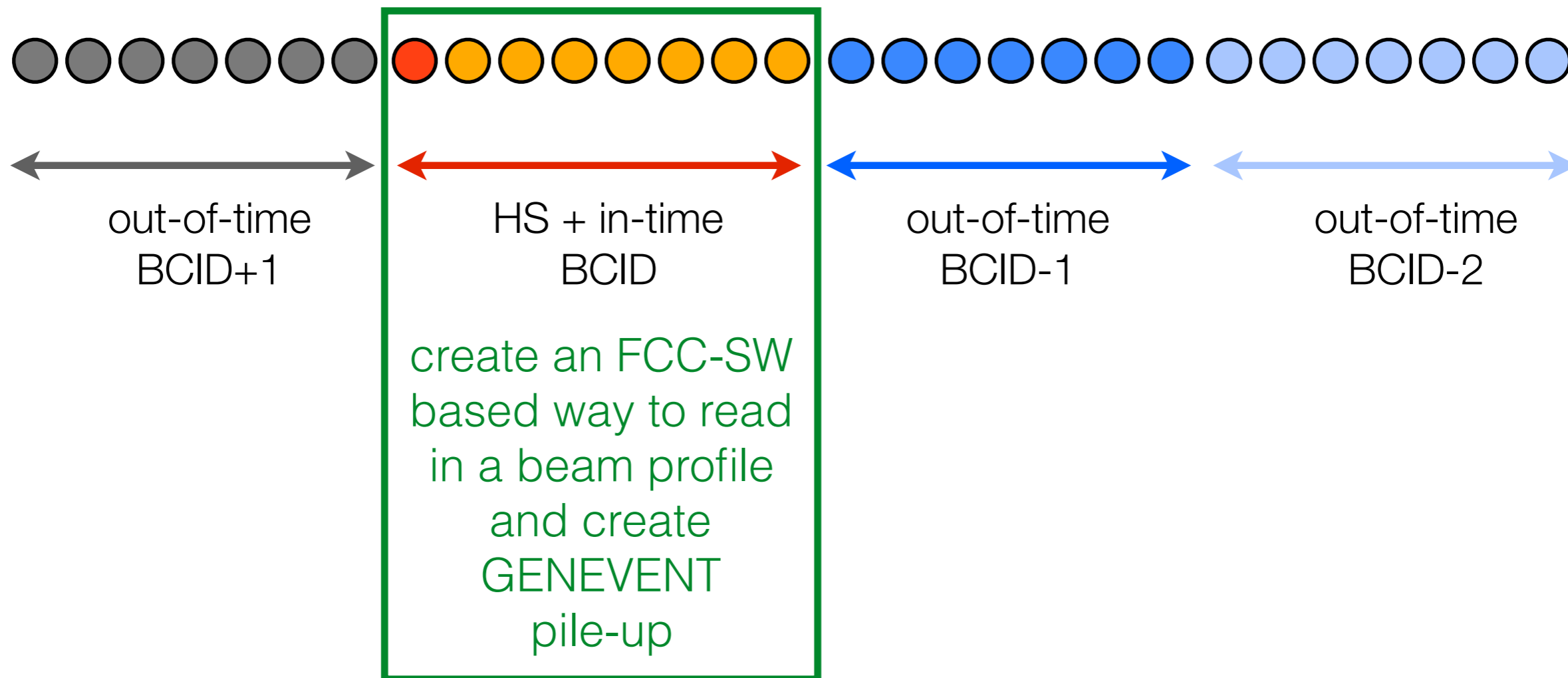




Part 1 - truth studies

Pile-up infrastructure

- ▶ Focus only on in-time pile-up for tracking



expand the module to deal with out-of time pile-up

expand the module to deal with hit-based pile-up



Pile-up & vertex reconstruction

- ▶ Truth based vertexing study, part 1
 - use GENEVENT pile-up to create HS+pile-up gen events
 - use average vertex smearing to analyse merging & shadowing effects

- ▶ Truth based vertexing study, part 2
 - use GENEVENT from part 1
 - add smearing from analytical track resolution program
 - add track inefficiencies from analytical tracker program
 - import an existing vertex reconstruction program and run it, study resolution, merging, splitting & shadowing effects

- ▶ Further integrate the vertexing package into the FCC SW suite



not covered yet

Truth-based global/local track densities

▶ Global occupancy studies

- use GENEVENT pile-up to create HS+pile-up gen events
- (straight-line) extrapolate them to detector layers
- estimate global track density/occupancy from in-time primaries
- add an estimate on secondaries

▶ Local occupancy studies

- use the same framework as for global studies
- repeat with high-pT jet events, boosted objects
- estimate the local track density
- estimate the min $\langle \text{track}, \text{track} \rangle$ distance



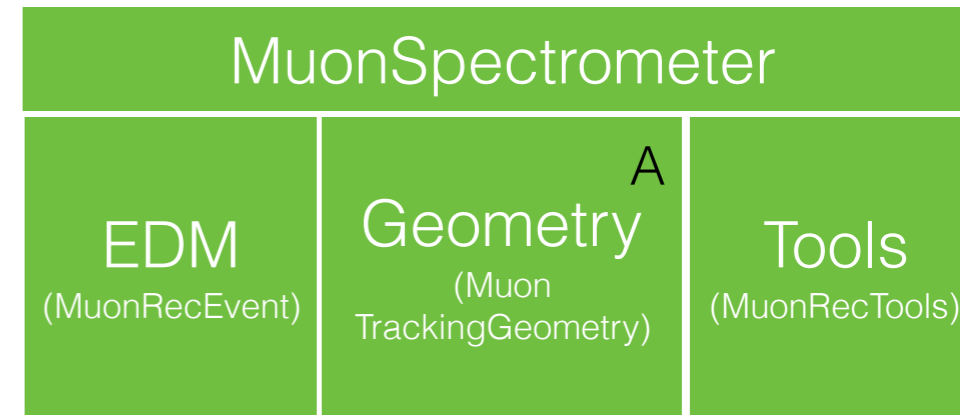
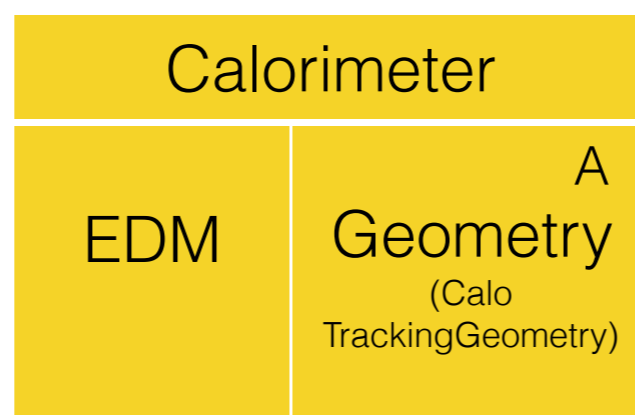
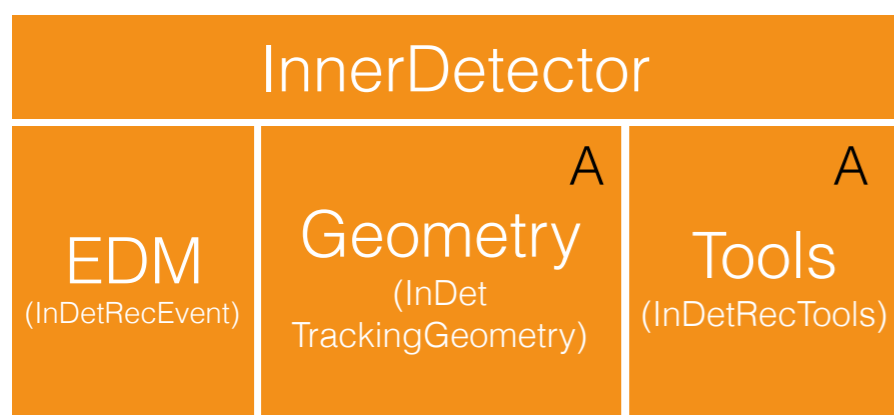
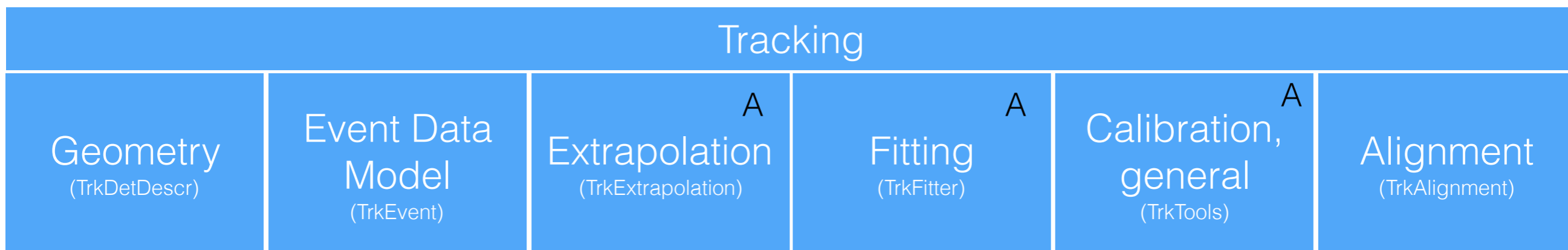
not covered yet



Part 2 - emulation studies

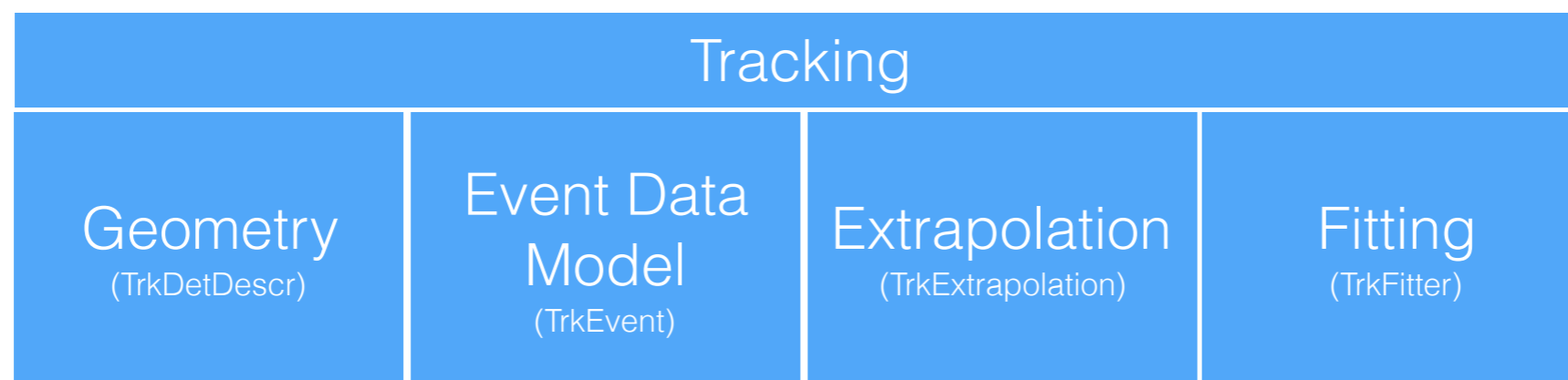
Import of the ATLAS Tracking SW (1)

- ▶ Encapsulation of the core of ATLAS tracking SW is ongoing



Import of the ATLAS Tracking SW (2)

- ▶ Encapsulation of the core of ATLAS tracking SW is ongoing
 - aim to decouple core components in Tracking container from ATLAS
 - still keep possibility to build it against ATLAS to keep development in sync



- add possibility to build an TrackingGeometry based detector from DD4Hep
- allow the Extrapolation & Navigation for Inner Detector
- add fast simulation support & geometric digitization
- add fitting modules & truth tracking




- convert output to FCC PODS event data

- expand to Calorimeter and MS support (combined track fitting)

a-tracking-sw

Attempt to encapsulate the ATLAS Tracking software from ATLAS

currently restricted access

★ 0 | 1 | SSH | HTTPS | ssh://git@gitlab.cern.ch:7999/ |  |  | + |  GLOBAL ▾

65 COMMITS

5 BRANCHES

0 TAGS

13.21 MB

ADD CHANGELOG

ADD LICENSE

ADD CONTRIBUTION GUIDE



A Tracking Software (ATS) Project

This library is based on the track reconstruction software developed by the [ATLAS Collaboration](#).

The main philosophy is to provide high-level track reconstruction modules that can be specified for detector technologies by simple extension.

- Event Data Model (EDM)
- Geometry
- track reconstruction tools

double build philosophy

The library is attempted to build against  and , while additional external dependencies are kept at a minimum.

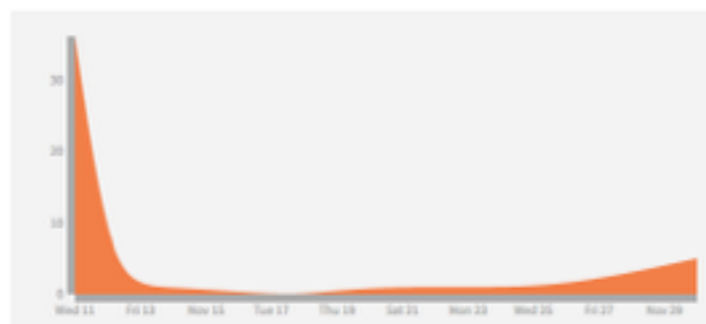
Commits to master, excluding merge commits. Limited by 6,000 commits



Andreas Salzburger

51 commits

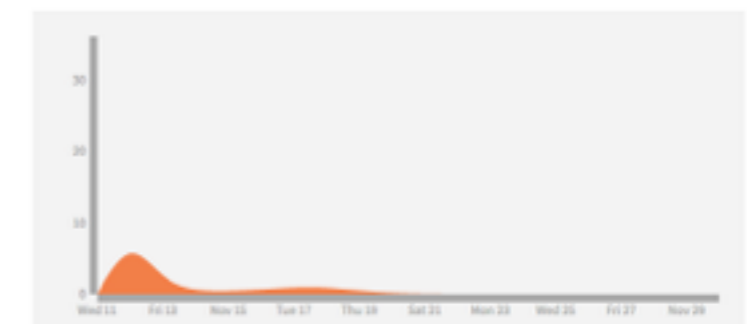
andreas.salzburger@cern.ch



Christian Gumpert

12 commits

cgumpert@cern.ch



ATS - Status, plans and timescales

- ▶ Prototype for geometry building decoupled from ATHENA
 - builds against Gaudi (Cmake) and Athena (cmt) : **done for core classes**
 - next step: dummy detector building in Gaudi/ATHENA and compare: **ongoing**
 - geometry building from DD4Hep in FCC context : **started**
- ▶ Integration of Extrapolation engine and basic EDM
 - proof of extrapolation concept and little fast track sim: **started**
- ▶ Integration of Truth tracking module: **not started**
- ▶ Integration of KalmanFitter: **started**



Julia Hrdinka, Noemi Calace, Christian Gumpert, Sharka Todorova, AS



Part 2 - reconstruction

First steps towards reconstruction

- ▶ Use fast simulation & GENEVENT pile-up files to produce hit files
 - start using/importing pattern recognition approaches
 - seeding & track following approach
 - global pattern recognition approach (e.g. conformal mapping)



not covered yet