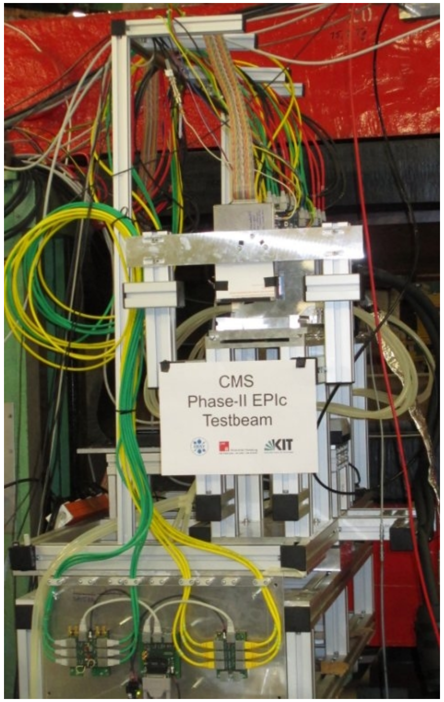
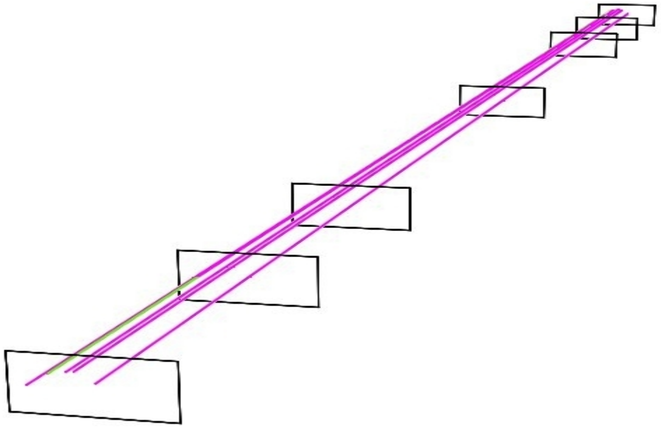
Test Beam Analysis with a Telescope and Irradiated Strip Sensors

Combining an analog ALiBaVa strip DUT with a digital EUDET/AIDA pixel telescope





Thomas Eichhorn

3rd Telescopes and Test Beams Workshop 2015

21.01.2015

Outline

- Test beam setup & measurements
  - Epitaxial strip sensors for the CMS Phase-II upgrade
  - ALiBaVa & telescope setup
- Integrating an analog strip sensor into a digital pixel telescope
  - ALiBaVa processors for EUTelescope
  - Challenges and changes to EUTelescope
- Sensor analysis and results

Test Beam Setup and Measurements

Measurement Goals & Program

- Phase-II upgrade of the CMS tracker:
  - Investigate possible future sensor materials

    → Could epitaxial silicon be a HL-LHC radiation-hard sensor material?

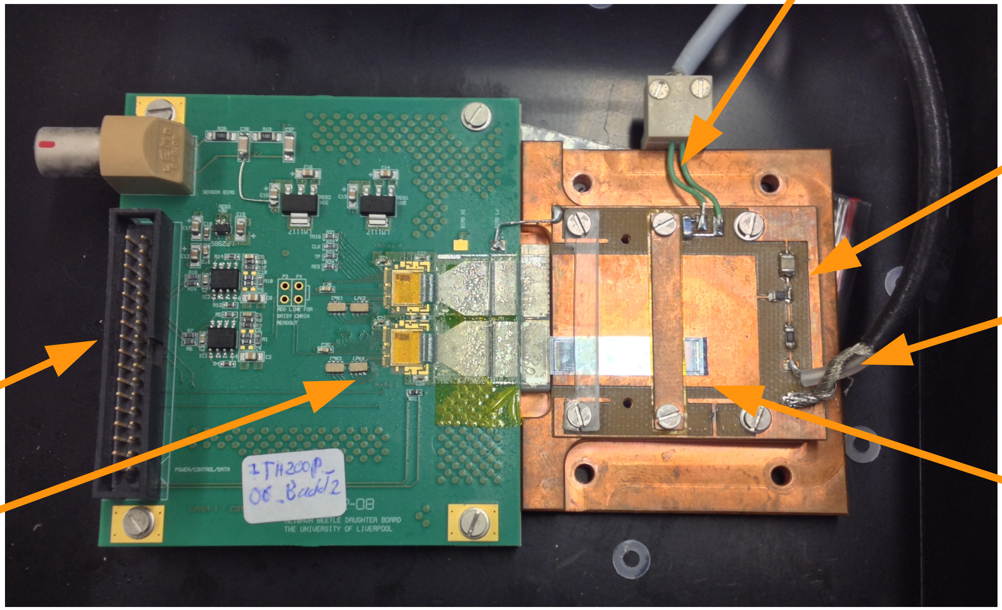
- Sensors / irradiations:
  - 19 epitaxial, float-zone and magnetic czochralski silicon strip sensors from a CMS vendor run: 64 strips, 80µm pitch, active thickness 70µm – 200µm
  - Proton irradiations up to Φ = 1.3e16 n_{eq}/cm^2 at CERN (23 GeV) and Los Alamos (800 MeV)
  - Both n- and p-type material, with p-stop and p-spray isolation
  - 5 GeV e^+/e^- from beams 21 & 22 with DATURA and ACONITE telescopes

- Measurement program:
  - 3 beam incidence angles (0° to 51.3°) to investigate charge sharing
  - How are signal/noise levels at different bias voltages (up to ±1000V)?
  - Can the sensors still be used after extremely high irradiations?

ALiBaVa Setup

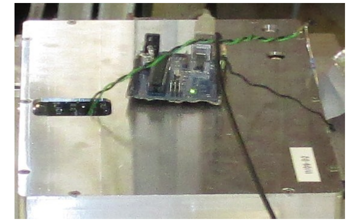
- A Liverpool Barcelona Valencia Read-out System
  - DAQ in-a-box system based on LHCb Beetle chip to read out strip sensor data
  - Motherboard for PC connection and data processing
  - Sensor wire-bonded to Daughterboard with Beetle ROC
  - Analog read-out of the 2 * 128 channels every event
  - Serial communication on two differential lines
  - Rate limited to ~ 150 Hz

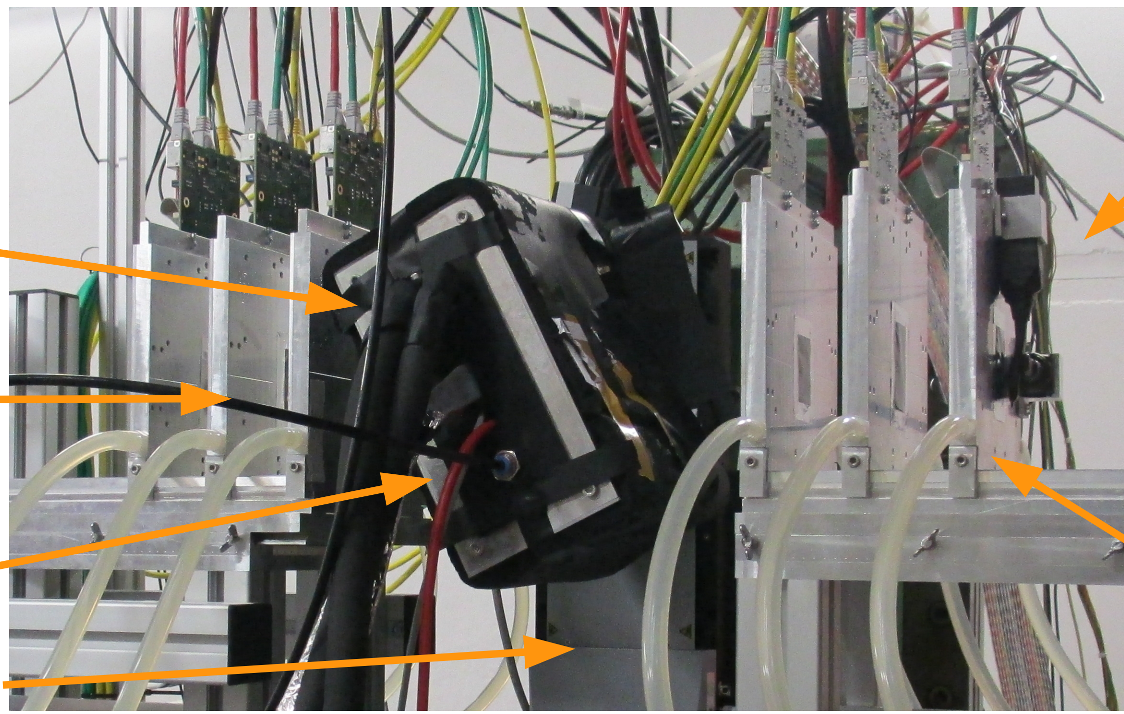
- Coldbox to mount setup in beam telescope



ALiBaVa Setup in the Test Beam

- Difficult setup goals:
  - Reach -27 °C sensor temperature for highest irradiations
  - Minimize scattering material in beam
  - Movable setup for different beam incidence angles
  - Limit access to beam area: use an Arduino for hardware resets

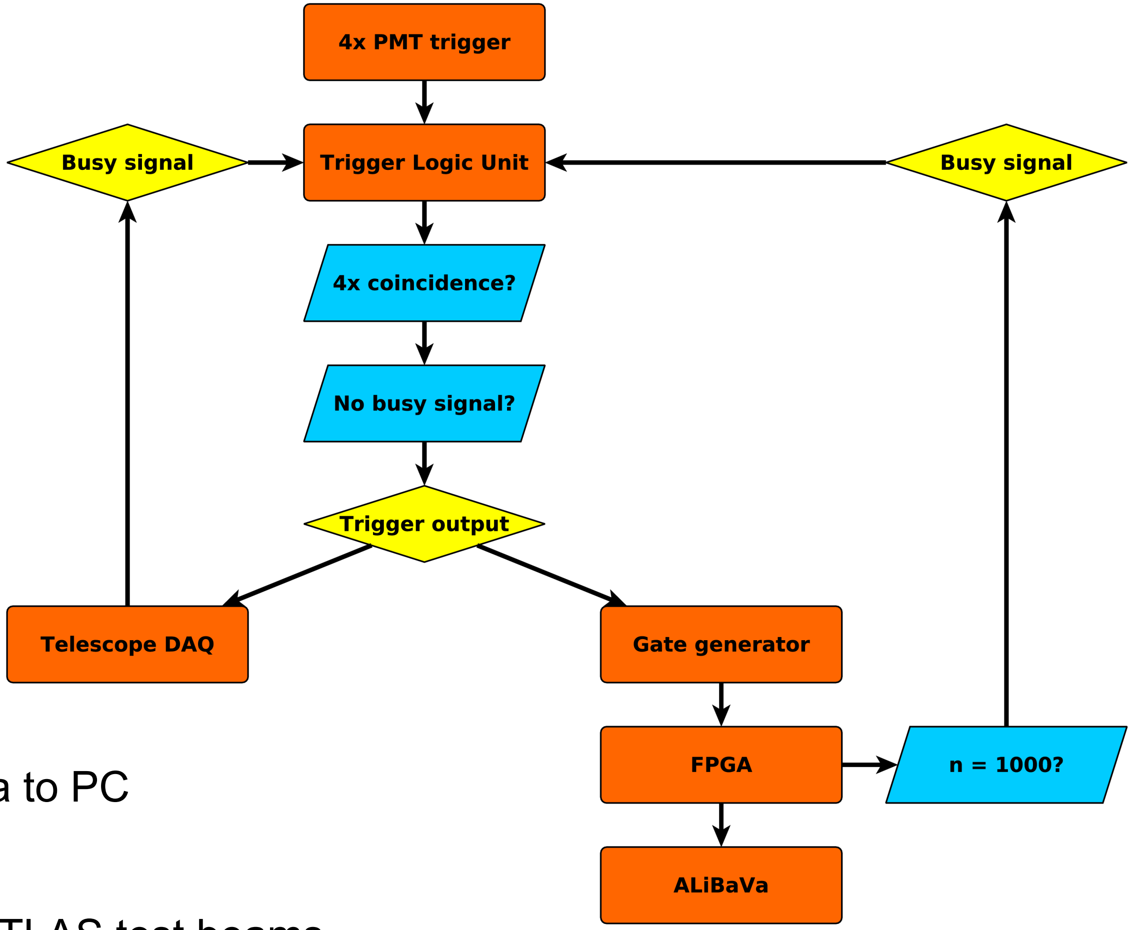




Trigger Synchronisation

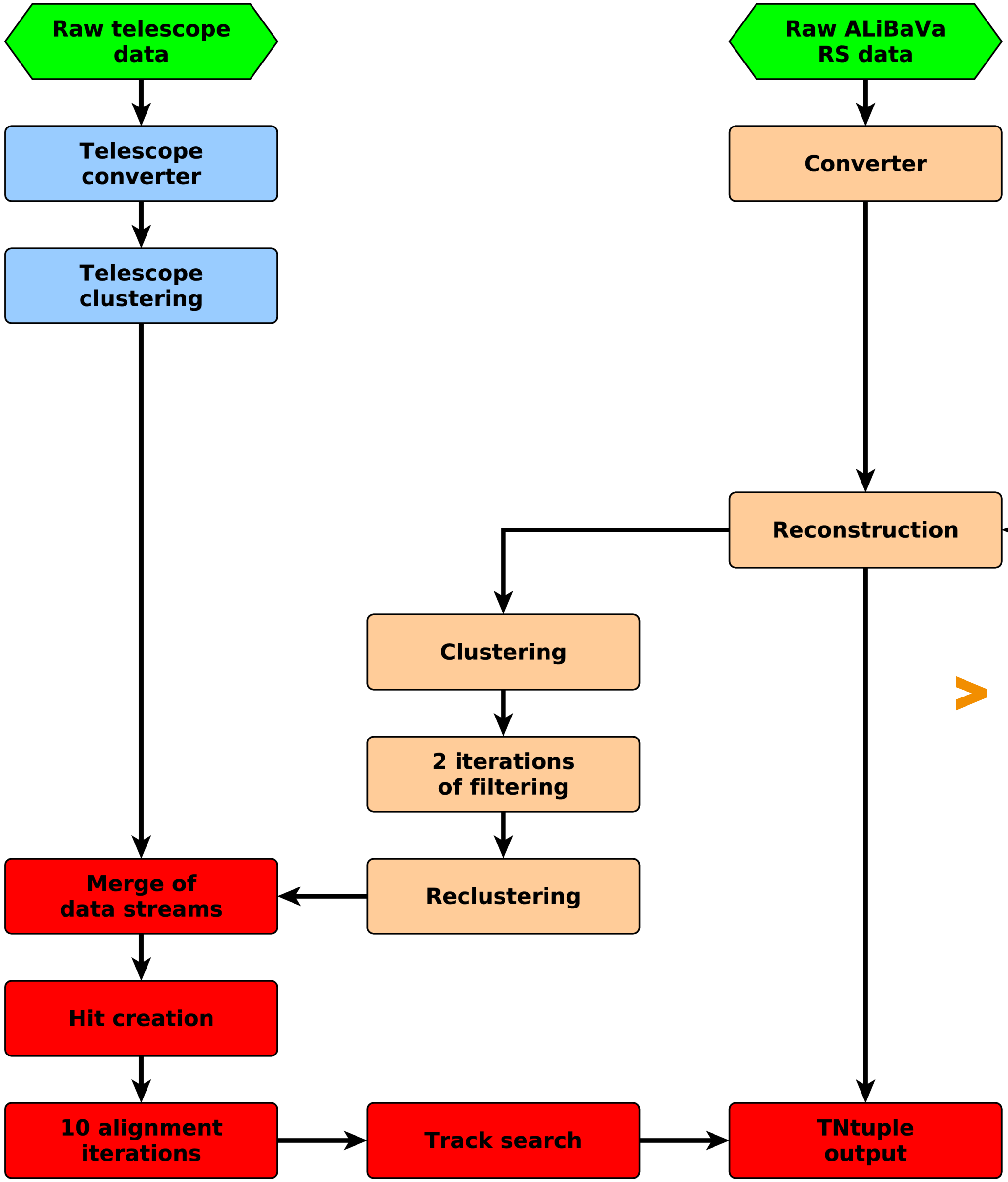
- Combination of 2 DAQ systems:
  - Different read-out speeds & memory/buffer sizes
  - Two clocks
  - How to match events offline?

- Custom gate generator & FPGA solution:
  - Set a global 2.5s 'busy' state every 1000 events
  - Time for ALiBaVa to send data to PC
  - Allows precise event counting
  - Developed by E. Yildirim for ATLAS test beams



Integrating an analog strip sensor into a digital pixel telescope

Reconstruction and Analysis Workflow

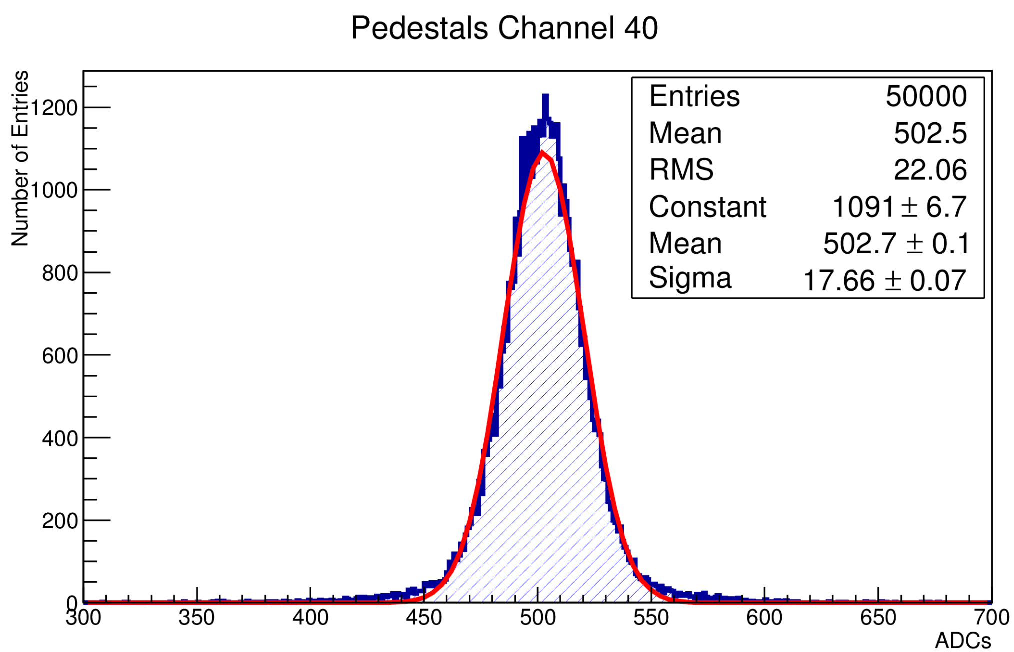


- Goal: Use telescope to pin-point tracks through the ALiBaVa sensor
  - Strategy: Include DUT as a 7th telescope plane
  - DUT clusters for alignment, *not* analysis
  - EUTelescope v00-09-02 and `datura-notdut` example as a basis

Data Conversion and Pedestals

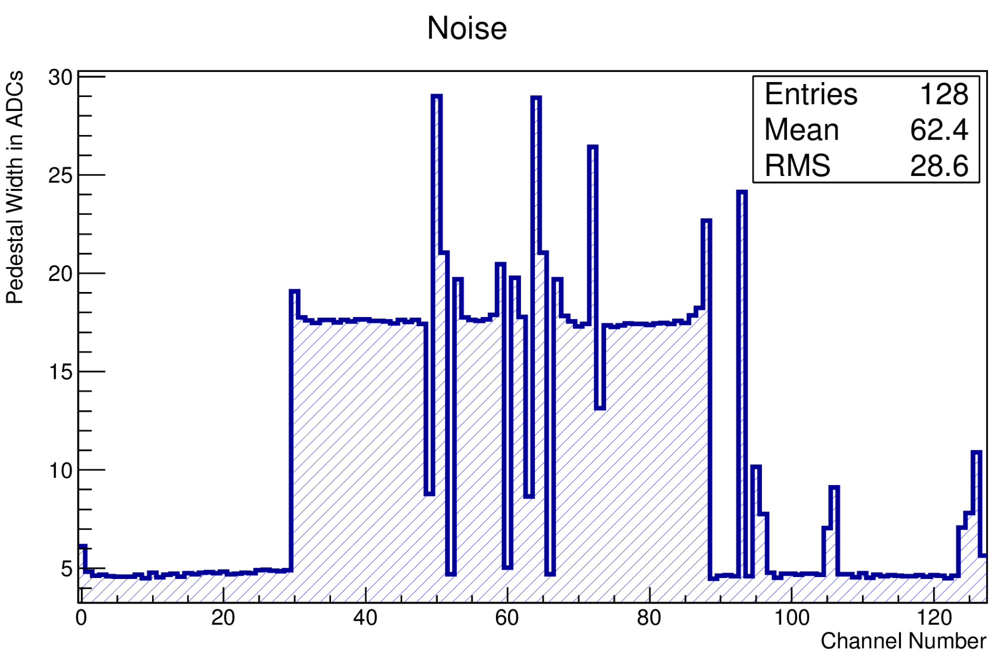
- Convert ALiBaVa data to LCIO

- For off-beam runs: calculate pedestals of each channel
  - Fill signals into a histogram and fit with a gaussian
  - Mean → pedestal value
  - Width → preliminary noise



Channel Masking

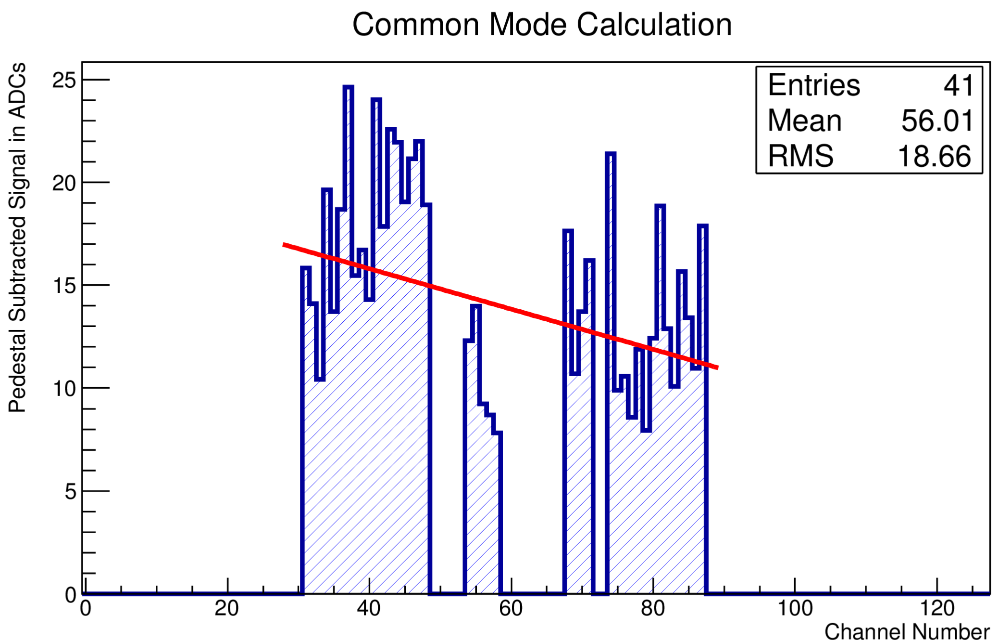
- Not all channels are connected to the sensor
  - The ALiBaVa has 2 * 128 channels – our sensor only has 64
  - Some wire bonds might be faulty
- Bad channels have to be masked → look at preliminary noise
  - Flag outliers from here on out



Common Mode Subtraction

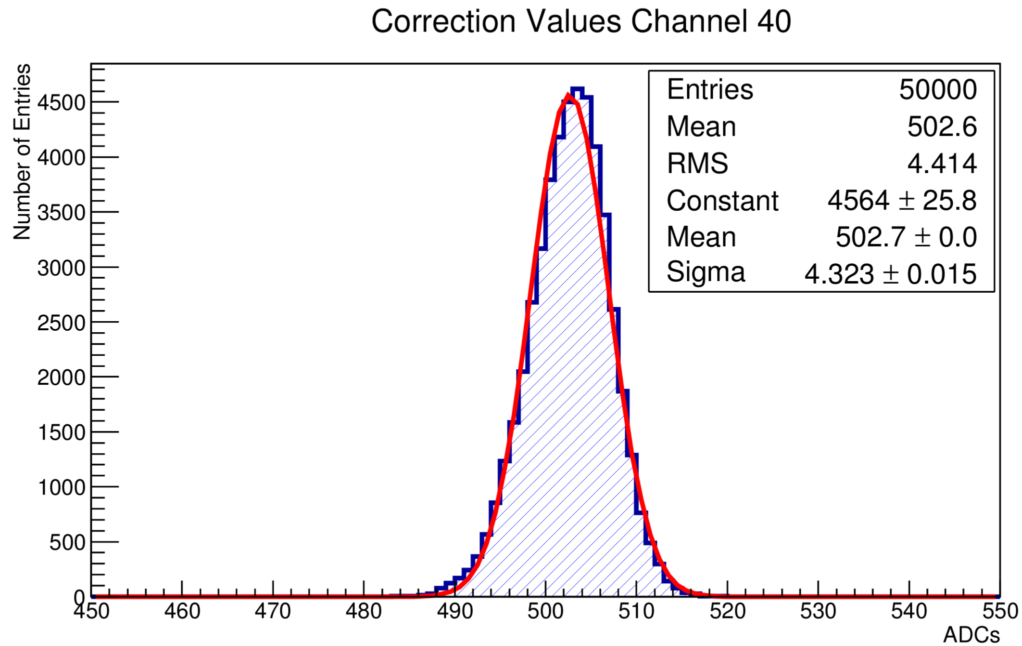
- Subtract pedestals, mask bad channels

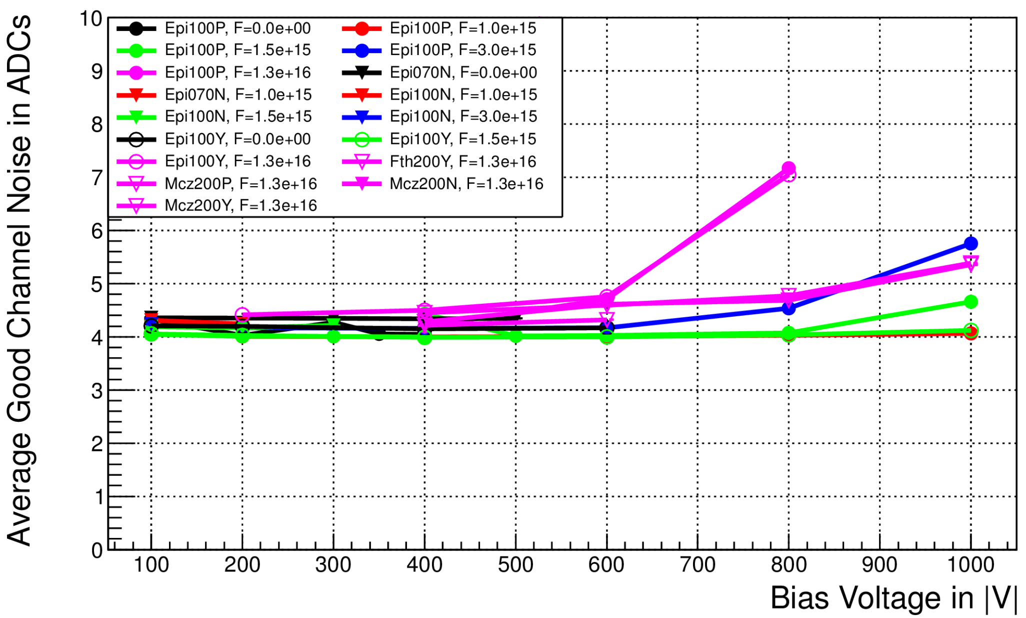
- Calculate common mode
  - Synchronous movement of all (good) channels in an event
  - For each event: exclude outliers and fit a slope → common mode



Data Correction and Noise

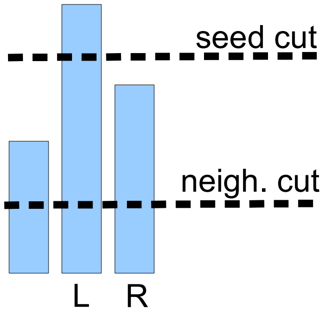
- Correction values (pedestal + common mode) for each channel and event
  - Correct the 'real' on-beam data
- Get the noise values for each channel
  - Will be used later on for clustering, signal-to-noise, etc.
  - Gaussian fit on correction values
  - For unirradiated sensors or Φ < 1.5e16 n_{eq}/cm^2
    ~ 4.5 ADCs
  - After Φ = 1.3e16 n_{eq}/cm^2:
    ~ 7.1 ADCs for d = 100 µm
    ~ 5.5 ADCs for d = 200 µm

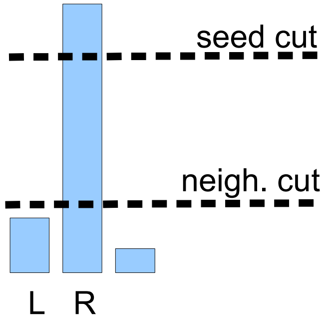




Clustering

- On corrected data: search for channels with signal > 5 * noise → 'seed'
  - If neighbours have a signal > 2.5 * noise → add to cluster
  - Cluster size: for 0° beam incidence: ~1.1, for 51° incidence ~ 1.8

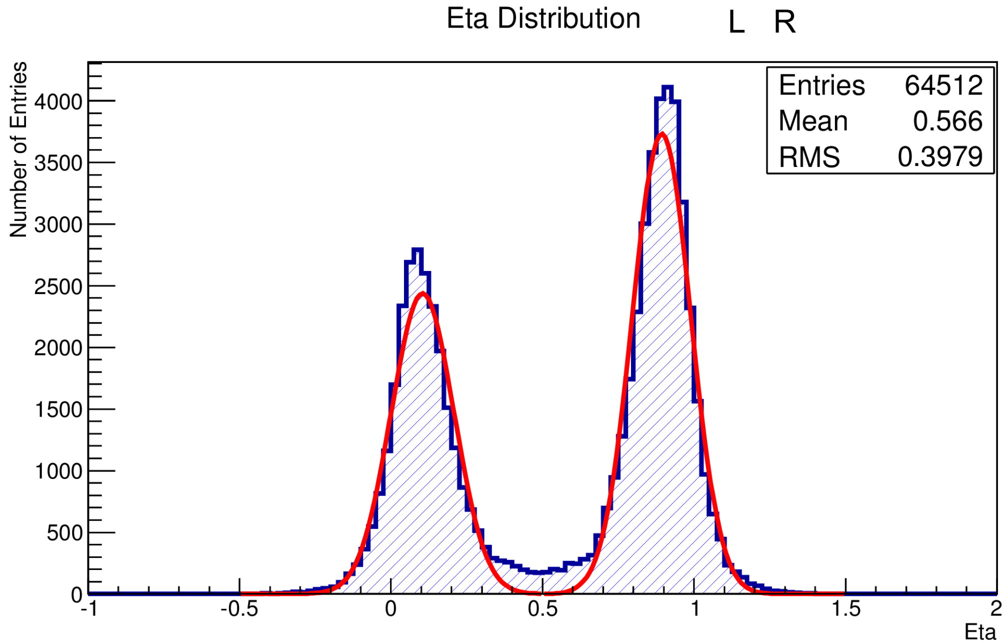




- Write same format as telescope clusters → need an X-coordinate!
  - Look for cluster on preceding telescope plane nearest the relative Y-position
    → copy X CoG from here

- Control plots: η-distribution
  - $\eta = Q_l / (Q_l + Q_r)$
  - Left / right is assigned to the seed and it's neighbour with the highest signal

    → asymmetric!

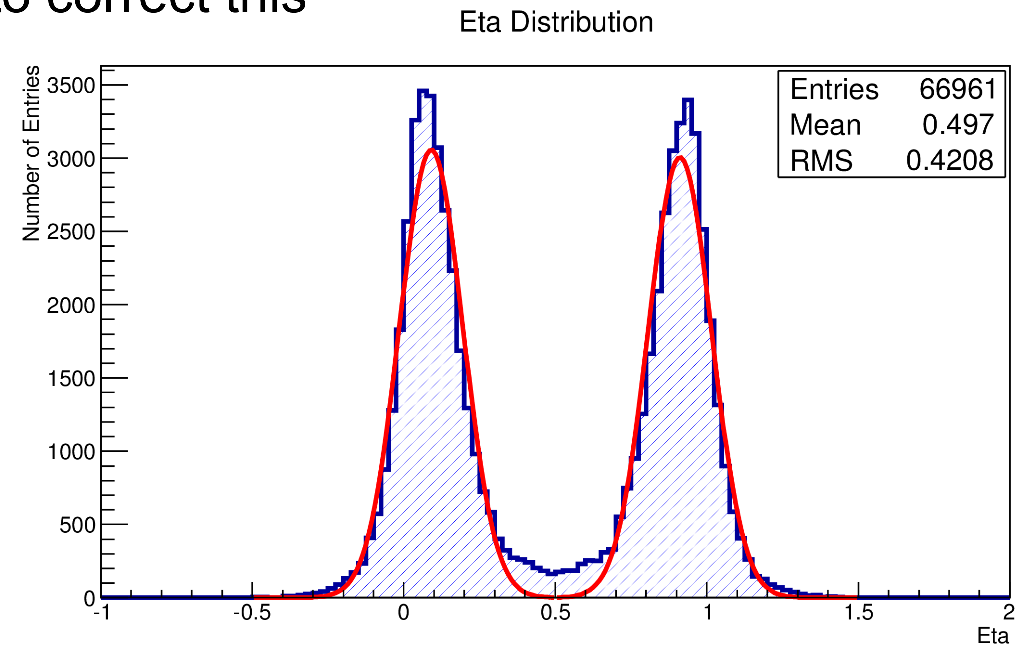


Filtering

- Asymmetric cross-talk:
  - Charge is spread to the next channel in the serial read out
  - Compare charge distributions of seed neighbours:
  - Channels on the right of a seed have ~6.4%, on the left only ~ 2.6% of the seed's charge
  - Also seen in the ALiBaVa chip header and with ATLAS sensors → no sensor effect

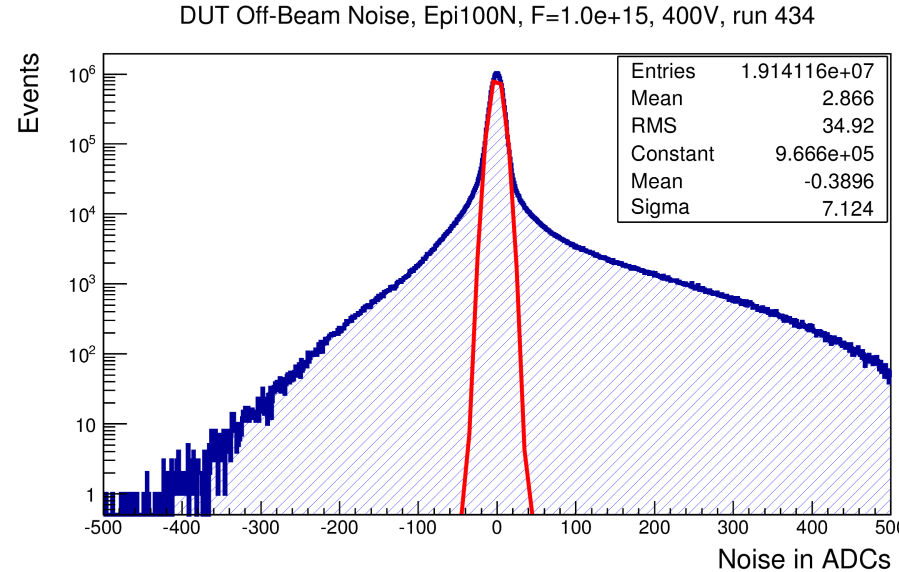
- **F**inite **I**mpulse **R**esponse filter to correct this
  - Run through all events:
    $Q_{new}[\ i\ ] = Q_{old}[\ i\ ] - a*Q[\ i-1] - b*Q[\ i-2]$
  - Calculate a,b from seed neighbour charge ratios
  - Conserve charge: add the cross-talking charge back to where it originated
  - After 2 filter iterations: recluster data

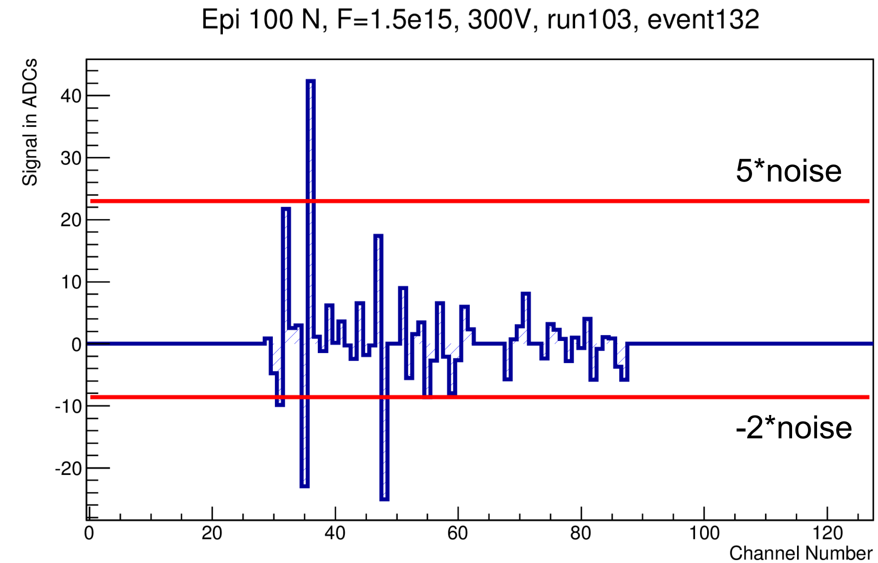
  → symmetric η! :-)



More Filtering

- Feature of irradiated HPK n-bulk sensors: non-gaussian noise
  - Show-stopper for a tracking detector
  - Try to remove this by another filter
  - Mask all signals > 150 ADCs
  - p-bulk data has ~ 0.2 clusters/event → mask events with > 1 cluster
  - If a seed candidate has a neighbour with a large negative signal → probably a discharge → mask event
  - Finally recluster data
- Try to salvage some n-bulk runs

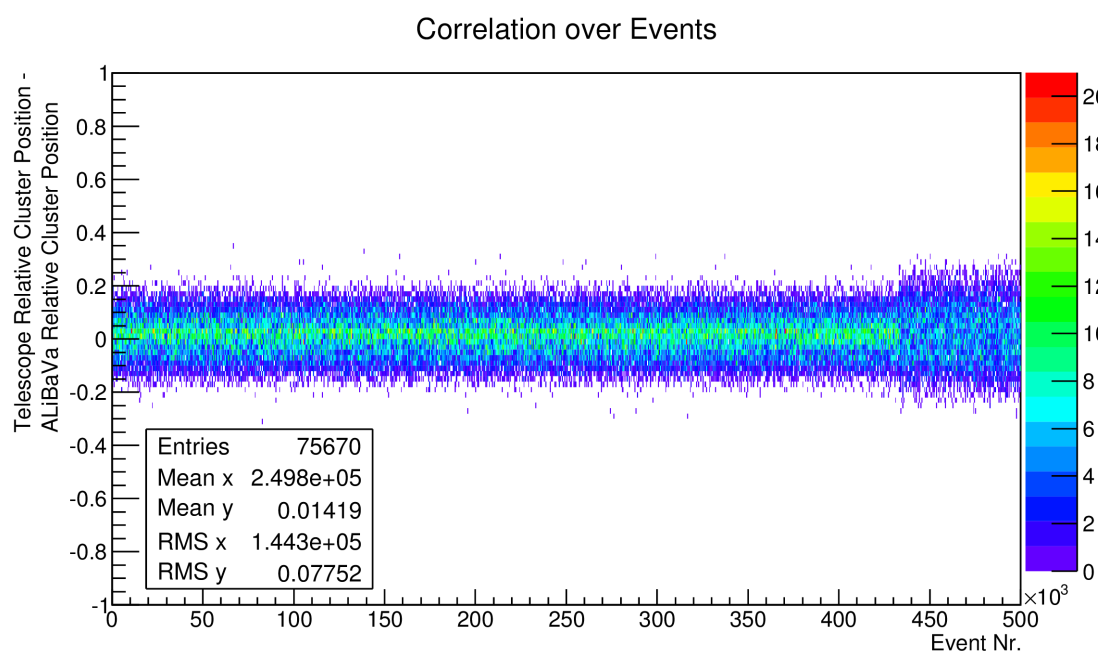




Merging Data Streams

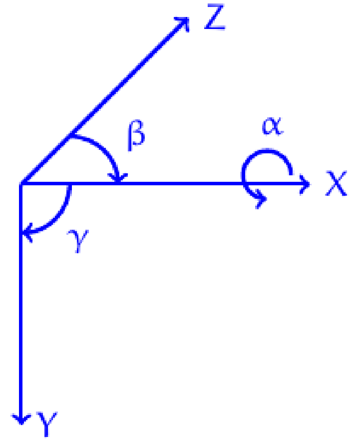
- Both ALiBaVa and Telescope data streams are in sync on an event basis and have the same number of events
  - Create new `TrackerPulse` and `TrackerData` collections with clusters from both systems

- Check synchronisation over time / events:
  - In some runs severals telescope runs for one ALiBaVa run → does concatenation work?
  - Beam rate not always steady, sometimes hour-long gaps (e.g. user access)
  - Example: loss of sync after ~430k events

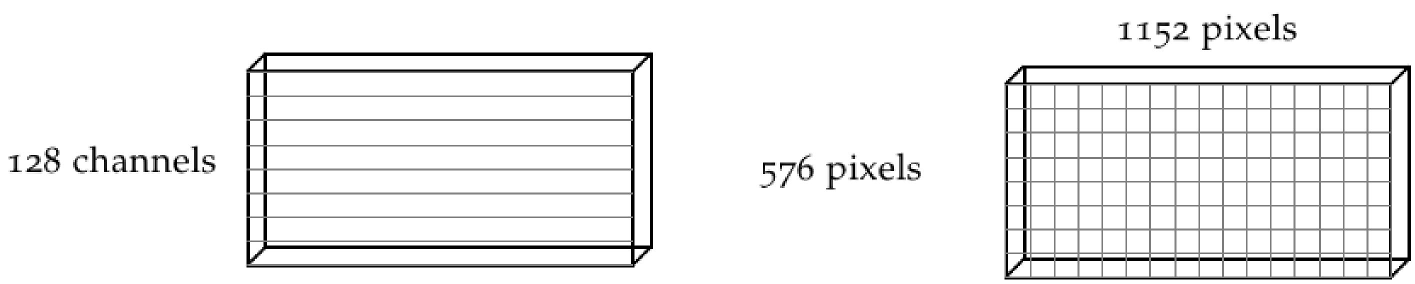


Hitmaking

- First 'major' change:
  - `EUTelHitmaker.cc` expects ZS binary data and disregards actual analog CoG position
  - Fixed by an optional processor parameter to identify analog planes by their `sensorid` in the cluster
  - Has this been fixed in newer versions?

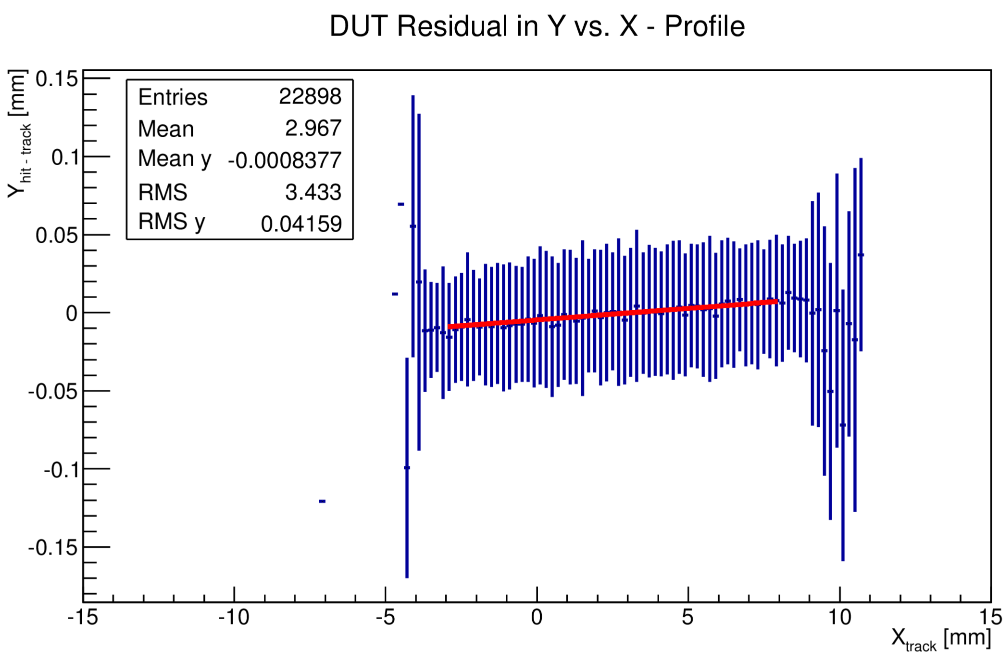


- Setup rotations are applied:
  - Turns in -α direction
- Rough prealignment is performed



Alignment Strategy

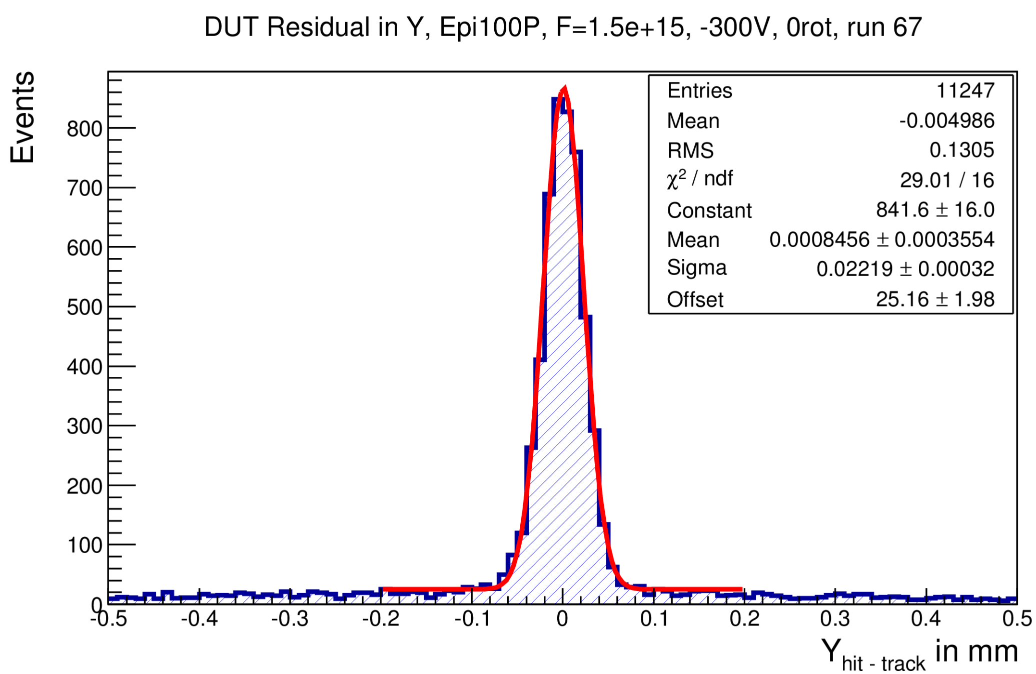
- 5 iterations with DAF fitter:
  - Narrow down residual cuts and resolutions
  - Fix front and end plane, all Z coordinates
- Use DUT Residual in Y vs. track X for additional γ rotation
  - Since no real DUT X information this can be a critical rotation
  - Inclination gives γ
  - Rotation not necessarily around plane center
  - additional X/Y shift afterwards
  - 2 iterations
- With final Z-alignment: 10 steps



Track Search

- Rewrite of `EUTelFitTuple.cc`:
  - Search tracks on upstream planes only, extrapolate to DUT
  - Translate track impact point back into DUT channels → undo alignment & hitmaker steps
  - Write tracks and reconstructed DUT sensor data to a `ROOT Ntuple`

- Track resolutions:
  - Assuming σ_{tele} = 5 μm, a DUT resolution of σ_{DUT} = 21.62 μm can be achieved (σ_{binary} ~ 22.55 μm)
  - Improves for rotated data
  - But: not for all runs :-(



Track Display

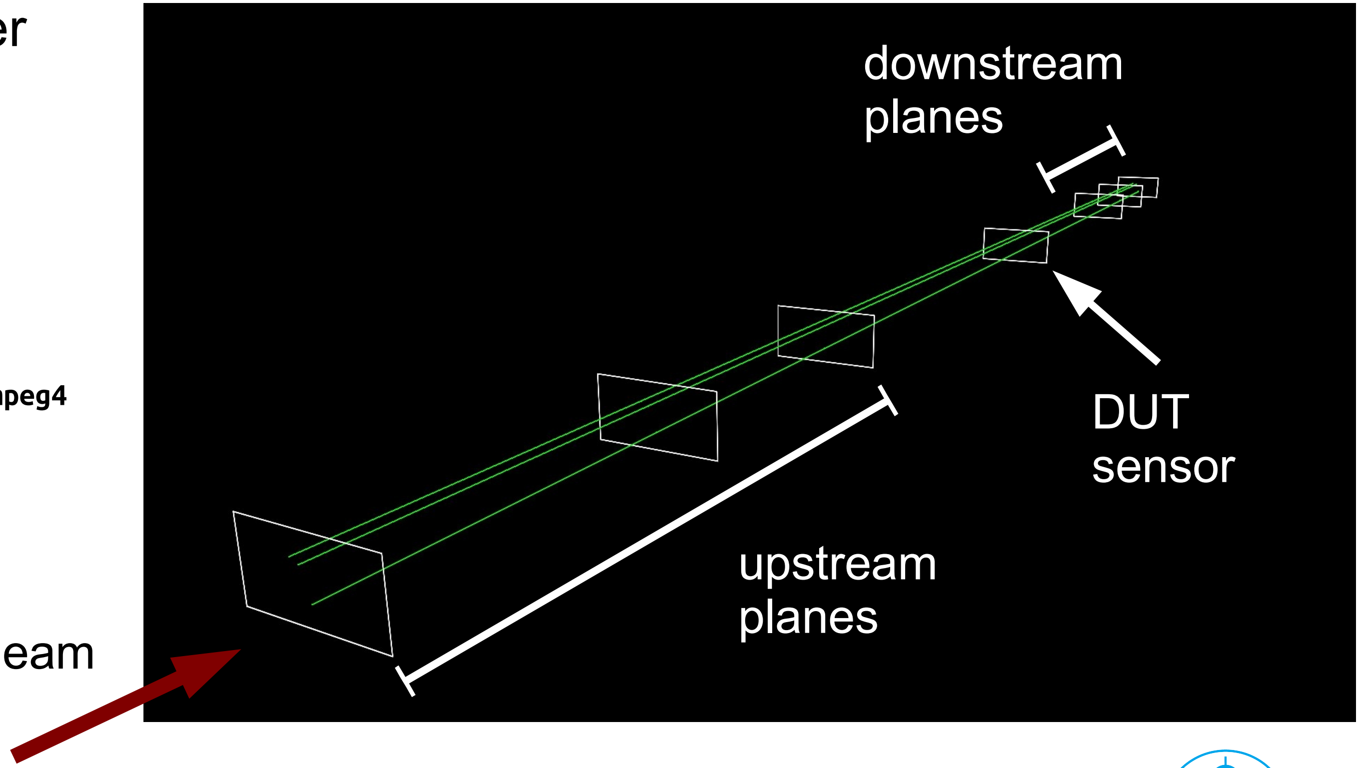
- CED to visualize tracks
  - Rebuild your EUTelescope installation with (more details in backup):

```
ilcsoft.install(CED(CED_version))
ilcsoft.module("CED").envcmake['CED_SERVER']='ON'
ilcsoft.install(CEDViewer(CEDViewer_version))
```

- EUTelEventViewer processor

FYI: screen recording in linux:

```
avconv -f x11grab -r 25 -s 1920x1080 -i :0.0 -vcodec mpeg4 file.avi
```

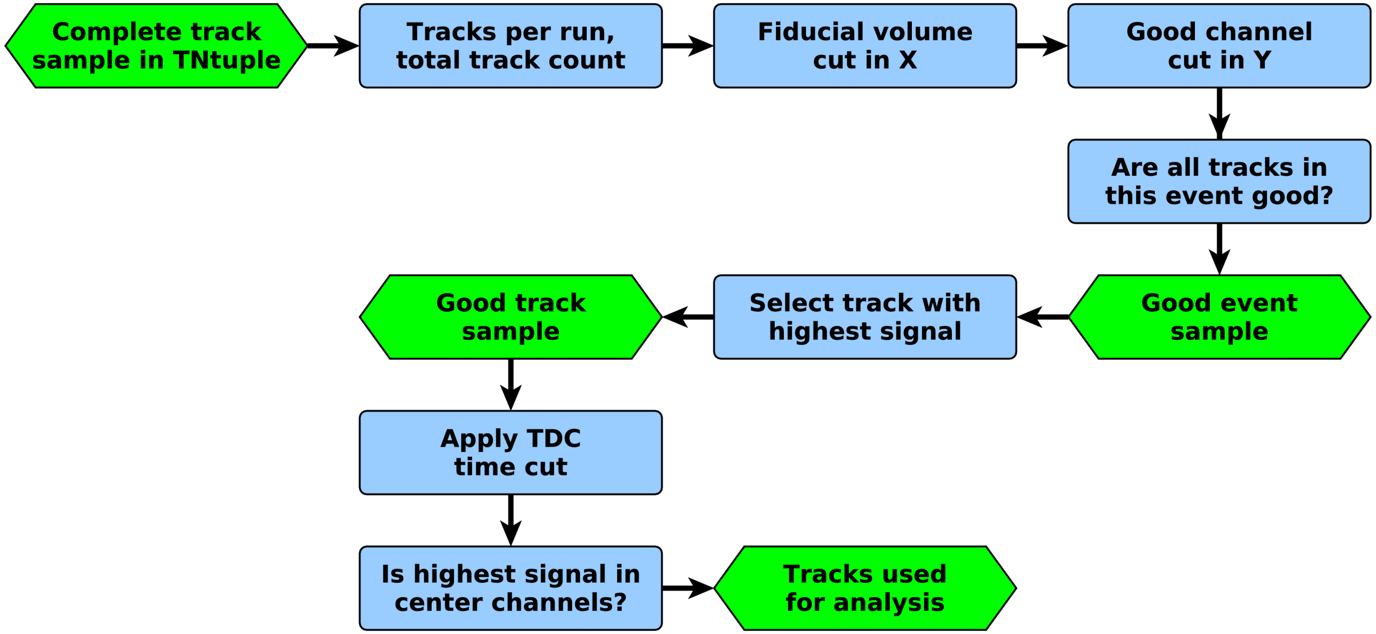


Sensor analysis with tracks from EUTelescope

Track Selection

- Many analysis steps done already by EUTelescope processors
  - Run on track `NTuple`
- Telescope is much slower in read out (115µs) than ALiBaVa (25ns)
  - Many tracks in a (telescope) event → much less in the ALiBaVa

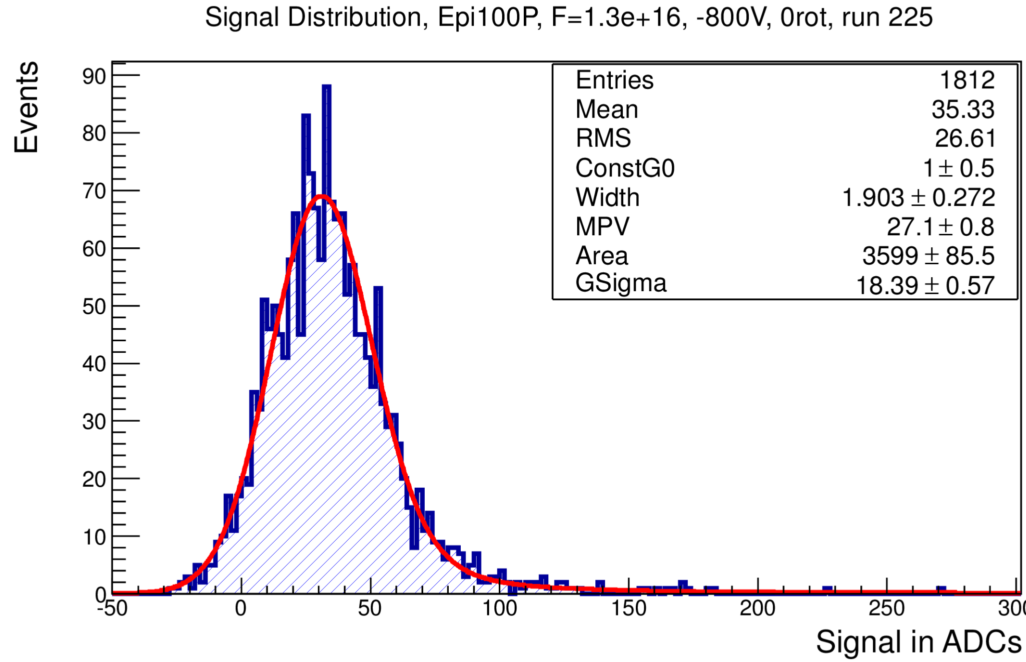
    → select the track that the ALiBaVa has most probably measured



Results: Charge Collection

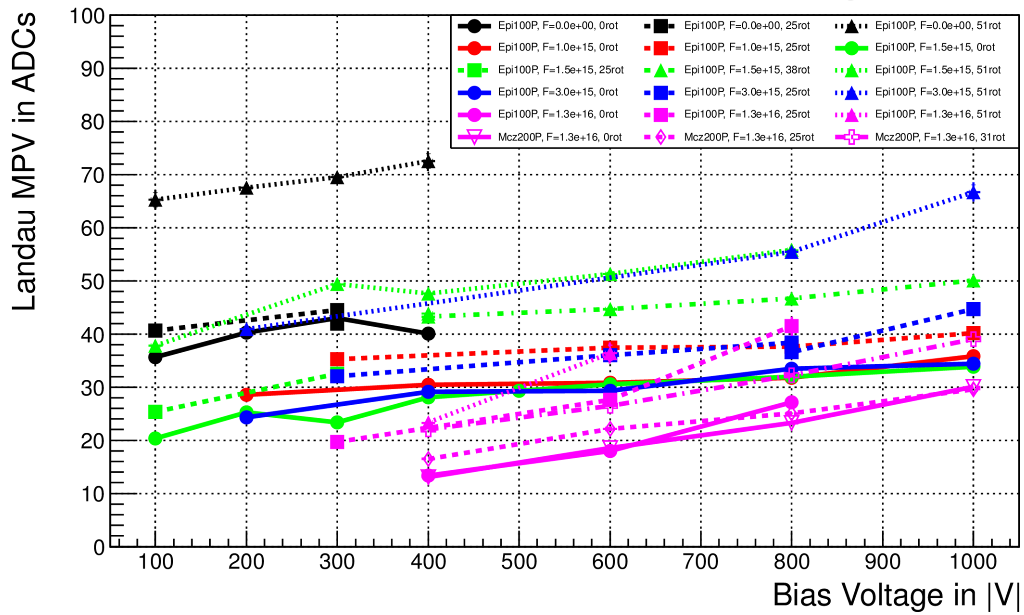
> **Signal definition:**

- Consider signal in 5 channels around extrapolated track impact
- Sum channel ADCs, fit Landau-Gaussian



> **Charge collection**

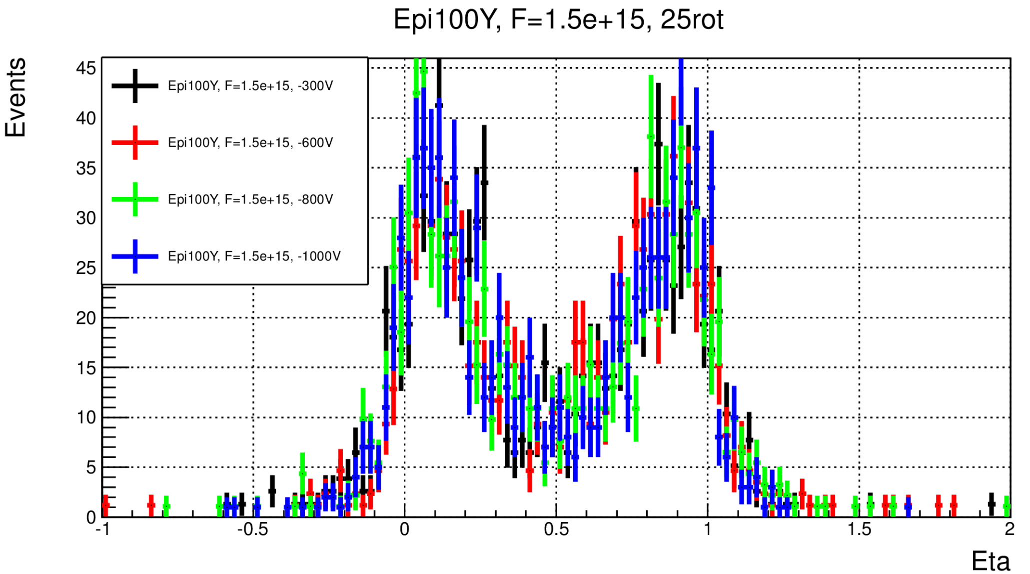
- Increase with beam incidence angles and bias voltage
- For $\Phi < 3e15\ n_{eq}/cm^2$:
  ~85% of unirr. signal collected
- For $\Phi = 1.3e16\ n_{eq}/cm^2$:
  ~70% of unirr. signal collected

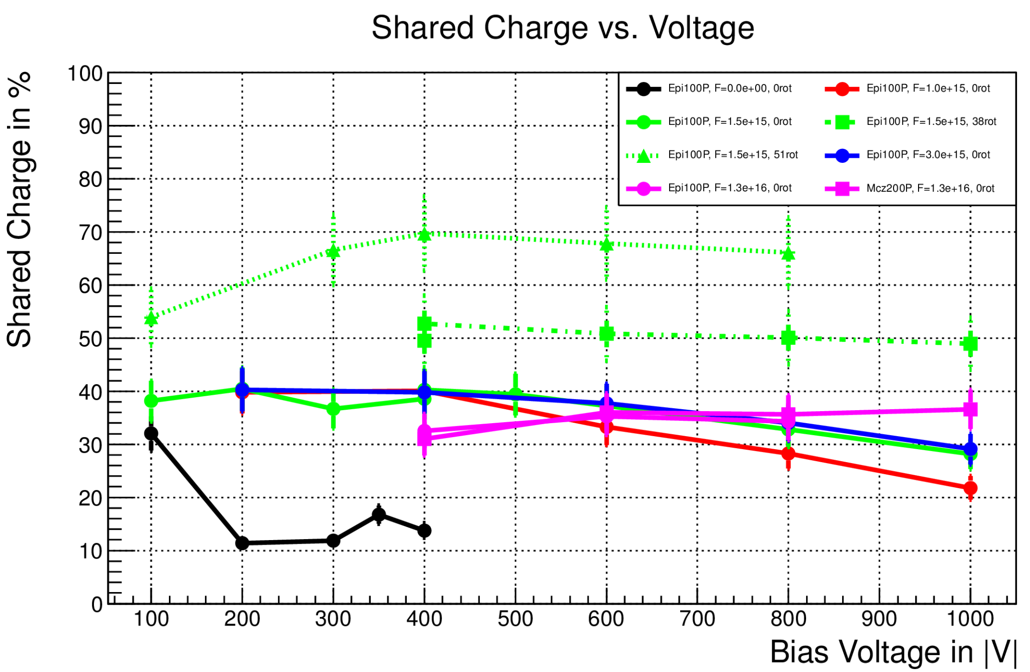


Results: Charge Sharing

- Charge sharing:
  - Integrate η-distributions
  - Consider entries > 0.2 and < 0.8 as 'shared charge'
  - Sharing increases with rotation and fluence, declines at higher voltages
  - Alternate definition:

    count channels next to track impact over a threshold → similar results, but introduces cuts





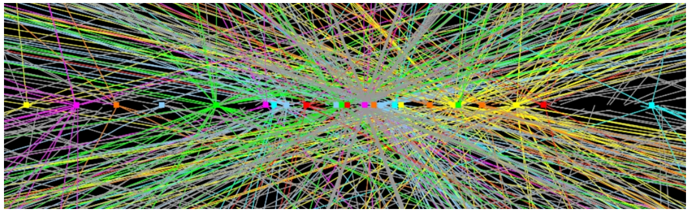
Summary

- 3 runs in TB21, 2 parasitic runs in TB22 resulted in ~300h of data taking
  - Setup installed, connected and running within minutes → real plug and play :-)
- Successful integration of an analog strip sensor a digital pixel telescope
- With custom EUTelescope processors many analysis steps can be performed during reconstruction
- Results show epitaxial silicon is a promising material for future detectors

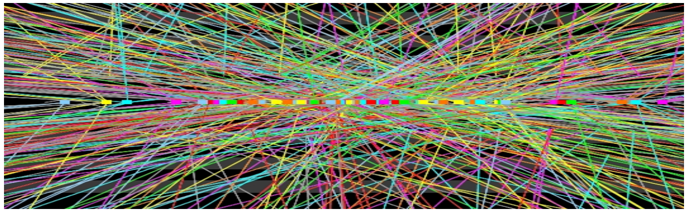
Backup

Phase-II Upgrade Motivation

- In 2023: upgrade of the LHC to a High Luminosity-LHC
  - Increase the luminosity by a factor of 5
  - Even harsher radiation environment



ATLAS $t\bar{t}$ event simulation: <μ> = 40

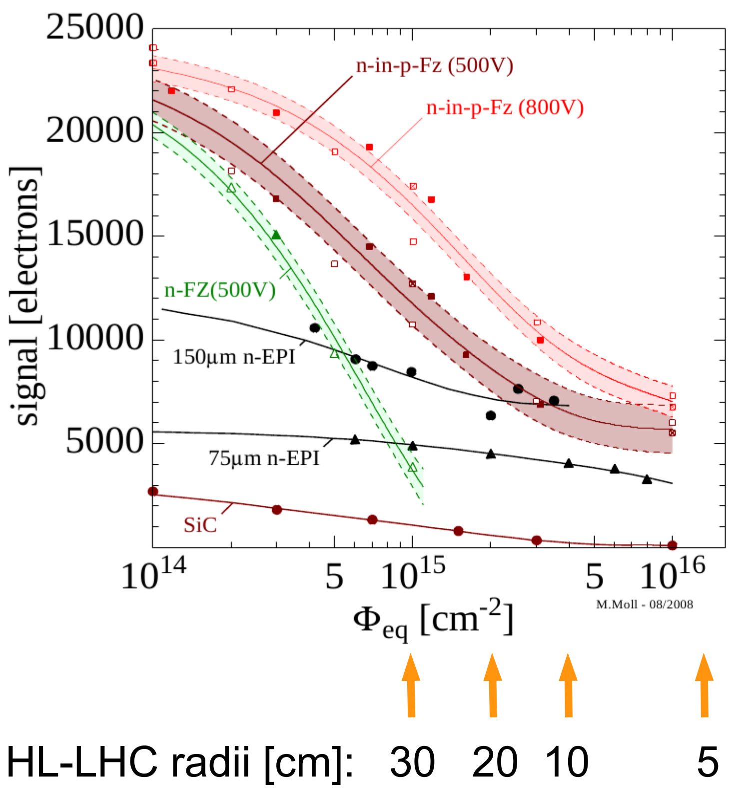


ATLAS $t\bar{t}$ event simulation: <μ> = 140

- LHC experiment trackers need to be upgraded to maintain excellent current performance
  - Improve granularity to keep low detector occupancy
  - CMS: tracker to contribute to the Level-1 trigger → new module concept
  - Develop radiation-hard sensors → new sensor technologies

Expected Fluences

- Is epitaxially grown silicon a radiation-hard sensor material?
  - Could it be used for a HL-LHC pixel sensor?
- Current CMS tracker:
  - Design fluence: 4e14 n_{eq}/cm^2 @ r = 20 cm
- HL-LHC tracker:
  - Expected fluence 2e15 n_{eq}/cm^2 @ r = 20cm
  - Fluences > 1e16 n_{eq}/cm^2 for pixel sensors
- Desired signals:
  - 6000 e^- for strip regions
  - 3000 e^- could be sufficient for pixel



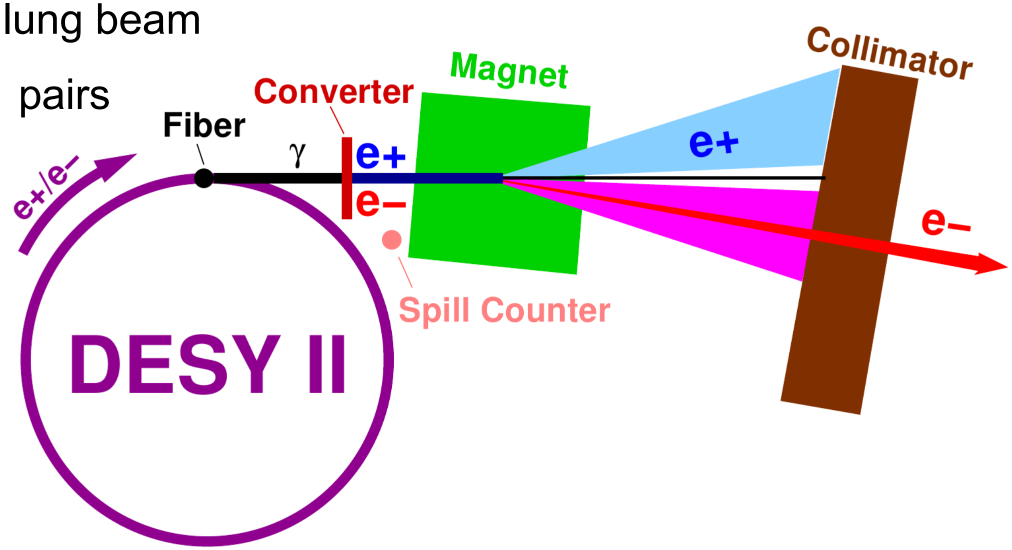
Sensor List

- List of successfully measured sensors
  - Epi = epitaxial
  - FTH = float-zone
  - MCZ = magnetic czochralski

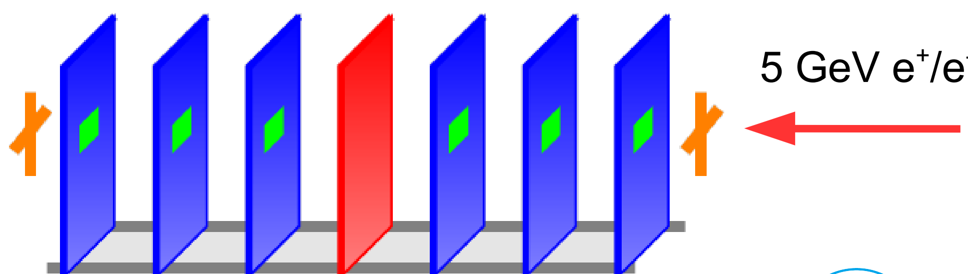
Fluence	Epi 100P	Epi 100Y	Epi 100N	Epi 70N	FTH 200Y	MCZ 200P	MCZ 200Y	MCZ 200N
0	√	√		√				
1e15 Los Alamos	√	√	√	√				
1.5e15 CERN	√	√	√					
3e15 CERN	√		√					
1.3e16 CERN	√	√	√		√	√	√	√

DESY-II Testbeam Setup

- Test beam from DESY-II synchrotron
  - Carbon fibre generates a bremsstrahlung beam
  - Metal plate converts photons to e^+/e^- pairs
  - Dipole magnet spreads out beam
  - End collimator cuts out final beam



- DESY beam telescopes
  - Series of parallel Si sensor planes inserted into the particle test beam
  - Reconstruct particle tracks through planes wrt. time and position
  - Use gained information to evaluate *device under test* (DUT) performance



- Use telescope to investigate epitaxial sensors

Event Viewer

- How to install and use event-viewer:

  1) Go to your `ilcinstall` directory. In `examples/eutelescope`, edit `release-standalone.cfg`

     Uncomment or make sure these lines exist:

     ```
     ilcsoft.install( CED( CED_version ))
     ilcsoft.module("CED").envcmake['CED_SERVER']='ON'
     ilcsoft.install( CEDViewer( CEDViewer_version ))
     ```

  2) Rebuild `ilcsoft`: Make sure `$ILCSOFT` is known, e.g. do "`export ILCSOFT=/path/to/where/you/are`", in the ilcinstall directory do: `./ilcinstall -i examples/eutelescope/release-standalone.cfg`

  3) Wait...

  4) Source the environment: in `ilcsoft/VERSION/Eutelescope/VERSION`: `source build_env.sh`

  5) Start the display: `glced &`

  6) In a new terminal, repeat 4) and run the eutelescope event-viewer: `jobsub -c config.cfg -csv runlist.csv event-viewer RUNNUMBER`

  7) `Enter` moves to the next event, `q` quits.

     Pro tip for the display window: zoom in and turn the background to black.

  8) The event-viewer needs hits and tracks. The collections are specified in the steering file or in the `config.cfg`. Make sure these exist.