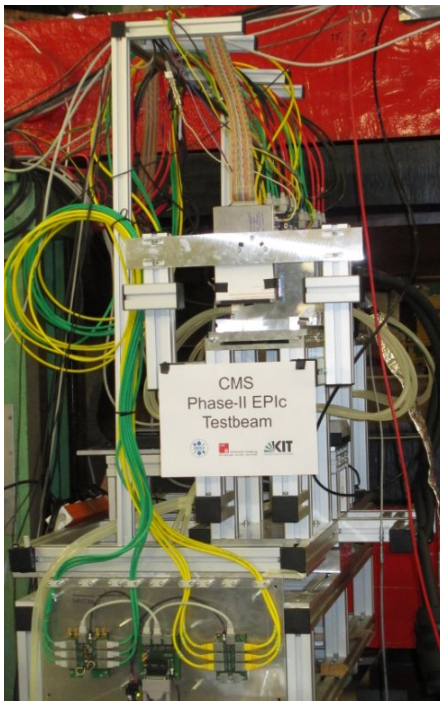
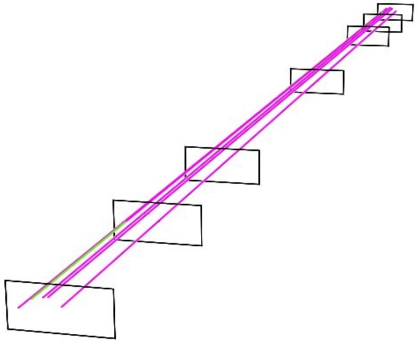
The CMS EPIc Test Beams

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





Thomas Eichhorn
Telescopes and Testbeams WS
01.07.2014

Outline

- Testbeam setup & measurements
  - Epitaxial strip sensors
  - ALiBaVa & telescope setup

- Integrating an analog strip sensor into a digital pixel telescope
  - ALiBaVa processors for EUTelescope
  - Challenges and changes to EUTelescope

Testbeam 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?

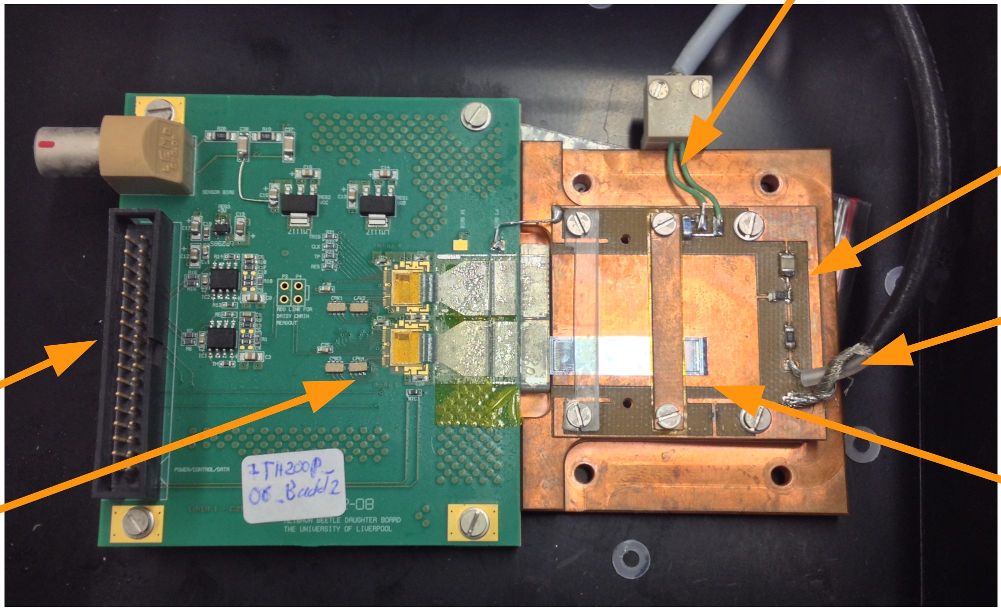
- Sensors / irradiations:
  - 19 epitaxial, float-zone and magnetic czochralski silicon strip sensors from a CMS vendor run: 64 strips, 80μm pitch, thickness 70μm – 200μm,
  - Proton irradiations up to $\Phi = 1.3e16\ n_{eq}/cm^2$ at CERN and Los Alamos
  - Both n- and p-type material, with p-stop and 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

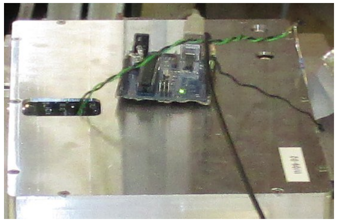
- A Liverpool Barcelona Valencia Read-out System
  - DAQ in-a-box system based on LHCb Beetle chip to read out strip sensor data
  - Motherboard for PC connection and data processing
  - Sensor wirebonded to Daughterboard with Beetle ROC
  - Analog read-out of the 2 * 128 channels every event
  - Serial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

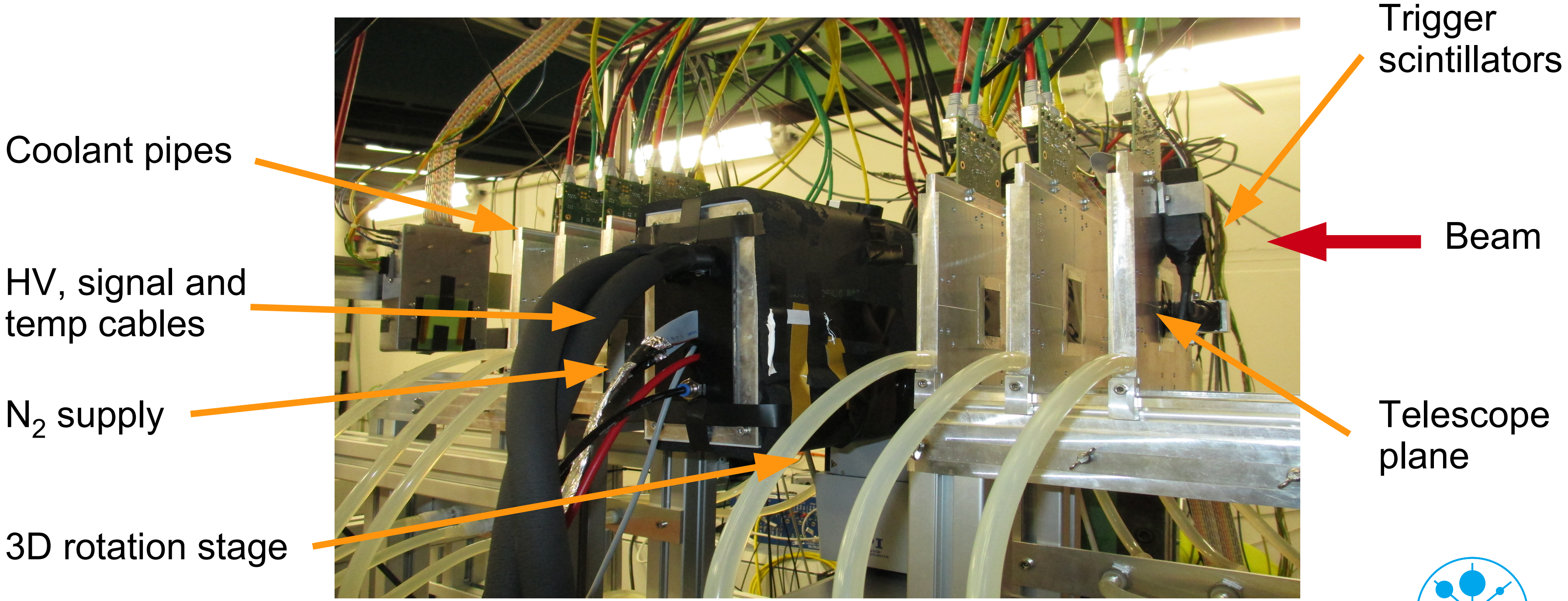
- Coldbox to mount setup in beam telescope



ALiBaVa Setup in the Testbeam

- 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:

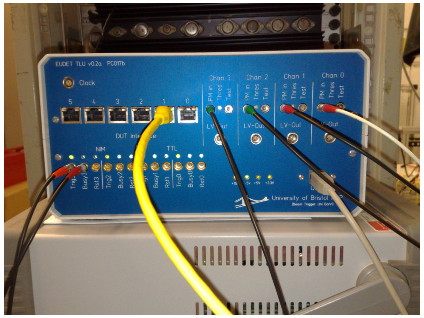
- Different read-out speeds & memory/buffer sizes
- Two clocks
- How to match events offline?

> 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 testbeams

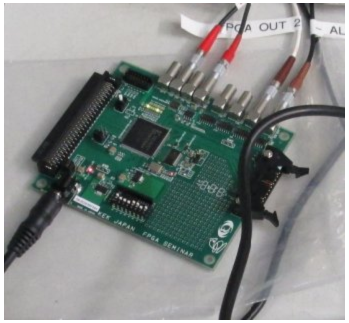
PMT Hit

Trigger Logic Unit:
Coincidence of all 4 PMTs → Trigger out



Gate generator

Trigger to telescope



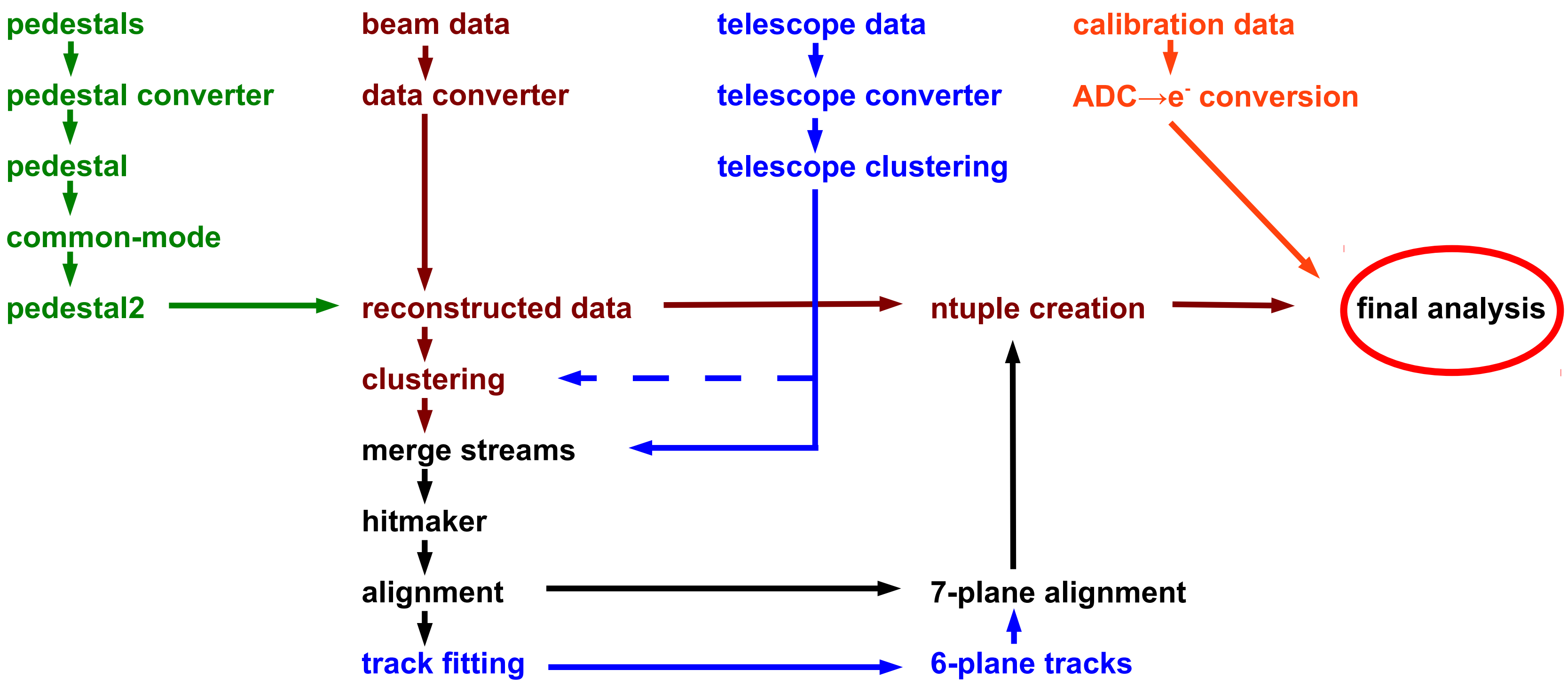
FPGA:
If (n=1000)
→ send busy

Trigger to ALiBaVa

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



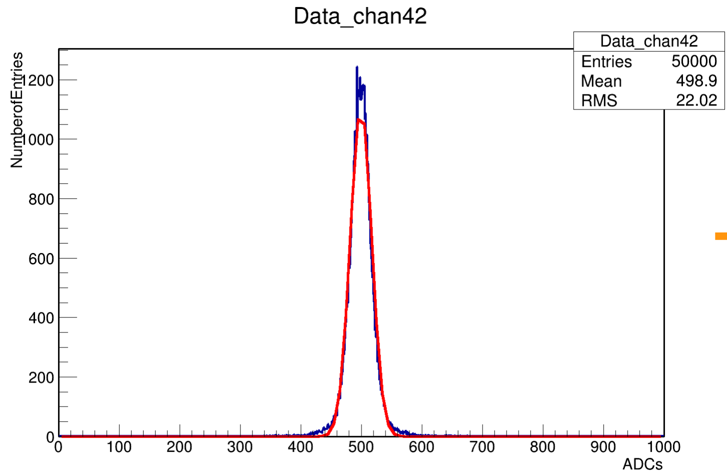
- EUTelescope v00-09-02 and `datura-notdut` example as a basis
  - Additional processors & some modifications

Data Converter and Telescope Clustering

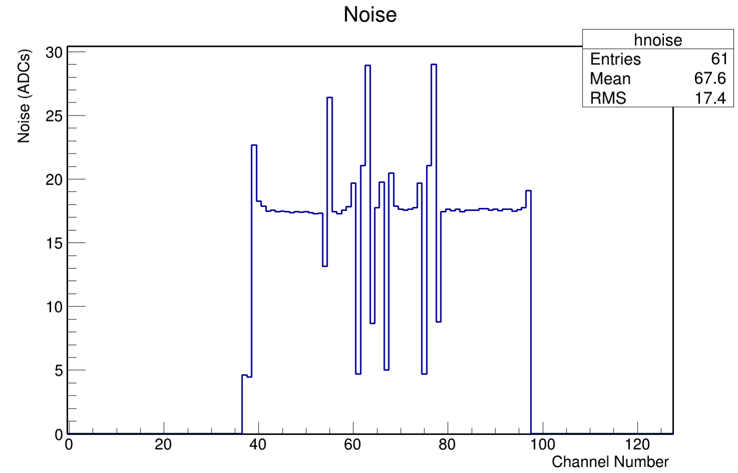
- ALiBaVa raw data stored in a binary format
  - Convert to LCIO:Raw
- Use existing telescope analysis chain (à la datura-nodut example)
  - Convert to LCIO:Raw
  - Clustering and hot pixel supression
- Multiple telescope runs for one ALiBaVa run:
  - Concatenate option from jobsub:
  - Modify `@LCIOInputFiles@` in the steering file → `@concatinput@`
  - `jobsub -c config.cfg --concatenate -option concatinput=.../output/lcio/run@RunRange@-converter.slcio -csv runlist.csv telescope-clustering-concat <startrun>-<endrun>`

Reconstruction – Pedestal

- Pedestal: signal in a channel w/o an event

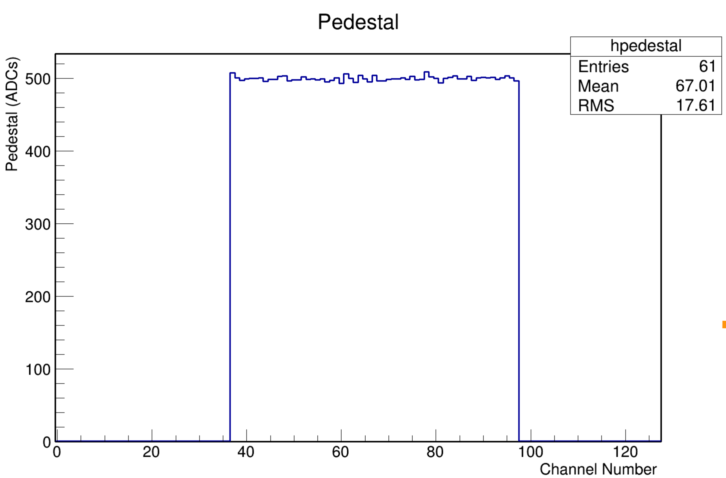


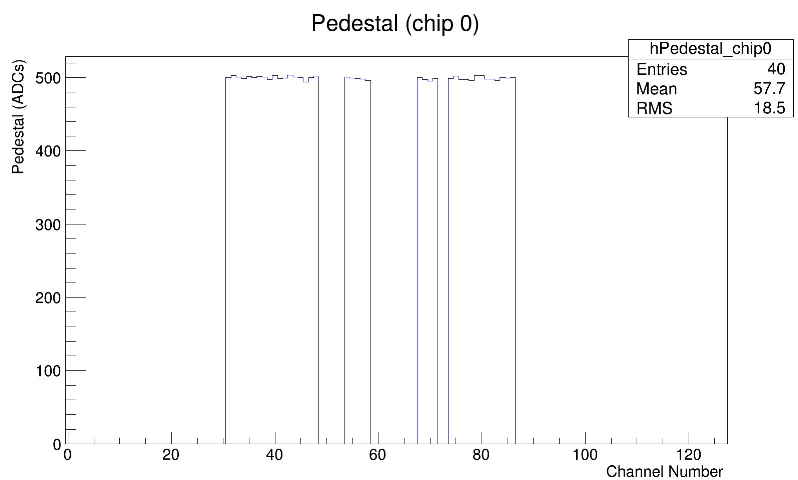
width



mean

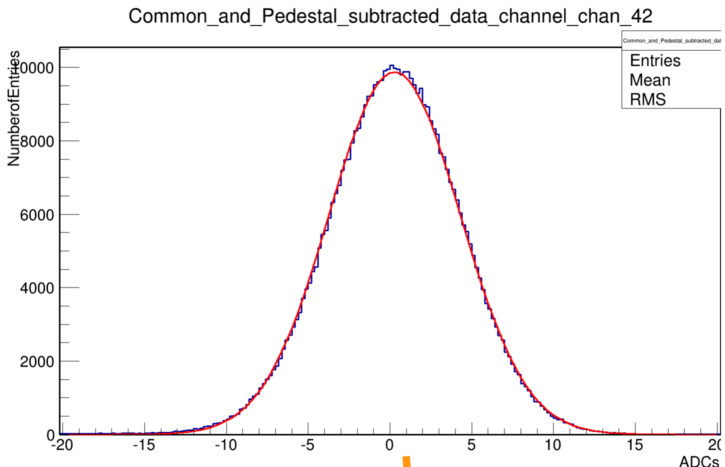
Mask channels with bonding issues

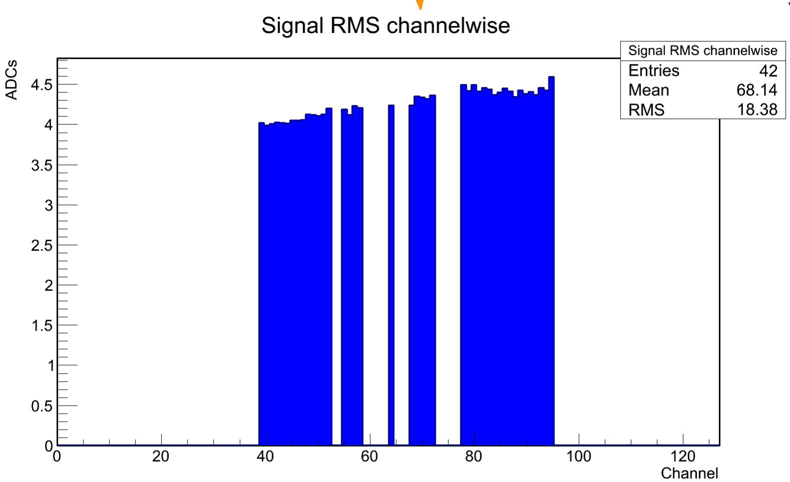


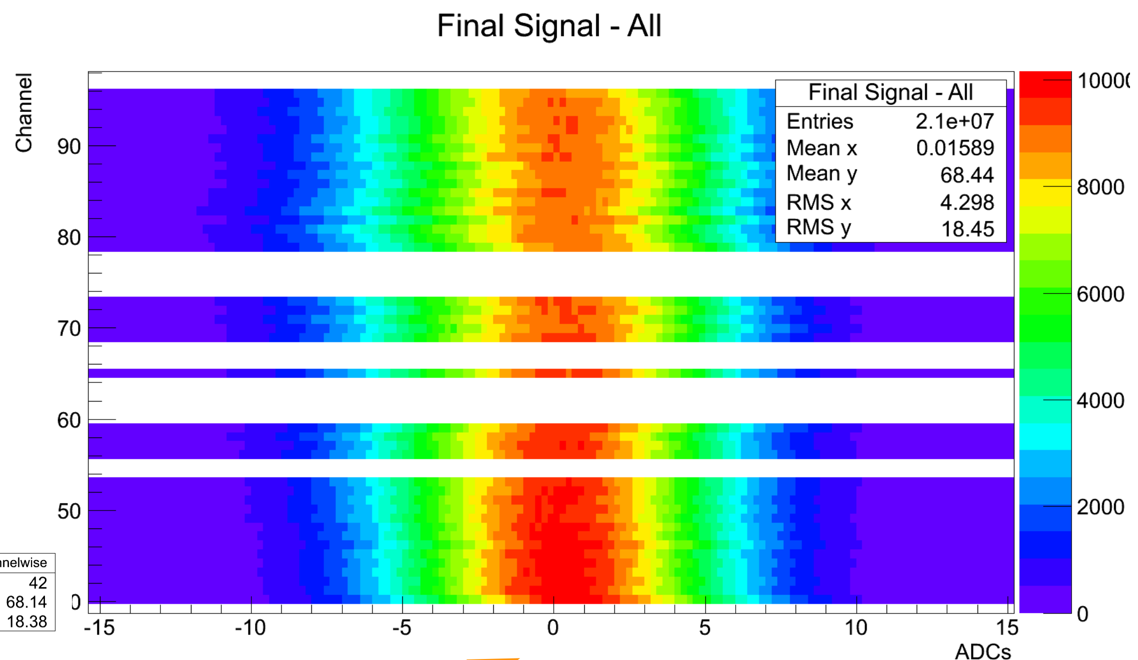


Reconstruction – Common Mode

- Common mode: synchronous movement of all channels in an event
  - Final RMS gives sensor noise level

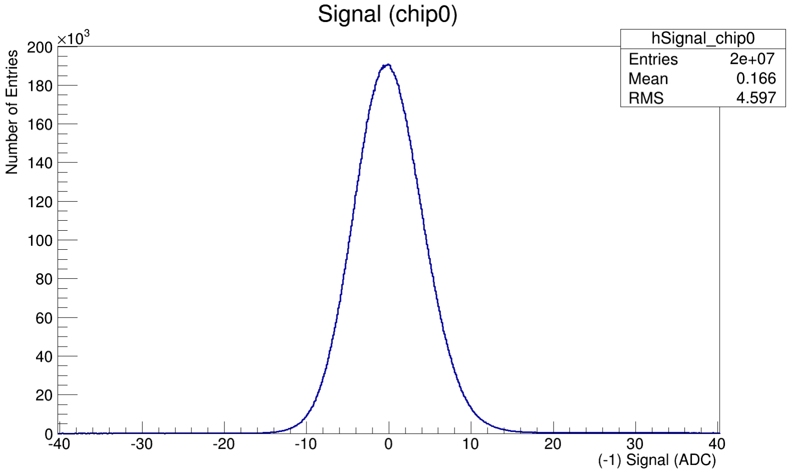


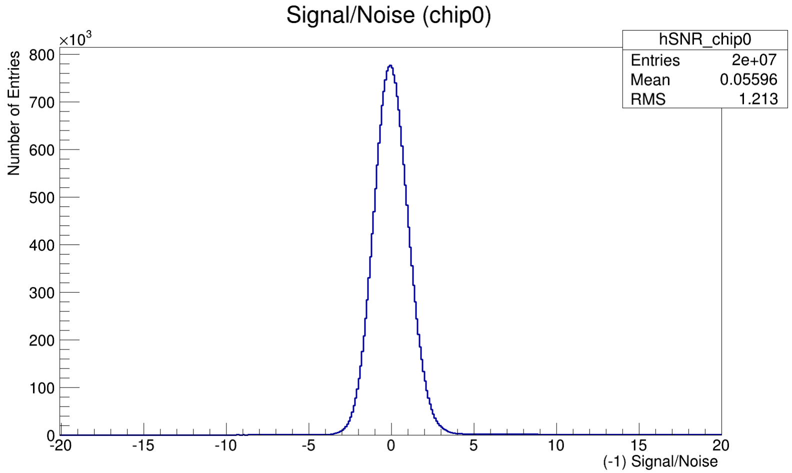




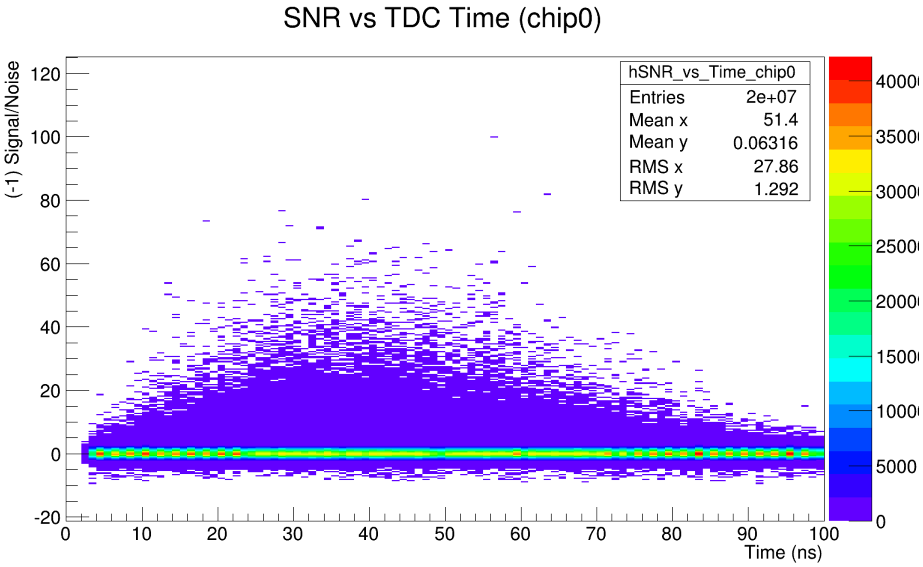
Reconstructed Data

- After pedestal subtraction and common-mode correction:
- Collection(s) of LCIO:TrackerData





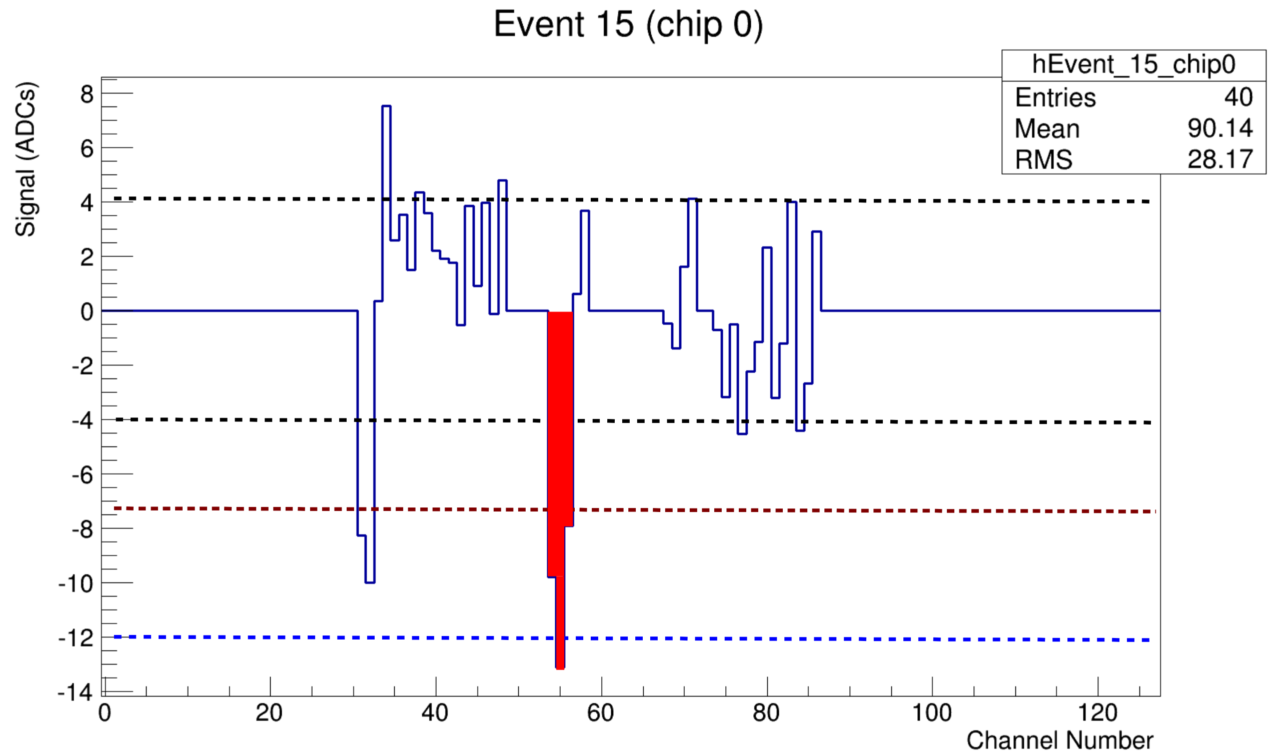
- TDC time cut can be applied
  - Here e.g.: 30ns – 50ns



Strip Clustering

- Search for clusters on reconstructed data
- Clusters only used to create hits → alignment
- S/N cuts for seed (3) and cluster (1.85)

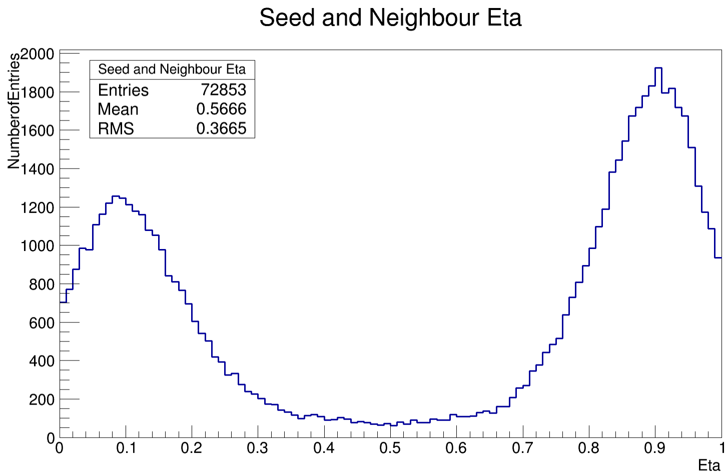
  - Noise ~ 4 ADCs
  - Cluster cut: ~ 7.4 ADCs
  - Seed cut: ~ 12 ADCs

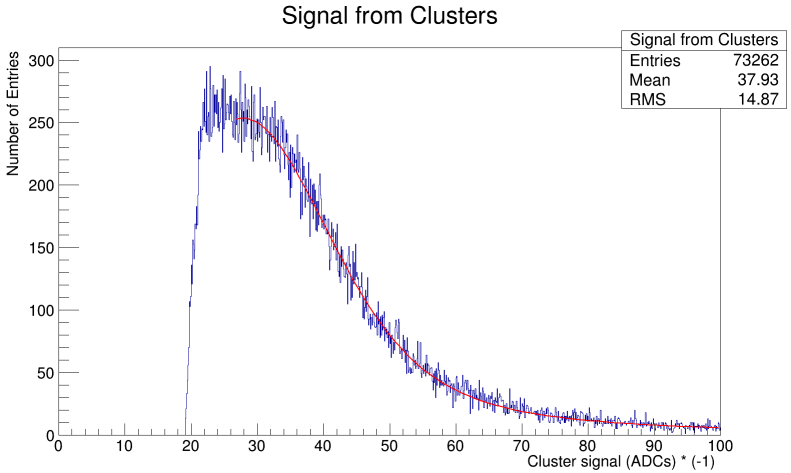


p-type sensor: negative signals

Strip Clustering – cont.

- η-distribution: $\eta = Q_{left} / (Q_{left} + Q_{right})$





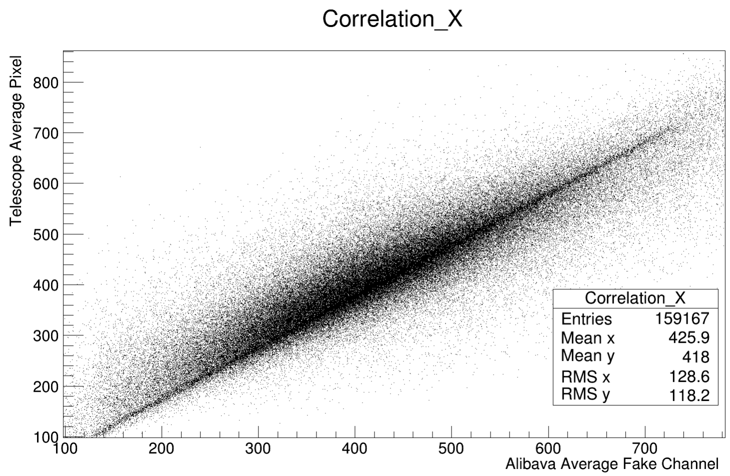
- Make merge easy: write same cluster format as telescope does:
  - `CellIDEncoder <TrackerPulseImpl> zsDataEncoder ("sensorID:5, clusterID:12, xSeed:12, ySeed:12, xCluSize:5, yCluSize:5, type:5", clusterCollectionName);`
  - `CellIDEncoder <TrackerDataImpl> idClusterEncoder ("sensorID:5, clusterID:12, sparsePixelType:5, quality:5", sparseClusterCollectionVec);`

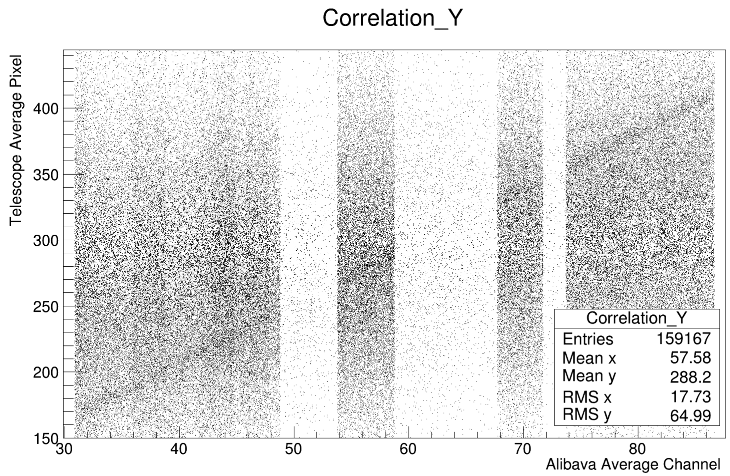
- Caveat: charge is stored as a `short` :-(
  - Clustersize: ~1.2 for 0° beam incidence angle, 1.5 if rotated to 51°

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
- EUTelescope and LCIO framework written for pixel sensors

  → 'fake' missing strip sensor coordinate from the average telescope cluster coordinate on neighbour plane in an event
- Correlations:



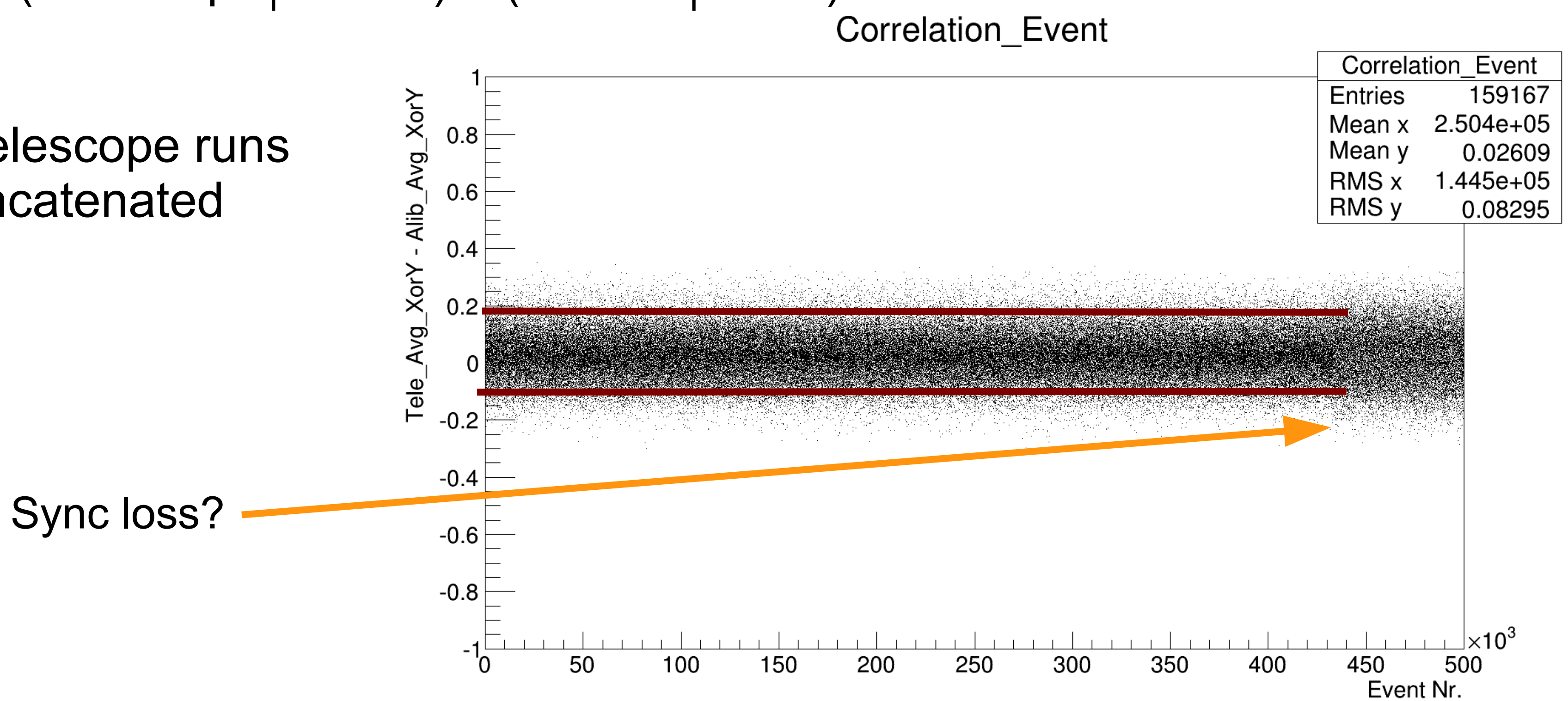


Merging Data Streams – cont.

- 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)

- $\Delta = (\text{Telescope}_Y / 1152) - (\text{Alibava}_Y / 128)$

- 4 telescope runs concatenated

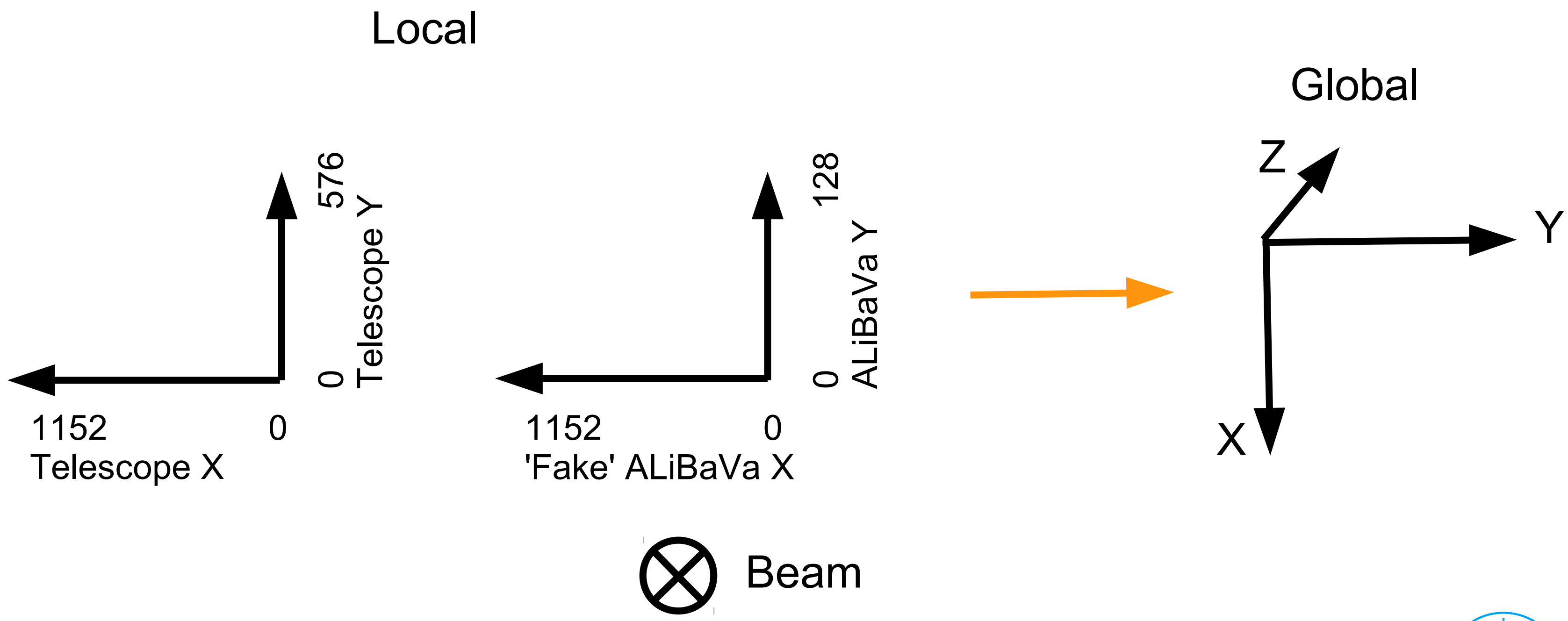


Hitmaking

- 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

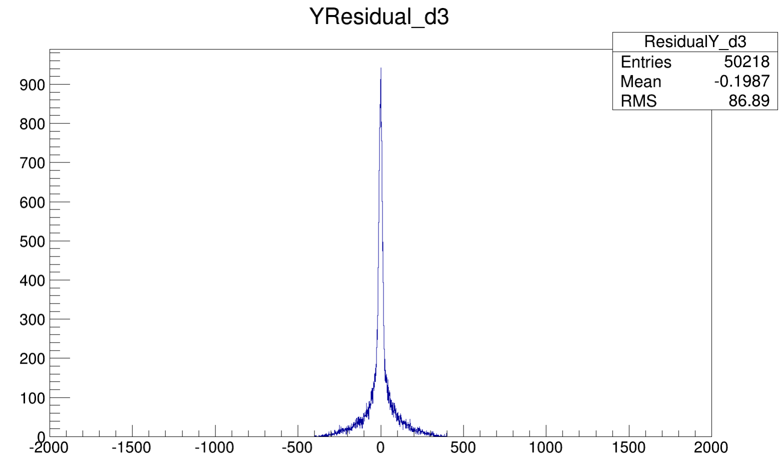
- Coordinate systems:

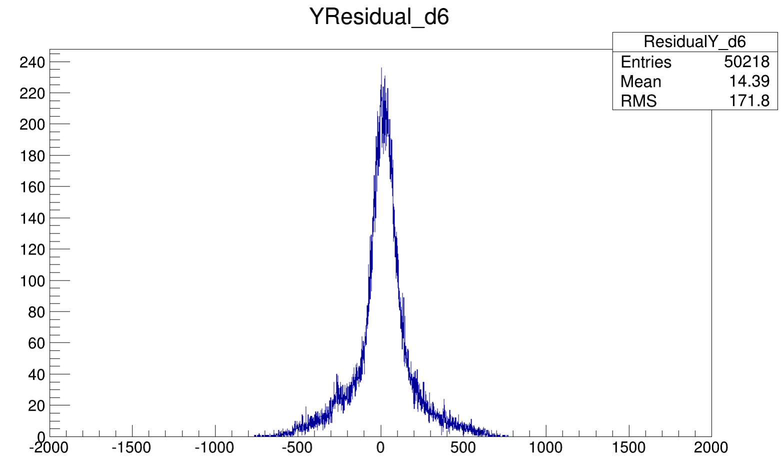
gl. pos = pitch * (- loc. pos + channels/2 - 0.5)



Alignment

- After hitmaking, perform 7-plane alignment
- Straight-line track search as input to MillepedeII
  - Fix front and end planes to avoid weak modes
  - Shifts in X,Y and α, β, γ rotations considered
- Residual cuts at ± 300µm for telescope planes, ±800µm for DUT plane
- Resulting alignment errors in 5D:
  - Telescope X/Y: ± 400nm
  - DUT X (fake): ± 85 µm
  - DUT Y: ± 1 µm
  - Telescope α, β, γ rotation: ± 0.001 rad
  - DUT α, β, γ rotation: ± 0.005 rad



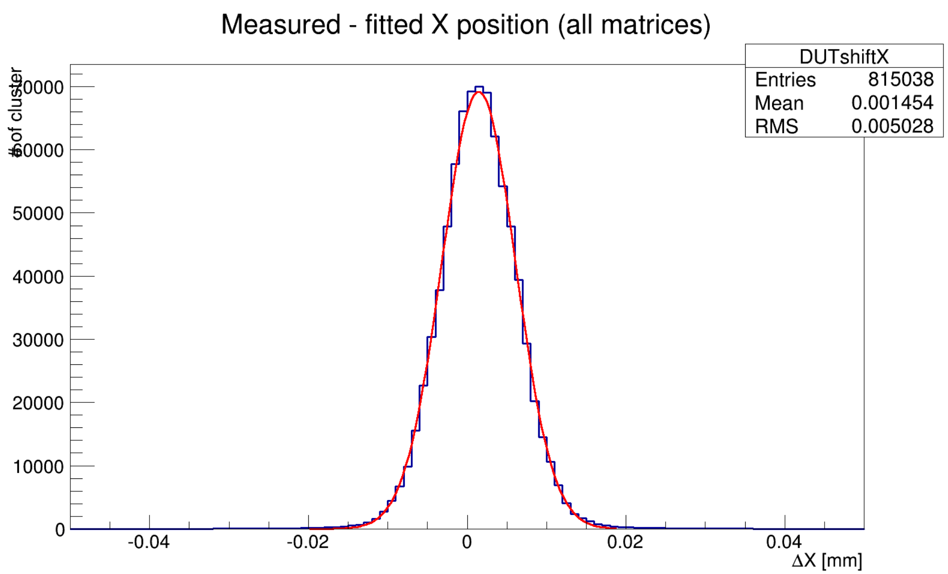


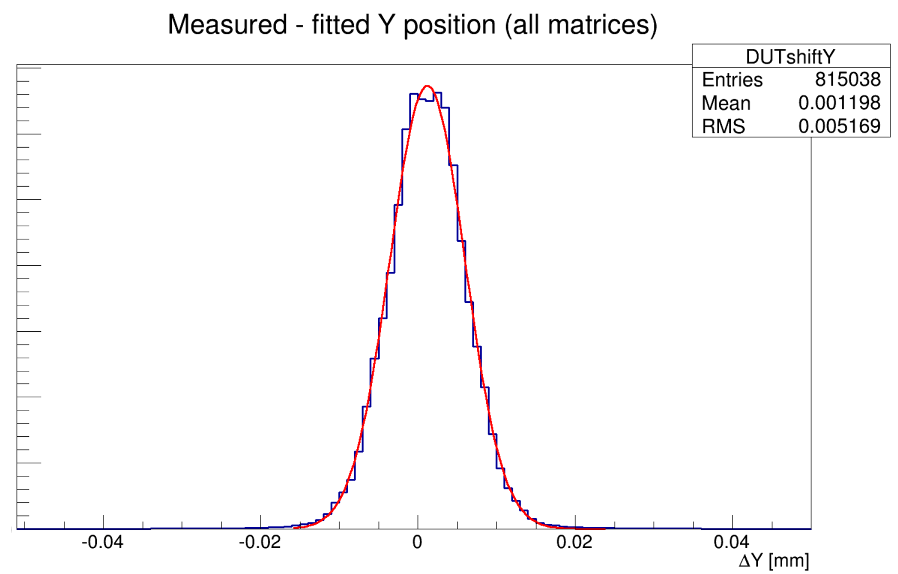
Track Fitting

- Telescope & DUT aligned → search for tracks in telescope only!
  - Cuts: $X^2 < 20$, require hit in all 6 ('of 7') planes

- At 5 GeV: ~ 1.55 Tracks / Event

- Resulting unbiased telescope residuals: σ_{meas} ~ 5 μm





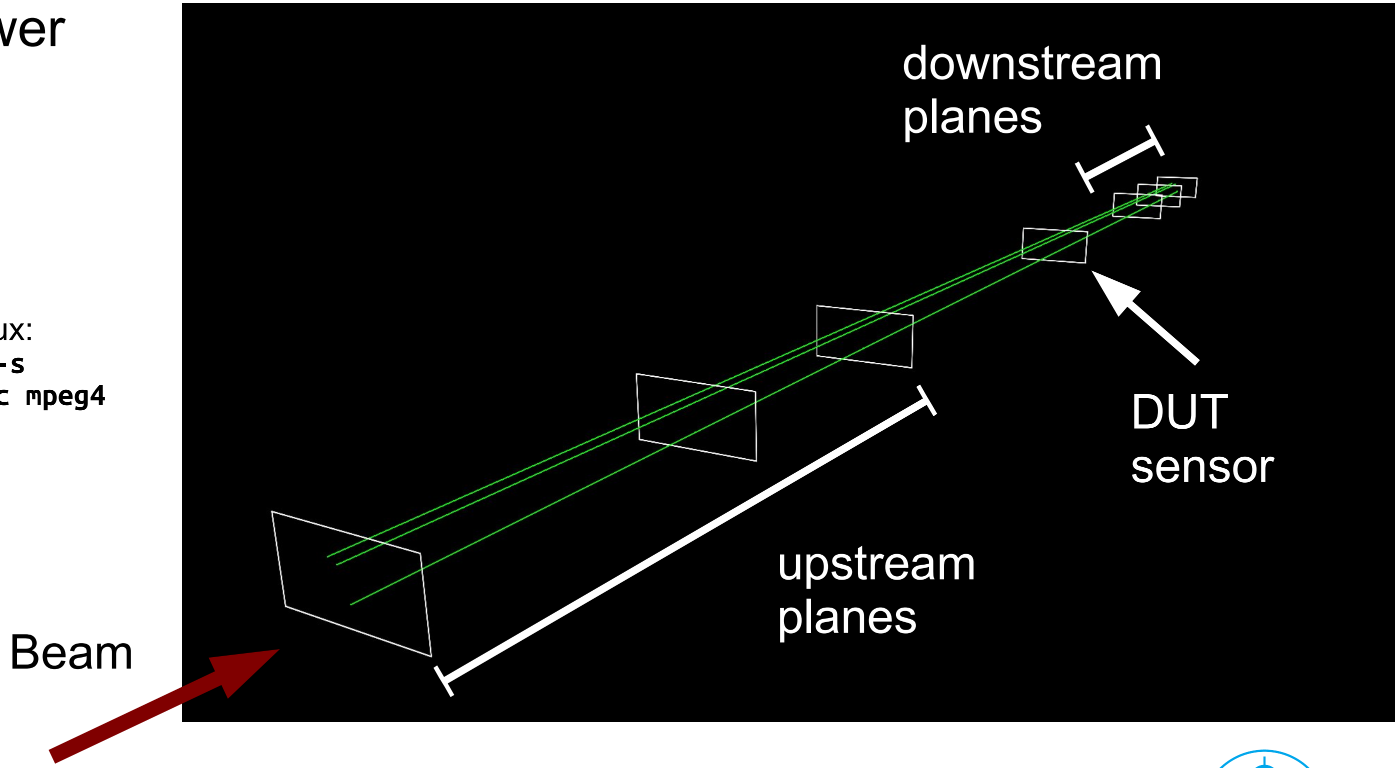
Track Display

- 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
```



NTuple filling & ALiBaVa reconstruction

- Rewrite EUTelFittuple.cc
  - Write telescope hits and tracks to ROOT Ntuple
  - Include the reconstructed DUT ADC data

- Less hits on DUT than in telescope (noise, high cluster cut, etc.)

  → Predict track impact position

  → Look into DUT data at impact position

- Impact position:
  - Assume straight line track between telescope planes before and after the DUT
  - Calculate expected hit position of track vector with (rotated) DUT plane
  - Revert rotations, alignment and 'hitmaker' to get expected pixel/strip position
  - Include this information in the Ntuple

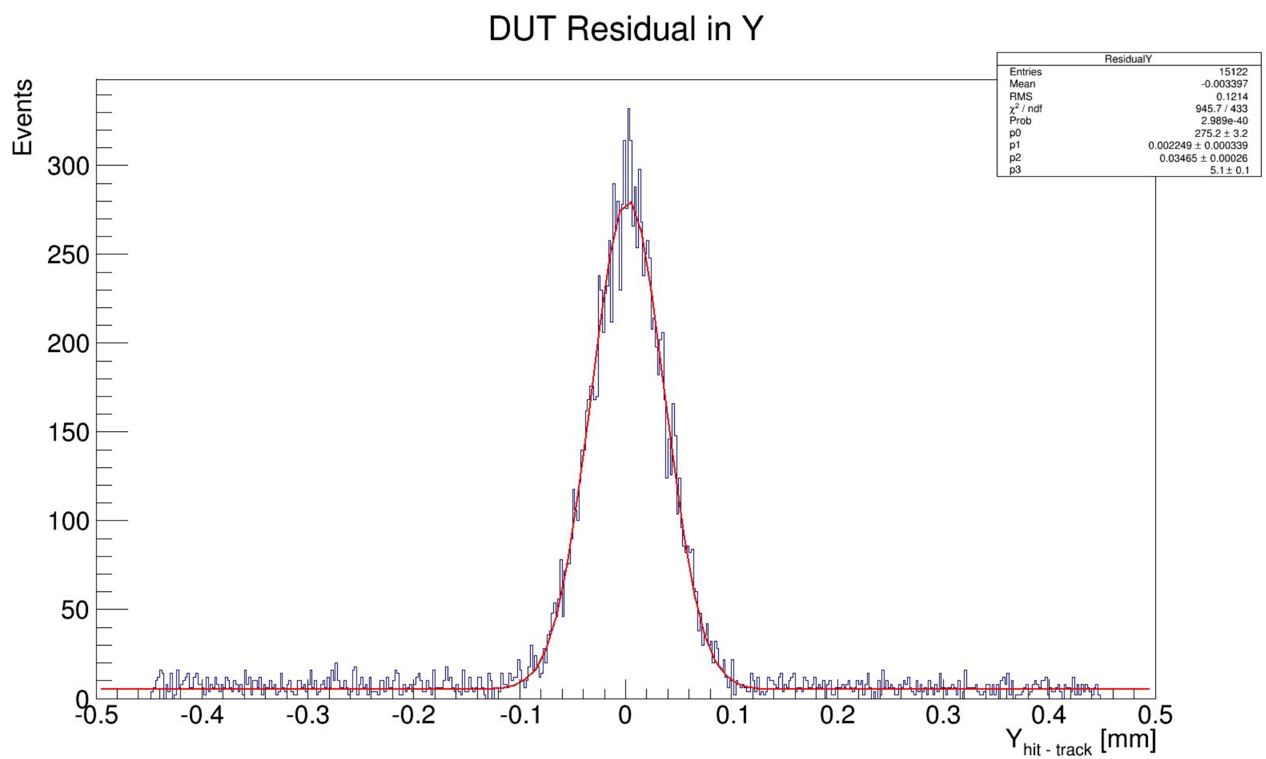
NTuple filling & ALiBaVa reconstruction – cont.

- DUT resolution: If a hit is matched on the DUT
  - Calculate residual from (straight line) track with rotated (γ), aligned (X, Y, α, β, γ) and prealigned (X,Y) DUT plane

- But: σ_{DUT} only 34 μm!
  - Expected:

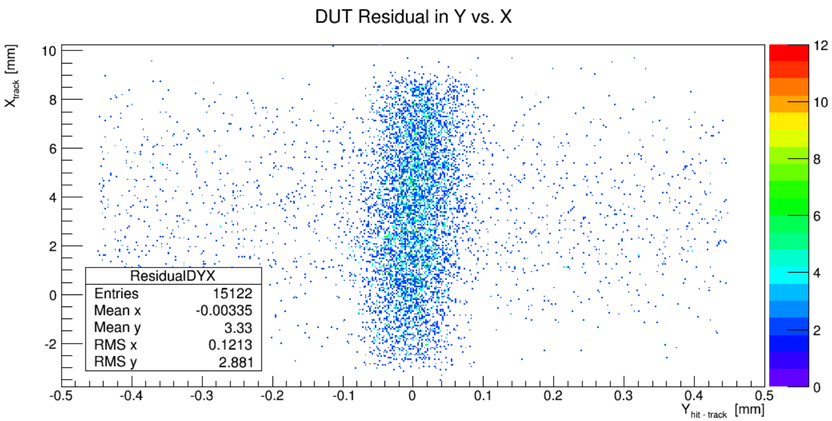
    80/sqrt(12) ~ 24 μm

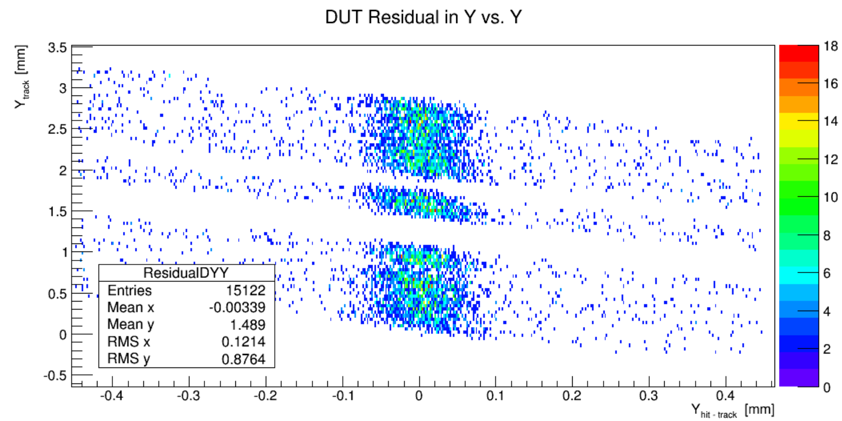
- Improve this how?
  - Non-integer cluster charge
  - Improve 'fake' coordinate selection
  - Iterative alignment

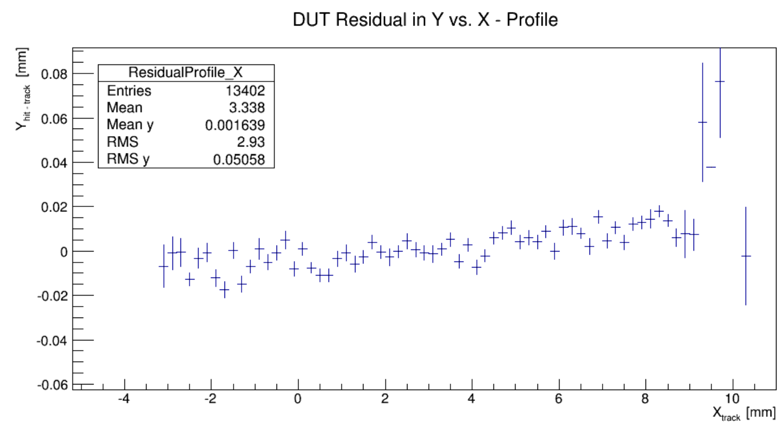


NTuple filling & ALiBaVa reconstruction – cont.

- 100µm unirr. Epi sensor, 51° rotation
- Residual in Y vs. track impact position



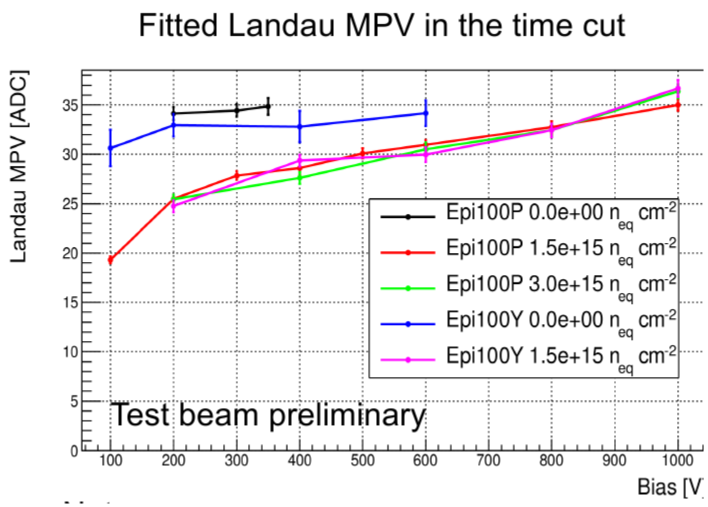




- Some rotations still not aligned away
- Work in progress

Summary

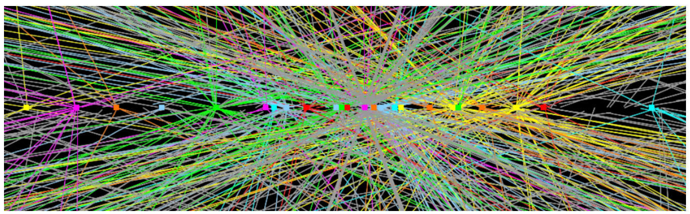
- 3 runs in TB21, 2 parasitic runs in TB22 resulted in ~300h of data taking
  - Setup installed, connected and running within minutes → real plug and play :-)
- Successful integration of (ALiBaVa) strip sensor data into EUTelescope pixel analysis software framework
  - Minor fixes and changes still needed
  - Final residual not as good as expected yet
  - Add more control plots
  - Code clean-up
- Move to new EUTelescope version after release?
- Final analysis done separately
  - Charge multiplication after irradiation can be seen



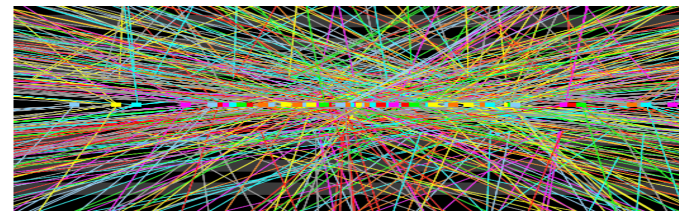
Backup

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: $\langle\mu\rangle = 40$

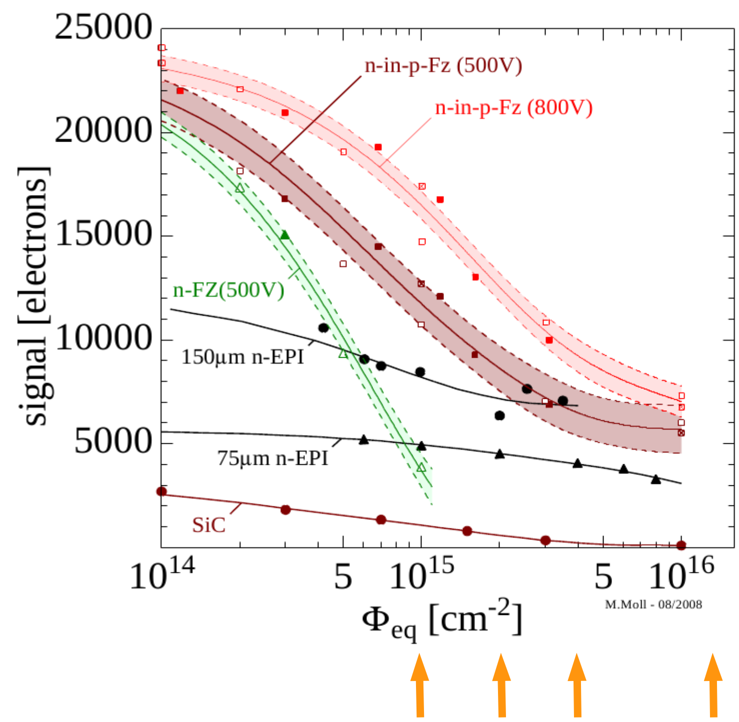


ATLAS $t\bar{t}$ event simulation: $\langle\mu\rangle = 140$

- 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



HL-LHC radii [cm]: 30 20 10 5

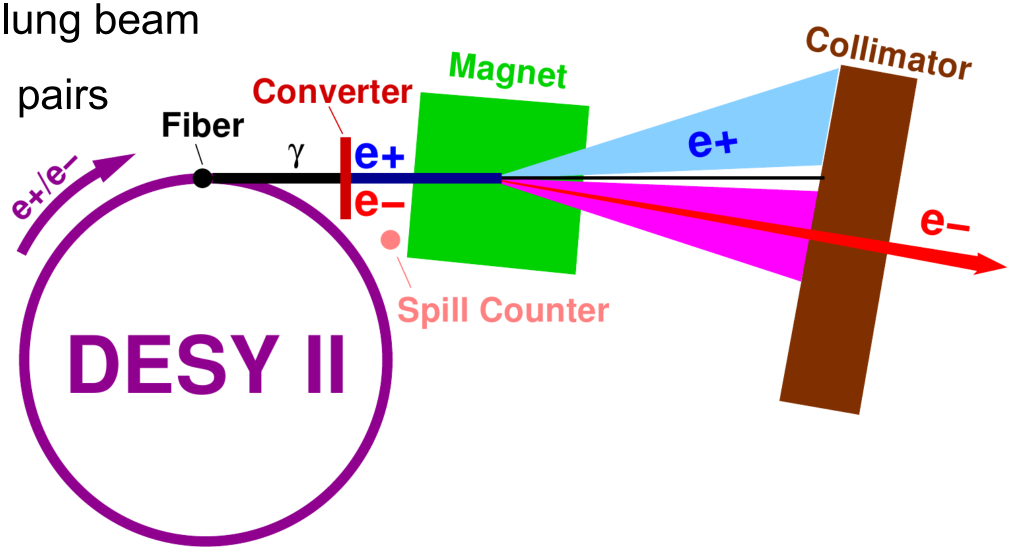
Sensor List

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

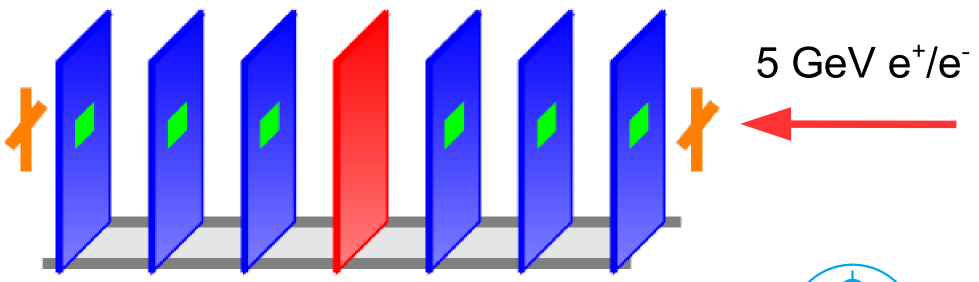
Fluence	Epi 100P	Epi 100Y	Epi 100N	Epi 70N	FTH 200Y	MCZ 200P	MCZ 200Y	MCZ 200N
0	√	√		√				
1e15 Los Alamos	√	√	√	√				
1.5e15 CERN	√	√	√					
3e15 CERN	√		√					
1.3e16 CERN	√	√	√		√	√	√	√

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.