Tracking performance in high multiplicities environment at ALICE

David Rohr ***for the ALICE Collaboration***

drohr@cern.ch, CERN

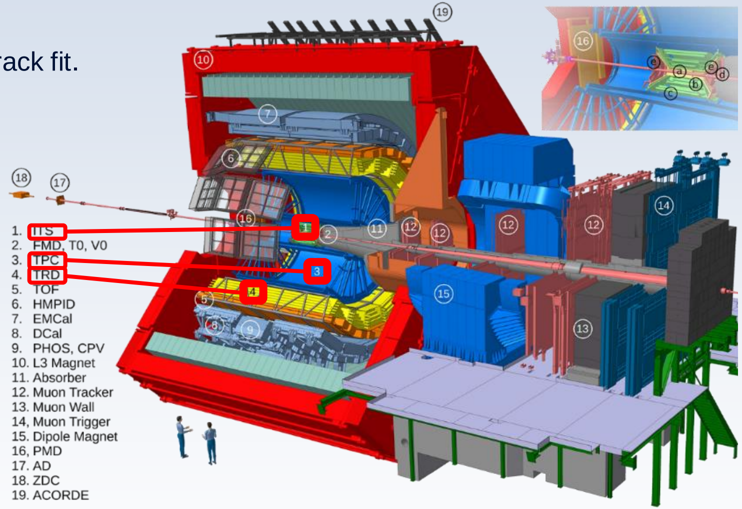
LHCP Conference 2017, Shanghai

18.5.2017

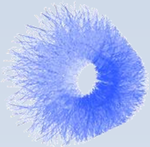
Tracking in ALICE

- **ALICE currently has two tracking implementations for the TPC:**
  - A fast online tracker in the HLT.
  - The offline TPC tracker as reference.
  - Both use the Kalman filter for track following and track fit.
  - The HLT uses a Cellular Automaton for seeding.

- **Also for ITS (silicon detector), there are distinct HLT and offline trackers.**
- **For the TRD there is only an offline tracker yet, prolonging TPC tracks.**

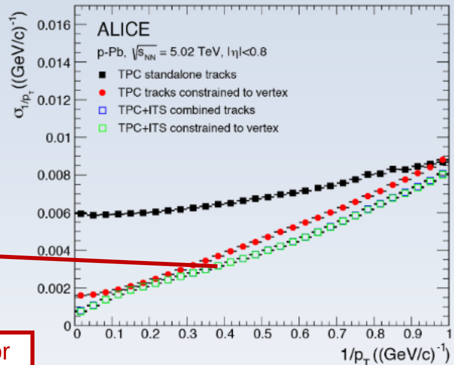


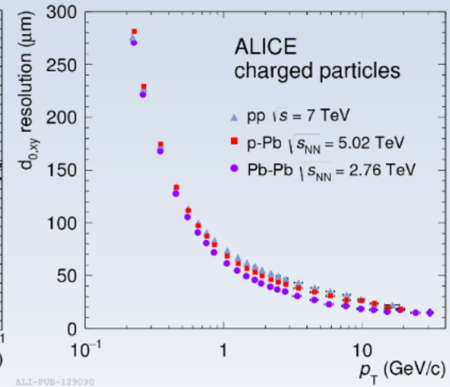
- **Luminosity will increase significantly in Run 3.**
  - We want to adapt the HLT tracking for Run 3.
  - Ideally, create one tracker as fast as the HLT version with competitive resolution to the offline version.



ALICE Offline Tracking Performance

- **Overview of offline tracking results from Run 1 / 2.**
  - We want to maintain the efficiency and resolution for Run 3.

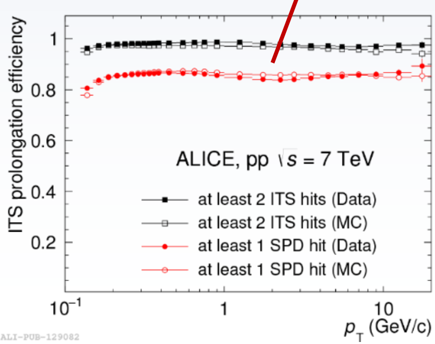


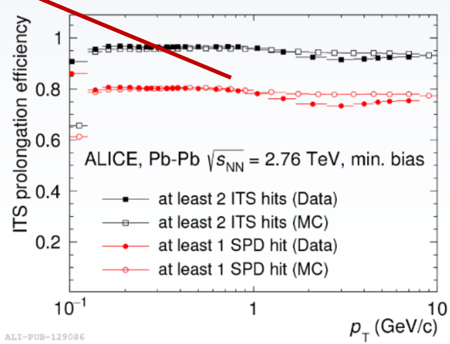


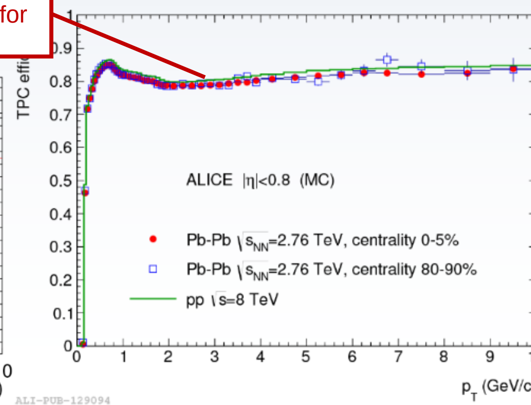
p_T resolution improved by ITS significantly

TPC ITS matching efficiency > 80% for at least 2 ITS hits. Identical for pp and Pb-Pb, reproduced in MC.

Same aggregate tracking efficiency for pp and Pb-Pb





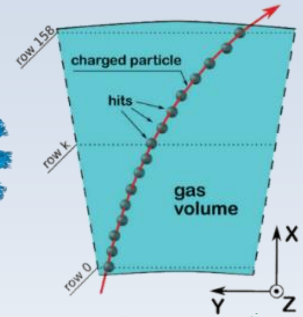


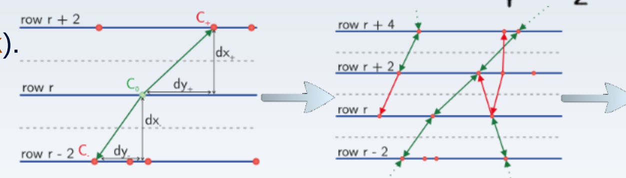
Run 3 Tracking derived from current tracking in ALICE HLT

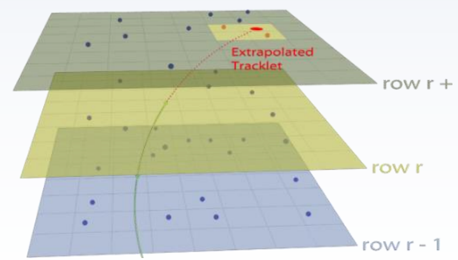
- **TPC Volume is split in 36 sectors.**
  - The tracker processes each sector individually.
  - Each sector has 159 read out rows in radial direction.
  - Tracking runs in 2 phases:

- **1. Phase: Sector-Tracking (within a sector)**
  - Heuristic, combinatorial search for track seeds using a **Cellular Automaton**.
    - A) Looks for three hits composing a straight line (**link**).
    - B) Concatenates links.

  - Fit of track parameters, extrapolation of track, and search for additional clusters using the **Kalman Filter**.

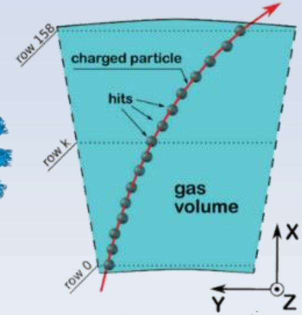




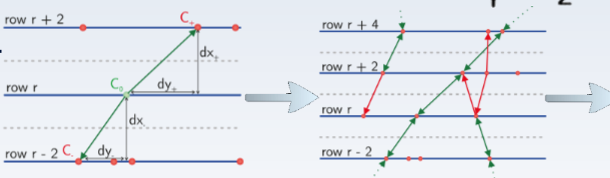


Run 3 Tracking derived from current tracking in ALICE HLT

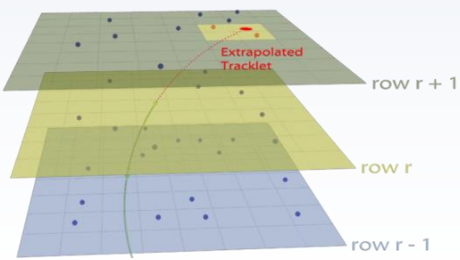
- **TPC Volume is split in 36 sectors.**
  - The tracker processes each sector individually.
  - Each sector has 159 read out rows in radial direction.
  - Tracking runs in 2 phases:



- **1. Phase: Sector-Tracking (within a sector)**
  - Heuristic, combinatorial search for track seeds using a **Cellular Automaton**.
    - A) Looks for three hits composing a straight line (**link**).
    - B) Concatenates links.



  - Fit of track parameters, extrapolation of track, and search for additional clusters using the **Kalman Filter**.

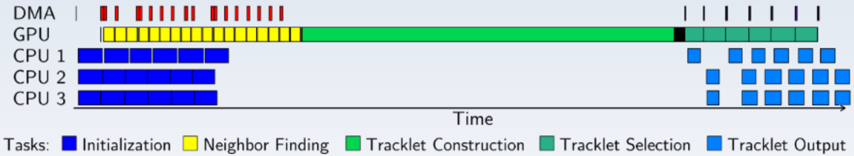


- **2. Phase: Track-Merger**
  - Combines the track segments found in the individual sectors.
- **Phase 1 track finding implemented in a common generic source code, that runs on CPU and GPU, supporting CUDA, OpenCL, and OpenMP.**

GPU Track Finding Performance Results

- **GPU tracking running since 2012 in the ALICE HLT.**
- **Currently using 180 AMD S9000 GPUs.**
- **HLT can reconstruct and fit up to 40,000,000 tracks per second.**

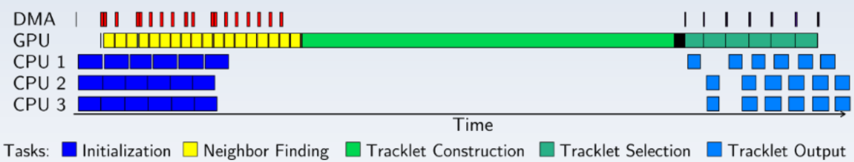
- Individual TPC sectors are processed in a pipeline, such that pre- and postprocessing, data transfer, and tracking of sectors overlap.
- Still, central Pb-Pb is unable to fully load modern GPU.
  - → Process 2 events in parallel: 2 * 145 ms → 220 ms (110 ms per event).
- **Significant speedupof two orders of magnitude (up to 150x) by using the Cellular Automatoon approach and the GPU.**



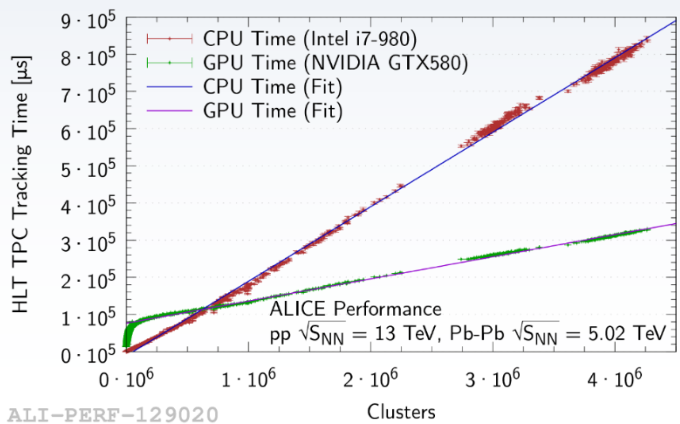
GPU Track Finding Performance Results

- **GPU tracking running since 2012 in the ALICE HLT.**
- **Currently using 180 AMD S9000 GPUs.**
- **HLT can reconstruct and fit up to 40,000,000 tracks per second.**

- Individual TPC sectors are processed in a pipeline, such that pre- and postprocessing, data transfer, and tracking of sectors overlap.
- Still, central Pb-Pb is unable to fully load modern GPU.
  → Process 2 events in parallel: 2 * 145 ms → 220 ms (110 ms per event).
- **Significant speedupof two orders of magnitude (up to 150x) by using the Cellular Automatoon approach and the GPU.**



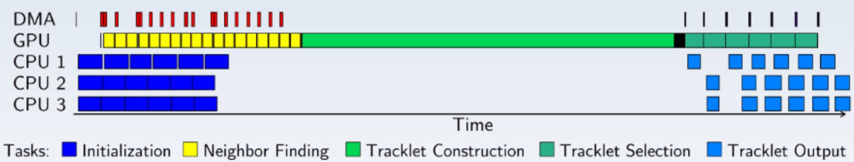
Tracking times scales linearly with input data size.



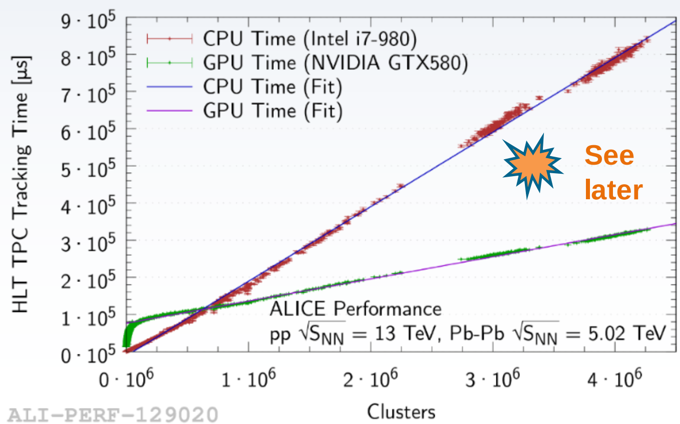
GPU Track Finding Performance Results

- **GPU tracking running since 2012 in the ALICE HLT.**
- **Currently using 180 AMD S9000 GPUs.**
- **HLT can reconstruct and fit up to 40,000,000 tracks per second.**

- Individual TPC sectors are processed in a pipeline, such that pre- and postprocessing, data transfer, and tracking of sectors overlap.
- Still, central Pb-Pb is unable to fully load modern GPU.
  → Process 2 events in parallel: 2 * 145 ms → 220 ms (110 ms per event).
- **Significant speedupof two orders of magnitude (up to 150x) by using the Cellular Automatoon approach and the GPU.**



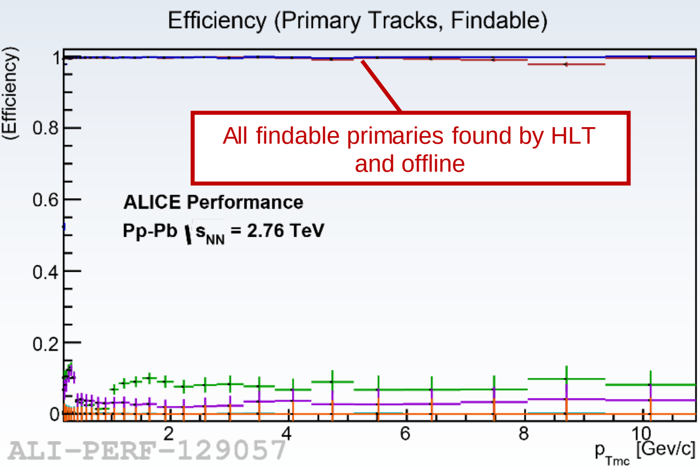
Tracking times scales linearly with input data size.

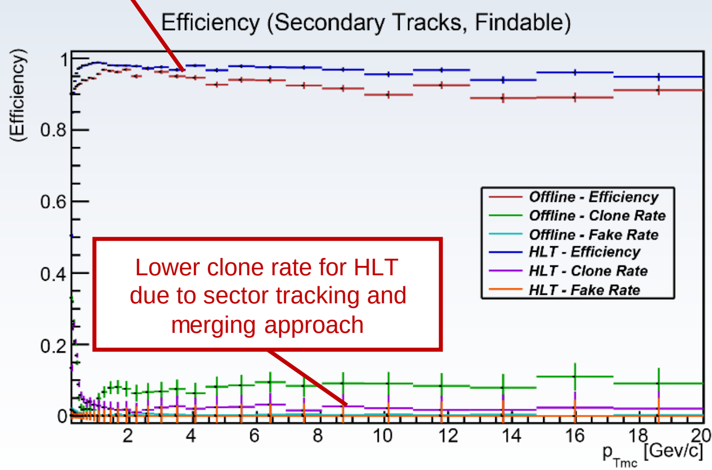


TPC Tracking efficiency

- **Simulation based on 2011 Pb-Pb, 2.76 TeV**
- **Findable tracks: min 70 TPC hits**

High efficiency for findable secondaries, advantage for HLT due to Cellular Automaton seeding without vertex constraint.

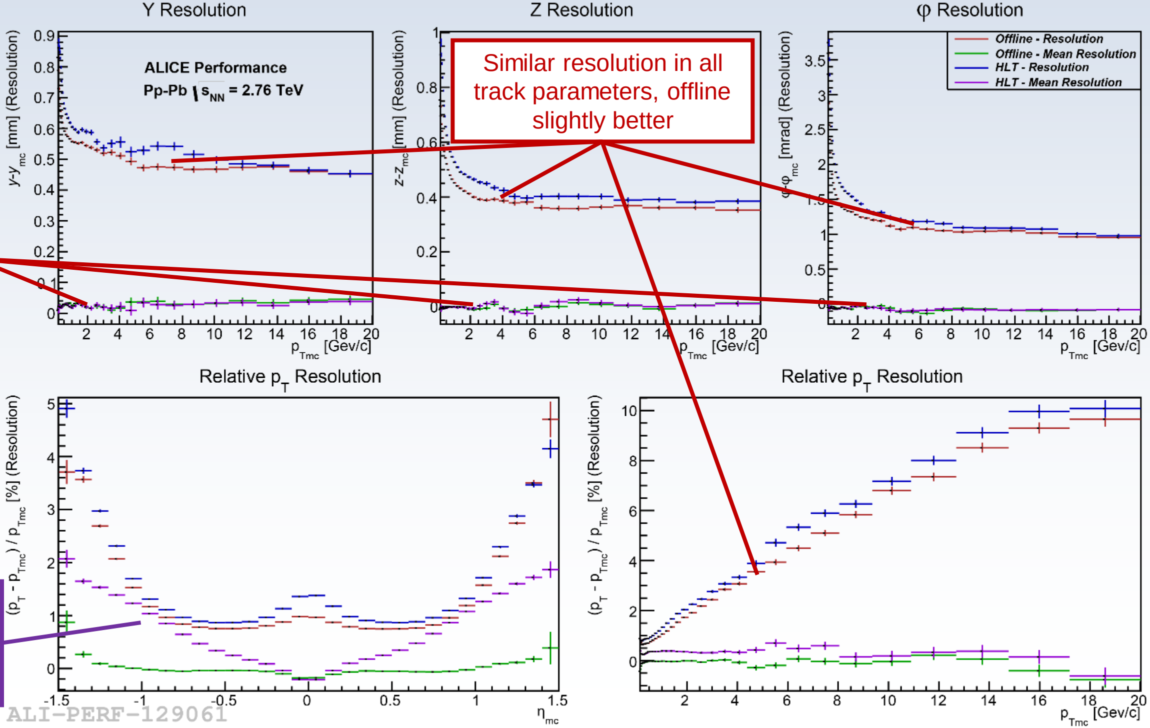




TPC Tracking resolution

- **Simulation based on 2011 Pb-Pb, 2.76 TeV**
- **TPC only fit without ITS or TRD**

- p_T bias in HLT tracks due to missing B_X and B_Y components of magnetic field.

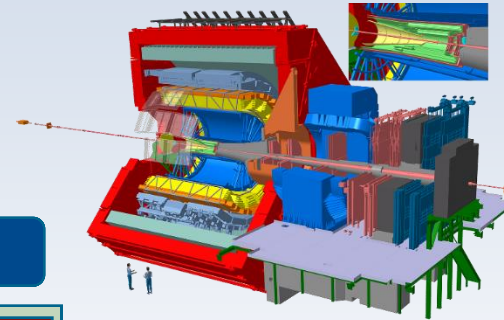


No significant bias in Y, Z, and ϕ for offline and HLT

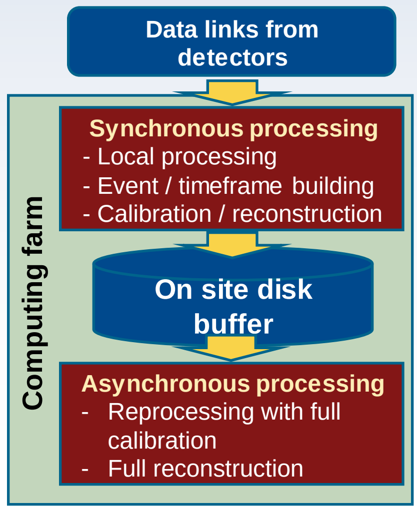
p_T bias in HLT tracks that depends on pseudorapidity

Implications of the ALICE Upgrade for Tracking

- **ALICE will perform a major upgrade for high luminosity data taking.**
  - Continuous read-out instead of triggered read-out using a GEM TPC.
    - Reconstruction based on time frames not events.
    - **50 kHz** Pb-Pb min-bias data taking.
      (compared to **8 kHz** IR, **500 Hz** to **1 kHz** trigger)

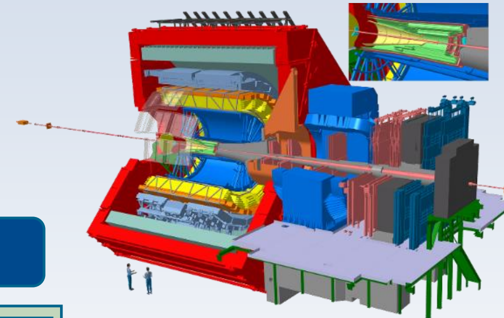


  - Major update / rewrite of tracking software.
    - Relevant detectors for track reconstruction:
      ITS (6 *silicon detector, 7 in Run 3*),
      TPC,
      TRD, TOF *(for refit and calibration)*

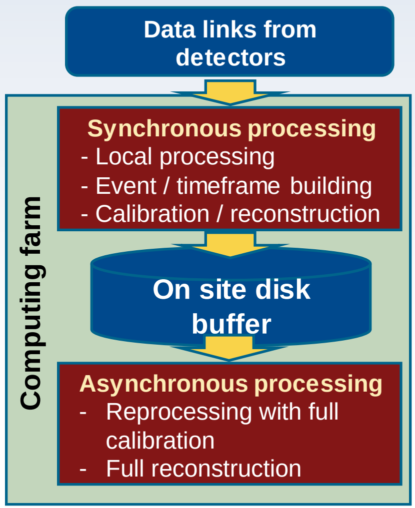


Implications of the ALICE Upgrade for Tracking

- **ALICE will perform a major upgrade for high luminosity data taking.**
  - Continuous read-out instead of triggered read-out using a GEM TPC.
    - Reconstruction based on time frames not events.
    - **50 kHz** Pb-Pb min-bias data taking.
      (compared to **8 kHz** IR, **500 Hz** to **1 kHz** trigger)



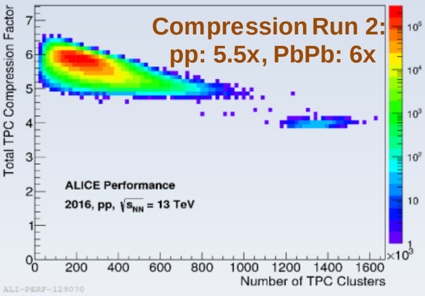
  - Major update / rewrite of tracking software.
    - Relevant detectors for track reconstruction:
      ITS (6 *silicon detector, 7 in Run 3*),
      TPC,
      TRD, TOF *(for refit and calibration)*



- **Challenges:**
  - Tracking time frames with overlapping events.
  - Continuous readout.
  - Reconstruct and store 50 times more events.
    (same time, similar storage)
    → Faster reconstruction
    → Better data compression based on tracks.
    → Online calibration.

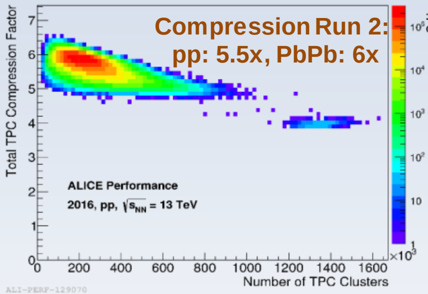
Data reduction

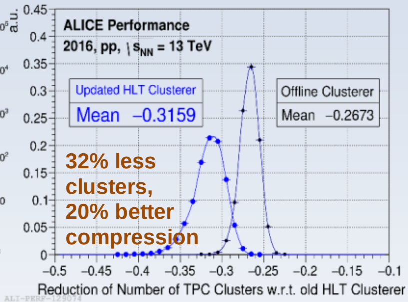
- **In the TDR for the ALICE Online Offline (O^2) computing upgrade, we foresee a data compression factor of 20 (with respect to Run 1 raw data size).**
  - Infeasible with lossless compression.
  - Online TPC clusterization with FPGA in HLT.
  - Only clusters, no raw data stored (2011).
  - Huffman compression in Run 1 (**factor 4x**).
  - Reduce cluster entropy by storing differences (2016).



Data reduction

- **In the TDR for the ALICE Online Offline (O²) computing upgrade, we foresee a data compression factor of 20 (with respect to Run 1 raw data size).**
  - Infeasible with lossless compression.
  - Online TPC clusterization with FPGA in HLT.
  - Only clusters, no raw data stored (2011).
  - Huffman compression in Run 1 (factor 4x).
  - Reduce cluster entropy by storing differences (2016).



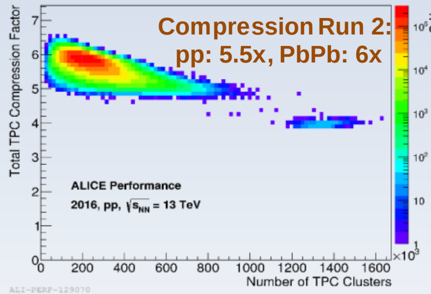


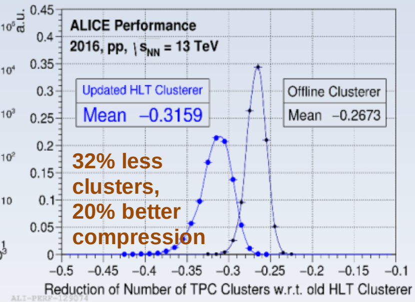
- **Additional compression studied for Run 2 / 3.**
  - TPC noise suppression → reject fake clusters.
  - Arithmetic compression instead of Huffman compression.
  - Reduce number of significant bits for cluster charge and size
- **Additional compression (use track information).**
  - Store cluster to track residual (less entropy).

Total compression factor of 9.1x demonstrated in a proof of concept prototype on data from 2016, commissioning this year.

Data reduction

- **In the TDR for the ALICE Online Offline (O^2) computing upgrade, we foresee a data compression factor of 20 (with respect to Run 1 raw data size).**
  - Infeasible with lossless compression.
  - Online TPC clusterization with FPGA in HLT.
  - Only clusters, no raw data stored (2011).
  - Huffman compression in Run 1 (**factor 4x**).
  - Reduce cluster entropy by storing differences (2016).





- **Additional compression studied for Run 2 / 3.**
  - TPC noise suppression → reject fake clusters.
  - Arithmetic compression instead of Huffman compression.
  - Reduce number of significant bits for cluster charge and size

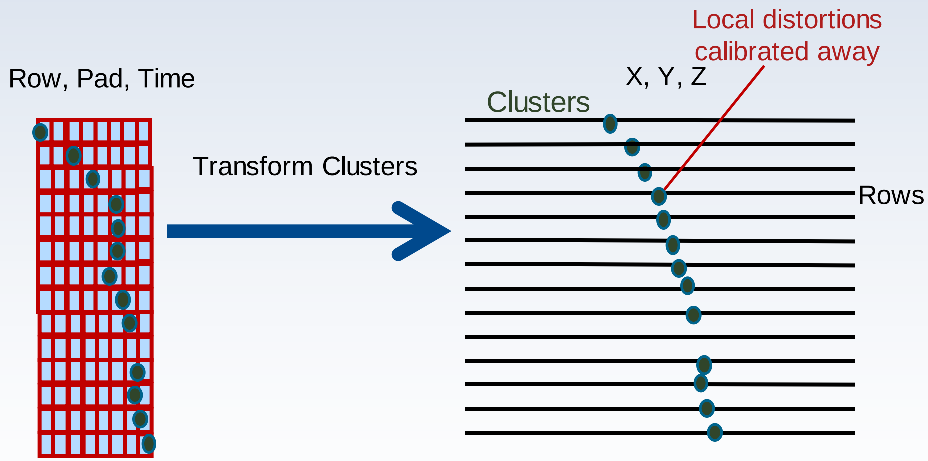
Total compression factor of 9.1x demonstrated in a proof of concept prototype on data from 2016, commissioning this year.

- **Additional compression (use track information).**
  - Store cluster to track residual (less entropy).
  - Positive identification of clusters excluded from storage: (expected **2x** reduction, summing to **18.2x**, close to the **20x we need**)
    - Clusters of looping tracks below 50 MeV (not used for physics), at higher p_T only 1st leg preserved.
    - Clusters of track segments with high inclination angle (bad quality, not used for track fit).

Track model compression

Work in Progress

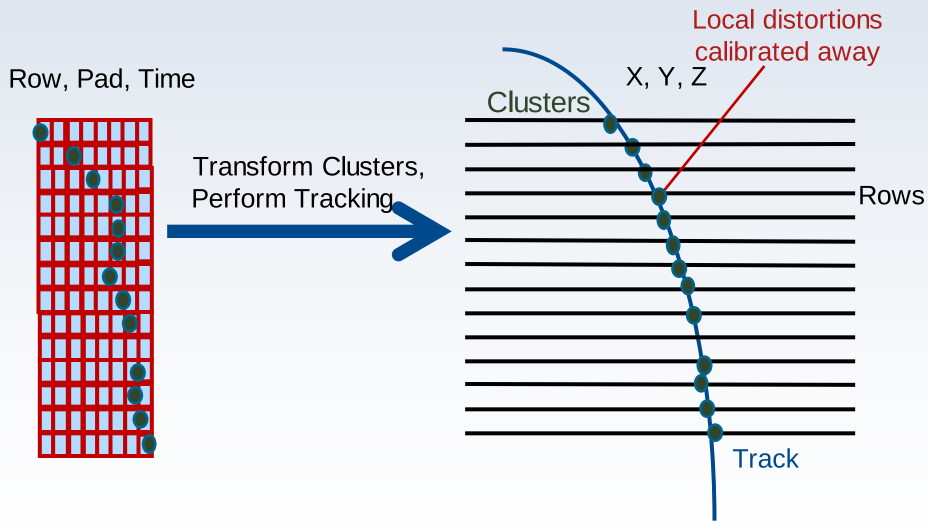
- **Idea: store only residuals of clusters to extrapolated track.**
  - **Smaller entropy** than absolute (differential) coordinates → **Better Huffman compression**.
  - **Constraint**: Clusters shall be stored in native TPC coordinates (Row, Pad, Time), independent from calibration.



Track model compression

Work in Progress

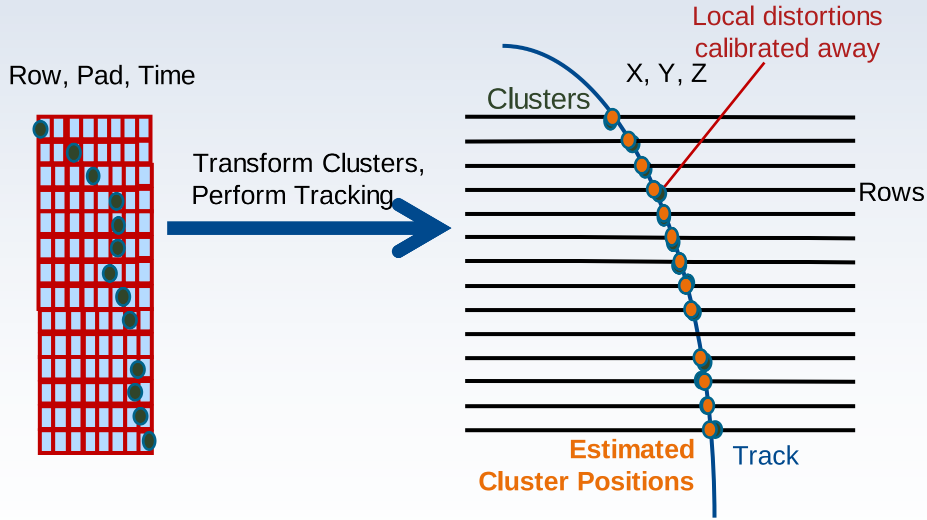
- **Idea: store only residuals of clusters to extrapolated track.**
  - **Smaller entropy** than absolute (differential) coordinates → **Better Huffman compression**.
  - **Constraint**: Clusters shall be stored in native TPC coordinates (Row, Pad, Time), independent from calibration.



Track model compression

Work in Progress

- **Idea: store only residuals of clusters to extrapolated track.**
  - **Smaller entropy** than absolute (differential) coordinates → **Better Huffman compression**.
  - **Constraint**: Clusters shall be stored in native TPC coordinates (Row, Pad, Time), independent from calibration.

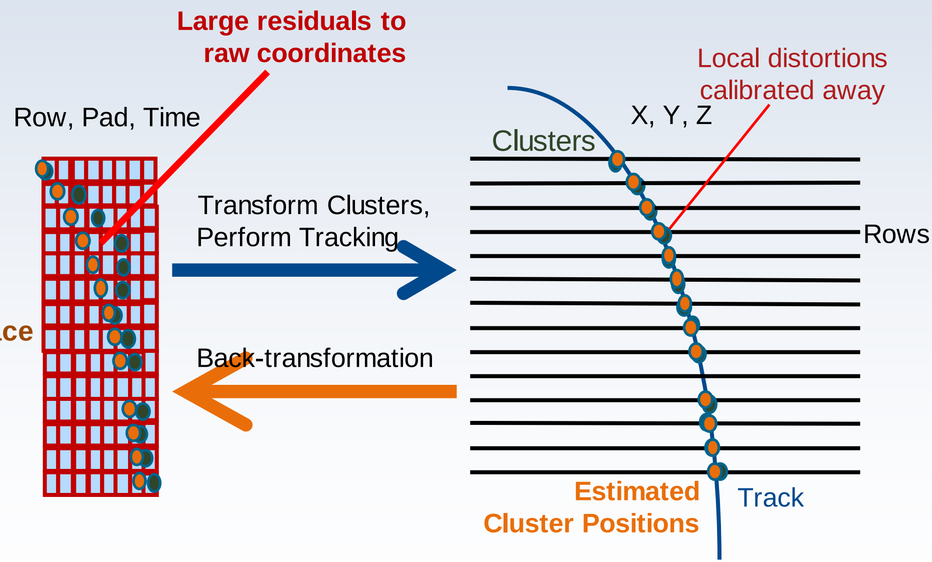


Track model compression

Work in Progress

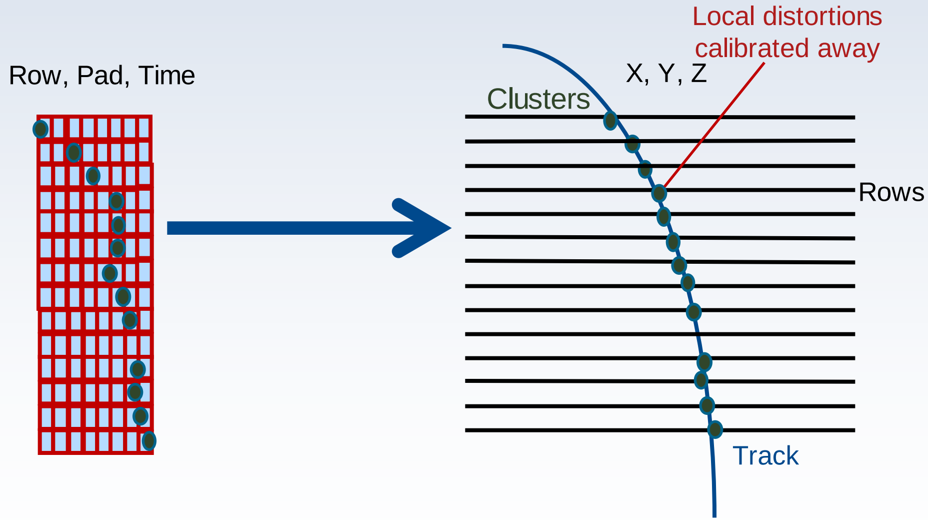
- **Idea: store only residuals of clusters to extrapolated track.**
  - **Smaller entropy** than absolute (differential) coordinates → **Better Huffman compression**.
  - **Constraint**: Clusters shall be stored in native TPC coordinates (Row, Pad, Time), independent from calibration.

- **Problems**:
  - Helix prolongation yields **large residuals**
    - → inefficient compression.
  - Linear back-transtransformation **cannot revert transformation** based on full calibration.
  - Helix **cannot accommodate space charge distortions** in the TPC.



Track model compression

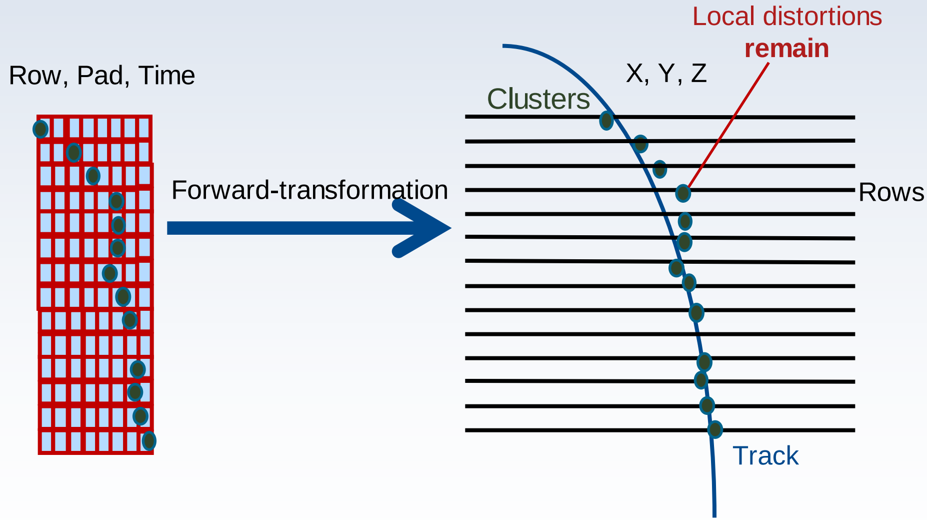
Work in Progress



Track model compression

Work in Progress

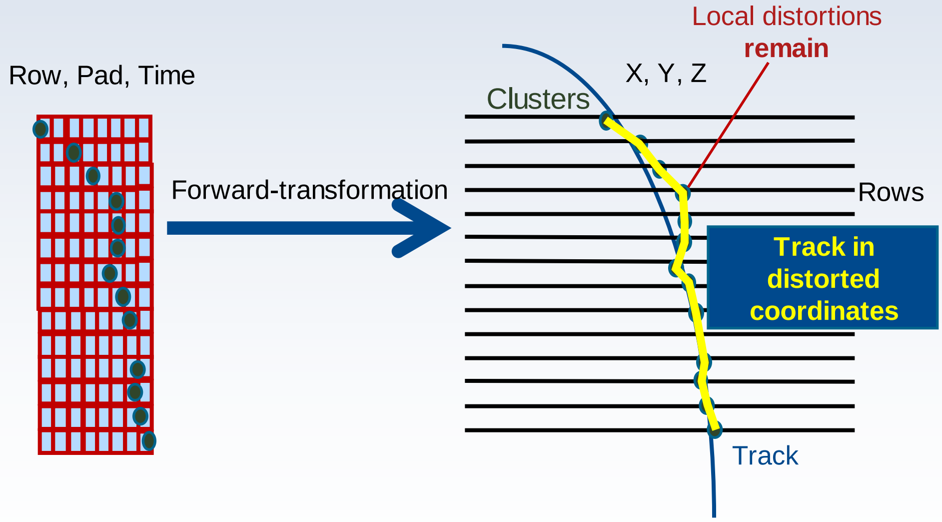
- **Solution:**
  - Employ **fast, reversible polynomial approximation**. *(In principle, every transformation works, but the closer the better!)*



Track model compression

Work in Progress

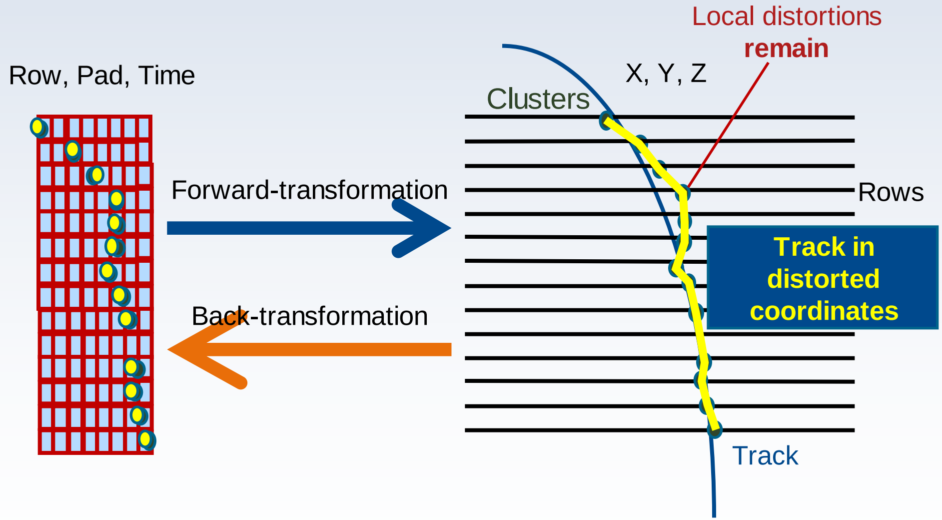
- **Solution:**
  - Employ **fast, reversible polynomial approximation**. *(In principle, every transformation works, but the closer the better!)*
  - **Refit track in distorted coordinate system**.



Track model compression

Work in Progress

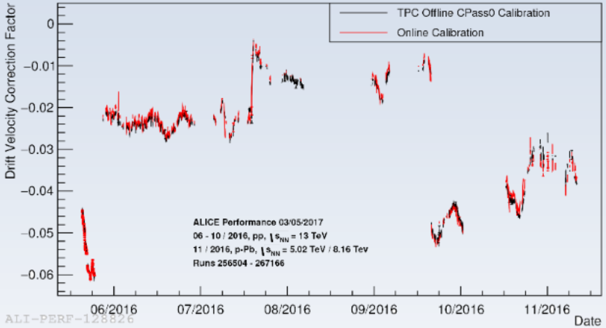
- **Solution:**
  - Employ **fast, reversible polynomial approximation**. *(In principle, every transformation works, but the closer the better!)*
  - **Refit track in distorted coordinate system**.



- **Non-associated clusters still compressed with differences scheme.**
- **Additional benefit: Cluster to track association is stored → Track found in HLT / synchronous phase available later.**

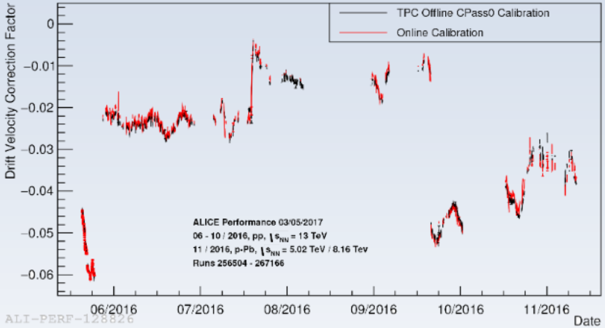
Online calibration in the HLT

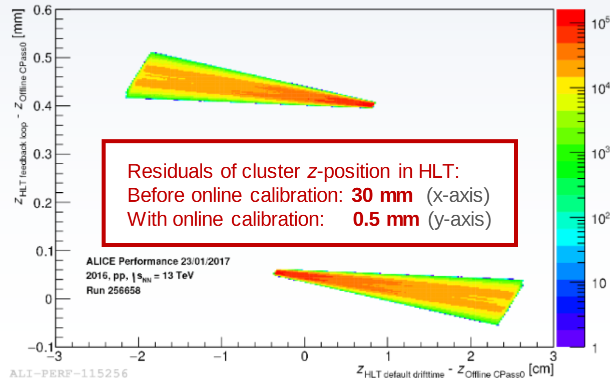
- **No significant difference in the drift velocity computed online and offline between May and December 2016.**
  - Gaps in between are periods without beam.
  - **Different ambient pressure and TPC temperature values online and offline are corrected by calibration** yielding slightly different factor.



Online calibration in the HLT

- **No significant difference in the drift velocity computed online and offline between May and December 2016.**
  - Gaps in between are periods without beam.
  - **Different ambient pressure and TPC temperature values online and offline are corrected by calibration** yielding slightly different factor.
- **Online calibration cluster z-position residual compared to offline below 0.5 mm (below the intrinsic TPC resolution).**
  - 30 mm without online calibration.
    - → HLT tracking sufficient for calibration purpose today.

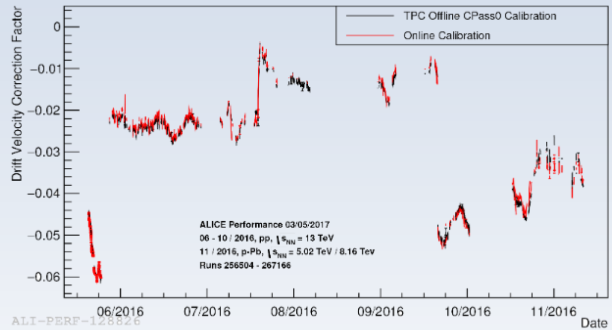


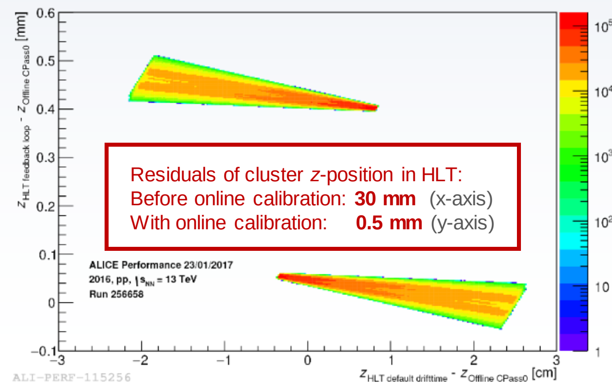


Online calibration in the HLT

- **No significant difference in the drift velocity computed online and offline between May and December 2016.**
  - Gaps in between are periods without beam.
  - **Different ambient pressure and TPC temperature values online and offline are corrected by calibration** yielding slightly different factor.

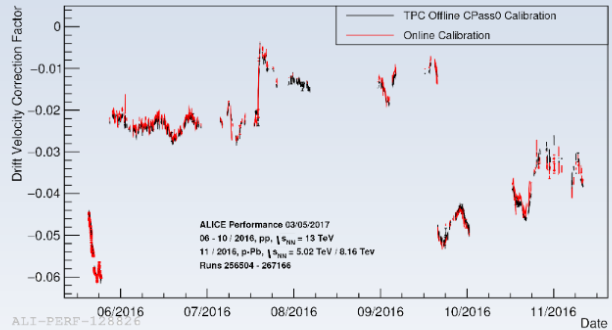
- **Online calibration cluster z-position residual compared to offline below 0.5 mm (below the intrinsic TPC resolution).**
  - 30 mm without online calibration.
    - → HLT tracking sufficient for calibration purpose today.
  - Gathering calibration data takes 2 minutes per interval.
  - Preparation and distribution takes 30 seconds.
    - – Calibration of the previous two minutes used for following time. (**feedback loop**).
    - → The first 2.5 minutes of a run are without online calibration.



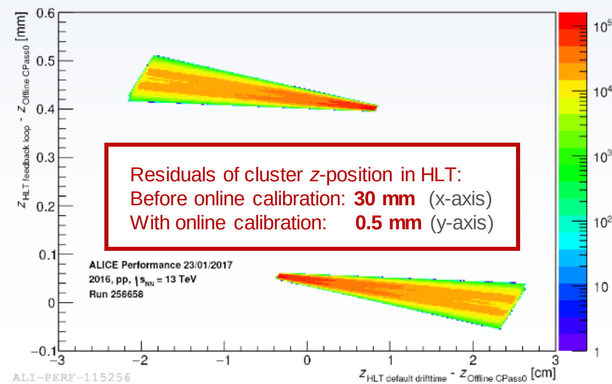


Online calibration in the HLT

- **No significant difference in the drift velocity computed online and offline between May and December 2016.**
  - Gaps in between are periods without beam.
  - **Different ambient pressure and TPC temperature values online and offline are corrected by calibration** yielding slightly different factor.

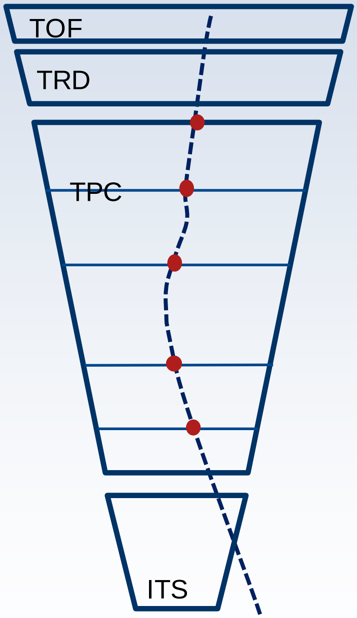


- **Online calibration cluster z-position residual compared to offline below 0.5 mm (below the intrinsic TPC resolution).**
  - 30 mm without online calibration.
    - → HLT tracking sufficient for calibration purpose today.
  - Gathering calibration data takes 2 minutes per interval.
  - Preparation and distribution takes 30 seconds.
    - – Calibration of the previous two minutes used for following time. (**feedback loop**).
    - → The first 2.5 minutes of a run are without online calibration.



- **Online calibration frameworks enables execution of most tasks with minor modification.**
  - Gain calibration commissioned at the moment.
  - Space charge distortion calibration foreseen online.

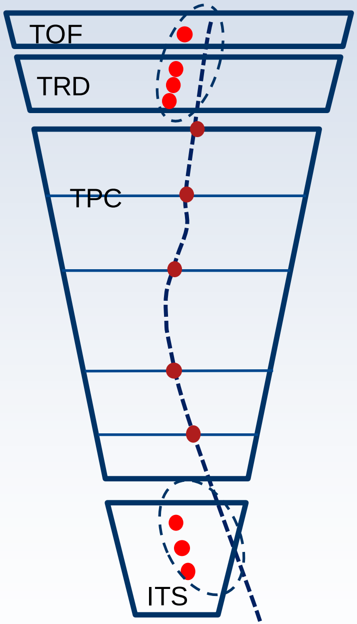
Space Charge Distortions corrections (in Pb-Pb Run 2)



- Reconstruct TPC tracks with relaxed tolerances (applying "default distortion maps" if available)
- Match to ITS and TRD/TOF with relaxed tolerances
- Refit ITS-TRD-TOF part and interpolate to TPC as a **reference** of the **true track position** at every pad-row
- Collect Y, Z differences between **distorted clusters** and **reference** points in sub-volumes (voxels) of TPC
- Extract 3D vector of distortion in every voxel
- Create smooth parameterization (fast interpolation by Chebyshev polynomials) to use for correction during following reconstruction

- Distortions change with time: 40 min intervals. (min 15-20 min for statistics)
- 15 (in Y/X) x 5 (in Z/X) voxels per padrow ⇒ ~430K in total

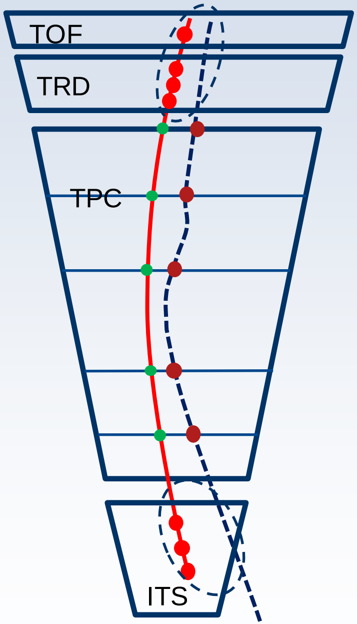
Space Charge Distortions corrections (in Pb-Pb Run 2)



- Reconstruct TPC tracks with relaxed tolerances (applying “default distortion maps” if available)
- Match to ITS and TRD/TOF with relaxed tolerances
- Refit ITS-TRD-TOF part and interpolate to TPC as a **reference** of the **true track position** at every pad-row
- Collect Y, Z differences between **distorted clusters** and **reference** points in sub-volumes (voxels) of TPC
- Extract 3D vector of distortion in every voxel
- Create smooth parameterization (fast interpolation by Chebyshev polynomials) to use for correction during following reconstruction

- Distortions change with time: 40 min intervals. (min 15-20 min for statistics)
- 15 (in Y/X) x 5 (in Z/X) voxels per padrow ⇒ ~430K in total

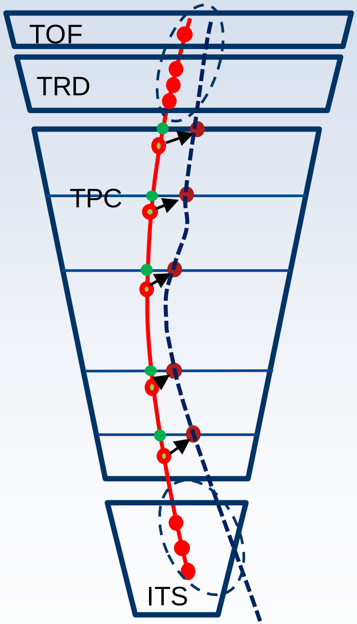
Space Charge Distortions corrections (in Pb-Pb Run 2)



- Reconstruct TPC tracks with relaxed tolerances (applying "default distortion maps" if available)
- Match to ITS and TRD/TOF with relaxed tolerances
- Refit ITS-TRD-TOF part and interpolate to TPC as a **reference** of the **true track position** at every pad-row
- Collect Y, Z differences between **distorted clusters** and **reference** points in sub-volumes (voxels) of TPC
- Extract 3D vector of distortion in every voxel
- Create smooth parameterization (fast interpolation by Chebyshev polynomials) to use for correction during following reconstruction

- Distortions change with time: 40 min intervals. (min 15-20 min for statistics)
- 15 (in Y/X) x 5 (in Z/X) voxels per padrow ⇒ ~430K in total

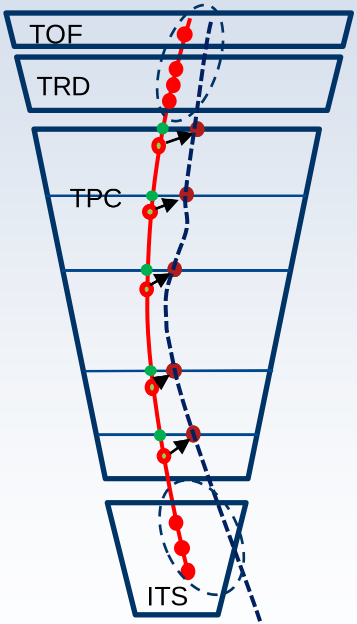
Space Charge Distortions corrections (in Pb-Pb Run 2)



- Reconstruct TPC tracks with relaxed tolerances (applying “default distortion maps” if available)
- Match to ITS and TRD/TOF with relaxed tolerances
- Refit ITS-TRD-TOF part and interpolate to TPC as a **reference** of the **true track position** at every pad-row
- Collect Y, Z differences between **distorted clusters** and **reference** points in sub-volumes (voxels) of TPC
- Extract 3D vector of distortion in every voxel
- Create smooth parameterization (fast interpolation by Chebyshev polynomials) to use for correction during following reconstruction

- Distortions change with time: 40 min intervals. (min 15-20 min for statistics)
- 15 (in Y/X) x 5 (in Z/X) voxels per padrow ⇒ ~430K in total

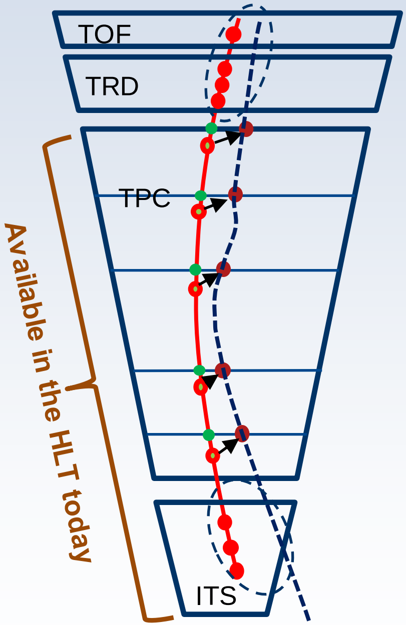
Space Charge Distortions corrections (in Pb-Pb Run 2)



- Reconstruct TPC tracks with relaxed tolerances (applying “default distortion maps” if available)
- Match to ITS and TRD/TOF with relaxed tolerances
- Refit ITS-TRD-TOF part and interpolate to TPC as a **reference** of the **true track position** at every pad-row
- Collect Y, Z differences between **distorted clusters** and **reference** points in sub-volumes (voxels) of TPC
- Extract 3D vector of distortion in every voxel
- Create smooth parameterization (fast interpolation by Chebyshev polynomials) to use for correction during following reconstruction

- Distortions change with time: 40 min intervals. (min 15-20 min for statistics)
- 15 (in Y/X) x 5 (in Z/X) voxels per padrow ⇒ ~430K in total

Space Charge Distortions corrections (in Pb-Pb Run 2)



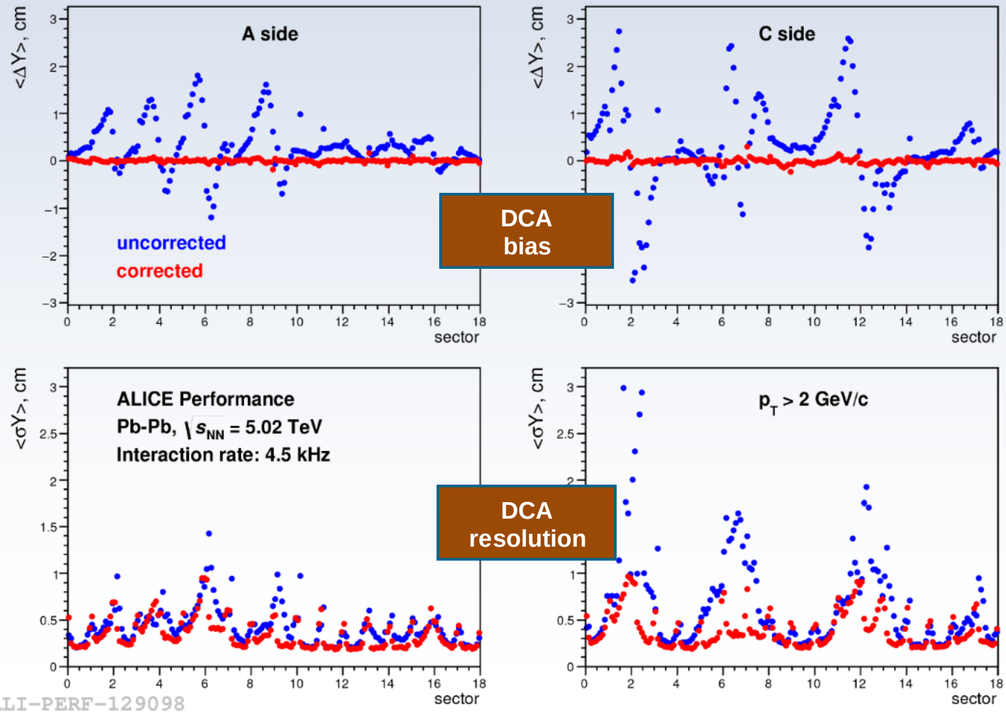
- Reconstruct TPC tracks with relaxed tolerances (applying “default distortion maps” if available)
- Match to ITS and TRD/TOF with relaxed tolerances
- Refit ITS-TRD-TOF part and interpolate to TPC as a **reference** of the **true track position** at every pad-row
- Collect Y, Z differences between **distorted clusters** and **reference** points in sub-volumes (voxels) of TPC
- Extract 3D vector of distortion in every voxel
- Create smooth parameterization (fast interpolation by Chebyshev polynomials) to use for correction during following reconstruction

- Distortions change with time: 40 min intervals. (min 15-20 min for statistics)
- 15 (in Y/X) x 5 (in Z/X) voxels per padrow ⇒ ~430K in total

- **Running offline**
- **To be implemented for HLT during Run 2 as prototype for Run 3**
- **Needs TRD and TOF reconstruction in the HLT (not available yet)**

Space Charge Distortions corrections results

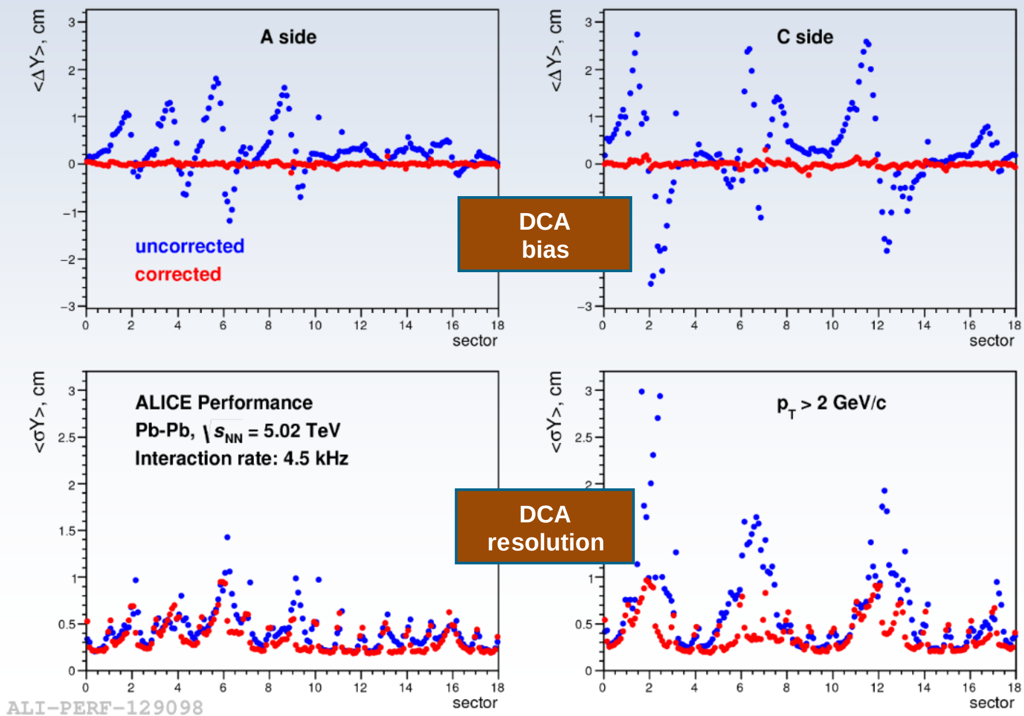
- **Corrections compensate bias of the track position.**
- **Due to long interval of 40 minutes to gather statistics, cannot correct for short term luminosity fluctuations.**
  → Degradation of resolution.



Space Charge Distortions corrections results

- **Corrections compensate bias of the track position.**
- **Due to long interval of 40 minutes to gather statistics, cannot correct for short term luminosity fluctuations.**
  → Degradation of resolution.

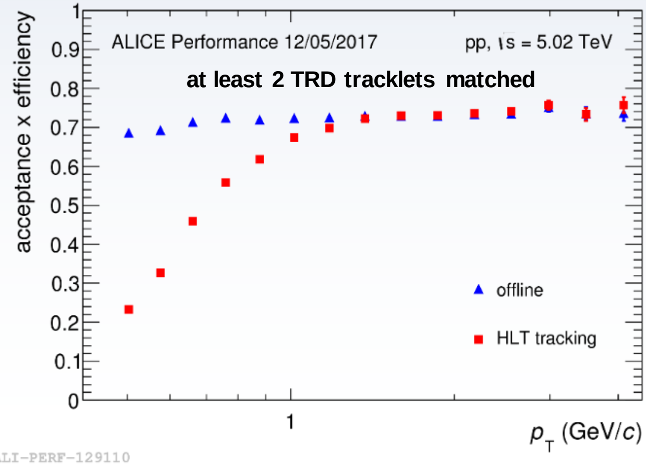
- **2 calibration algorithms for Run 3:**
  - ITS – TRD interpolation as presented here.
  - Integration of currents in the TPC and direct computation of distortion maps.
    – Cannot be tested during Run 2 since it does not work in triggered readout mode (needs to integrate all current in continuous mode).



Online TRD tracking in the HLT and for Run 3

Work in Progress

- **Space Charge Distortion Calibration requires online TRD tracking.**
  - Offline tracking uses **offline tracklets** created from hits around TPC tracks.
  - **Not available in Run 3**, use **online tracklets** created by TRD readout.
- **Extrapolates TPC track into TRD layers, attaches closest tracklet.**
  - Decision-tree based algorithm foreseen if needed for high occupancy.
- **Same matching efficiency for online and offline tracklets > 1 GeV/*c*.**
  - Drop of efficiency at low p_T is caused by the absence of online tracklets (currently optimized for electron trigger at $p_T > 3$ GeV/*c*)
- **Extending matching to $p_T \sim 0.6$ GeV/*c* is important for**
  - Disentangling between radial and rϕ distortions.
  - Bridging TPC tracks to TOF.



Online TRD tracking in the HLT and for Run 3

Work in Progress

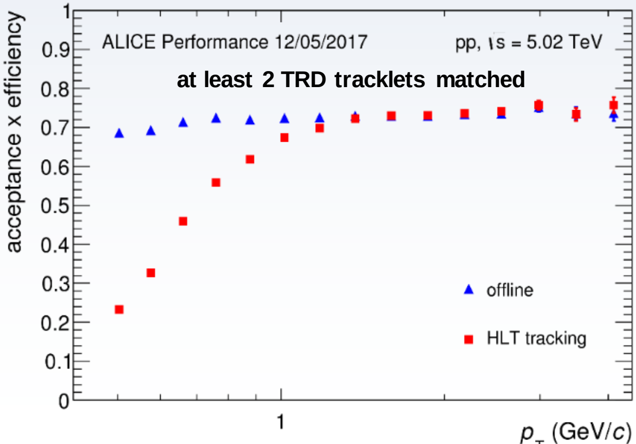
- **Space Charge Distortion Calibration requires online TRD tracking.**
  - Offline tracking uses **offline tracklets** created from hits around TPC tracks.
  - **Not available in Run 3**, use **online tracklets** created by TRD readout.

- **Extrapolates TPC track into TRD layers, attaches closest tracklet.**
  - Decision-tree based algorithm foreseen if needed for high occupancy.

- **Same matching efficiency for online and offline tracklets > 1 GeV/*c*.**
  - Drop of efficiency at low p_T is caused by the absence of online tracklets (currently optimized for electron trigger at p_T > 3 GeV/*c*)

- **Extending matching to p_T ~ 0.6 GeV/*c* is important for**
  - Disentangling between radial and rϕ distortions.
  - Bridging TPC tracks to TOF.

- **Early work on matching the TOF hit in the HLT started.**
  - Emphasis on TRD, which is more complicated.



ALI-PERF-129110

Tracking with GEM TPC, plan for Run3

- **Changes for TPC tracking in ALICE during Run 3**
  - Continuous read-out
  - Higher interaction rate
    - → Time frames instead of events
    - → Increase of distortions

Tracking with GEM TPC, plan for Run3

- **Changes for TPC tracking in ALICE during Run 3**
  - Continuous read-out
  - Higher interaction rate
    - → Time frames instead of events
    - → Increase of distortions

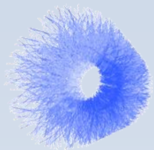
- **HLT TPC tracker operates on events as a whole.**
- **Can also process timeframe (or slice of timeframe) as a whole.**
  - → Increased data size per event / time frame.
    - Better GPU utilization through more parallelism.
    - **Feasible since HLT tracking time goes linearly with input size.**
    - Only limited by GPU memory – up to 16 GB today.
    - Slicing into smaller volumes possible, as we do today.

Tracking with GEM TPC, plan for Run3

- **Changes for TPC tracking in ALICE during Run 3**
  - Continuous read-out
  - Higher interaction rate
    - → Time frames instead of events
    - → Increase of distortions

- **HLT TPC tracker operates on events as a whole.**
- **Can also process timeframe (or slice timeframe) as a whole.**
  - → Increased data size per event / time frame.
    - – Better GPU utilization through more parallelism.
    - – **Feasible since HLT tracking time goes linearly with input size.**
    - – Only limited by GPU memory – up to 16 GB today.
    - – Slicing into smaller volumes possible, as we do today.

- **Hits in time frame have row, pad, time coordinates**
  - Currently, the HLT cluster transformation transforms all clusters to x, y, z coordinates beforehand.
  - No longer possible!
  - Transforming time to z needs the time of the interaction.
  - The vertex is only determined during tracking.
    - → Could apply the transformation multiple times, once per vertex, and run the tracking multiple times.
      - → Avoid if possible, very compute intense.
    - → **Joint tracking and transformation.**
  - Requires online calibration for the transformation.



Tracking with GEM TPC, plan for Run3

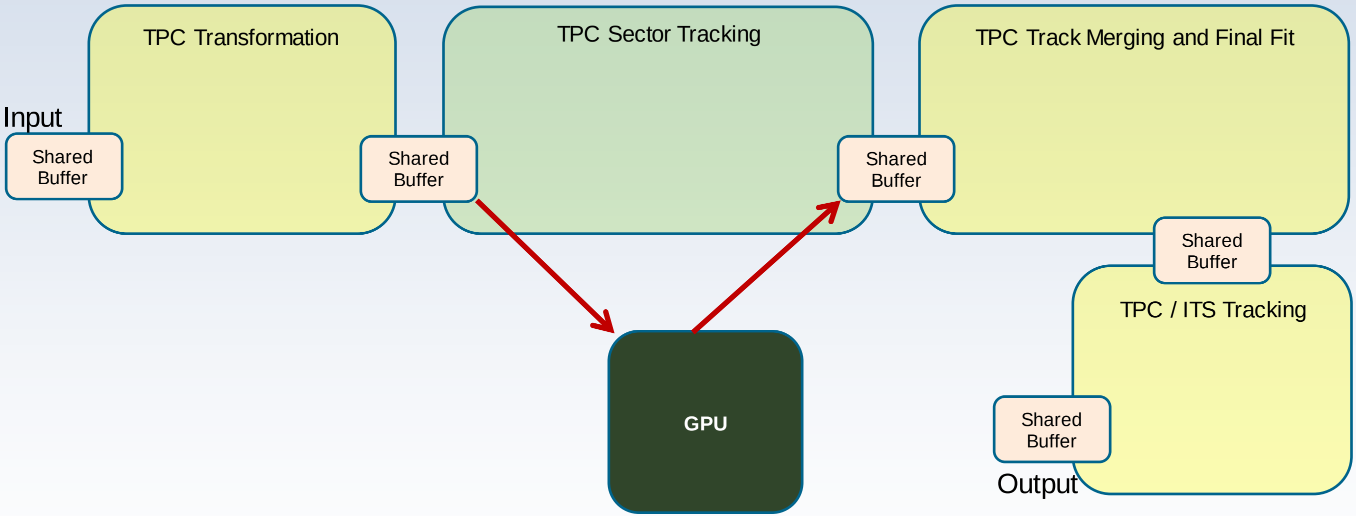
- **Changes for TPC tracking in ALICE during Run 3**
  - Continuous read-out
  - Higher interaction rate
    - → Time frames instead of events
    - → Increase of distortions

- **HLT TPC tracker operates on events as a whole.**
- **Can also process timeframe (or slice timeframe) as a whole.**
  - → Increased data size per event / time frame.
    - – Better GPU utilization through more parallelism.
    - – **Feasible since HLT tracking time goes linearly with input size.**
    - – Only limited by GPU memory – up to 16 GB today.
    - – Slicing into smaller volumes possible, as we do today.

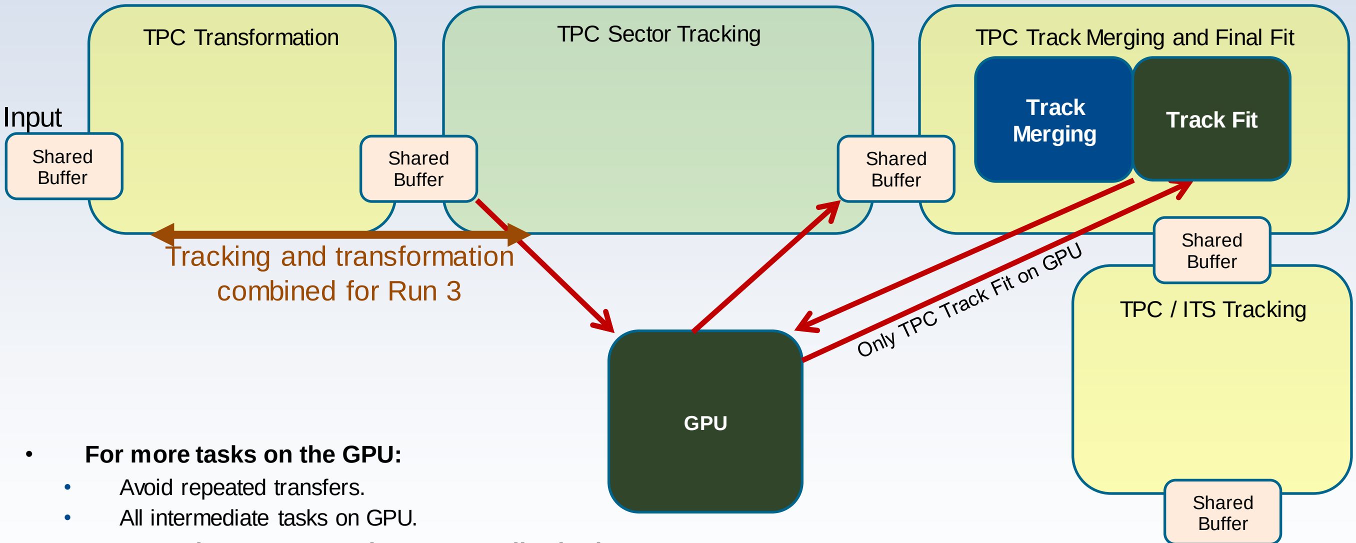
- **Hits in time frame have row, pad, time coordinates**
  - Currently, the HLT cluster transformation transforms all clusters to x, y, z coordinates beforehand.
  - No longer possible!
  - Transforming time to z needs the time of the interaction.
  - The vertex is only determined during tracking.
    - → Could apply the transformation multiple times, once per vertex, and run the tracking multiple times.
      - → Avoid if possible, very compute intense.
    - → **Joint tracking and transformation.**
  - Requires online calibration for the transformation.

- **Needs consideration during tracking:**
  - Multiple vertices per event.
  - Radial distortions for track extrapolation.

Current HLT TPC / ITS Tracking

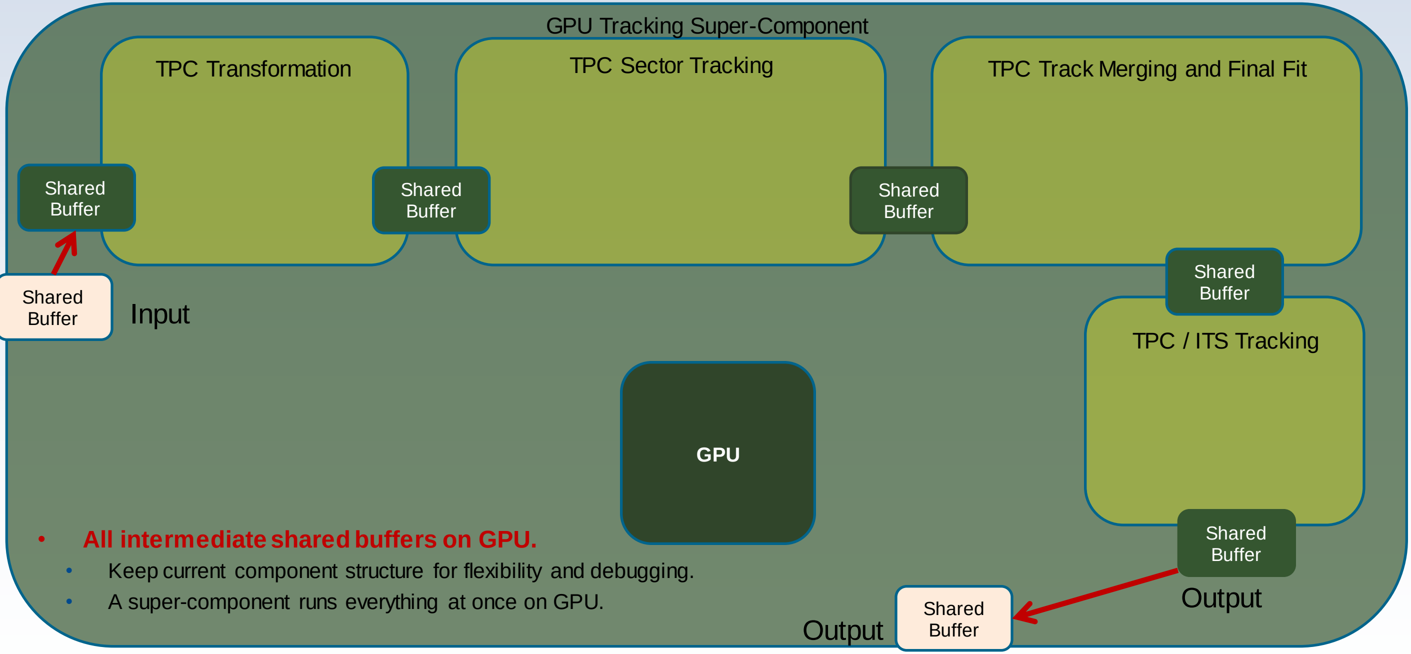


Current HLT TPC / ITS Tracking



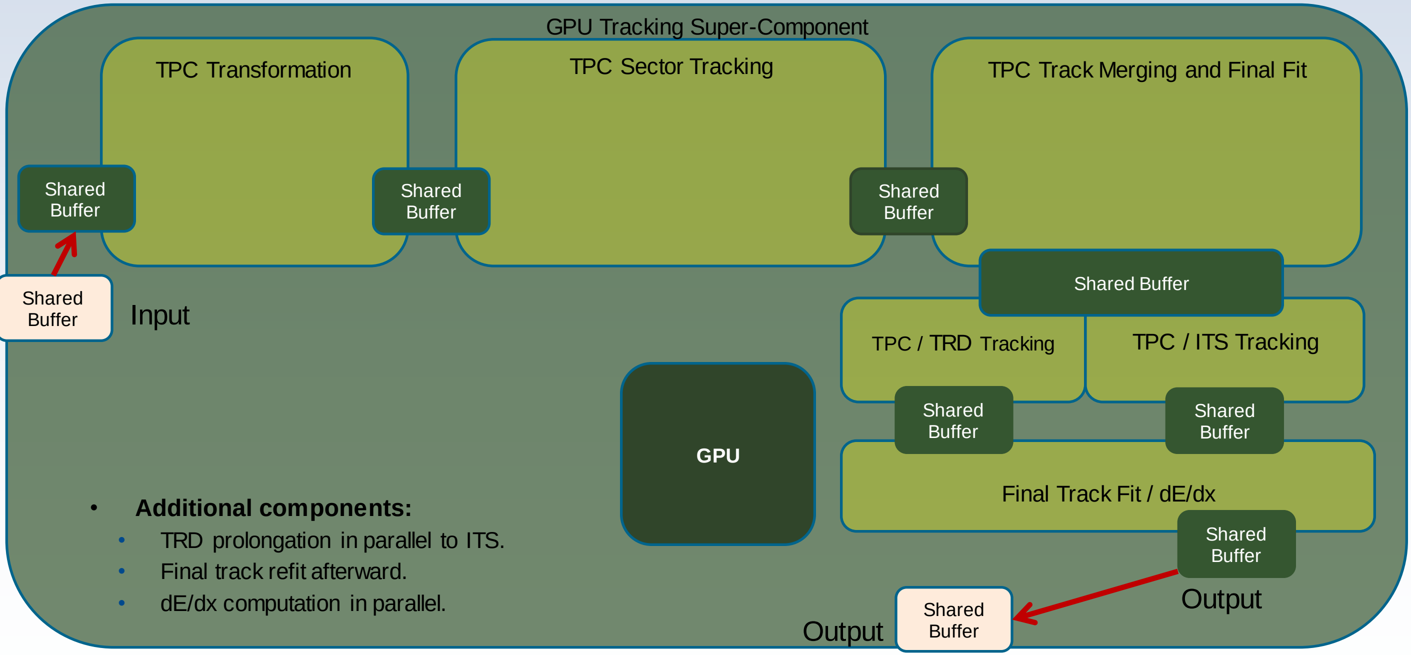
- **For more tasks on the GPU:**
  - Avoid repeated transfers.
  - All intermediate tasks on GPU.
- **Consecutive reconstruction steps well suited.**
  - The entire tracking chain seems a good candidate.

GPU Tracking Scheme for Run 3



- **All intermediate shared buffers on GPU.**
  - Keep current component structure for flexibility and debugging.
  - A super-component runs everything at once on GPU.

GPU Tracking Scheme for Run 3



- **Additional components:**
  - TRD prolongation in parallel to ITS.
  - Final track refit afterward.
  - dE/dx computation in parallel.

Summary

- **Today, different tracking algorithms online (in HLT farm) and offline.**
  - Comparable efficiency.
  - HLT has few per-cent worse resolution, B_X and B_Y components of the magnetic field missing.
  - Significant speedup in HLT through Cellular Automaton and GPU usage.
- **In Run 3: Higher rates: 50 kHz Pb-Pb with continuous read-out.**
  - Needs faster reconstruction with online calibration, and improved data compression.
- **Data compression (factor 20x needed)**
  - TPC data compression in 2016: 5.5x through Huffman compression.
  - Factor 9.1 achieved with improved prototype (arithmetic compression, track model compression).
  - Additional factor 2x estimated from removing clusters not used for physics.
- **Online calibration:**
  - TPC online drift time calibration deployed (online tracking z-residuals (w.r.t. offline) reduced from 30mm to 0.5 mm).
  - Space charge distortion calibration (SCD) deployed offline, work ongoing to run it online.
  - Work in progress: Extend online tracking to TRD and TOF (needed for SCD calibration).
- **Tracking in Run 3**
  - Reconstruct timeframe instead of events at once → improves GPU utilization, tracking time goes linear with data size.
  - Speed up reconstruction by moving more steps to GPU.

BACKUP

Structure of HLT TPC Tracking

- **Tracking split in 4 main (abstract) steps.**
  - Each step is internally split in technical substeps.
  - Phase 1 (Steps 1 and 2) on GPU, Phase 2 (Steps 3 and 4) and CPU!

#	Task	How	Locality	Description	Time	Device
1	Seeding	Cellular Automaton	Very Local (hit and adjacent hits)	Find short track candidates of 3 to about 10 clusters.	Ca 30%	GPU or CPU
2	Track following	Kalman Filter (simplified)	Sector-local	Fit parameters to candidate, find full track segment in one sector via track following with simplified Kalman filter (e.g. constant B-field, y and x uncorrelated, …)	Ca 60%	
3	Track Merging	Combinatorics / Mathematics	Global	Merge track segments within a sector and between sectors	Ca 2%	CPU only (GPU version available but not used
4	Track Fit	Kalman Filter (full)	Global	Full track fit with full Kalman filter (polynomial approximation of B-field)	Ca 8%	

Common tracker source code

- **CPU and GPU tracker (in CUDA) share common source files.**
- **Specialist wrappers for CPU and GPU exist, that include these common files.**

common.cpp:

```
__DECL FitTrack(int n) {
….
}
```

cpu_wrapper.cpp:

```
#define __DECL void
#include ``common.cpp``

void FitTracks() {
  for (int i = 0;i < nTr;i++) {
    FitTrack(n);
  }
}
```

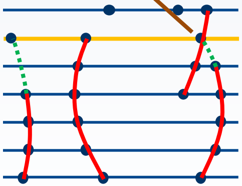
cuda_wrapper.cpp:

```
#define __DECL __device void
#include ``common.cpp``

__global void FitTracksGPU() {
  FitTrack(threadIdx.x);
}

void FitTracks() {
  FitTracksGPU<<<nTr>>>();
}
```

→ **Same source code for CPU and GPU version**
- The macros are used for API-specific keywords only.
- The fraction of common source code is above 90%.

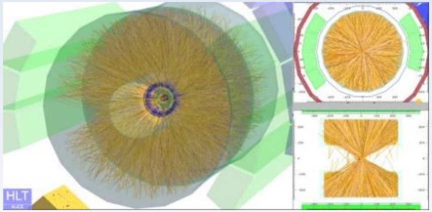
Consistency of Tracking Results

- **Cluster to track assignment**
  - Problem: Cluster to track assignment was depending on the order of the tracks.
    - Each cluster was assigned to the longest possible track. Out of two tracks of the same length, the first one was chosen.
    - Concurrent GPU tracking processes the tracks in an undefined order.
  - Solution: We need a continuous (floating point) measure of the track quality.
    - Two 32-bit floats can still be identical, but that is unlikely.
- **Similar problem in track merging, which depended on track order.**



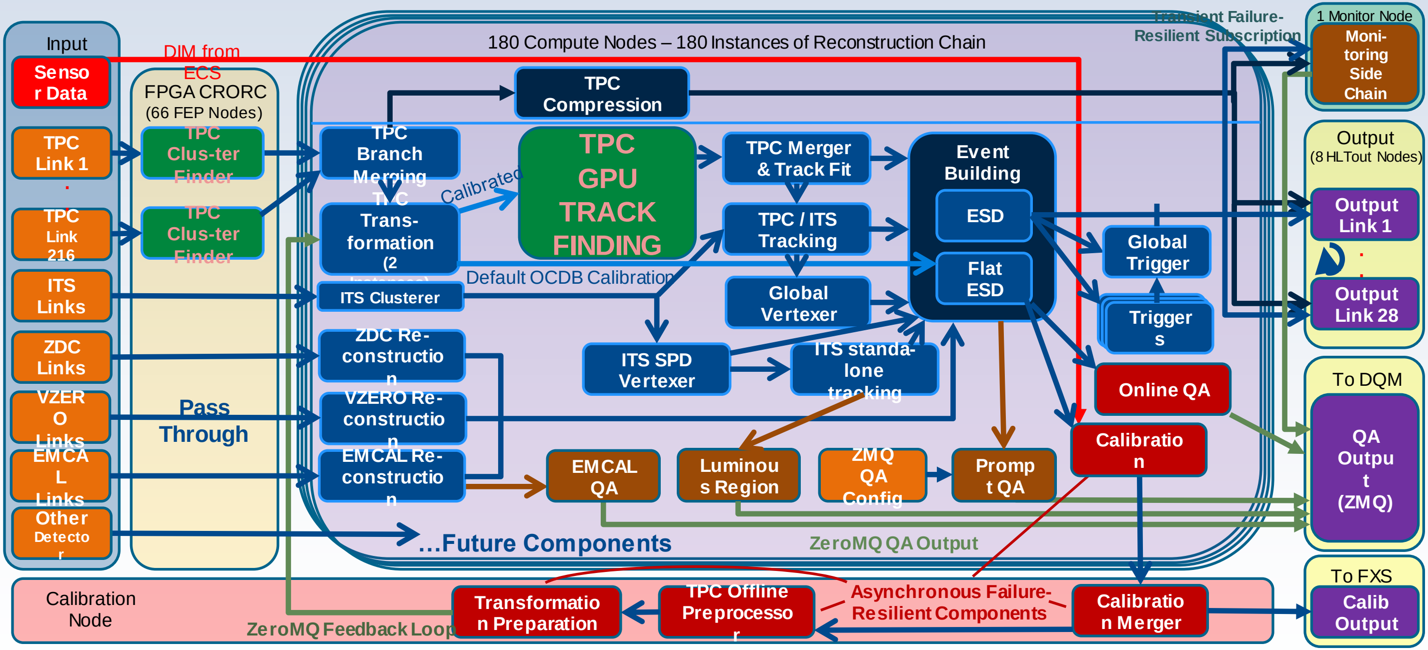
ALICE HLT TPC Tracking on GPU

- **HLT track reconstruction fast enough to cope with all trigger scenarios in Run 2 and with the maximum TPD optical link rate.**
- **Tracker has a common source code for CPU / OpenCL / CUDA yielding consistent results.**
- **180 compute nodes with GPUs in the HLT**
  - Since 2012 in 24/7 operation, no problems so far.
- **Cost savings compared to an approach with traditional CPUs:**
  - About 500.000 CHF during ALICE Run I.
  - Above 1.000.000 CHF during Run II.
  - Mandatory for future experiments, e.g. CBM (FAIR, GSI) and ALICE upgrade with >1TB/s data rate.
  - Can be used to test new online tracking features for Run III.



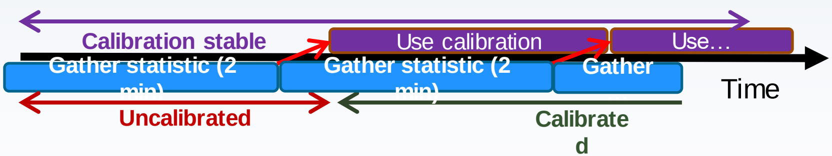
- **We are now looking into optimizations for new GPU architectures, but not yet specific to one model.**
  - Plan to bring more components onto the GPU, reduce PCIe transfer, keep component structure.
  - Using GPUs with more memory, we are confident to process timeframes similarly to events today.

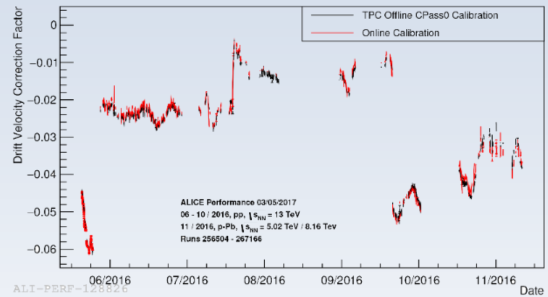
Overview ALICE online reconstruction components (2016)

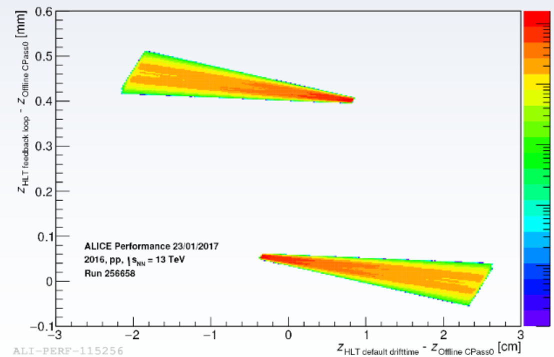


Online calibration in the HLT

- **No significant difference in the drift velocity computed online and offline between May and December 2016.**
    - Gaps in between are periods without beam.
    - **Different ambient pressure and TPC temperature values online and offline are corrected by calibration** yielding slightly different factor.
- **Online calibration cluster z-position residual compared to offline below 0.5 mm (below the intrinsic TPC resolution).**
    - 30 mm without online calibration.
        - → HLT tracking in its current state is fully sufficient for calibration purpose.
    - Gathering calibration data takes 2 minutes per interval.
    - Preparation and distribution takes 30 seconds (**feedback loop**).
        - → The first 2.5 minutes of a run are without online calibration.





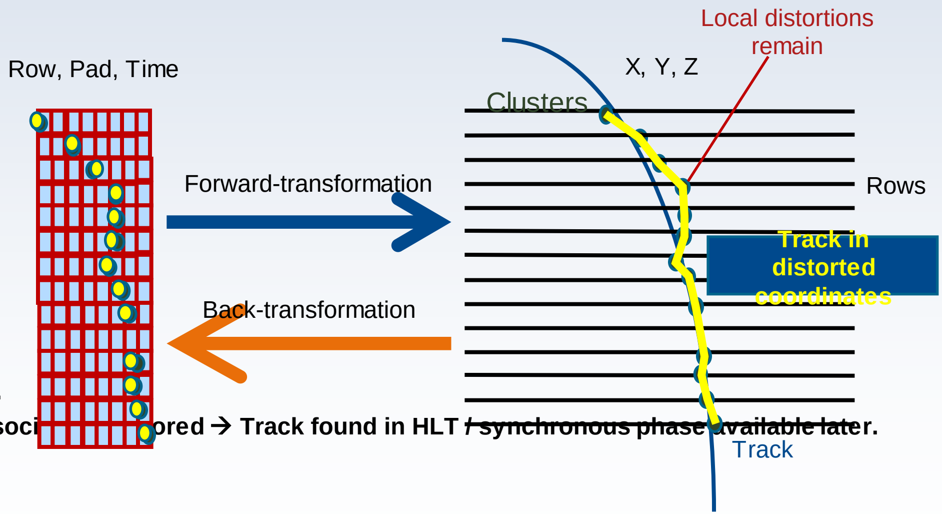


Track model compression

- **Solution:**
  - Employ […] **Work in Progress** […]mation. *(In principle, every transformation works, but the closer the better!)*
  - **Refit track in distorted coordinate system**.

- **Encoding:**
  - Transform hits from row, pad, time to distorted x, y, z using fast transformation.
  - Refit the track in distorted frame.
  - Propagate to the next row.
  - Transform back to row, pad, time and store residual to next cluster in row, pad time.

- **Decoding:**
  - Propagate track in distorted frame to X-row.
  - Transform track position to row, pad, time.
  - Add residual and store cluster position.
  - Propagate back to X, Y, Z, and refit track.



- **Non-associated clusters still compressed with differences scheme.**
- **Additional benefit: Cluster to track associ… ored → Track found in HLT ~~/ synchronous phase available later~~.**