Tracking at ATLAS and CMS

w/ special emphasis on Run2 performance and developments for Run3

mia tosi

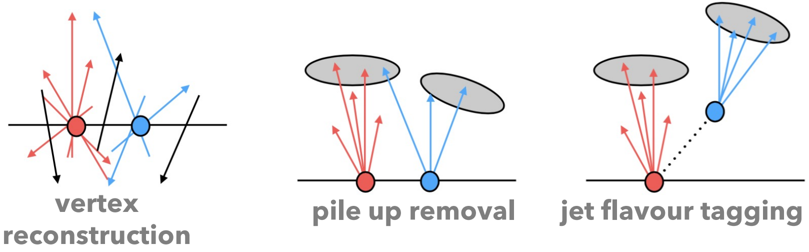
on behalf of the ATLAS and CMS collaborations

-- LHCP 2021 --

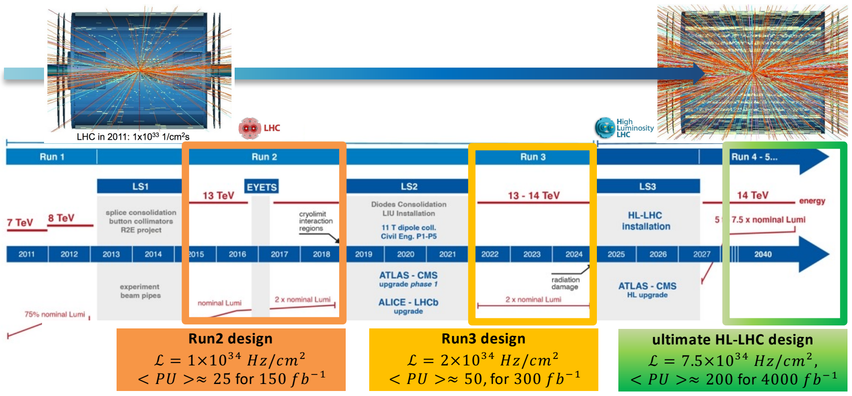
Jun 10th, 2021

the tracking challenge at the LHC

- tracking is a key ingredient of reconstructing the full event
  → used in almost every element of reconstruction
- these need to be reconstructed w/
  - very high **efficiency** ($> 90\%$ for $\sim GeV$ pions)
  - **precise** track parameters
  - very **low fake** rate: $O(\sim\%)$
  - **quickly** (stringent CPU limits)



- in a proton-proton collision in Run2, typically, $20 \div 30$ charged particles w/in the tracker acceptance and 40 collisions per bunch crossing: $\Rightarrow O(1000)$ charged particles per event



this represents a **complex combinatorial problem**, which increases in difficulty w/ pile-up

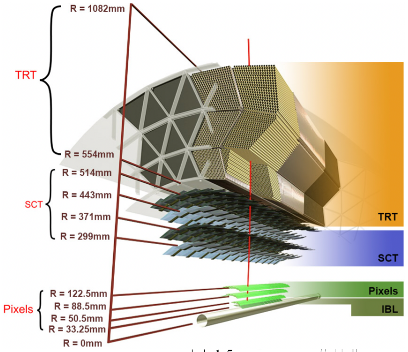
the quality of the reconstructed track candidates becomes challenging to maintain under high pile-up

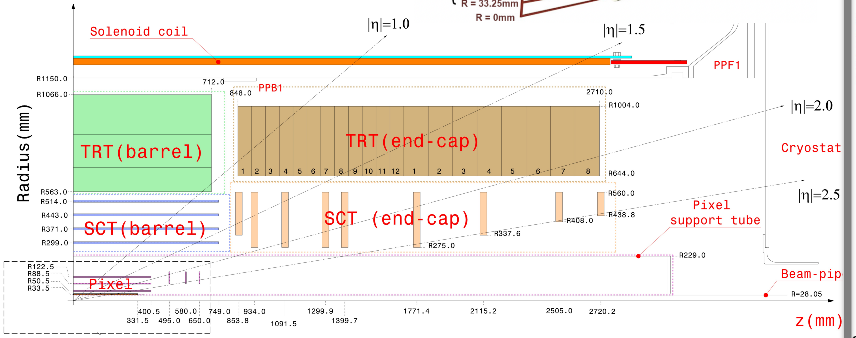
- high cluster density leads to incorrect cluster-to-track association, pulling the reconstructed trajectories from their true values
- w/ the increase of the available clusters, the random collections of clusters (fake tracks) increases
- the timing required for tracking scales rapidly w/ $< \mu >$

introduction : ATLAS and CMS tracker

✓ coverage $|\eta| < 2.5$
✓ axial B-field : $\mathbf{2}\ \boldsymbol{T}$

3 different technologies
- silicon pixels (pixel)
- silicon strips (SCT)
- gaseous straw tubes (TRT)





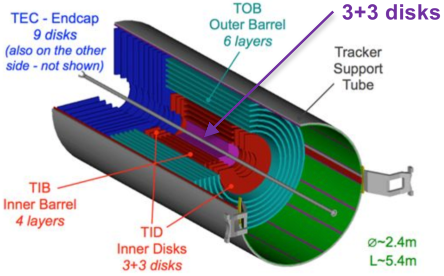
Subdetector	Element size	Intrinsic resolution [μm]	Radius barrel layers [mm]
IBL	50 μm×250 μm	8×40	33.2
Pixel	50 μm×400 μm	10×115	50.5, 88.5, 122.5
SCT	80 μm	17	299, 371, 443, 514
TRT	4 mm	130	from 554 to 1082

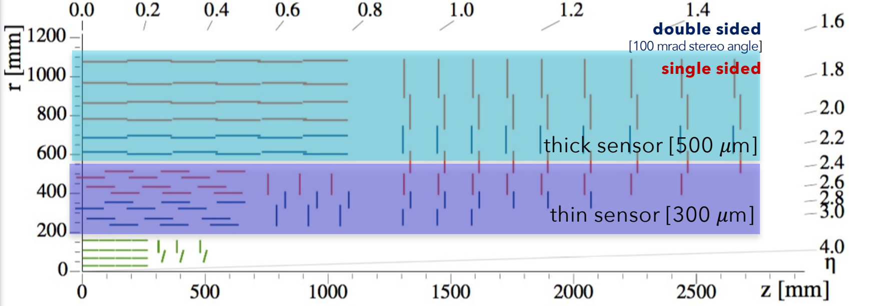
✓ coverage $|\eta| < 3.0$
✓ axial B-field : $\mathbf{3.8}\ \boldsymbol{T}$

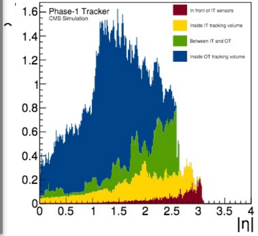
2 different technologies
- silicon pixels (pixel)
  - ✓ occupancy: $O(10^{-3})$
  - ✓ hit resolution: $(10,40)\times(230,530)\ \mu m$
- silicon strips (strip)
  - ✓ occupancy: $O(10^{-2})$
  - ✓ hit resolution: $10\times(20,40)\ \mu m$

until 2016
3 layers
2+2 disks

Pixel
4 layers
3+3 disks







r_{inner} [mm]	r_{outer} [m]	X_0 @ $\|\eta\| = 0$	p_T resolution @1(100) GeV, $\|\eta\| = 0$
29	1.1	0.4	0.7 (1.5)%

non negligible amount of dead material inside the tracker volume

ATL-SOFT-PUB-2007-007

track reconstruction procedure - ATLAS

Primary Tracking (INSIDE-OUT) → prompt tracks

seeds are formed **from triplets** of hits in **silicon detectors**

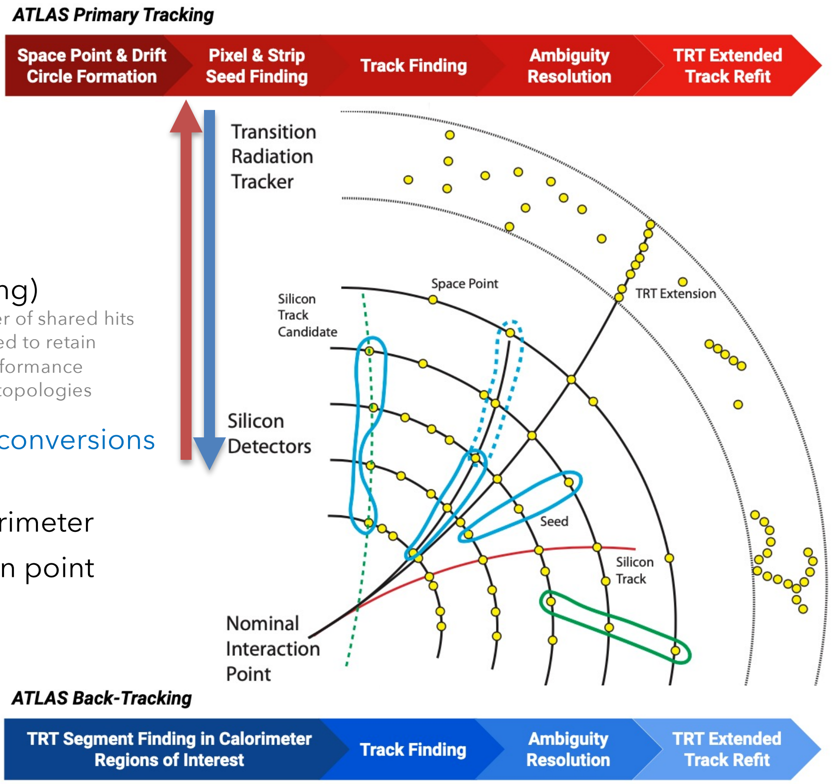
a combinatorial Kalman filter is used to extend the seeds and build track candidates

track candidates are **scored** according to their track parameters and hit topology (number of shared hits, holes, hits-on-track)
candidates w/ poor quality are then removed (Ambiguity Solving)

silicon **tracks** are **extended to** the **TRT** sub-detector

global χ^2 fit for precise track parameters evaluation

limited number of shared hits is permitted to retain high performance in dense topologies



Back-Tracking (OUTSIDE-IN) → secondary tracks and photon conversions

seeds are formed **from TRT** hits and **SCT** as well, only in Regions of Interest determined by deposits in the EM calorimeter

a Kalman filter is used to backport the seeds toward the interaction point and build track candidates

Other-Tracking → forward muons and short tracks

seeds are formed only from left-over hits

Displaced-Tracking → displaced tracks

dedicated version of the track reconstruction w/ wider search window in the transverse impact parameter on dedicated set of collision data in Run2 (by default in Run3)

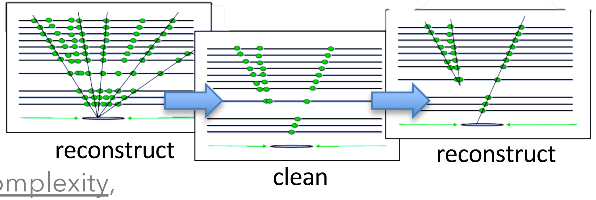
JINST 9 (2014) P10009

track reconstruction procedure - CMS

Combinatorial Track Finder : extension of the Kalman filter to perform both the pattern recognition and the track fitting in the same framework

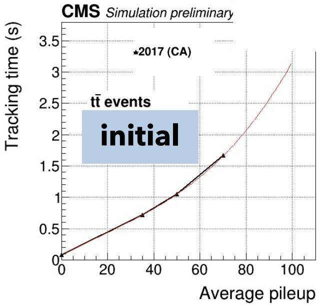
tracks reconstruction is an **iterative procedure**:

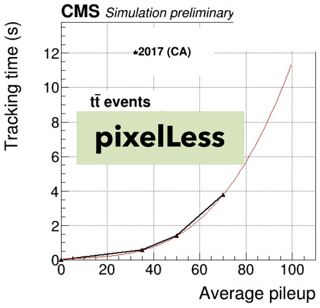
initial iterations search for tracks that are easiest (and fastest) to find, the corresponding hits are removed, thereby reducing the combinatorial complexity, and simplifying the subsequent iterations in search for more difficult classes of tracks



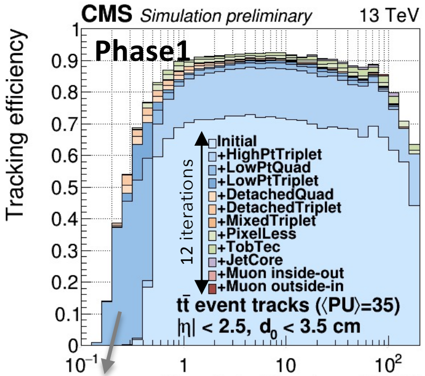
Iteration	Seeding	Target track
Initial	pixel quadruplets	prompt, high p_T
LowPtQuad	pixel quadruplets	prompt, low p_T
HighPtTriplet	pixel triplets	prompt, high p_T recovery
LowPtTriplet	pixel triplets	prompt, low p_T recovery
DetachedQuad	pixel quadruplets	displaced--
DetachedTriplet	pixel triplets	displaced-- recovery
MixedTriplet	pixel+strip triplets	displaced-
PixelLess	inner strip triplets	displaced+
TobTec	outer strip triplets	displaced++
JetCore	pixel pairs in jets	high-p_T jets
Muon inside-out	muon-tagged tracks	muon
Muon outside-in	standalone muon	muon

- the *InitialStep* makes use of high-p_T **quadruplets** coming from the beam spot region
- subsequent steps use **triplets** or improve the acceptance either in p_T or in displacement
- the later steps use seeds w/ hits from the strip detector to find detached tracks
- final steps are dedicated to special phase-space
  - highly dense environment (i.e., w/in jets)
  - clean environment (i.e., muons)

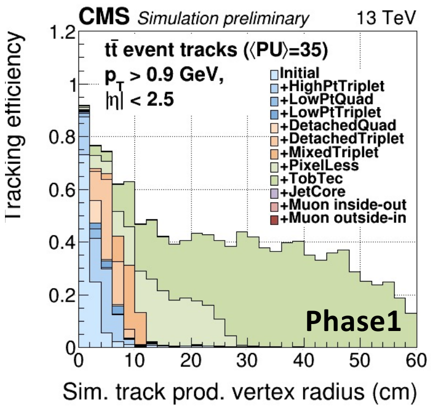




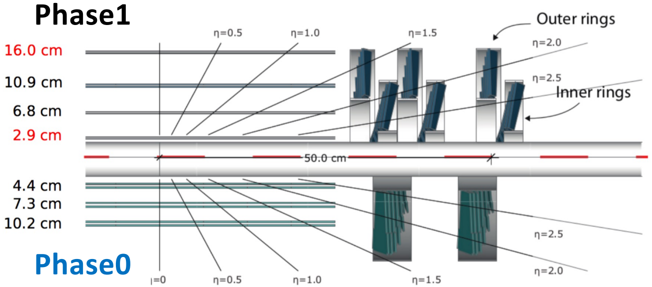
tracking performance : efficiency

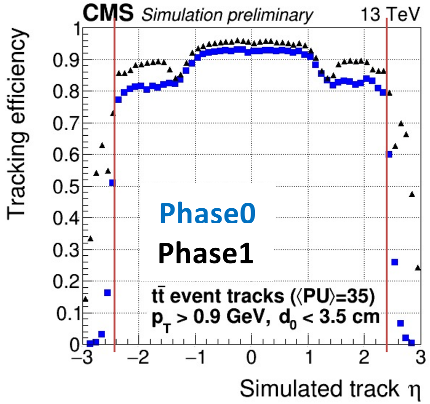


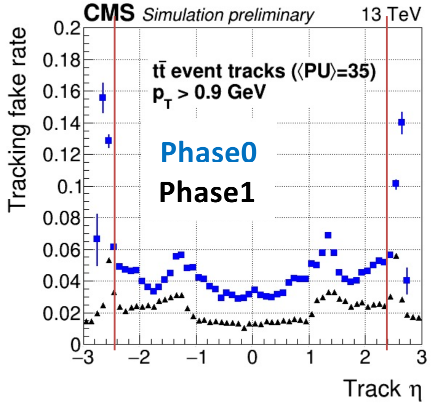
yes, down to 200 MeV ;)

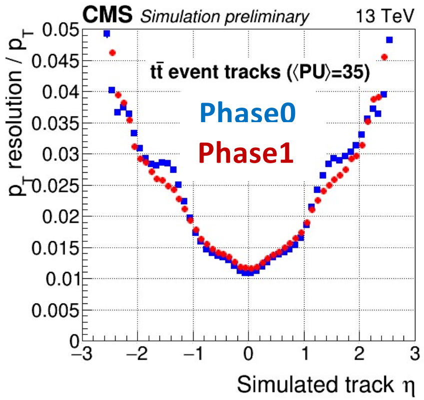


high tracking performance
thanks to significant improvements made during both the LS1 and Run2: new iterations, new tuning, PU mitigation, code re-engineering, new seeding framework, **Cellular Automaton seeding**









✓ increase efficiency
(above all at high pseudo-rapidity)
✓ decrease fake rate
✓ improve p_T resolution
(mainly in the transition region)

tracking efficiency in data using Z and J/Ψ into muons via tag-and-probe (see backup)

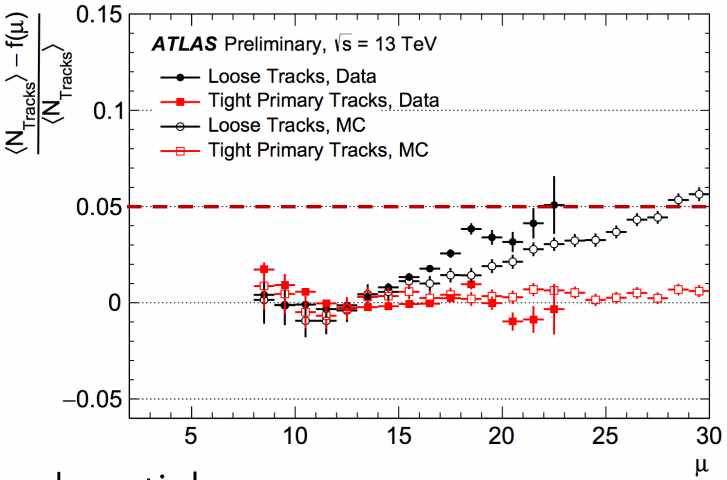
ATLAS has very similar behaviour

tracking performance : fake rate

higher pile-up ⇒ higher combinatorics ⇒ higher fake rate

fake rate of the order of <5% on large η/p_T range

➢ highly curved (very low- p_T)
  or very straight tracks (high- p_T)
  are more likely to be fakes



data driven technique to extract fake rate
exploits relation : $< N_{tracks} > \propto < \mu >$

deviations due to 2 effects:

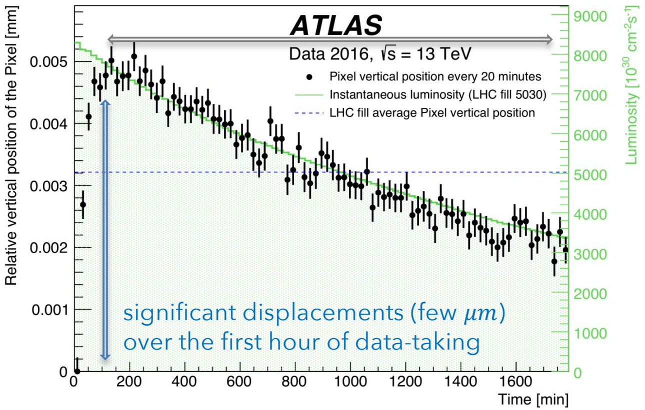
- **combinatorial** fakes
- increased number of **secondaries** that "give" hits to primary produced charged particles

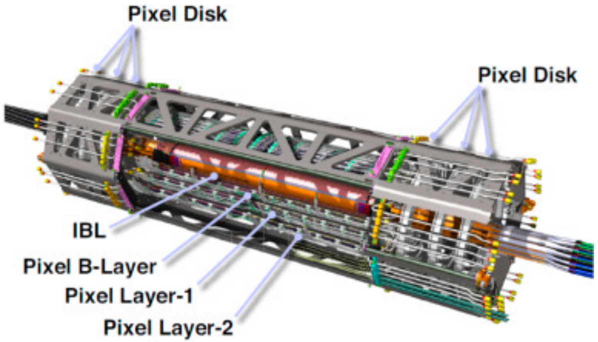
tracker aligment

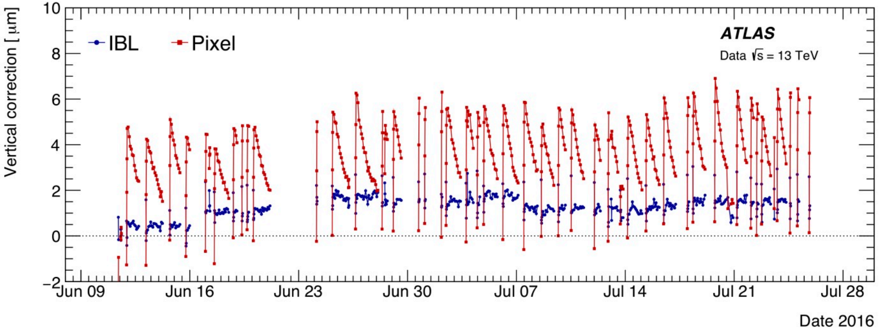
- the average inner detector position across an LHC fill does not describe the sub-detector's position!
  - ➢ short-timescale movements while recording data driven by thermal effect (electrical power consumption of on-module readout) at the centre of the ATLAS detector

  - ✓ automated alignment scheme for the Inner Detector
    → dynamic alignment update throughout each LHC fill calibrating the recorded data
    - every 20 minutes during the first hour of data-taking
    - every 100 minutes for the rest of the fill

@thermal equilibrium, as the LHC luminosity decreases, the sub-detector's overall system thermal mass increases, inducing it to drift slowly (in the opposite direction)

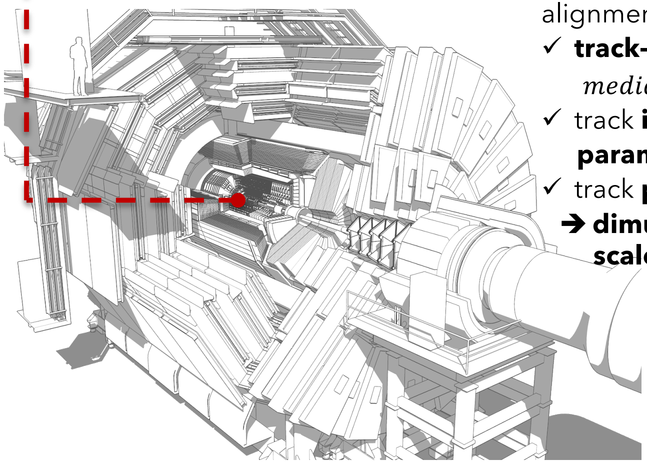






tracker aligment

- intense work to produce ultimate Run2 **alignment** (2016-2018), corresponding to ~ 140 fb^{-1}
  - for each year, the whole data is aligned in a <u>single, global fit</u>
    in order to accumulate enough cosmic rays (2÷4M), since this is the limiting factor
  - coping with <u>residual systematic changes in hit positions due to radiation</u>
  - largest alignment fits to date in terms of number of parameters to align, with up to **~700k parameters** !
    → ~220 geometries over the three years to cover significant changes over time



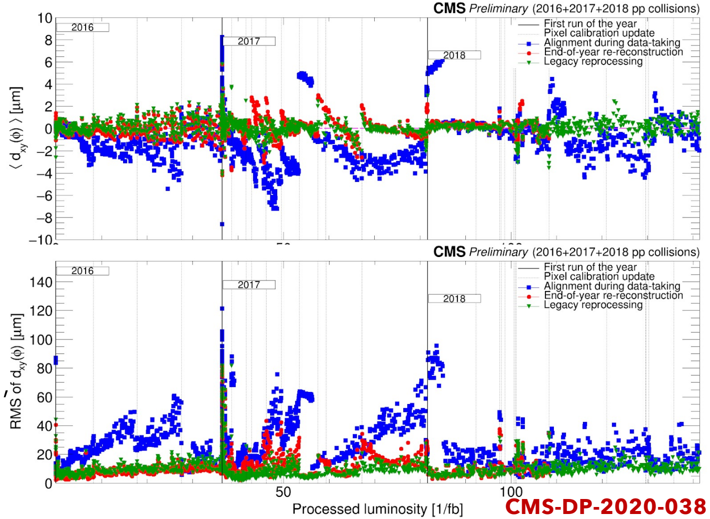
alignment flattens / improves

✓ **track-hit residual**, $median(x'_{pred} - x'_{hit})$

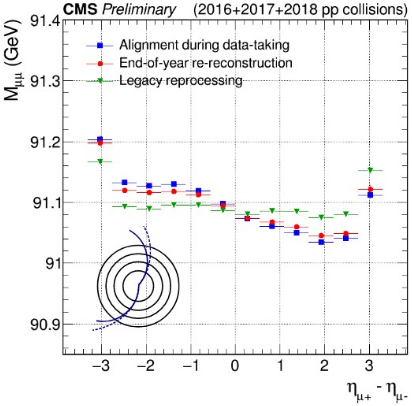
✓ track **impact parameter**, $< d_{xy} >$

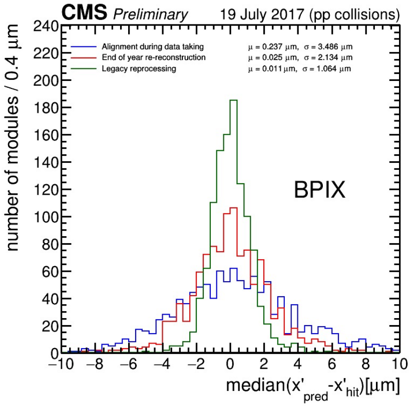
✓ track **p$_T$ response**

➔ **dimuon mass**, $M_{\mu\mu}$, **scale** and **resolution**



CMS-DP-2020-038





Run2 Legacy processing

Goal

- full exploitation of Run2 data, performance improvements and ultimate precision in calibration and alignment
- a homogenous set of data and MC for analysis

Performance improvements

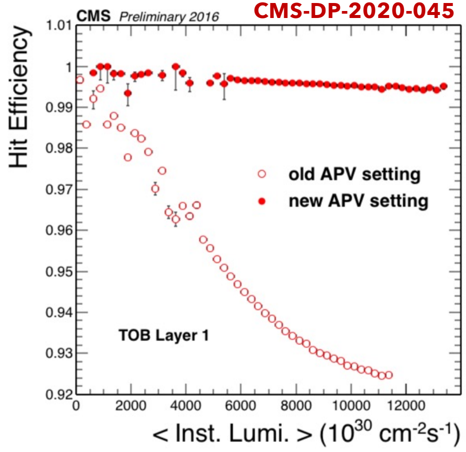
- data realignment and recalibration
- improved MC simulation and digitization model, reconstruction and physics objects

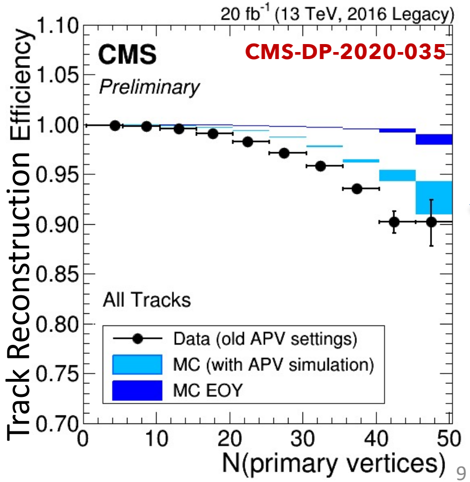
Example

2016 non-optimal setting of the parameter governing the drain speed of the preamplifier circuit **used in the Strip Tracker readout ASIC (APV25)**

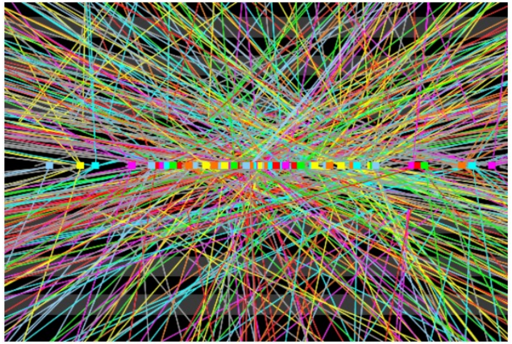
led to saturation effects in the pre-amplifier of the APV25 read-out chip
→ dynamic hit reconstruction inefficiency

for Legacy processing
→ APV simulation leads to improved data/MC agreement
→ mitigation in the pattern recognition used for the track reconstruction





beyond Run2



- future LHC upgrades offer the opportunity for an order of magnitude greater data samples
- to exploit fully the LHC luminosity upgrades, ATLAS and CMS must **preserve** (or even **enhance**) the current **performances**

…in a much challenging environment: **pile-up**

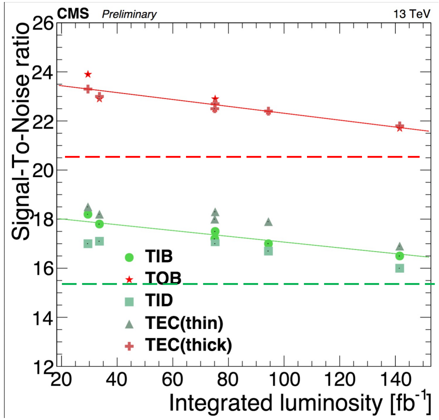
LHC demonstrated
the ability to deliver
even beyond expectations !

- Run3: $< PU > \approx 50$
- Run4: $< PU > \approx 140 \div 200\ (!)$

  - unprecedented challenges for pattern recognition
  - high radiation dose to detectors

during Run3, there will be a degradation of the detectors w.r.t. nominal performance due to
- the **extreme PU scenario**
- the **accumulated radiation**

$\Rightarrow$ **no showstoppers** have been identified
✓ PU conditions seem to impact more than the ageing
✓ impact on overall physics performance should be manageable
  - tolerable degradation of tracking and vertexing resolution

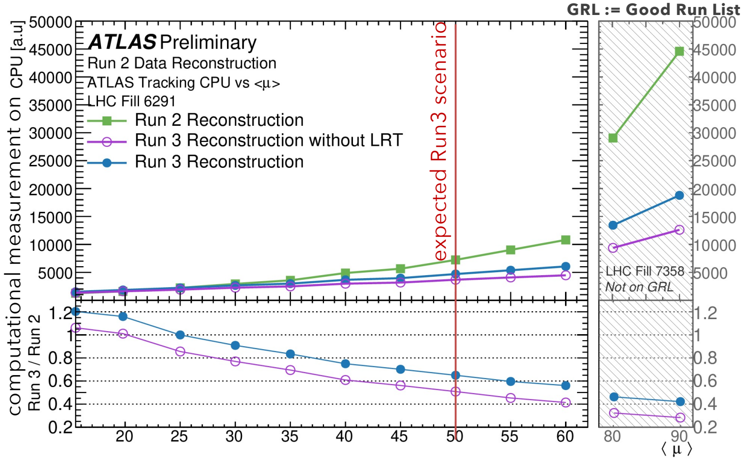


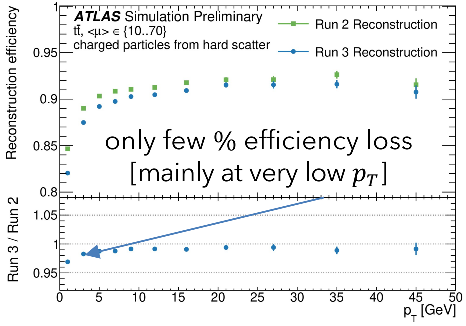
ATL-PHYS-PUB-2021-012

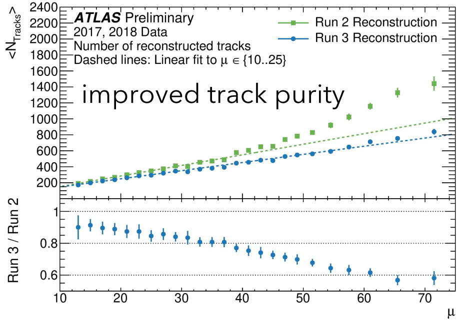
track reconstruction improvements for Run3

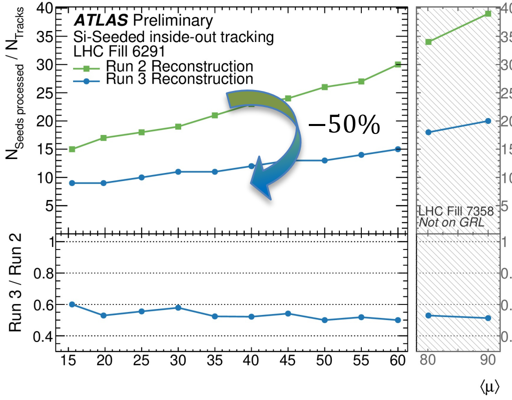
many **tunings** and **adjustments** have been developed for Run3
stricter requirements on the track candidates,
the window width, seeding and back-tracking, ..

- near linear scaling of the CPU consumption w/ $< \mu >$
- **timing** of the **pattern recognition reduced by a factor 4** (!)
  [the other improvements of about a factor $1.5 \div 2$]
- the fraction of the event reconstruction taken by the tracking reduced to 40% at $< \mu > = 50$ [it was 64%]
- **fake track** reconstruction **rates drastically reduced**, and the average quality of the tracks increased
  [large reduction in the overall number of tracks written to disk, reducing the needs for storage space]
- tracking efficiency is only marginally affected







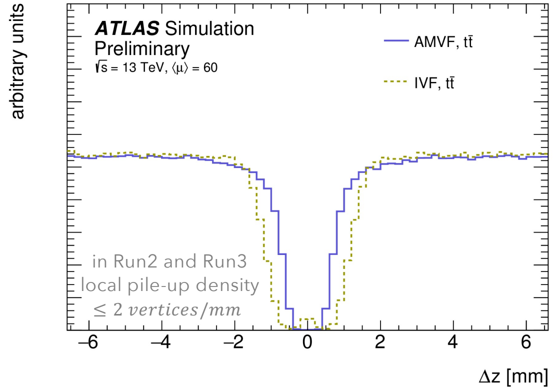


ATL-PHYS-PUB-2019-015

vertex reconstruction improvements for Run3

CMS Vertex Reconstruction is based on

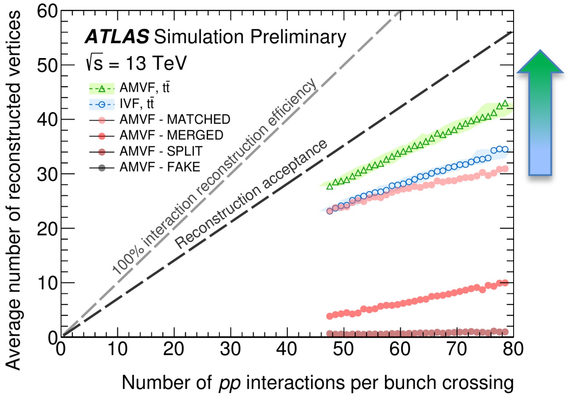
- ✓ **Deterministic Annealing for track clustering**
- ✓ **Adaptive Vertex Fitter** for best **position estimate**
- ➢ vertex reconstruction efficiency : ~75%
- ➢ for Run3 : significant improvements in the computing timing achieved by
  - mild approximations in the function evaluation
  - relaxed convergence criteria at high T
  - restricted the z–range for track–cluster assignments



ATLAS moved to **Adaptive Multi-Vertex Finder (AMVF)**

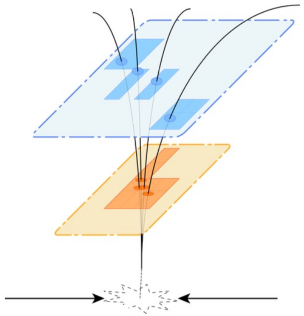
- change of seed finder to **Gaussian seed finder**
- tracks for vertex fitting are associated to seed according to impact parameters significance and constrained to the seed position in *z*
- tracks share weights w/ **multiple vertices**, which are **fit simultaneously**

- ➢ better overall vertex reconstruction efficiency
  - **- at high-μ, recovered 30% of reconstruction efficiency**
  - **- less dependence** of reconstruction efficiency **on pile-up density**
  - improved longitudinal separation
- ➢ in addition, ACTS-provided implementation brings a 40% reduction in the CPU timing ;)



tracking in dense environment

ATL-PHYS-PUB-2015-006



in ATLAS makes use of **Neural Networks (NNs)**:

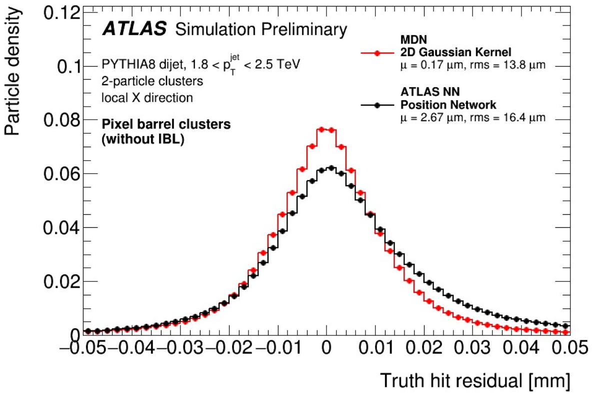
- hit multiplicity of a given cluster : **shared cluster splitting** (1 classification NN)
- hit position and associated uncertainties of split clusters

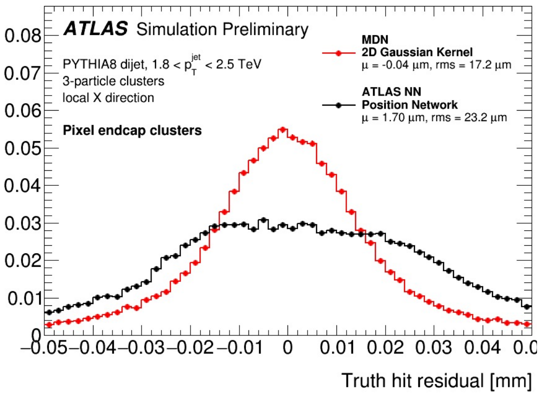
NN : 3 NNs for (x, y) **position** + 2x3 for (σ_x, σ_y) **uncertainties** NNs

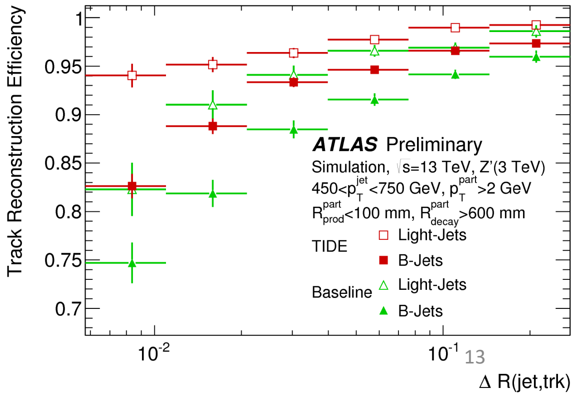
MDN : 3 Mixture Density Network for both (x, y) **position** and (σ_x, σ_y) **uncertainties**

in the core of high-p_T jets, average separation of tracks can be **smaller than sensitive elements size**
→ hits from different tracks can result in a **merged pixel cluster**
[the effect is even more pronounced for b-jets, because the B decay happens closer to the pixel detector]

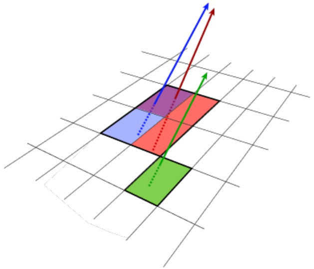
- ➢ **improved tracks reconstruction in dense environments**
- ➢ **residual inefficiencies** due to **not-fully efficient cluster splitting**





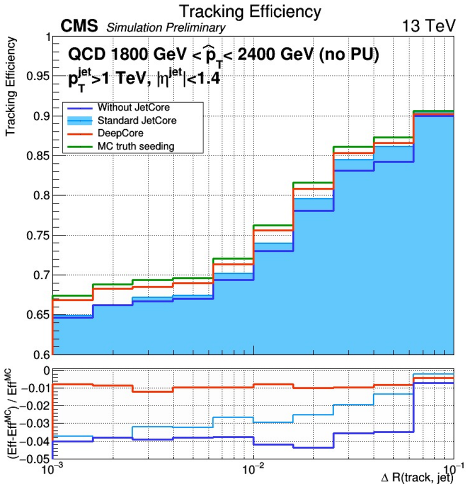


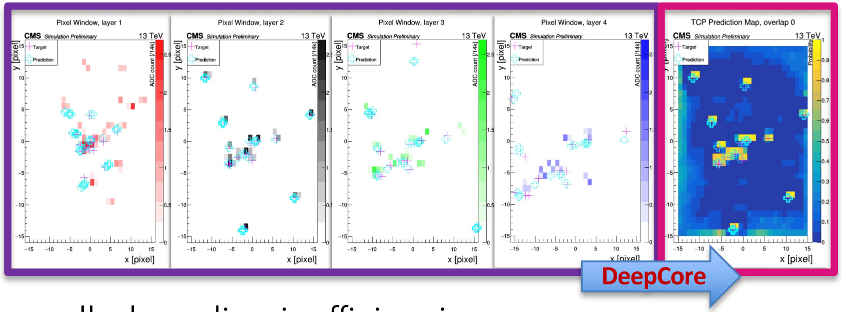
tracking in dense environment DeepCore



- in Run2, dedicated step has been added : **Jet Core**
  → the merged clusters in the pixel detector affect already the seeding step

- in Run3, the **DeepCore** will be deployed
  ⇒ basic idea is to **skip the pixel clustering**,
  exploit directly the RAW pixel data and **reconstruct the seed of tracks w/in the jets**
  → develop a convolutional Neural Network (cNN)
  to reproduce the «function»





✓almost cancelled seeding inefficiencies
✓**fake rate reduction up to 60%**
✓**seeding timing reduced by 85%**

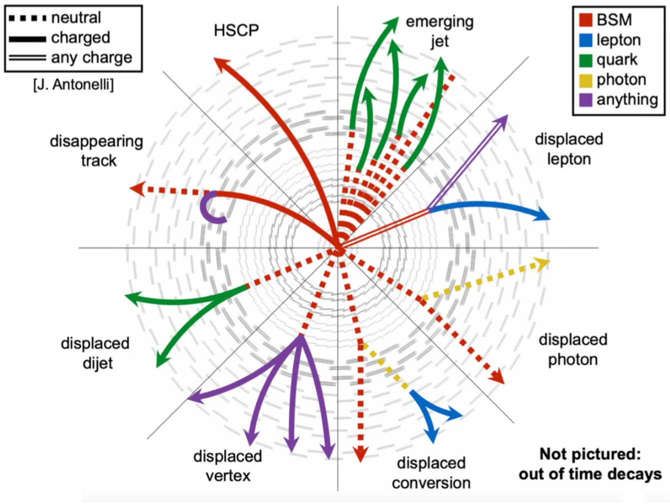
displaced tracking

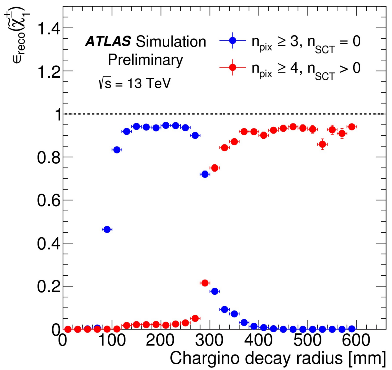
ATL-PHYS-PUB-2019-011

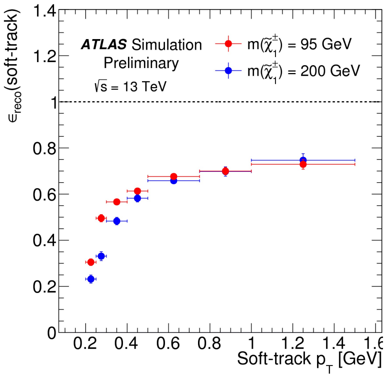
standard track reconstruction is optimised
to mainly reconstruct tracks from primary interactions

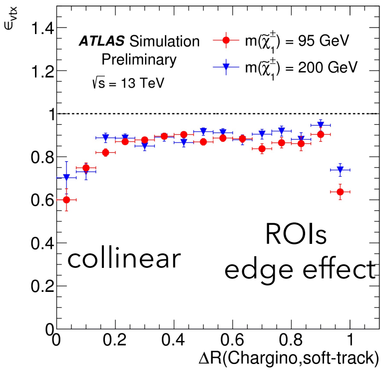
tracks originating from LLP decays can have:
- **large impact parameters**
- **fewer inner hits**, especially in the pixel sub-detectors
- **point** to a **largely displaced decay vertex**
- can be **soft** reaching $p_T < 300\ MeV$

both ATLAS and CMS developed dedicated reconstruction



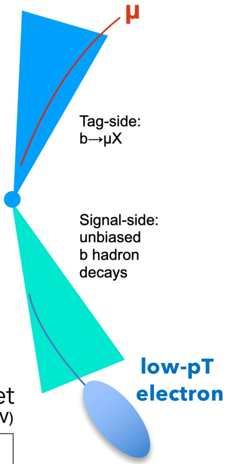






- second pass tracking on the left-over hits:
  - only ~5% efficiency loss on simulated chargino tracks
  - starting from 3 pixel hits, special requirements on SCT extension
- soft pion:
  - tracking in a Region of Interest (ROI)
  - last seed is the end of PIX tracklet, SCT only for extension
- 2 track vertex:
  - ~90% efficiency,
  - O(1mm) resolution

dedicated reconstruction for low pT electron



custom **low-pT electron reconstruction**

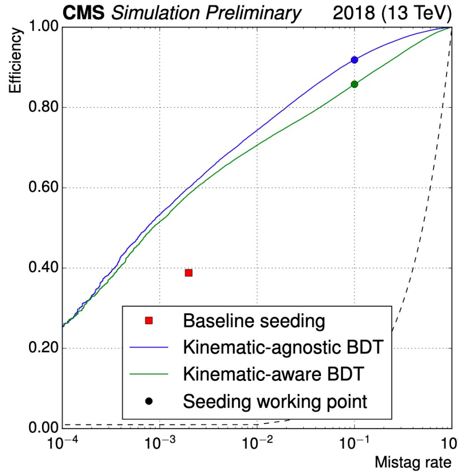
→Gaussian Sum Filter (GSF) tracking (computationally expensive)

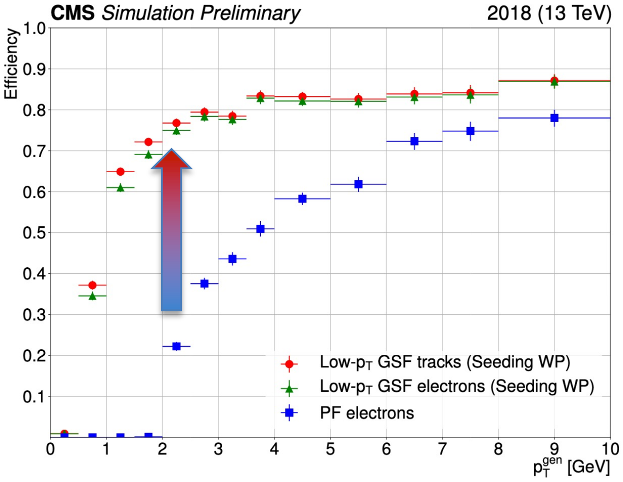
seeded by a more computationally efficient logic that identifies low-pT electron candidates

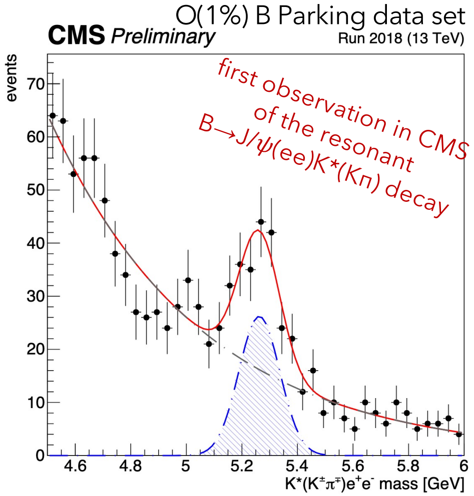
two independent boosted decision trees (BDT) that provide discrimination based on

- a "kinematically agnostic" BDT (exploits tracking and calorimeter information)
- a model-dependent "kinematically aware" BDT (utilizes the p_T, η, and the track impact parameter of an electron candidate)

→ a loose "seeding working point" yields a **10% mis-identification rate** while providing a factor **~2 gain in efficiency**







tracking at HLT : heterogenous computing

using other computing hardware
than regular CPUs (GPUs, FPGAs, ..)

Run3 can be "full commissioning"
to prepare for the Run4 step-change
[radical solutions will be necessary]

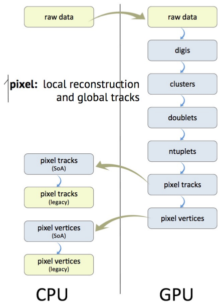
CMS foresees to move to an "heterogeneous" computing model
in particular, to **offload HLT algorithms to GPU already in Run3**

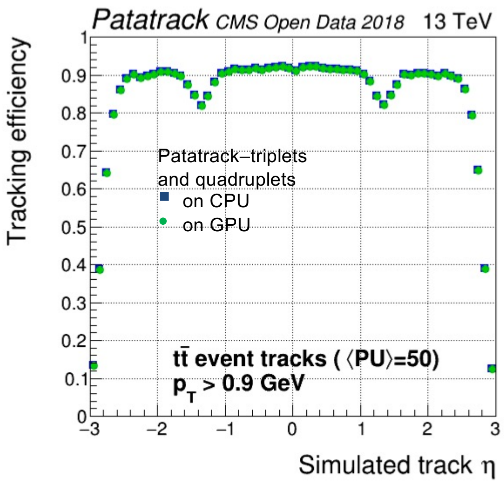
- gain experience with heterogenous architectures ahead of Phase2
- plan to equip each node with GPU,
  reduce the overall HLT farm CPU by the amount we can offload to the GPU

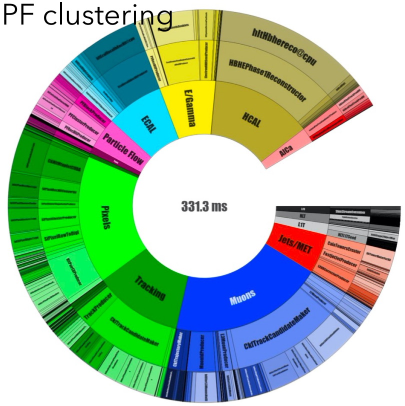
current **porting of codes to GPU** :

- ✓ pixel local reconstruction
- ✓ pixel-only tracking and vertexing
- ❑ strip clusterizer
- ✓ HCAL local reconstruction
- ✓ ECAL local reconstruction
- ❑ PF clustering

- developed 2 **Multiple Scattering-aware fits**:
  - Riemann Fit
  - Broken Line Fit
- developed a **seeds cleaner** "fishbone"
- implemented the **z-clustering by DBSCAN**



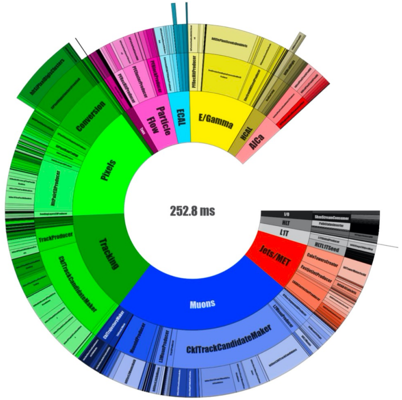




-24% cpu usage
+22% throughput

~20% of online reconstruction can be offload to GPU

successfully tested @P5
(still on cosmics data taking)



conclusions

tracking algorithms need to provide high-quality tracks efficiently and w/ an efficient use of resources

✓ **high tracking** and **vertexing performance** in Run2
[despite challenging conditions at the LHC]

- in order to provide more precise and accurate track reconstruction sophisticated algorithms, techniques and calibrations have been developed :
  - simulation accurate in predicting tracking behaviour
  - detailed studies of material
  - dynamic alignment
  - track efficiency from data-driven techniques

very good performance in Run2

commissioning of new strategies in Run3

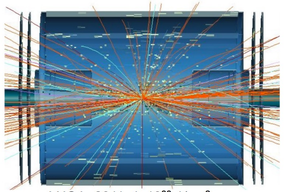
- **Run3 developments** include:
  - **mitigation strategy** for **detector ageing** and **pile-up handling**
  - **improvements** at tracking at **trigger level**
  - **improvements** for tracking **in dense environments**
  - **improvements** for **displaced tracking**
  - improvements in the vertex reconstruction

- the HL-LHC will provide unprecedented challenges in terms of track and vertex reconstruction
  → this open up a rich playground for future developments in both hardware and machine learning based tracking

challenge @ HL-LHC

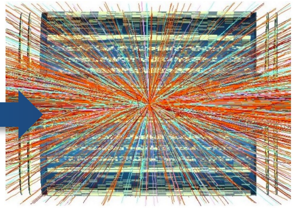
BACKUP

introduction

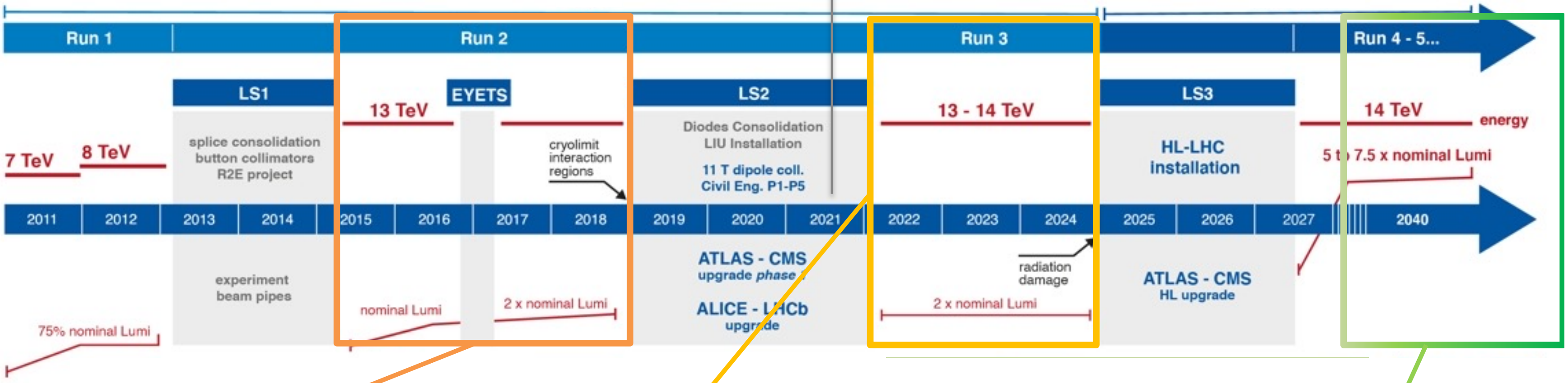


LHC in 2011: 1×10^{33} $1/cm^2s$

LHC



High Luminosity LHC



Run2 design
$\mathcal{L} = 1\times10^{34}\ Hz/cm^2$
for 150 fb^{-1}

Run3 design
$\mathcal{L} = 1.3 \div 2\times10^{34}\ Hz/cm^2$
for 300 fb^{-1}

nominal Run4 scenario
$\mathcal{L} = 5\times10^{34}\ Hz/cm^2,\ <PU> \approx 140$
for 3000 fb^{-1}

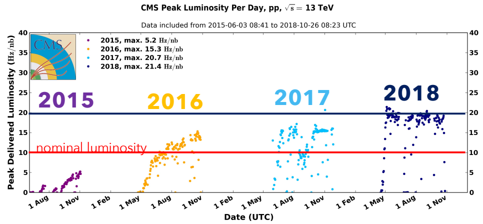
ultimate Run4 scenario
$\mathcal{L} = 7.5\times10^{34}\ Hz/cm^2,\ <PU> \approx 200$
for 4000 fb^{-1}

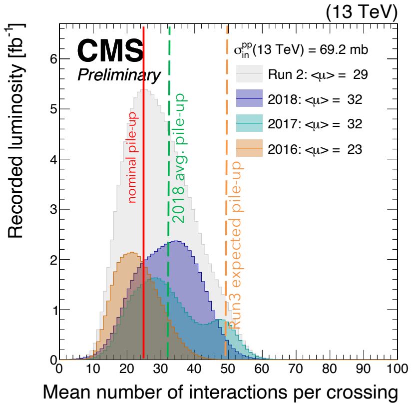
introduction : LHC environment

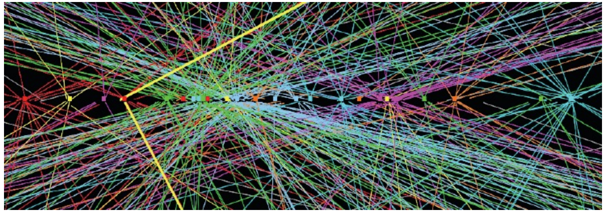
➢ **excellent performance** of **LHC at** $\sqrt{s} = 13\ TeV$
much **higher luminosity** and **Pile-Up** (PU)
(w.r.t. Run1 and nominal)

mean number of interactions $< \mu > := \frac{\mathcal{L} \cdot \sigma_{inel}}{N_{bunch} \cdot f_{LHC}}$

- reached $\mathcal{L} = 1.9 \times 10^{34}\ Hz/cm^2$
- peak PU $= 30 \div 60$







in a bunch crossing event $\sim \boldsymbol{O(50k)}$ **hits**
need to be processed, decoded, and combined into clusters,
and then combined into track seeds
that are subsequently attempted to be extended to identify the charged particles (tracks)
and precisely reconstruct their trajectories

this represents a **complex combinatorial problem**, which increases in difficulty w/ pile-up

the quality of the reconstructed track candidates becomes challenging to maintain under high pile-up

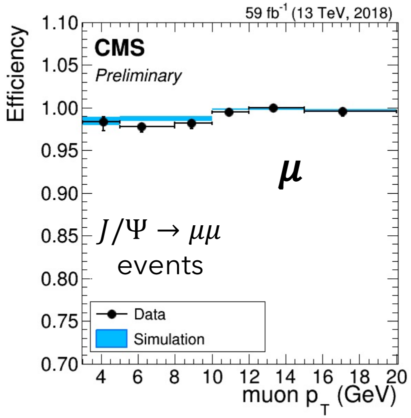
➢ high cluster density leads to incorrect cluster-to-track association,
pulling the reconstructed trajectories away from their true values

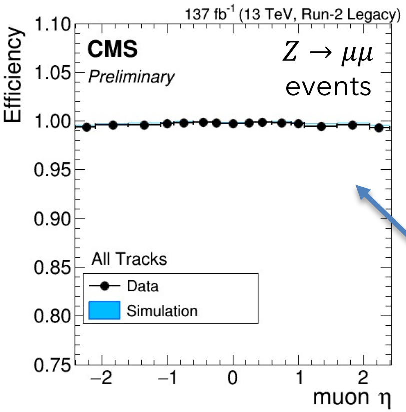
➢ w/ the increase of the available clusters, the random collections of clusters (fake tracks) increases

➢ the timing required for tracking scales rapidly w/ $< \mu >$

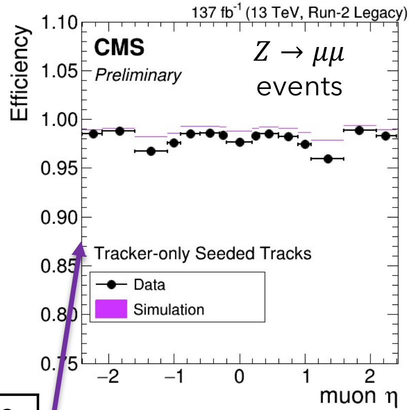
tracking performance : efficiency

tracking efficiency extracted from data
using tag-and- probe method from $Z \to \mu\mu$ and $J/\Psi \to \mu\mu$





	Run2
tracker + muon seeded	**99.0%**
tracker-only seeded	**97.5%**



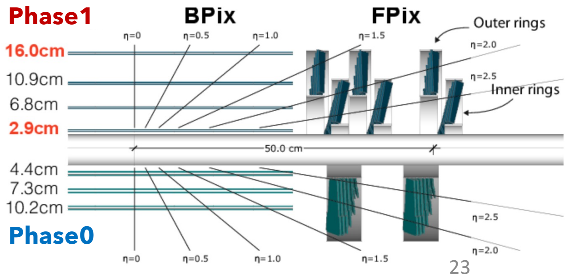
π, K, ..

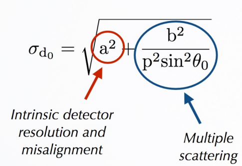
primary vertex and track impact parameter resolution

- thanks to the **Phase1 pixel detector** installed in 2017
  - the vertex reconstruction shows better performance than 2016 (Phase0) one
  - the track impact parameter on the transverse plane has resolution for tracks with $p_T = 1 \div 10\ GeV$

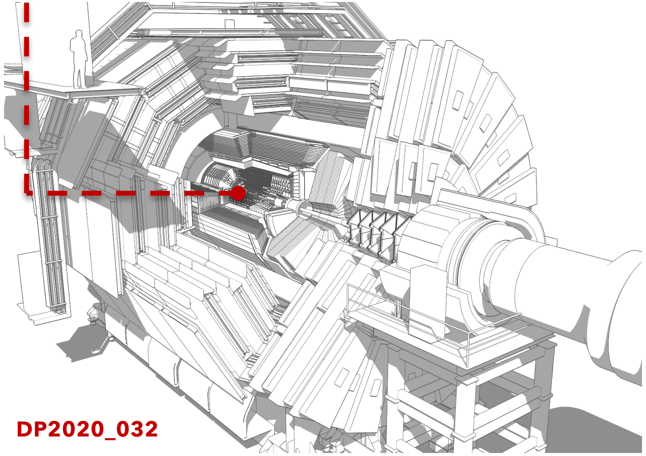
    $\boldsymbol{\sigma_{d_{xy}} = 20 \div 75\ \mu m}$, for |η| < 3.0

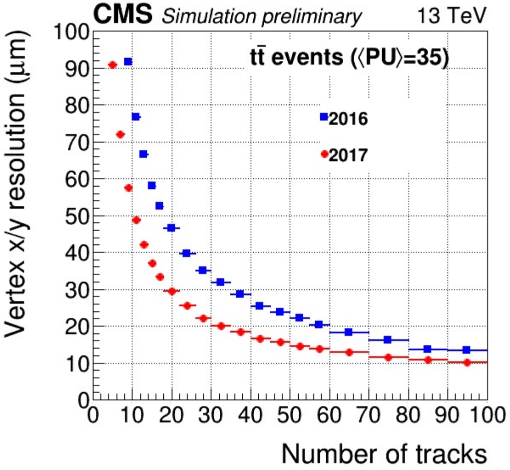
    $\sigma_{d_{xy}} = 20 \div 65\ \mu m$, for |η| < 1.4 [$25 \div 90\ \mu m$ for Phase-0 Pixel detector]

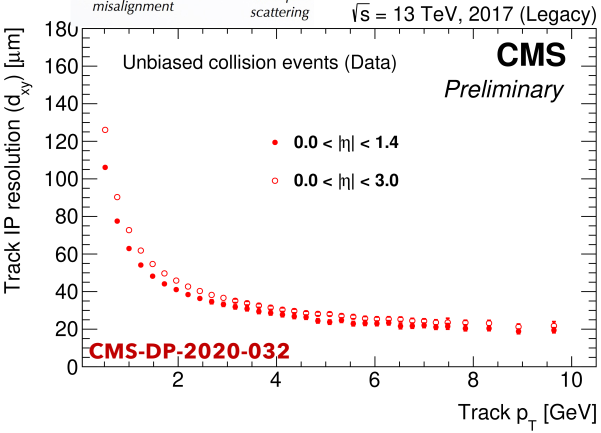




$a \approx 20\ \mu m$ @10 GeV







DP2020_032

track reconstruction in CMS

in each iteration, tracks are reconstructed in four steps:

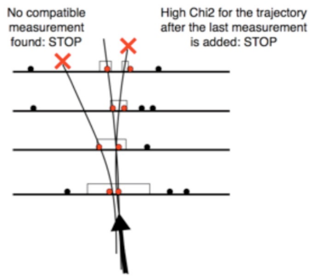
1. seeding: use combination of pixel, strip or mixed hits

provides track candidates, w/ an initial estimate of the trajectory parameters and their uncertainties

2. pattern recognition: alignment uncertainty taken into account

hit compatible w/ predicted track position are added (Kalman update)

to the trajectory track parameters are updated



3. final fit: taking into account the Field non uniformity and a detailed description of the material budget

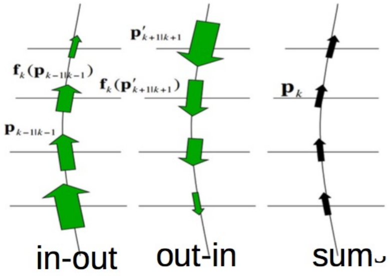
provides the best estimate of the parameters of each smooth trajectory after combining all associated hits [outlier hits are rejected]

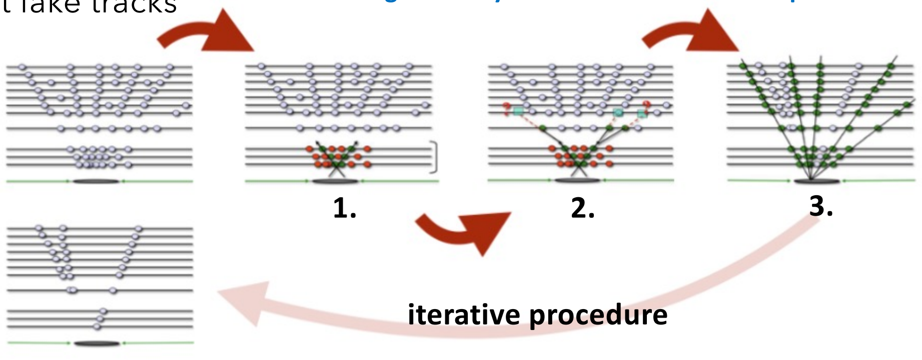
4. selection:

sets quality flags based on the fit χ^2 and the track compatibility w/ interaction region

aims to reject fake tracks

tracks sharing too many hits are also cleaned as duplicates

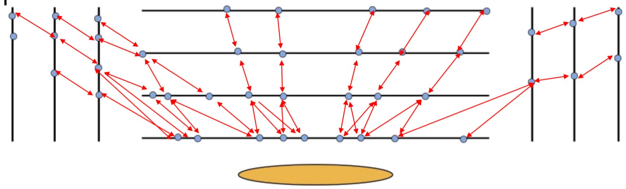




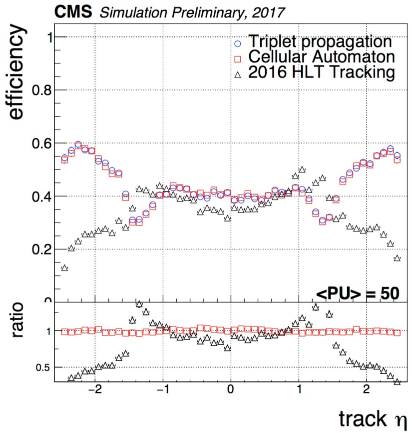
cellular automaton

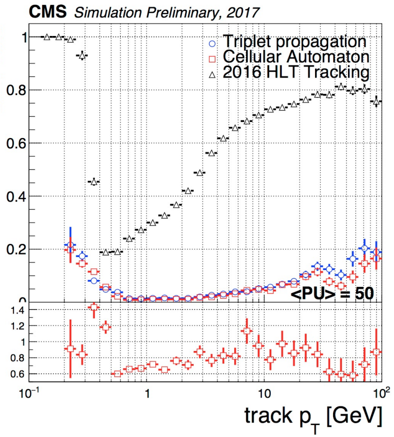
new track seeding algorithm based on **Cellular Automaton** (CA) technique

- it starts from a list of layers and their pairings
  - a graph of all possible connections between layers is created
  - doublets (cells) are created for each pair of layers [compatible with a region hypothesis]
- fast computation of the compatibility between 2 connected cells



Quadruplet Algorithm	Time per event	speedup wrt. 2016
GPU Cellular Automaton	(1.2 ± 0.9) ms	24.4×
CPU Cellular Automaton	(14.0 ± 6.2) ms	2.1×
Triplet Propagation	(72.1 ± 25.7) ms	0.4×
2016 Pixel Tracks	(29.3 ± 13.1) ms	1×





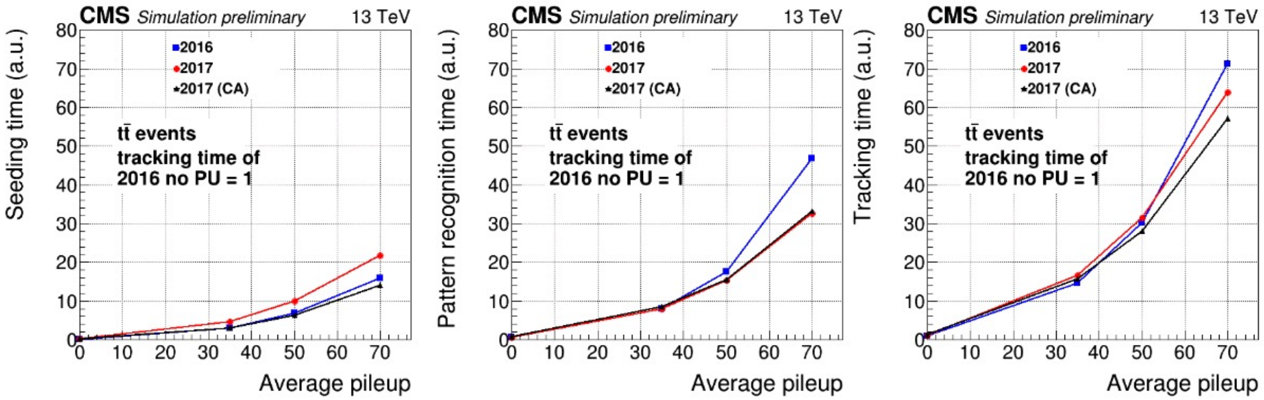
✓timing

- **x5 faster** than old algorithm
  x2 faster than 2016 configuration

✓performance
(with new pixel detector)

- almost same efficiency in the barrel
- **50% efficiency gain in the endcap**
- **x4 reduction in fakes** !

cellular automaton



cellular automaton seeding

- more robust
- smaller complexity vs PU than 2016 track seeding
  despite the increased number of layer combinations involved
- in pattern recognition, no additional gain

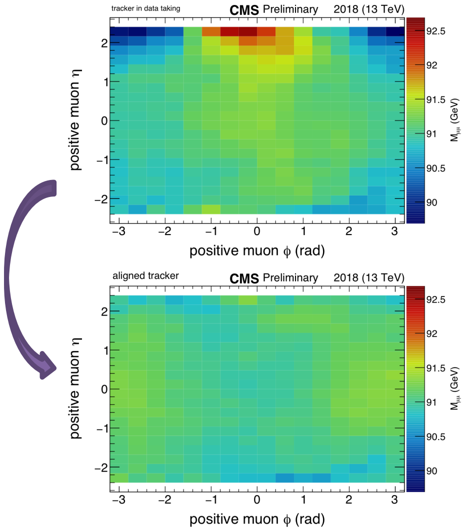
despite the increase in the number of pixel layers
~20% faster track reconstruction wrt to 2016 tracking @ <PU> = 70

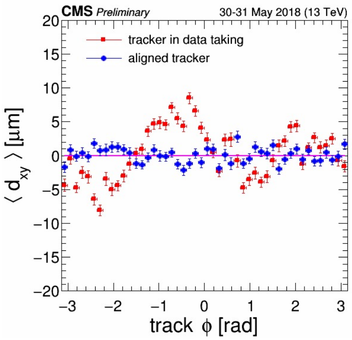
tracker aligment

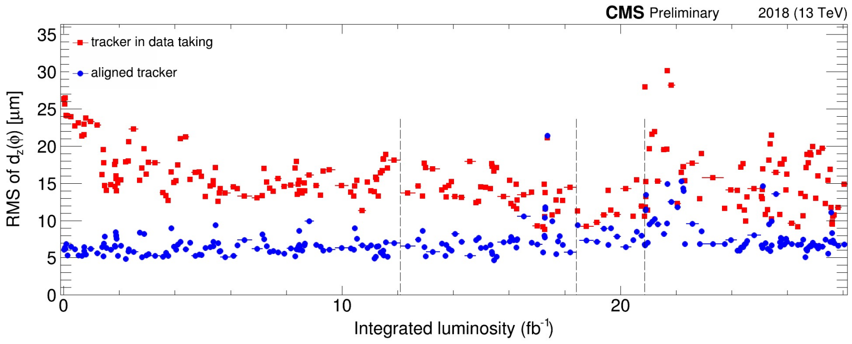
- **precise** knowledge of the **detector alignment** crucial for track reconstruction
- detector components have been observed to move
  e.g. tension of support structure due to thermal transients or magnetic field changes
- CMS employs an **automatic procedure**
  to monitor movements of top level hierarchical mechanical structures
  (half-barrels and half-disks)
  when appropriate, detector geometry is updated based on these online results

➢ ϕ-dependent **mass bias in $m_{\mu\mu}$ @high muon rapidity**
  → greatly reduced w/ the update of the alignment

➢ track impact parameters (d_{xy} and d_z) are sensitive to
  - Lorentz Angle mis-calibrations
  - misalignment in the pixel detector
  → residual bias is nicely recovered by the time granularity alignment

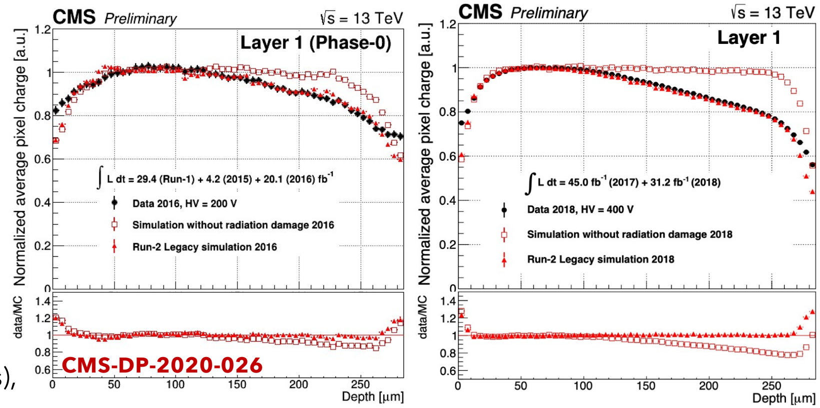


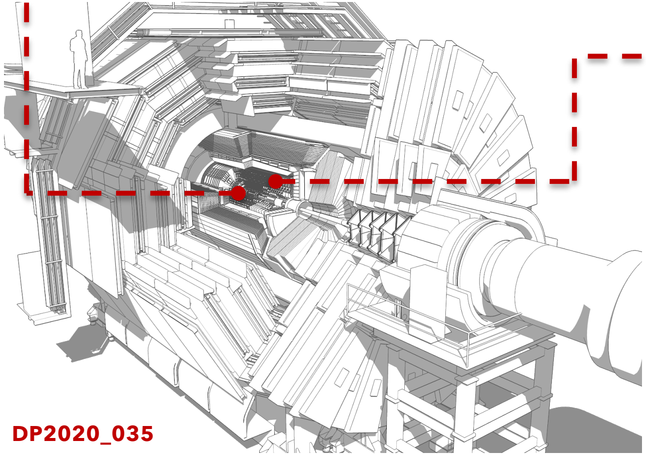




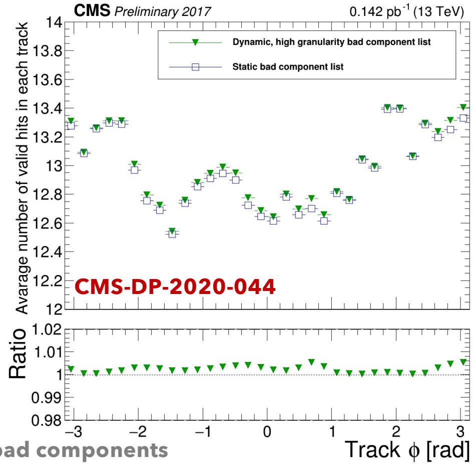
tracker calibrations

pixel:
- data:
  - conditions updates,
    automatic bad component determination procedure
    including improved Cluster Position Estimator (CPE)
- MC:
  - including bad components on FED channel basis (stuck TBMs),
    ultimate performance of MC,
    including radiation damage, dynamic inefficiencies





strips
- MC:
  - include dynamic hit inefficiency
- MC/Data:
  - updated gains for the 1st period in 2016,
    affected by APV saturation



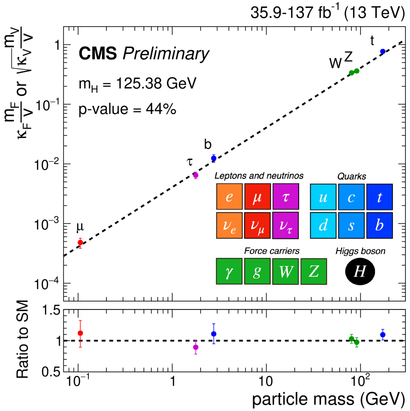
having a dynamically updated list of bad components
the missing hits can be recategorized as inactive hits
⇒ further propagation of the previously stopped tracks
due the maximum allowed number of missing hits

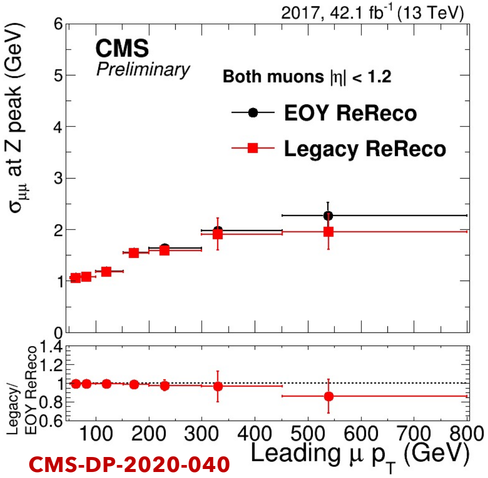
DP2020_035

muon performance

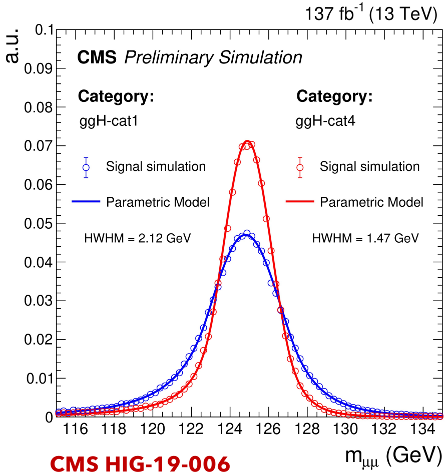
- muon reconstruction combines information from the tracker and muon systems
- different algorithms for the reconstruction, identification and isolation are implied
  - ✓ **efficiencies** are generally **high**, with small p_T dependence in all the probed momentum range and across all rapidity ranges
  - ✓ consistent measurements between the different standard candles
  - ✓ up to rather large $|\eta|$, no significant scale biases are observed
  - ✓ **excellent dimuon mass resolution** ⇒ **first evidence of H coupling to muons** 3.0σ obs (2.5σ expected)

    (also thanks to the very high magnetic field, and the tracking system performance)





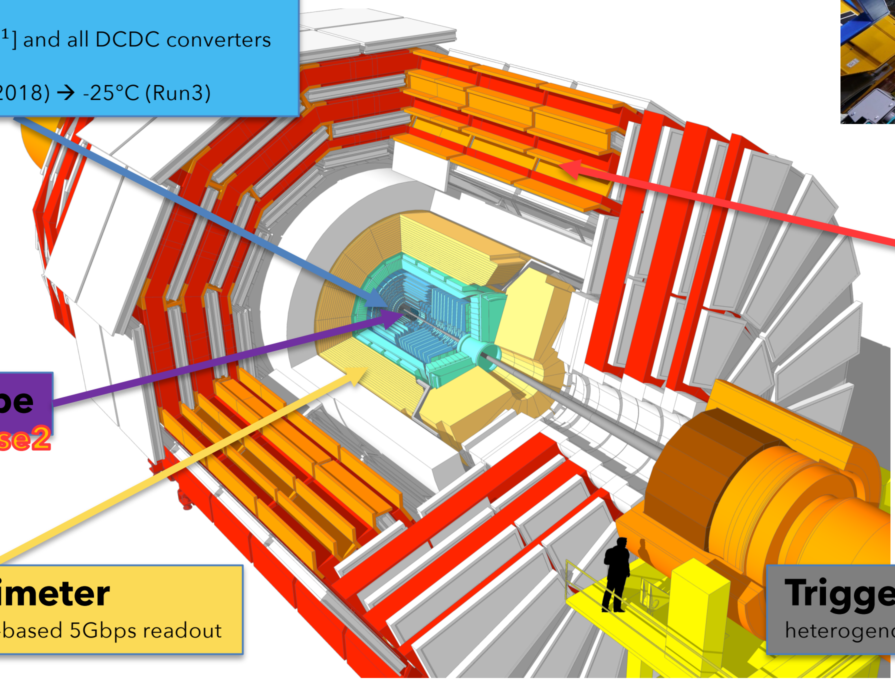
CMS-DP-2020-040



CMS HIG-19-006

preparing for Run3

"Phase-1" upgrades of CMS majority of the upgrades have been done in the past years



Silicon Tracker
- Pixel
  replace layer1 [250 fb^{-1}] and all DCDC converters
- Microstrips
  running colder -20°C (2018) → -25°C (Run3)

Muon Detectors
shielding against neutron background
- Drift tubes
  upgrade front-end electronics
- Resistive Plate Chambers
  leak repair
- Cathode strip chambers
  upgrade front-end electronics
- GEM
  installed GE1/1 chambers

for phase2

new beampipe
for phase2

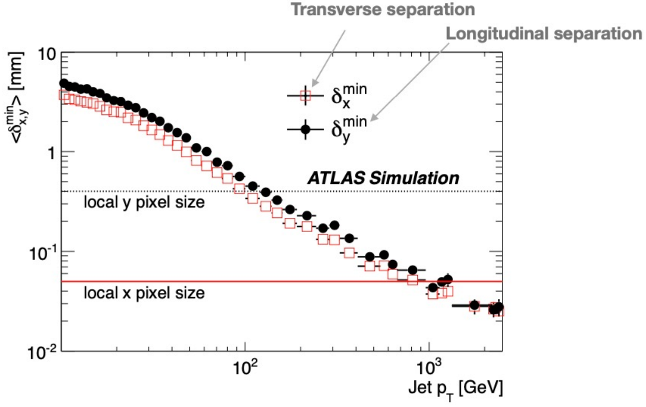
Hadron Calorimeter
install new SiPM+QIE11-based 5Gbps readout

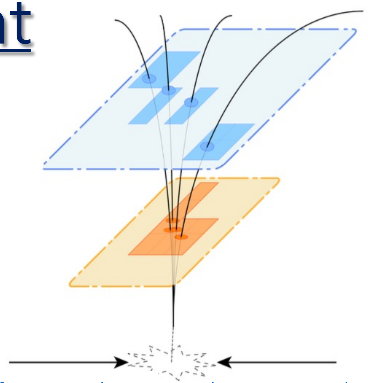
Trigger System
heterogenous HLT farm (CPU+GPU)

tracking in dense environment

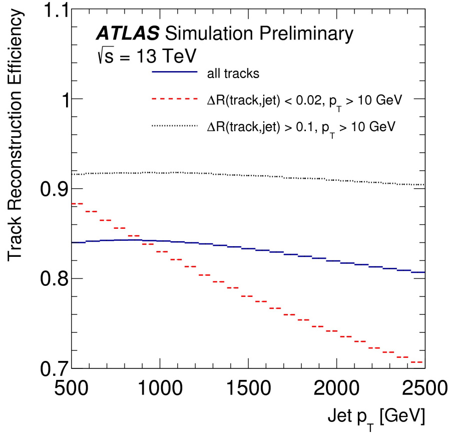
algorithms like the Particle Flow, the b-tagging and the jet sub-structure rely on the good performance of the track reconstruction w/in jets

- average separation of tracks inside high-p_T jet cores can be **smaller than sensitive elements size**





hits from different tracks can result in a merged pixel cluster in the core of high-p_T jets
[the effect is even more pronounced for b-jets, because the B decay happens closer to the pixel detector]



⇒ increase of shared clusters between track candidates
⇒ **lower track reconstruction efficiency** w/ standard algorithms

track reconstruction at HLT

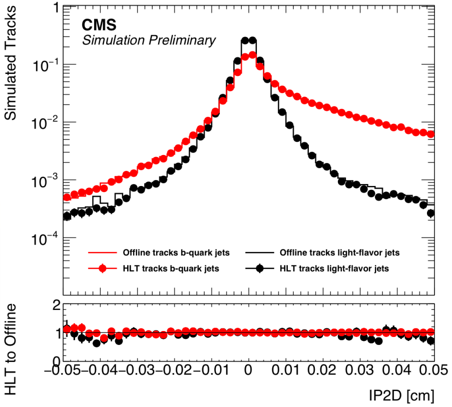
in Run2, we decide to keep the Tracking@**HLT as close as possible** to the **offline** track reconstruction on both the algorithms & configurations but **time** is a constraint

⇒ **reduce #iterations** wrt offline tracking

⇒ **constrain** tracking regions (i.e., **regional** tracking)

CA seeding since 2017

N	Step Name	Seeding	Target Track
0	Iter0	pixel tracks (from **quadruplets**)	prompt, high p_T
1	Iter1	pixel quadruplets	prompt, low p_T
2	Iter2	pixel triplets	recovery (not only high p_T)
3	Triplet recovery	pixel triplets in η-ϕ region	static triplet recovery
4	Doublet recovery	Pixel doublets in η-ϕ region	static doublet recovery

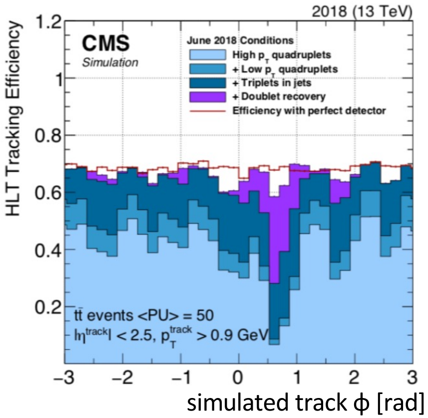


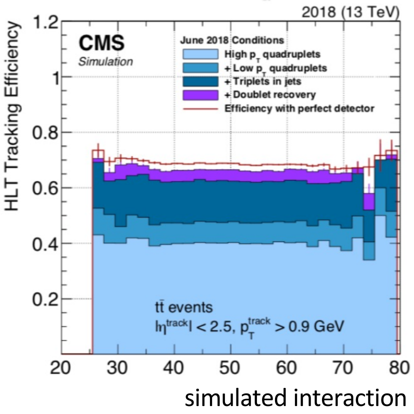
during Phase1 pixel commissioning,
failures observed mostly geometrically contained

⇒ in 2017, adopted a **Static mitigation** via dedicated iterations in specific η-φ regions

however, recovery is insufficient for additional (dynamic) pixel issues [like the DC/DC converter issue]

⇒ in 2018, adopted the **Dynamic mitigation** of pixel issues [trade off among efficiency/fake and timing]





nearly ideal efficiency is achieved

efficiency is almost flat as a function of #PU