Report on HLT Phase-2 performance

Task 3.1.1

Luca Ferragina, Marco Musich, Marco Rovere, Jan Schulz, Davide Valsecchi

Table of Contents

Introduction

Disclaimer

The results presented in this talk reflect the combined efforts of various contributors:

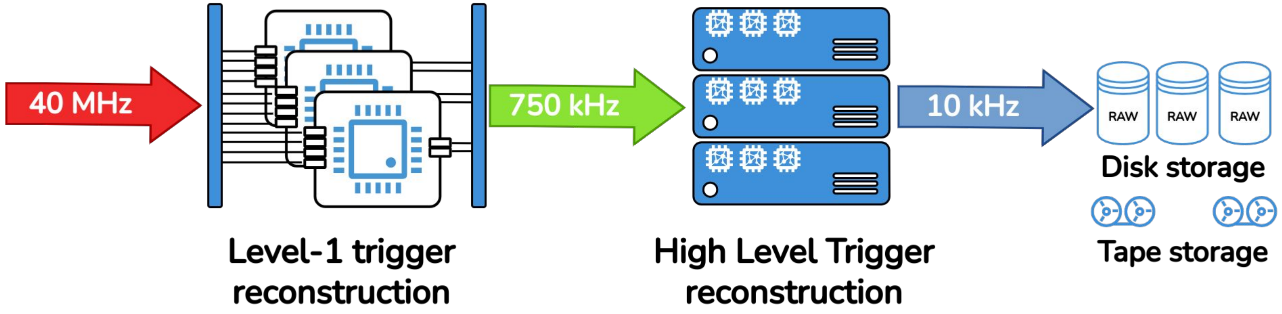
- **Next Generation Trigger** (**NGT**) Team: Dedicated development and analysis.
- **CMS colleagues** : Foundational tools, samples and implementations.
- **Trigger Study Group** (**TSG**): Performance profiles and supporting studies.

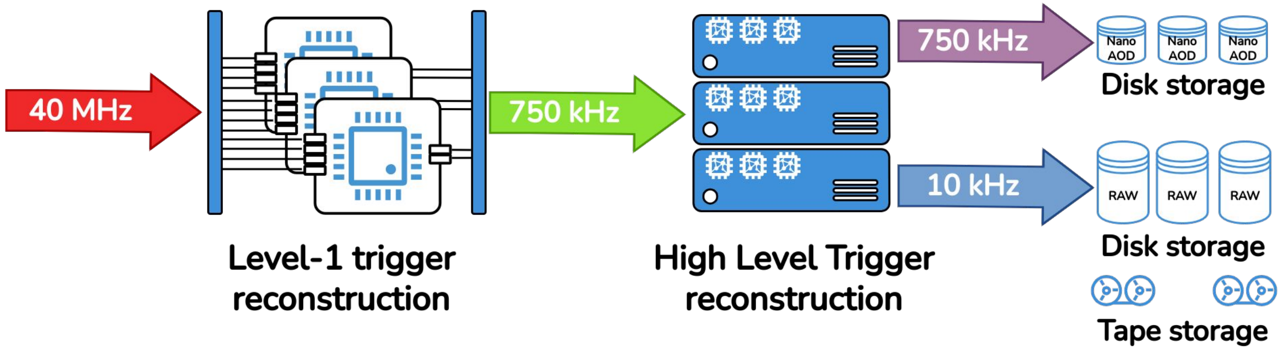
We extend our gratitude to all collaborators and to CMS as a whole for their significant contributions and support.

Task 3.1.1: R³ Faster Reconstruction

- The successful **Patatrack experience** in CMS has shown that it is possible to **improve the physics quality and reconstruction throughput of selected physics objects (pixel tracks) by leveraging heterogeneous architectures**
- This required ~4 years of development to:
  - Study the performance of the current algorithm and identify bottlenecks
  - **Rethink the algorithms** and **data structures** targeting heterogeneous architectures
  - Develop, integrate and validate the results in CMSSW
  - Propagate the new objects to the rest of the reconstruction
- The R³ project will use a similar approach to redesign the most important physics objects:
  - Muons
  - Electrons and photons
  - Taus
  - Jets, MET and Particle Flow Global Event interpretation
- Perform offline-like full event reconstruction,

  in addition to the traditional event selection





Reminder and goals

- 2024 contractual milestone for Task 3.1.1 is to write a report on the performance of online reconstruction.
  - identify bottlenecks,
  - propose targeted improvement solutions,
  - outline the necessary features for the generic CMS Structure of Arrays (SoA) (See Felice's talk).
- **Our main goal is to understand the missing factor that would be needed in order to reach the ambitious goal of this task**.

The "Missing Factor" for Reaching NGT Goals

- **Context**: CMS has a defined budget for acquiring hardware in Run-4 and Run-5. This budget directly impacts the amount of computational resources we can allocate for the High-Level Trigger (HLT) and reconstruction tasks. Our goal is to determine the missing factor — the performance improvements needed in the HLT reconstruction process to meet the ambitious goals of the Next Generation Trigger (NGT).
- **Fixed-Budget Model**: Given that CMS's budget is constrained, the amount of hardware available is fixed. This means *we must focus on improving the efficiency of the HLT to meet the ambitious NGT goals*. The missing factor refers to how much we need to speed up the HLT reconstruction to compensate for any hardware limitations and still meet the desired performance targets.
- **Extrapolation**: the process of deriving the “Missing Factor”, in different conditions/scenarios.

Key performance Metrics

- Current HLT Phase-2 Performance
  - Measure the current performance of the simplified HLT Phase-2 menu for the High Luminosity LHC (HL-LHC).
- Offline Phase-2 Reconstruction
  - Measure the current performance of the offline Phase-2 reconstruction system.
  - Account for ongoing developments in the offline reconstruction, including improvements from Run-3.
- Run-3 Performance & Future Projections
  - Identify missing algorithms in the Phase-2 sequence that could impact performance.
  - Estimate their relative CPU impact in Run-3, extrapolate to 200 pile-up (PU) conditions, and integrate these estimates into the overall performance analysis, where this makes sense.

From TDR…

Results and Recommendations from the TDR

In the DAQ and HLT TDR we measured performances (throughput/timing) using L1-accept and TTbar events at both 140 and 200 PileUp scenarios

- Extrapolation based on L1-accept measurements.
  - +20% to account for missing Tau paths in the menu.
  - +50% to account for the “simplified” L1 and HLT menu.
- Assume 500 kHz input rate for Run-4 (2028), 750 kHz for Run-5 (2032).
- Assume 50% code runs on GPU by Run-4, 80% on GPU by Run-5.
- Assume flat +20% improvements in performance/CHF for both CPU and GPU.
- Performance/CHF for CPU measured using HS06.
- Performance/CHF for GPU measured using Pixel reconstruction code on NVIDIA T4.

Results and Recommendations from the TDR

Scenario	PU	Year Start	Throughput	Gain/year	Missing Factor CPU-Only	Fraction On GPU	Missing Factor w/ GPUs
Run 4	140	2028	32.2 ev/s	+20%	2.5×	50%	1.6×
Run 5	200	2032	13.4 ev/s	+20%	5.0×	80%	2.5×

- Results summarized in the table above
- All corrections accounted for
- Challenging goals for CPU-only scenario
- Achievable with the help of heterogeneous computing.

… into the future …

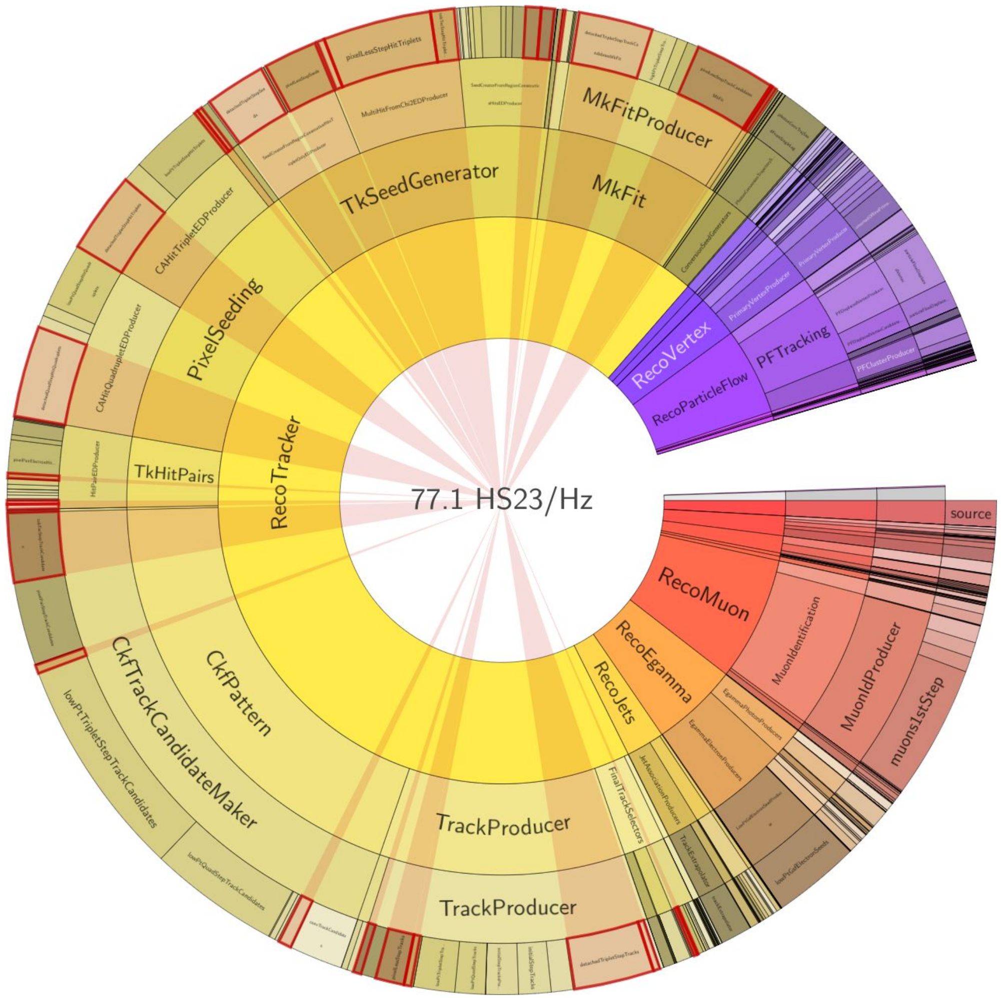
Measurements’ Setup

- For evaluating HLT Phase-2 Menu performances:
  - **TTbar sample simulated with 200 pileup (PU) interactions**
  - **L1-accept skim from Minimum Bias sample, 140PU**
  - **L1-accept skim from Minimum Bias sample, 200PU**
- **Machine used**:
  - AMD EPYC “**Bergamo**” 9754
    - HS23: 7450.248 (more info here)
    - Cores: 2×128×2 (number of sockets × physics cores × logical cores)
  - Offline Phase-2 Configuration
    - 1 socket only, 4 jobs, 64 threads, 64 streams, mainly due to memory constraints.
    - The throughput measured has been scaled by a factor 2, as if the machine was fully occupied.
- **All numbers and measurements will be expressed in HS23**.
- **Google Sheet** with all measurements and extrapolation: link

Extrapolations for NGT

- NGT extrapolations based on two different scenarios:
  - **run Offline Phase-2 reconstruction at 500 kHz (Run-4) and 750 kHz (Run-5)**
  - **scale HLT Phase-2 Menu as if run w/o filters**
    - We do still apply, also in this case, the +50% to account for the “simplified” nature of the Menu.
    - Maybe “overly aggressive”, but it’s the more conservative assumption we can make.
- Configurations and assumptions:
  - **keep the same fraction of the code to run on GPU in Run-4 (50%) and Run-5 (80%)**
  - **use up-to-date HL-LHC Schedule**
  - **extrapolations based on L1-accept skim at 140 and 200PU** (TTbar numbers computes as well)
- **After conducting measurements and making necessary considerations, we decided not to apply any correction factor from Run-3 reconstruction.**

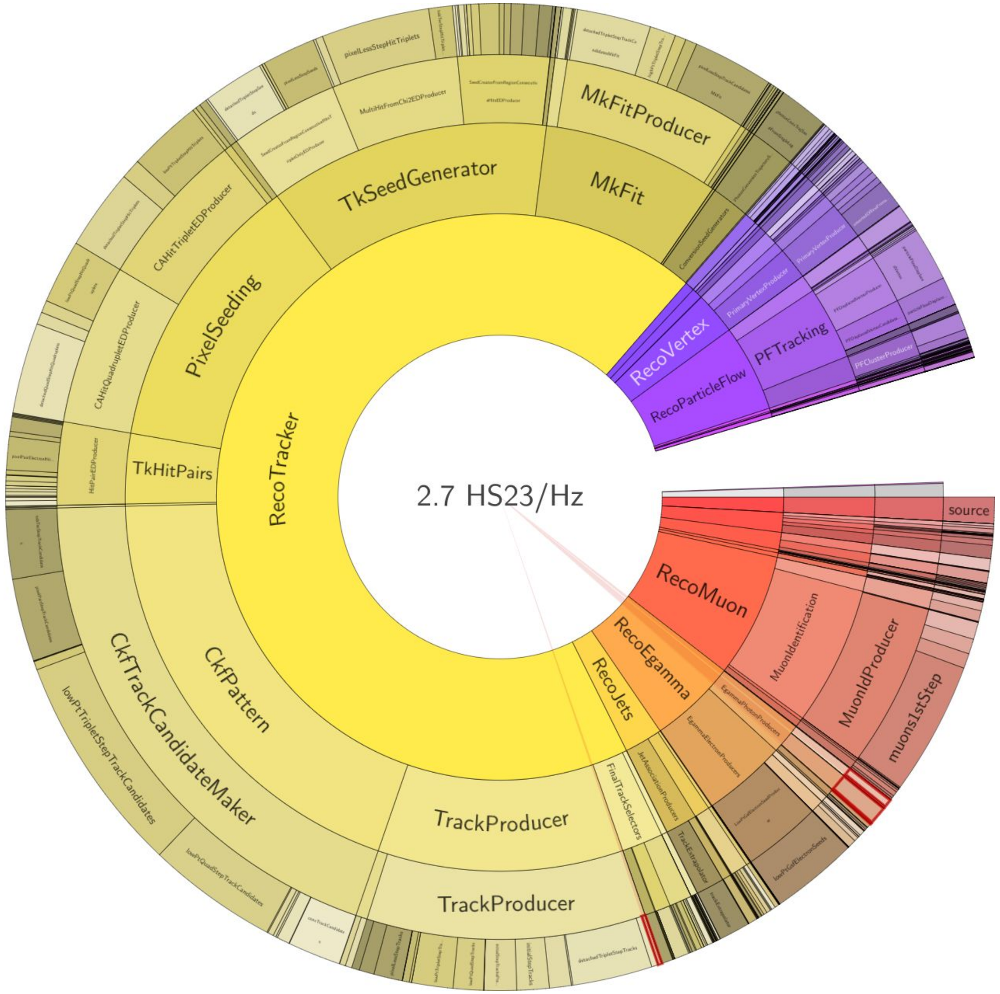
Offline Run-3, L1-accept (real data)



Missing from the Phase-2 reconstruction

- displaced tracks
    - ~ 30% time, efficient up to 50 cm
    - largely recovered **for free** by **new LST algorithm**

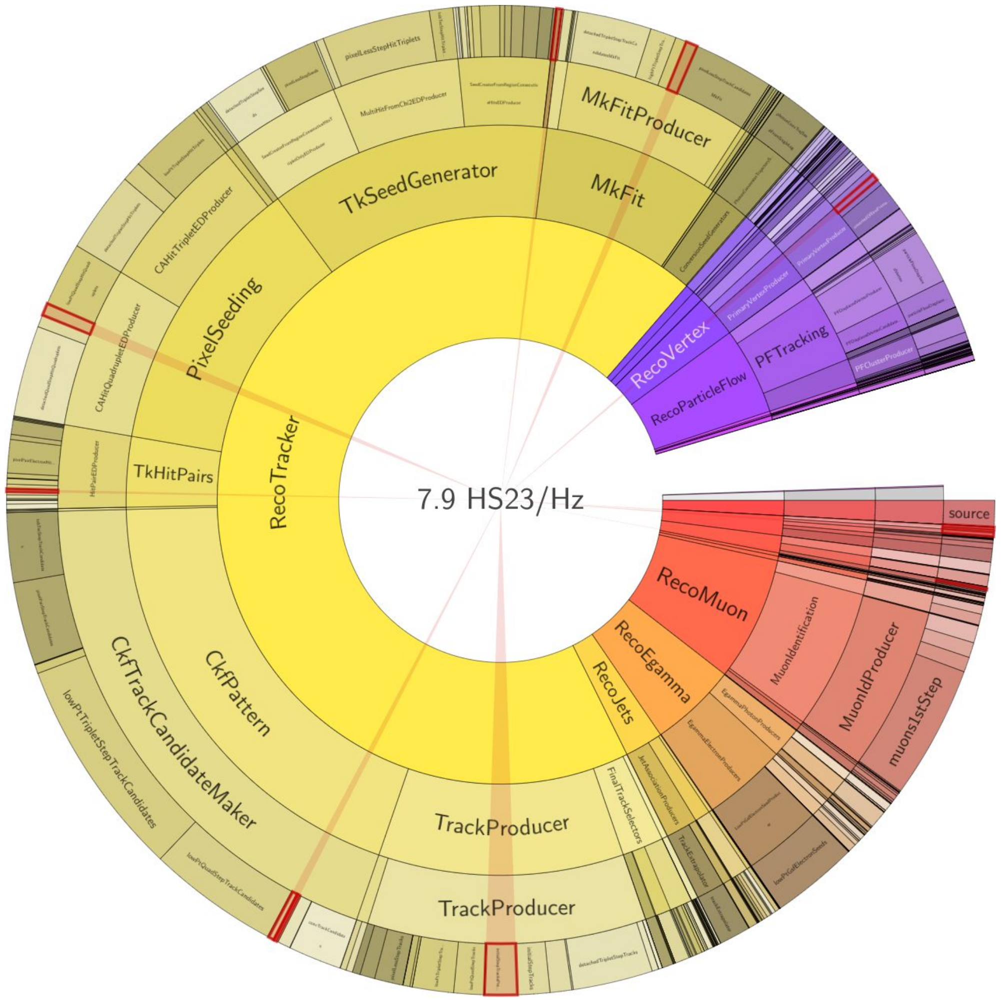
Offline Run-3, L1-accept (real data)



Missing from the Phase-2 reconstruction

- displaced tracks
  - ~ 30% time, efficient up to 50 cm
  - largely recovered **for free** by **new LST algorithm**
- conversions (*disabled*)
  - ~ 1% time

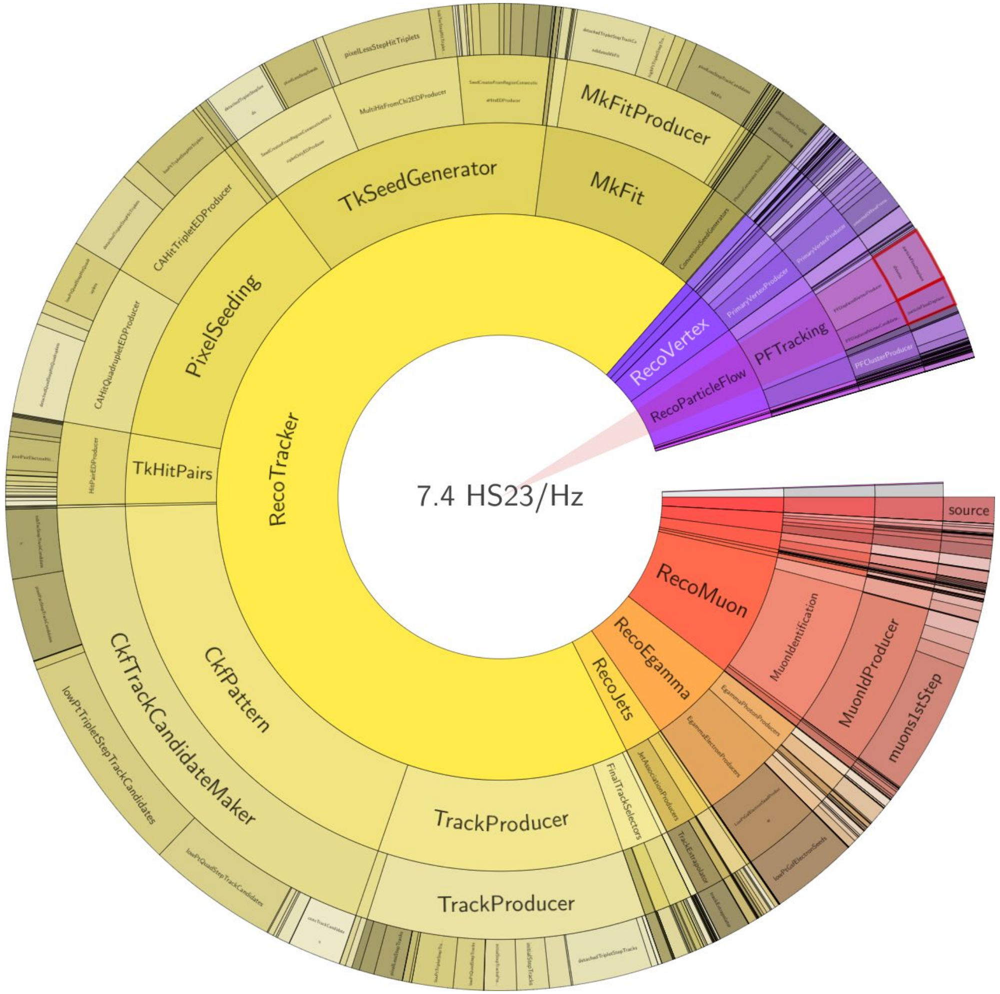
Offline Run-3, L1-accept (real data)



Missing from the Phase-2 reconstruction

- displaced tracks
  - ~ 30% time, efficient up to 50 cm
  - largely recovered **for free** by **new LST algorithm**
- conversions (*disabled*)
  - ~ 1% time
- "jet core" tracking
  - ~ 3% of time

Offline Run-3, L1-accept (real data)



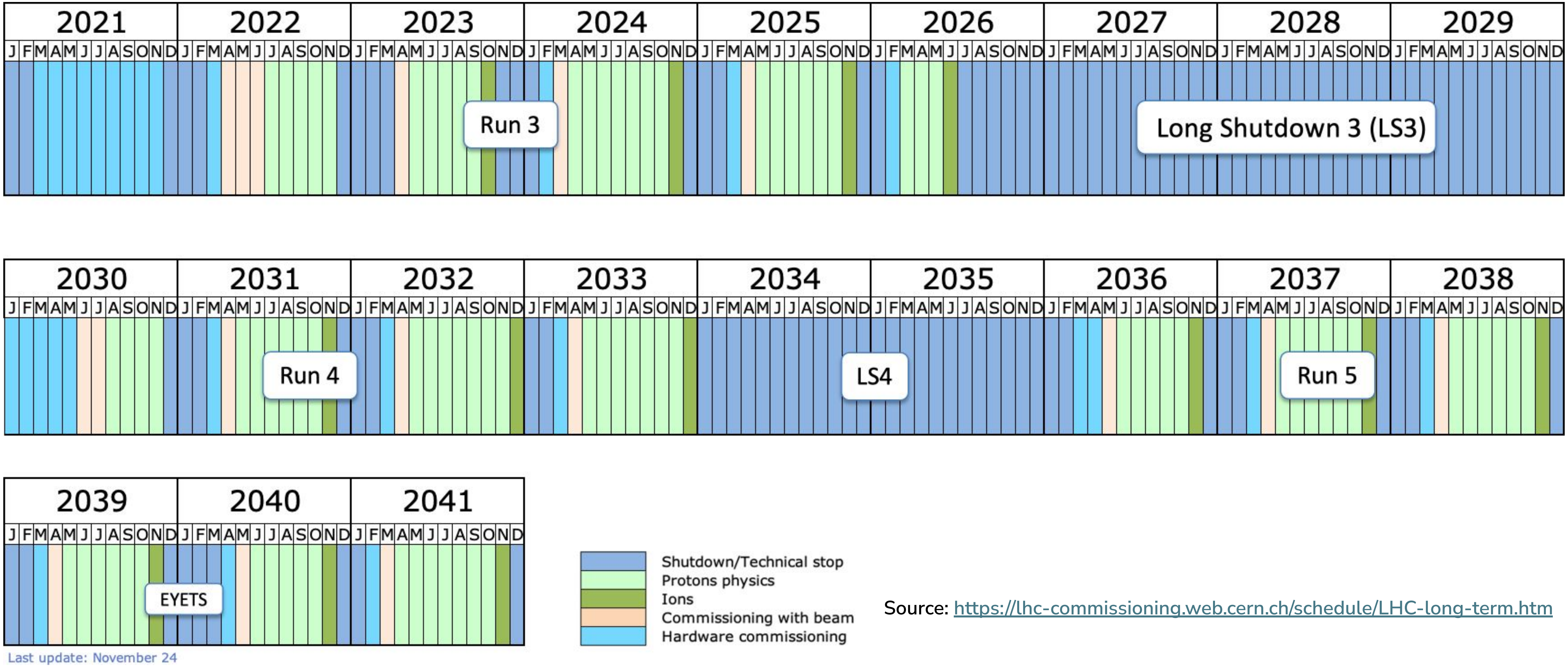
Missing from the Phase-2 reconstruction

- displaced tracks
  - ~ 30% time, efficient up to 50 cm
  - largely recovered **for free** by **new LST algorithm**
- conversions (*disabled*)
  - ~ 1% time
- "jet core" tracking
  - ~ 3% of time
- displaced particle flow interactions
  - ~2% time
- raw to digi step is negligible

overall **no Run-3 based corrections** to Phase-2 reconstruction time have been considered

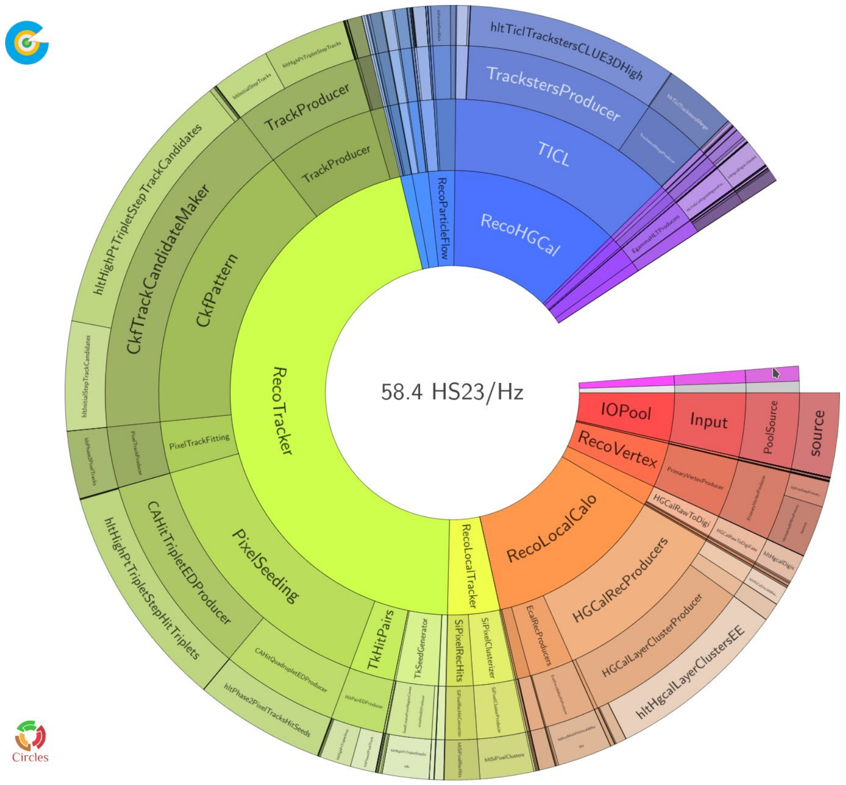
- negligible with respect to the other assumptions we make

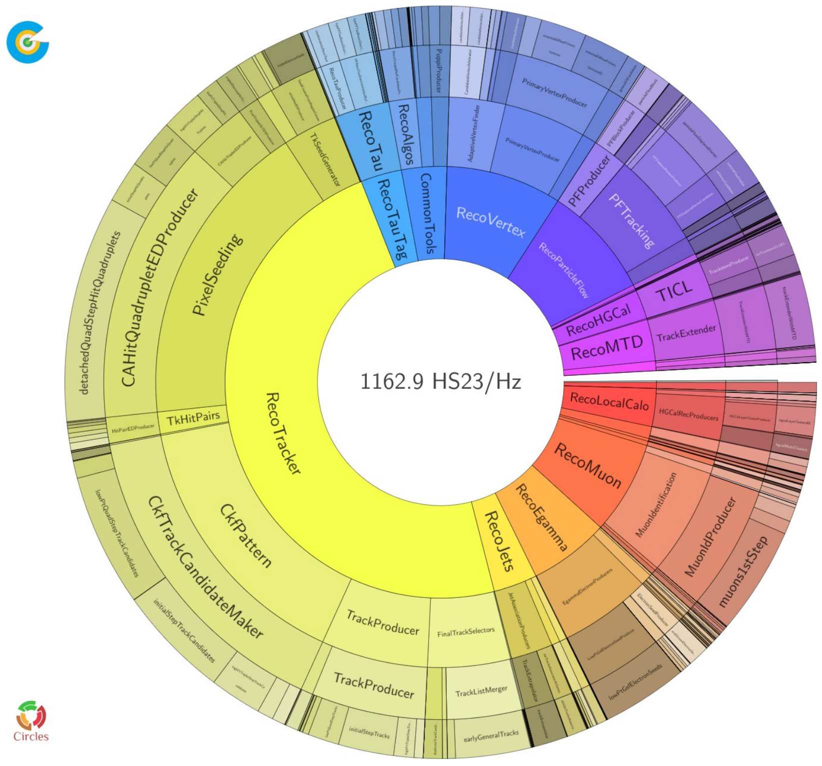
HL-LHC Schedule

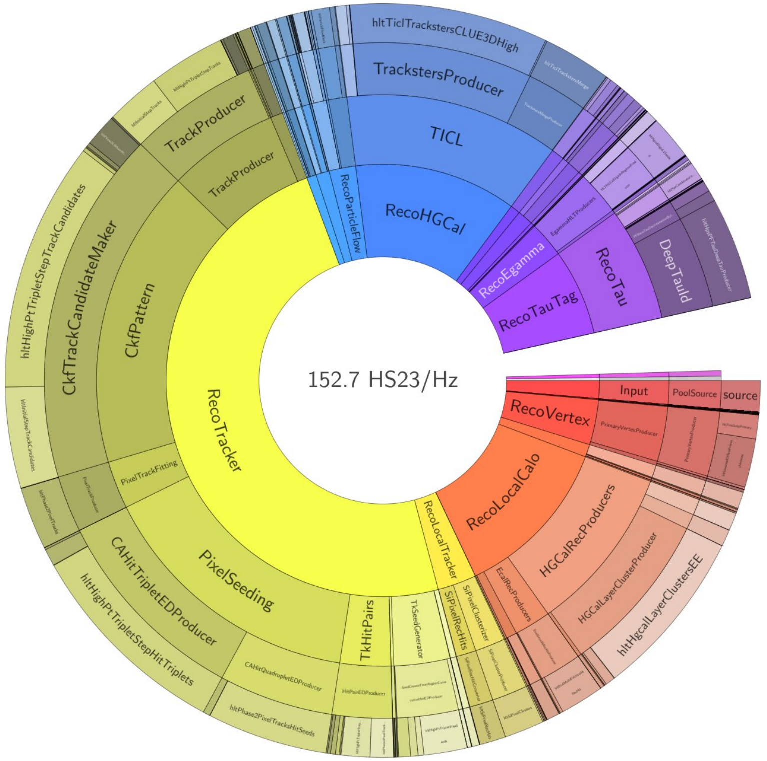


Source: https://lhc-commissioning.web.cern.ch/schedule/LHC-long-term.htm

Results used to derive NGTs extrapolations

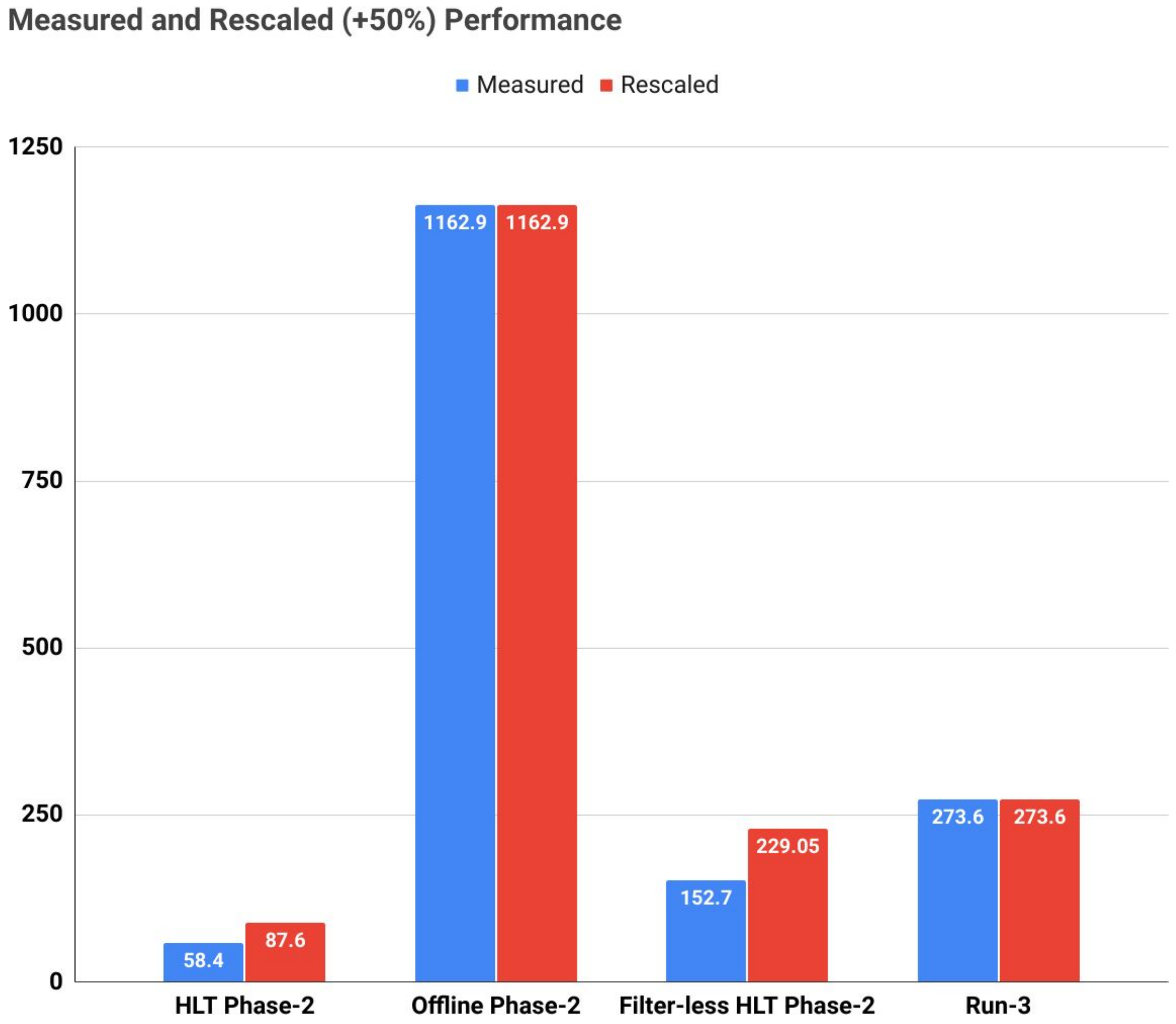






HLT Phase-2 | Offline Phase-2 | Filterless HLT Phase-2

Results used to derive NGTs extrapolations



Tables used to derive NGTs extrapolations

Module	HS23/Hz	Fraction
RecoTracker	26.8	45.9%
RecoLocalCalo	7.7	13.1%
RecoHGCal	7.3	12.5%
other	4.7	8.0%
RecoLocalTracker	2.2	3.8%
IOPool	2.1	3.5%
RecoVertex	1.9	3.3%
RecoParticleFlow	1.1	1.9%
EventFilter	0.9	1.5%
RecoEgamma	0.7	1.2%

HLT Phase-2

Module	HS23/Hz	Fraction
RecoTracker	558.1	48.0%
RecoVertex	100.8	8.7%
RecoParticleFlow	98.0	8.4%
RecoMuon	81.2	7.0%
RecoEgamma	69.8	6.0%
CommonTools	39.5	3.4%
RecoJets	38.1	3.3%
RecoMTD	36.4	3.1%
RecoTauTag	35.8	3.1%
RecoLocalCalo	33.1	2.8%

Offline Phase-2

Module	HS23/Hz	Fraction
RecoTracker	73.9	48.4%
RecoLocalCalo	18.7	12.2%
RecoHGCal	18.5	12.1%
RecoTauTag	9.8	6.4%
RecoVertex	5.1	3.4%
other	4.7	3.1%
RecoLocalTracker	4.4	2.9%
RecoEgamma	4.1	2.7%
RecoParticleFlow	3.0	2.0%
IOPool	2.1	1.3%

Filterless HLT Phase-2

NGT Extrapolations

NGT Extrapolations - Offline Reconstruction

Scenario	PU	Year Start	Throughput	Gain/year	Missing Factor CPU-Only	Fraction On GPU	Missing Factor w/ GPUs
Run 4	140	2030	12.0 ev/s	+20%	23.8×	50%	14.6×
Run 5	200	2036	6.2 ev/s	+20%	31.4×	80%	14.4×

- Results summarized in the table above
- Close to impossible goals for CPU-only scenario
- Extremely challenging even with heterogeneous computing.

NGT Tables w/ Phase 2 Offline reconstruction

Sample	TTbar		L1-accepted minimum-bias	
	Baseline	Ultimate	Baseline	Ultimate
	140	200	140	200
NGT throughput per node	11.2 ev/s	**6.8** ev/s	**12.0** ev/s	**6.2** ev/s
Throughput per kHS23	1.50 Hz/kHS23	0.91 Hz/kHS23	1.61 Hz/kHS23	0.83 Hz/kHS23
HLT input rate	500 kHz	750 kHz	500 kHz	750 k/Hz
Total processing power	332 MHS23	822 MHS23	310 MHS23	901 MHS23

	Run-4 (2030)		Run-5 (2036)	
Processing power needs				
current estimate (this Report)	310.4 HS23		901.2 (310.4 + 590.8) HS23	
w/ factor 14.6 (14.4) speedup	21.3 HS23		62.7 (21.3 + 41.4) HS23	
	Evolution Model			
	+20 %	+15 %	+20 %	+15 %
Price/performance ratio				
current value (2023)	2.61 CHF/HS23			
improvement factor until run start	3.0×	2.3×	9.0×	5.3×
ratio at run start, CPU only	0.88 CHF/HS23	1.13 CHF/HS23	0.35 CHF/HS23	0.56 CHF/HS23
with 50% offload to GPU	0.54 CHF/HS23	0.69 CHF/HS23	–	–
with 80% offload to GPU	–	–	0.08 CHF/HS23	0.13 CHF/HS23
HLT farm cost				
cost at run start, CPU only	18.6 MCHF	24.1 MCHF	12.1 MCHF	20.2 MCHF
with 50% offload to GPU	11.4 MCHF	14.7 MCHF	–	–
with 80% offload to GPU	–	–	4.6 MCHF	7.7 MCHF

NGT Extrapolations - Filter-less Phase2 HLT

Scenario	PU	Year Start	Throughput	Gain/year	Missing Factor CPU-Only	Fraction On GPU	Missing Factor w/ GPUs
Run 4	140	2030	47.6 ev/s	+20%	6.0×	50%	3.7×
Run 5	200	2036	32.5 ev/s	+20%	6.0×	80%	2.7×

- Results summarized in the table above
- Extremely challenging goals for CPU-only scenario
- Challenging, yet achievable with heterogeneous computing.

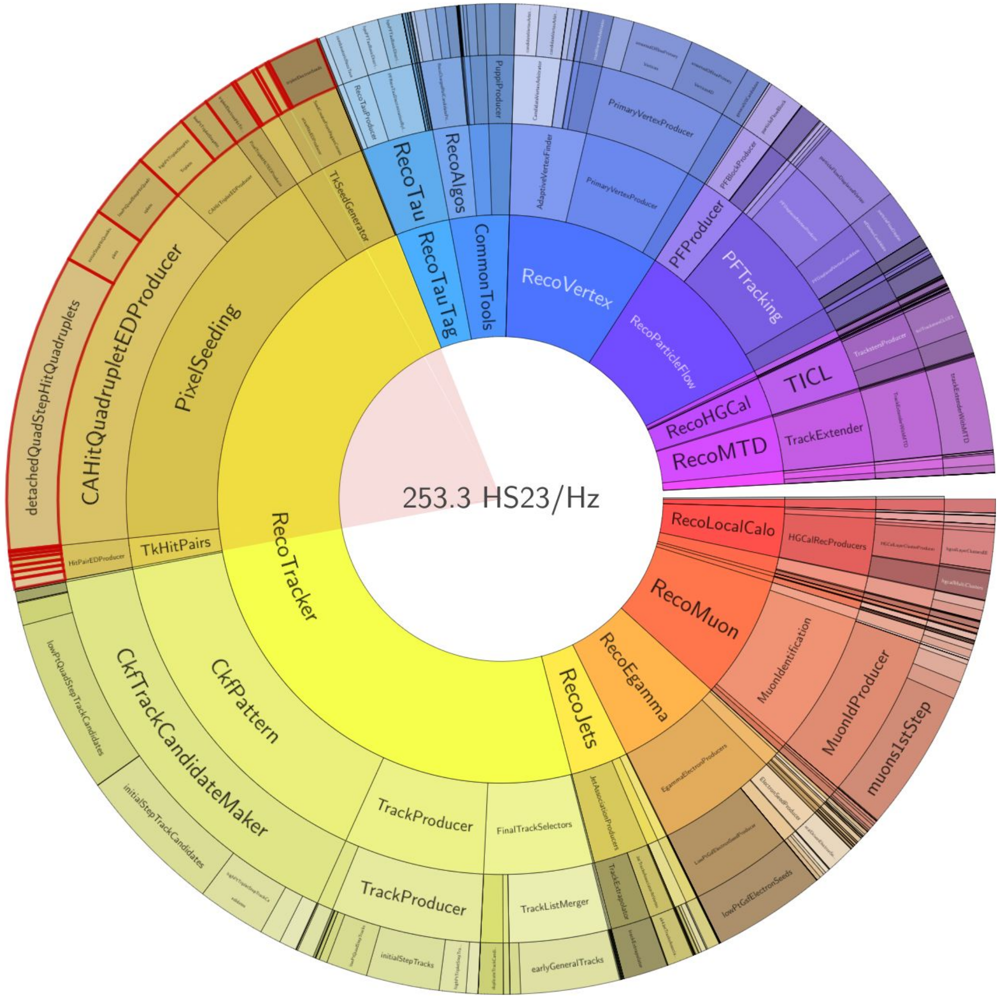
NGT Tables filter-less HLT reconstruction

Sample	TTbar		L1-accepted minimum-bias	
	Baseline 140	Ultimate 200	Baseline 140	Ultimate 200
NGT throughput per node	68.6 ev/s	**41.3** ev/s	**71.4** ev/s	**48.8** ev/s
+50 % contingency	45.8 ev/s	**27.8** ev/s	**47.6** ev/s	**32.5** ev/s
Throughput per kHS23	6.14 Hz/kHS23	3.73 Hz/kHS23	13.1 Hz/kHS23	8.96 Hz/kHS23
HLT input rate	500 kHz	750 kHz	500 kHz	750 kHz
Total processing power	81.4 MHS23	201.3 MHS23	78.3 MHS23	171.8 MHS23

	Run-4 (2030)		Run-5 (2036)	
Processing power needs				
current estimate (this Report)	78.3 HS23		171.8 (78.3 + 93.5) HS23	
w/ factor 3.7 (2.7) speedup	21.3 HS23		62.7 (21.3 + 41.4) HS23	
	Evolution Model			
	+20 %	+15 %	+20 %	+15 %
Price/performance ratio				
current value (2023)	2.61 CHF/HS23			
improvement factor until run start	3.0×	2.3×	9.0×	5.3×
ratio at run start, CPU only	0.88 CHF/HS23	1.13 CHF/HS23	0.35 CHF/HS23	0.56 CHF/HS23
with 50% offload to GPU	0.54 CHF/HS23	0.69 CHF/HS23	–	–
with 80% offload to GPU	–	–	0.08 CHF/HS23	0.13 CHF/HS23
HLT farm cost				
cost at run start, CPU only	18.6 MCHF	24.1 MCHF	12.1 MCHF	20.2 MCHF
with 50% offload to GPU	11.4 MCHF	14.7 MCHF	–	–
with 80% offload to GPU	–	–	4.6 MCHF	7.7 MCHF

Remarks/Future Developments

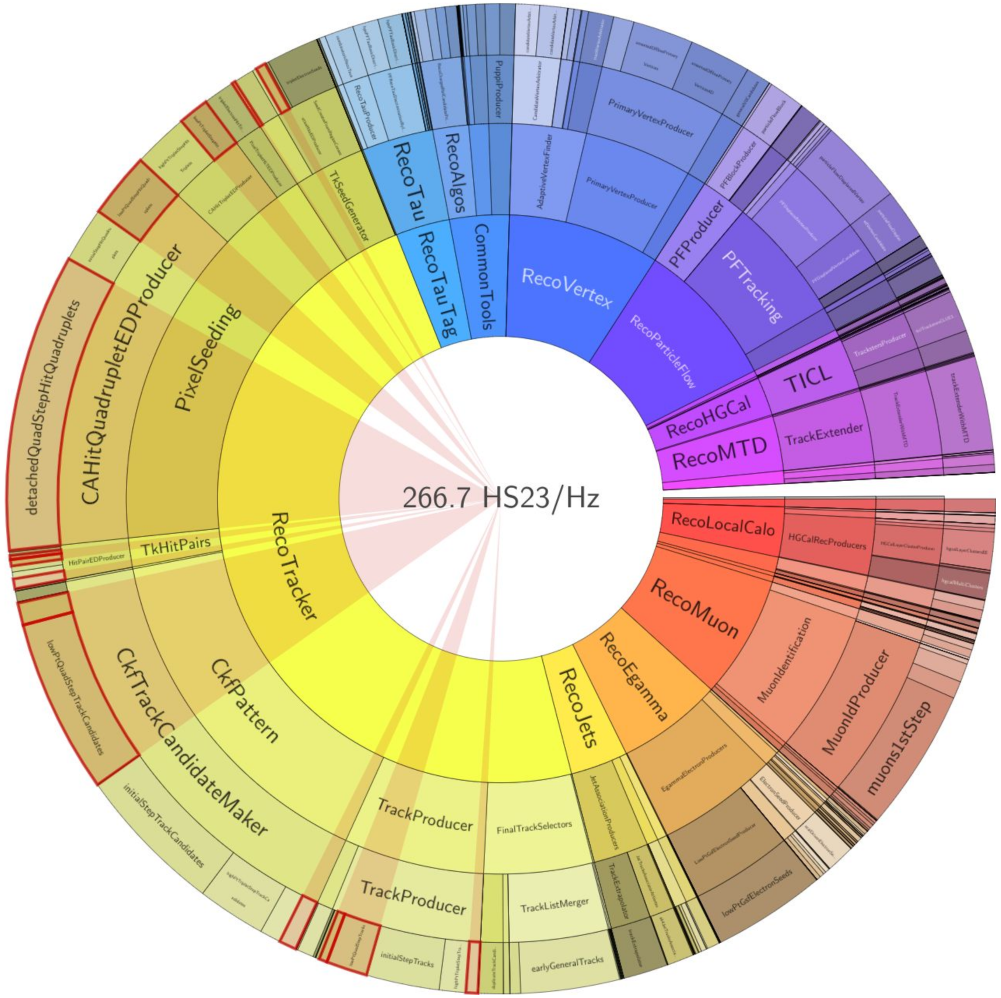
Offline Phase-2, L1-accept (simulation)



Generic Observations

- Pixel seeding still "full legacy"

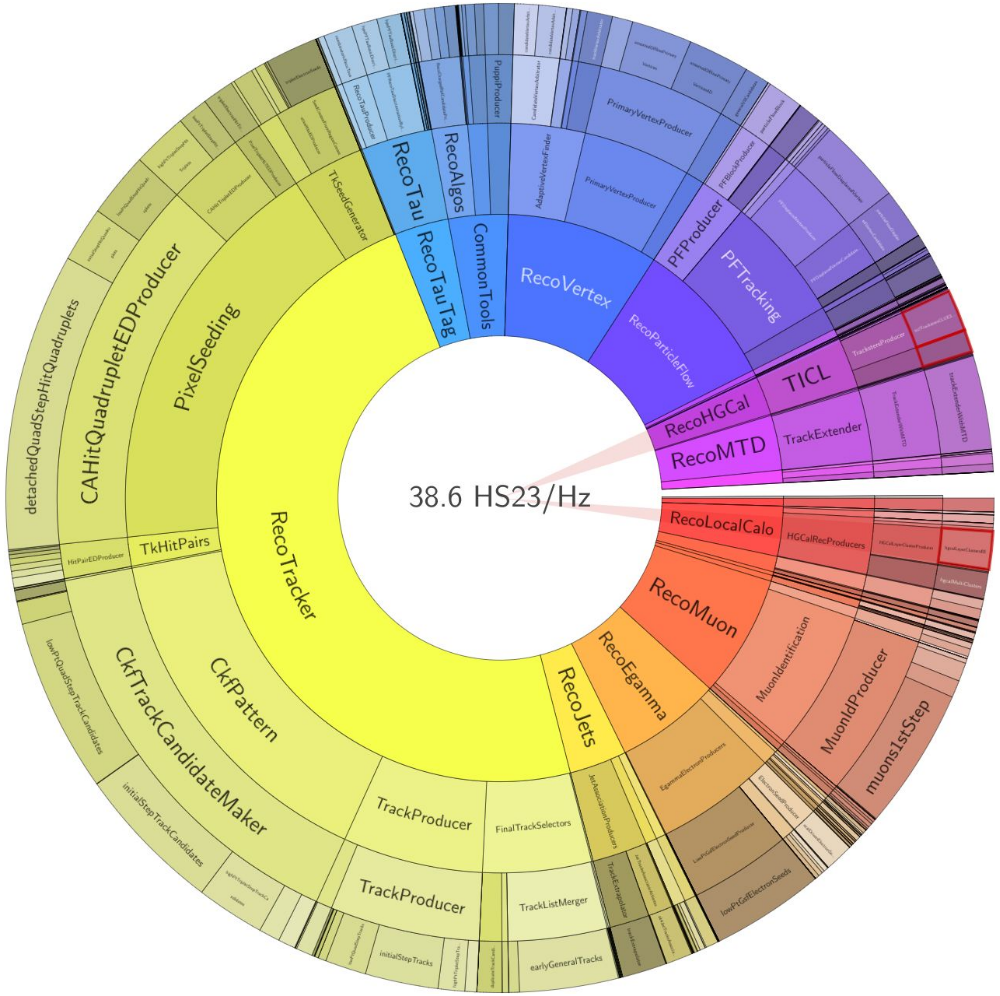
Offline Phase-2, L1-accept (simulation)



Generic Observations

- **Pixel seeding** still “full legacy”
- **Low p_T (300 MeV/c) & displaced (< 10 cm)** phase-space responsible for large fraction of the pie

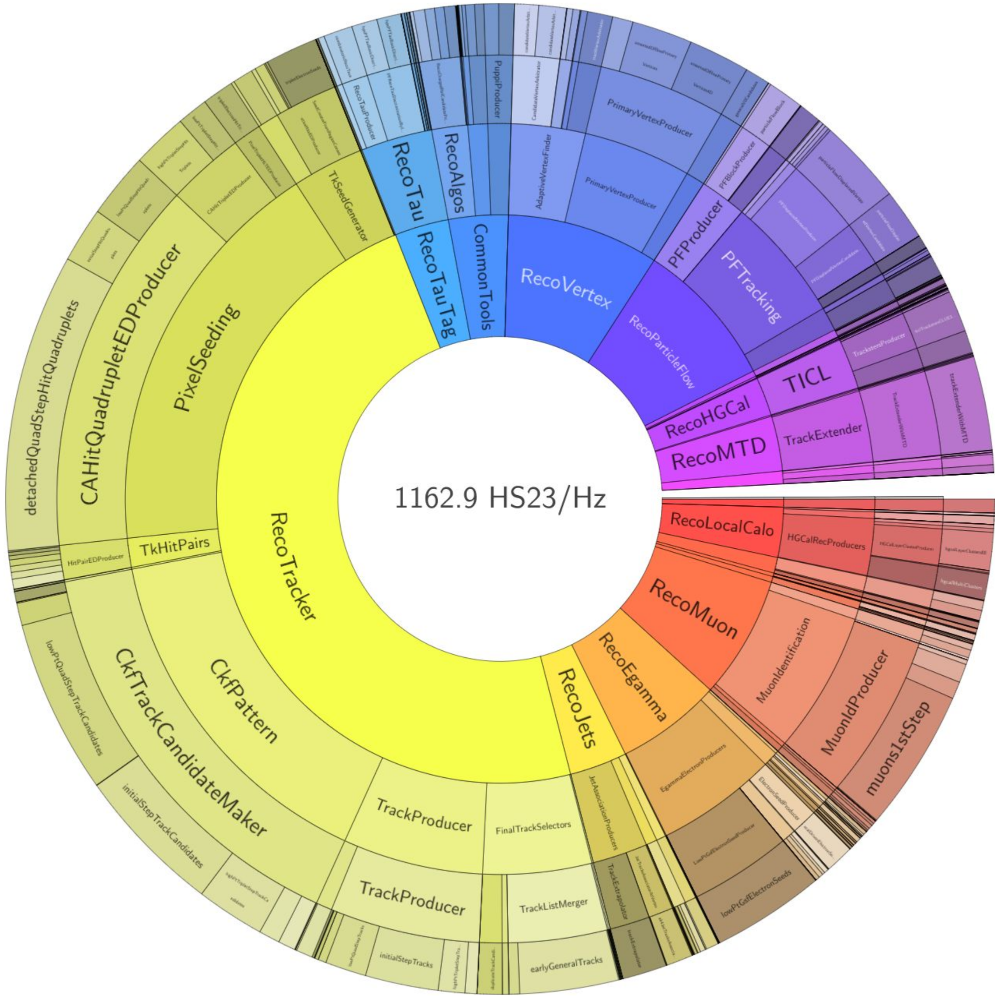
Offline Phase-2, L1-accept (simulation)



Generic Observations

- **Pixel seeding** still “full legacy”
- **Low p_T (300 MeV/c) & displaced (> 10 cm)** phase-space responsible for large fraction of the pie
- Brand new development, **HGCAL/TICL, extremely encouraging**

Offline Phase-2, L1-accept (simulation)



Generic Observations

- **Pixel seeding** still "full legacy"
- **Low p_T (300 MeV/c) & displaced (> 10 cm)** phase-space responsible for large fraction of the pie
- Brand new development, **HGCAL/TICL, extremely encouraging**
- **No usage of recent tracking developments** (LST)

Key Developments for Faster Reconstruction

- **Regardless of the extrapolation model, the NGT challenge is extremely ambitious**
  - Filterless Phase-2 HLT requires factors **3~4×** for Run-4 and Run-5
  - Pure Offline Phase-2 reconstruction requires factors **15×**
- **Pushing further on Heterogeneous Computing**
  - **Accelerate the integration of accelerators**.
  - Transition aligns with trends and complements SoA development.
- **Innovative Techniques**
  - Take inspiration and push forward on **The Iterative CLustering** (**TICL**) and **Line Segment Tracking** (**LST**).
    - Improve reconstruction performance and flexibility.
- **Low-p_T Region Coverage**
  - Balance computational cost vs. physics reach.
  - Extend capabilities for very displaced tracks.
- **AI-Driven Solutions**
  - Support novel AI methods for complex reconstruction challenges.
    - Some already started, e.g. DNN super-clustering in HGCAL, many yet to come.
  - Expand fast GPU inference to reduce latency and improve efficiency.

Key Developments for Faster Reconstruction

- **Cost-Benefit Analysis & Collaboration**
  - Foster continuous collaboration with physicists to align with experiment goals.
  - Evaluate costs and benefits of each initiative.
- **Develop a flexible Validation Framework based on solid Monte Carlo truth information**
  - Instrumental to understand in details physics performance, especially for “composite” objects and Particle Flow event interpretation
  - Essential to do any Machine Learning training.
- **Integrating Run-3 Improvements**
  - Leverage robust, tested features from Run-3 reconstruction
    - Heterogeneous Patatrack Pixel Tracks.
    - Heterogeneous ECAL, HCAL and partially Particle Flow
  - Aim for enhanced efficiency, accuracy, and reliability in the HLT Phase-2 framework.

CERN

NexTGen

Next Generation Triggers

Backup

Machines and running configurations

- **Release**: CMSSW_14_2_0_pre2 (latest pre-release available at the time)
- **Machines used**:
  - AMD EPYC "**Milan**" 7763
    - HS06: 3223.8 ± 33.8 (more info here)
    - HS23: 3629.334 (more info here)
    - Cores: 2×64×2 (number of sockets × physics cores × logical cores)
  - Offline Phase2 Configuration
    - 1 socket only, 2 jobs, 64 threads, 64 streams, mainly due to memory constraints.
    - The throughput measured has been scaled by a factor 2, as if the machine was fully occupied.
  - AMD EPYC "**Bergamo**" 9754
    - HS23: 7450.248 (more info here)
    - Cores: 2×128×2 (number of sockets × physics cores × logical cores)
  - Offline Phase2 Configuration
    - 1 socket only, 4 jobs, 64 threads, 64 streams, mainly due to memory constraints.
    - The throughput measured has been scaled by a factor 2, as if the machine was fully occupied.
- **All numbers and measurements will be expressed in HS23**.

Tables from the TDR

Sample	TTbar		L1-accepted minimum-bias	
	Baseline	Ultimate	Baseline	Ultimate
	140	200	140	200
Average time per event	4.7 s/ev	7.8 s/ev	2.2 s/ev	5.3 s/ev
+20% for tau lepton triggers	5.7 s/ev	9.3 s/ev	2.6 s/ev	6.3 s/ev
+50% contingency	8.5 s/ev	14.0 s/ev	4.0 s/ev	9.5 s/ev
Throughput per node	15.0 ev/s	9.1 ev/s	32.2 ev/s	13.4 ev/s
Throughput per kHS06	8.9 Hz/kHS06	5.4 Hz/kHS06	19.2 Hz/kHS06	8.0 Hz/kHS06
HLT input rate	500 kHz	750 kHz	500 kHz	750 kHz
Total processing power	55.9 MHS06	137.9 MHS06	26.1 MHS06	93.6 MHS06

	Run-4 (2028)		Run-5 (2032)	
Processing power needs				
current estimate (this TDR)	26.1 HS06		93.6 (26.1 + 67.4) HS06	
w/ factor 1.6 (2.5) speedup	16.3 HS06		37.3 (16.3 + 21.0) HS06	
	Evolution Model			
	+20 %	+15 %	+20 %	+15 %
Price/performance ratio				
current value (2020)	4.76 CHF/HS06			
improvement factor until run start	4.3×	3.1×	8.9×	5.4×
ratio at run start, CPU only	1.11 CHF/HS06	1.55 CHF/HS06	0.53 CHF/HS06	0.89 CHF/HS06
with 50% offload to GPU	0.70 CHF/HS06	0.99 CHF/HS06	–	–
with 80% offload to GPU	–	–	0.22 CHF/HS06	0.37 CHF/HS06
HLT farm cost				
cost at run start, CPU only	18.0 MCHF	25.3 MCHF	11.1 MCHF	18.5 MCHF
with 50% offload to GPU	11.4 MCHF	16.0 MCHF	–	–
with 80% offload to GPU	–	–	4.6 MCHF	7.7 MCHF

Offline configurations

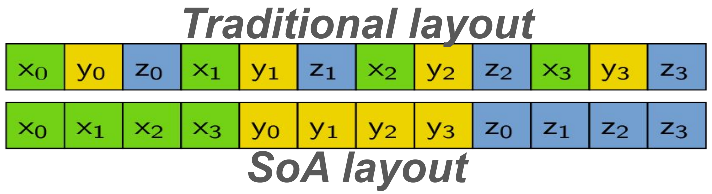
- All jobs have been configured to **run exclusively the reconstruction sequence**, w/o any DQM, Validation or anything else.
- **Output Module Disabled**:
  - Prevents filling up disk space; output module removed to minimize I/O impact.
  - Simply removing the output module causes unscheduled configuration to not run any modules.
  - Solution:
    - Use the `GenericConsumer` class, an `EDAnalyzer` to introduce artificial dependencies on EDM products.
    - **Configured similarly to a typical output module in CMSSW**.
- **Remove Monte Carlo dependent modules**
  - From `vertexrecoTask` remove
    - `process.quickTrackAssociatorByHits`
    - `process.tpClusterProducer`
    - `process.trackTimeValueMapProducer`
  - From `particleFlowRecoTask` remove
    - `process.quickTrackAssociatorByHits`
    - `process.simPFProducer`
    - `process.tpClusterProducer`
- **The goal is to evaluate how effective the reconstruction algorithms are, with any shortcuts taken here handled in other NGT tasks as necessary.**

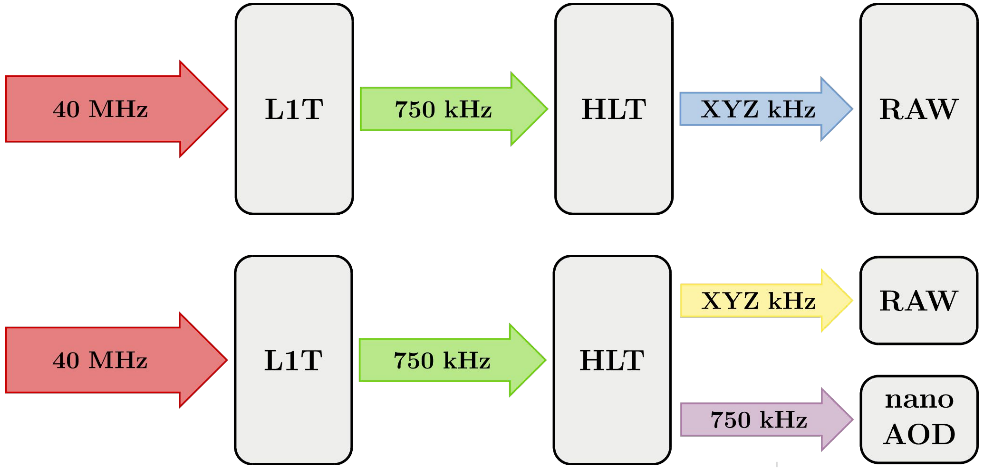
Full performance reports

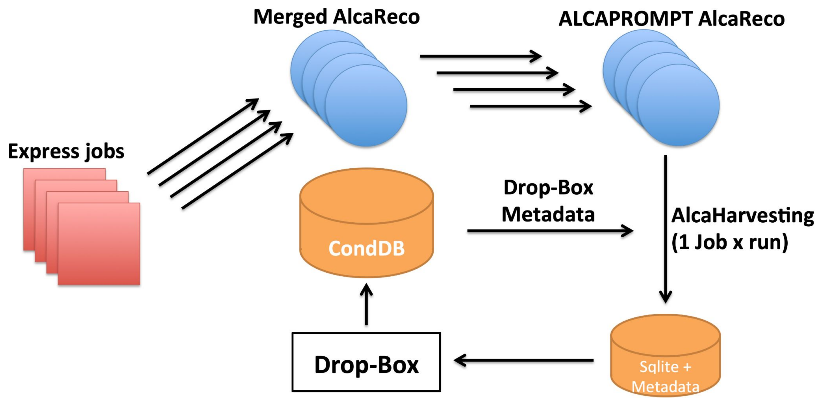
- Comprehensive performance reports are available at the following links:
  - HLTP2 TTbar, 200PU, Milan
  - HLTP2 TTbar, 200PU, Bergamo
  - HLTP2 L1-accept, 140PU, Milan
  - HLTP2 L1-accept, 200PU, Milan
  - HLTP2 L1-accept, 140PU, Bergamo
  - HLTP2 L1-accept, 200PU, Bergamo
  - OfflineP2 TTbar, 200PU, Milan
  - OfflineP2 TTbar, 200PU, Bergamo
  - OfflineP2 L1-accept, 140PU, Milan
  - OfflineP2 L1-accept, 200PU, Milan
  - OfflineP2 L1-accept, 140PU, Bergamo
  - OfflineP2 L1-accept, 200PU, Bergamo

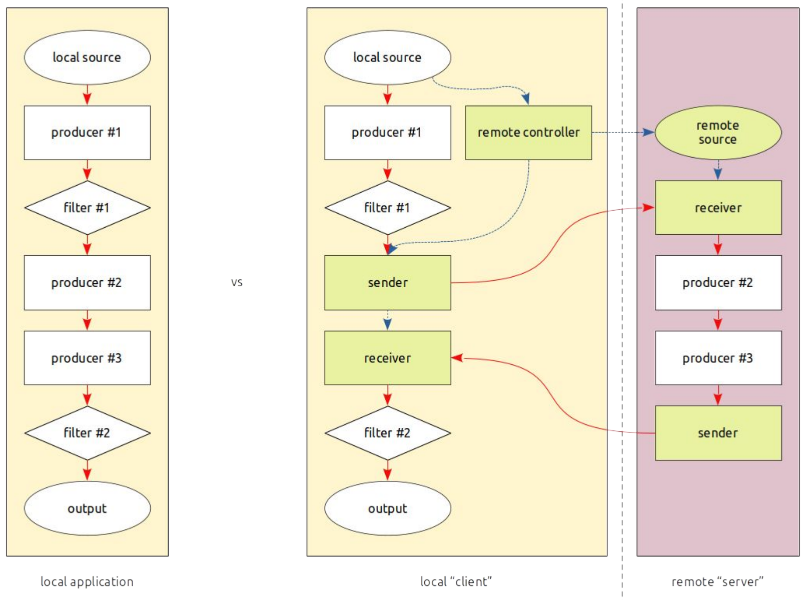
R^3 Has ambitious goals

- R^3 aims to transform the HLT event reconstruction by developing a suite of algorithms that rethink the process entirely, rather than just speeding up existing ones. Depending on the level of speed-up required, innovative approaches will be applied as needed to meet live physics analysis requirements. Key efforts include optimizing data structures for accelerators like GPUs, redesigning CMSSW as a distributed application with minimal code impact, and leveraging high-speed interconnects to reduce latency.
- R^3 will also reduce disk usage by compressing or simplifying data, and compute necessary conditions at HLT to match offline reconstruction quality, ensuring high physics performance with minimal disk space.







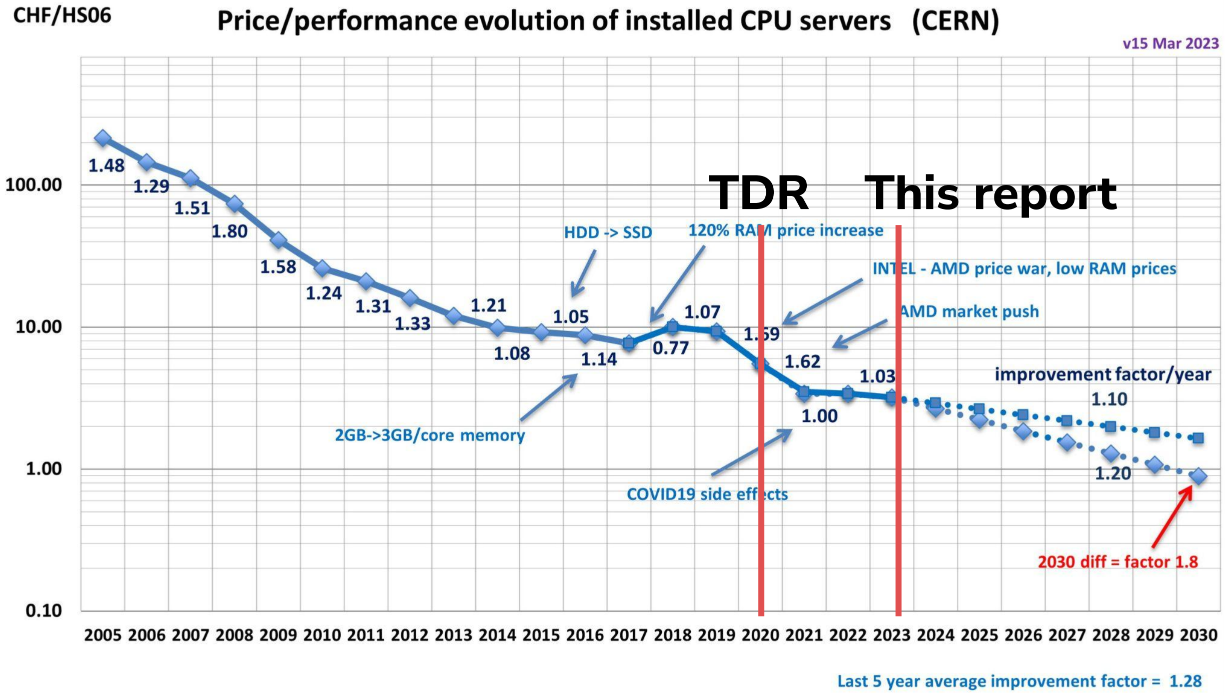


HLT Extrapolations

Updated on cost per performance CPU/GPU

- **TDR Estimate** (2020): GPU price-per-performance at **1.27 CHF/HS06**
- **Updated Configuration**: 2x AMD EPYC 9754 "**Bergamo**" processors + 3x NVIDIA L4 GPUs
- **Run-3 HLT Alpaka-only Workflow**: Pixel, ECAL, HCAL, partial particle flow
- **Measured Throughputs**:
  - CPU-only (256 cores / 512 threads): 1189 ± 6 events/second
  - GPU-only (3x L4 GPUs): 1915 ± 2 events/second
  - Combined CPU+GPU: 2783 ± 6 events/second
  - Contribution of 3 GPUs: Additional 1594 ± 9 events/second
  - Single GPU contribution: Additional 531 ± 3 events/second
- **Inferred Performance**: **Each L4 GPU adds 3327 HS23**
- **Updated Price-Per-Performance:** New estimate **0.58 CHF/HS23**

CPUs	Gain/Year	Current	CHF/HS23	Future	CHF/HS23	Speedup
2 × AMD EPYC “Bergamo” 9754	15 %	2023	2.6	2029	1.13	2.3×
	20 %	2023	2.6	2029	0.88	3.0×
	15 %	2023	2.6	2034	0.56	4.7×
	20 %	2023	2.6	2034	0.35	7.4×



Notes on HLT Extrapolations

- "**Bergamo**" machine is our current best-guess benchmarking machine
  - Evolution of performance/CHF realistically taken into account
- **The performance/CHF for the GPU has been updated using a Bergamo equipped with 3 NVIDIA L4**
- **HL-LHC schedule updated to be**
  - Run-4 starts in 2030, Run-5 starts in 2035
- L1-accept rate at 500kHz(Run-4) and 750kHz(Run-5)
- Optimistic scenario of 20% gain/year in performance/CHF (15% derived as well)
- 50% code on GPU (Run-4) and 80% code on GPU (Run-5)
- +50% of contingency applied to HLT Extrapolation due to simplified Menu
- **Google Sheet** with all measurements and extrapolation: link

HLT Extrapolations - Bergamo, L1-accept, Updated HL-LHC

Scenario	PU	Year Start	Throughput	Gain/year	Missing Factor CPU-Only	Fraction On GPU	Missing Factor w/ GPUs
Run4	140	2030	100.5 ev/s	+20.00%	2.8×	50.00%	1.7×
Run5	200	2036	68.6 ev/s	+20.00%	2.8×	80.00%	1.3×

Ongoing Development on accelerators: Line Segment Tracking (LST)

The LST algorithm

LST is a brand new **algorithm for building/seeding**:

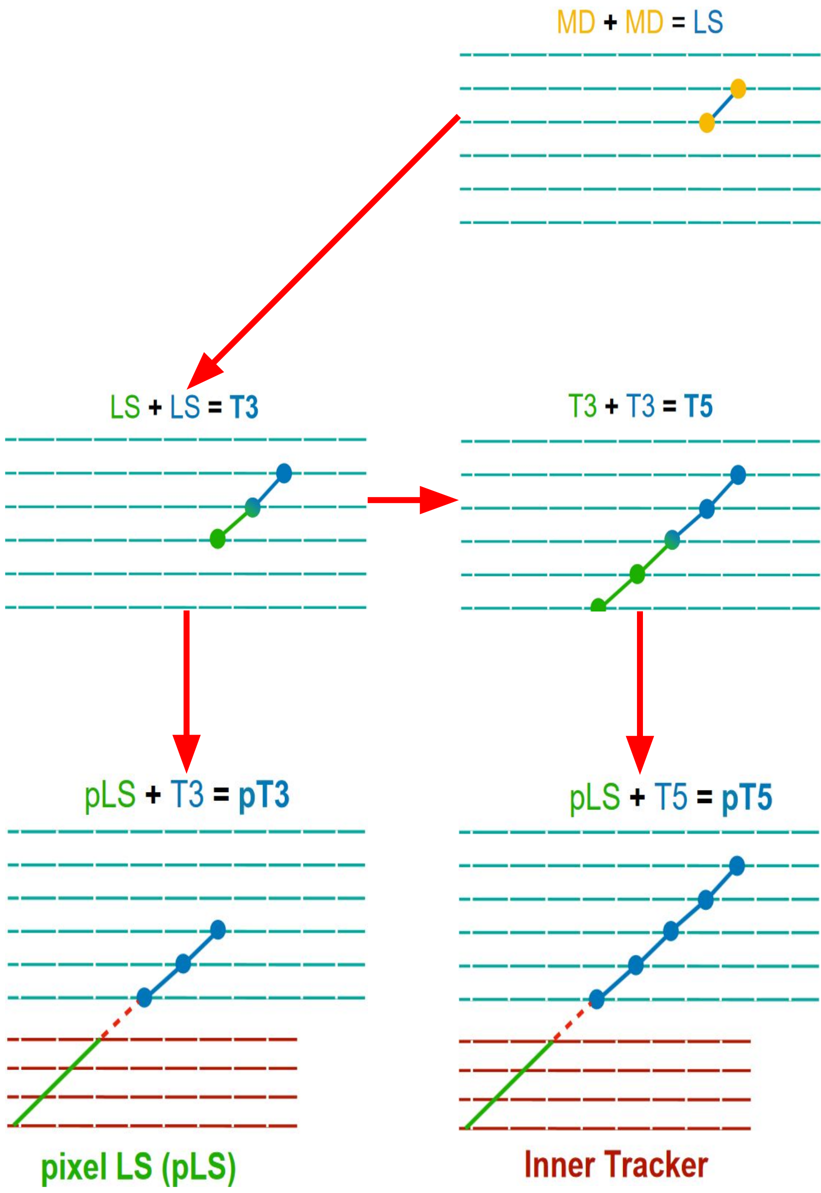
- Moves away from the Kalman filter logic.
- Relies on massive parallelization provided by accelerators.
- Written in the **Alpaka** framework to be hardware agnostic.

Algorithm logic:

- Start from pairs of hits in the tracker dual sensors. **Minidoublets (MDs)**: Similar L1 stubs but going down to p_T = 0.8 GeV.
- Link short objects to create longer ones.
- Objects independent of each other ⇒ **Massive parallelization**.

Improvements for both physics and computation

Performance under development.



Setup for comparison

Comparison between:

- "**Base CKF**":
  - **InitialStep:** **CKF building** on 4-hit Patatracks.
  - **HighPtTripletStep:** **CKF building** legacy 3-hit high-p_T recovery seeds.
- "**LST with CKF on LST Quads+Triplets**":
  - **InitialStep:** **LST building** on 4-hit Patatracks + legacy 3-hit high-p_T recovery seeds.
  - **HighPtTripletStep:** Recovery CKF building on 4-hit + 3-hit **LST seeds**.

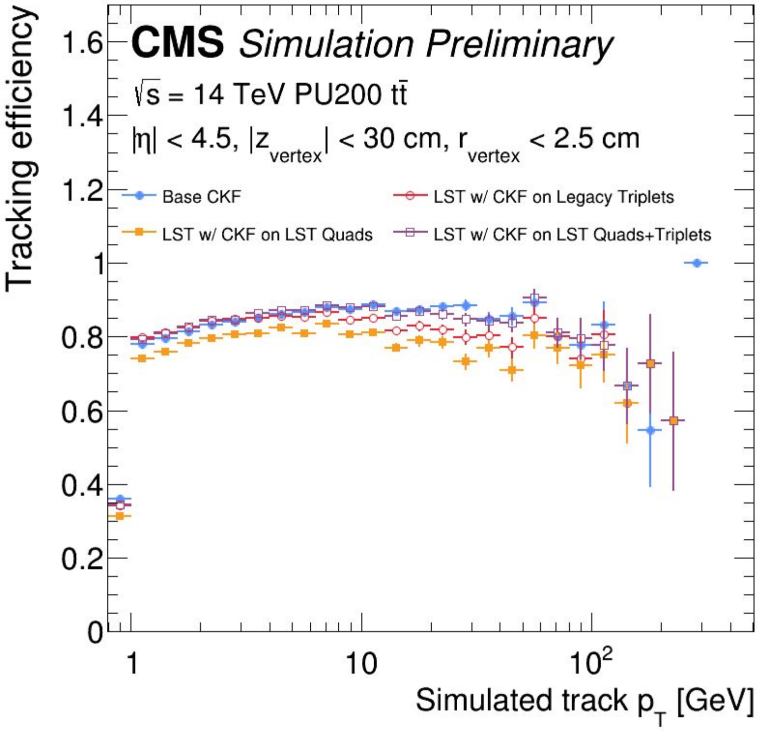
For the computing performance:

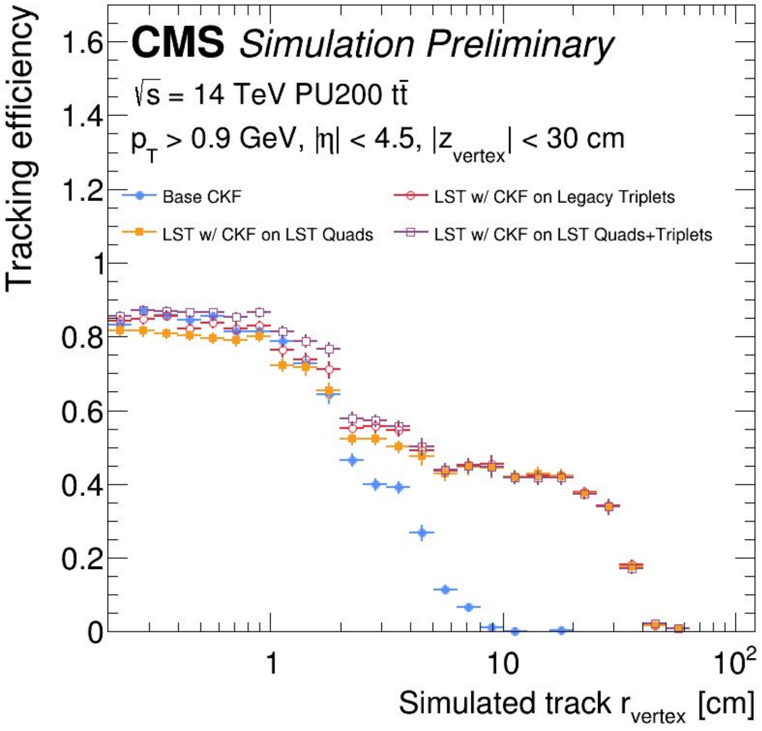
- Comparing only the HLT tracking sequence throughput on **1000 TTbar events at 200 PU**.
- Measurements with **2 threads** (for CPU = **AMD EPYC "Milan" 7763**), pinned to 2 specific CPU cores, and **2 streams** (for GPU = **NVIDIA "Ampere" A30 PCIe**) performed with local access to the input.

More details on the configurations in the backup and in DP2024/014.

Work in progress configurations with many developments/improvements expected.

Tracking efficiency

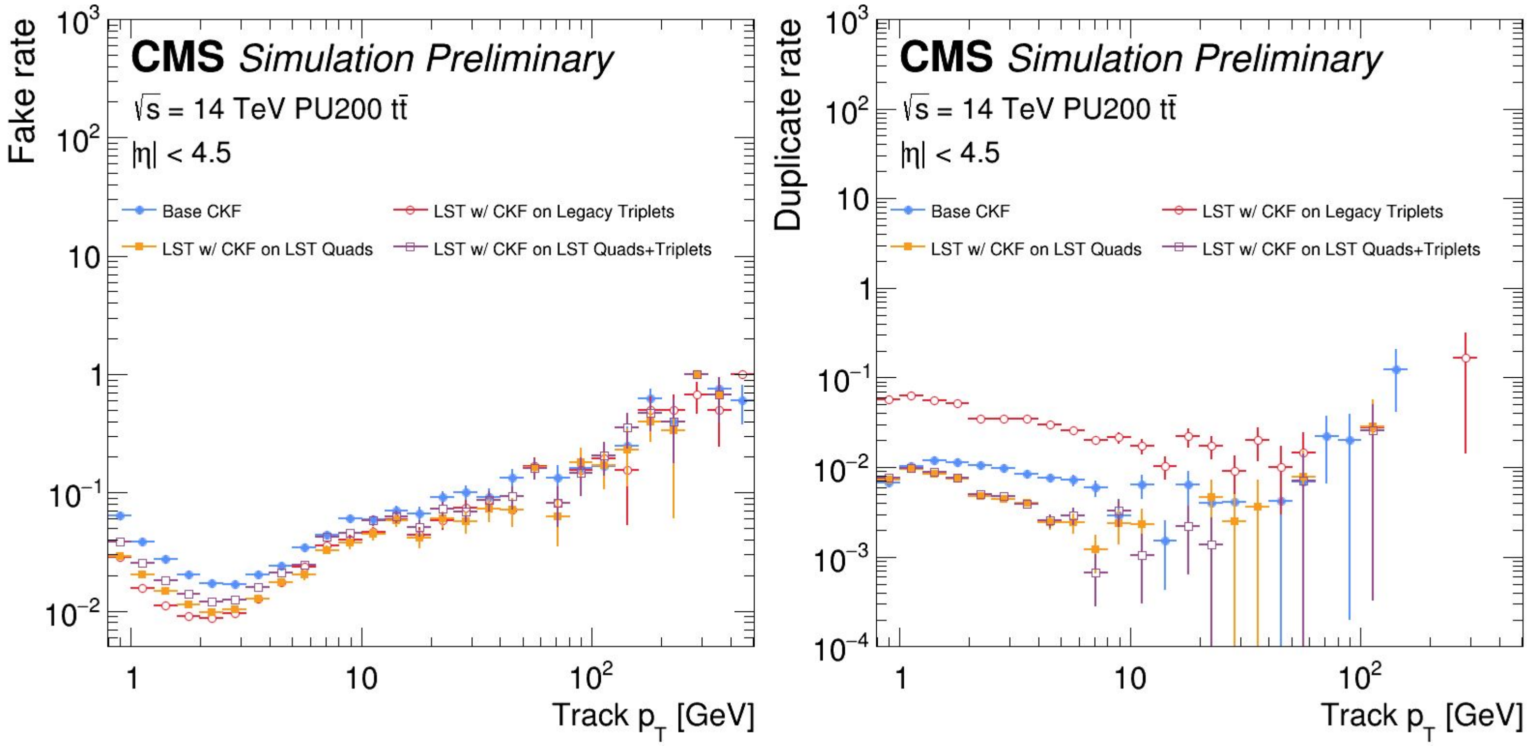




- Overall comparable efficiency vs. p_T:
  - Small **gains at low p_T**.
- Acceptance of **displaced tracks** (r_{vertex} > 5 cm) when **building with LST**:
  - Completely **new feature for HLT**.

Base CKF vs. **LST with CKF on LST Quads+Triplets**

Tracking fake & duplicate Rate



- Overall lower fake rate when **building with LST**:
  - Most **reduction at low p_T**, where the bulk of the tracks is.
- Overall lower duplicate rate when **building with LST**:
  - Most **reduction at high p_T**.
- Less tracks to process ⇒ Computing reduction downstream.

Base CKF vs. **LST with CKF on LST Quads+Triplets**

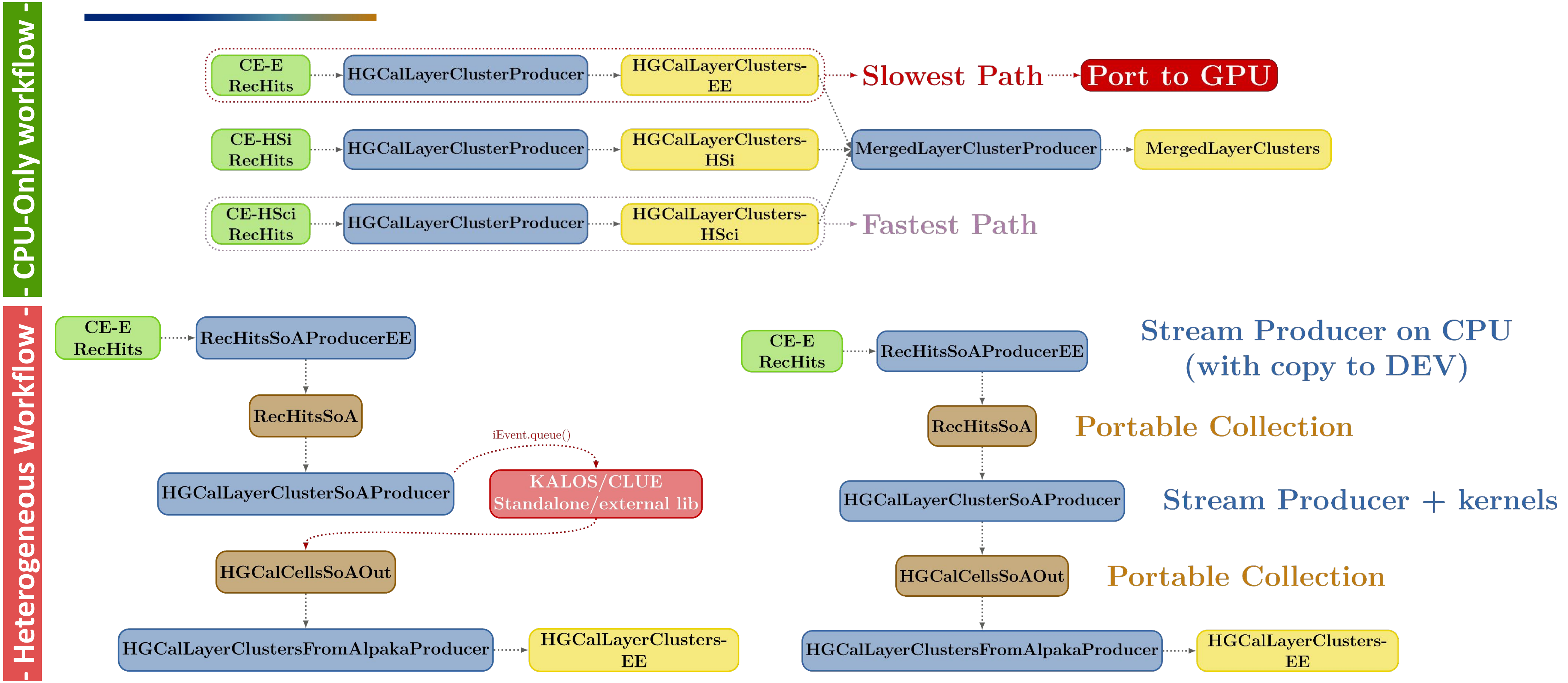
Measured computing Performance

- Both a **CPU and a GPU variant are available** for the LST algorithm.
- While the GPU variant of the LST algorithm was extensively tuned with profiling tools, the CPU variant may still benefit from an optimization.
- The **CPU variant runs serially** (no parallelization):
  - Currently no option for parallel CPU backend for Alpaka in CMSSW.
- The throughput value of the **Base CKF** configuration is used as a reference, i.e. **the values quoted are normalized** so that the **Base CKF** value is equal to 1.

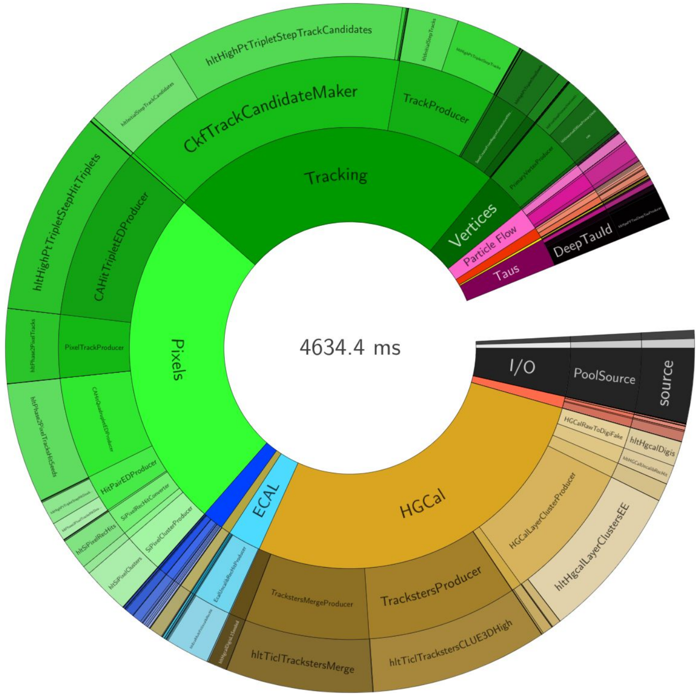
	LST with CKF on LST Quads+Triplets
LST on CPU Throughput / Base CKF	0.70 ± 0.09
LST on GPU Throughput / Base CKF	0.92 ± 0.09

Ongoing Development on accelerators: CLUE clustering

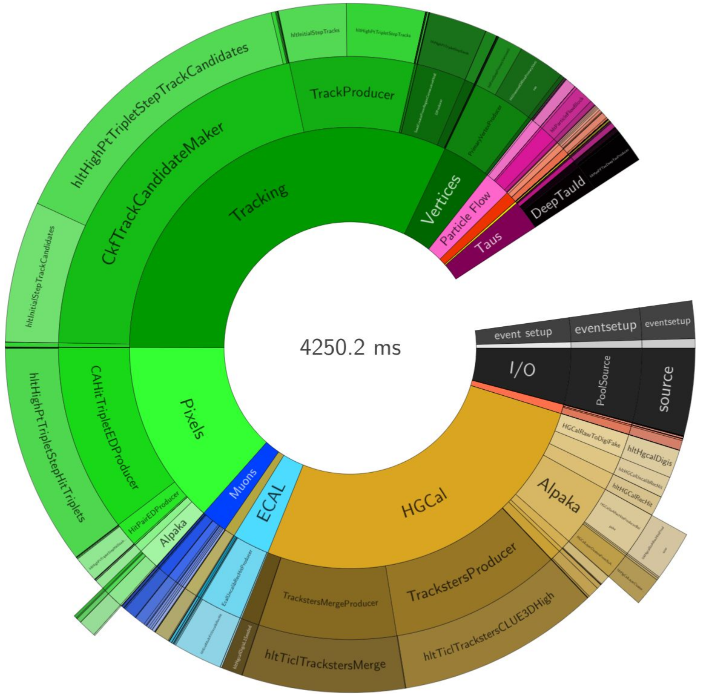
CLUE Heterogeneous Workflow in CMSSW



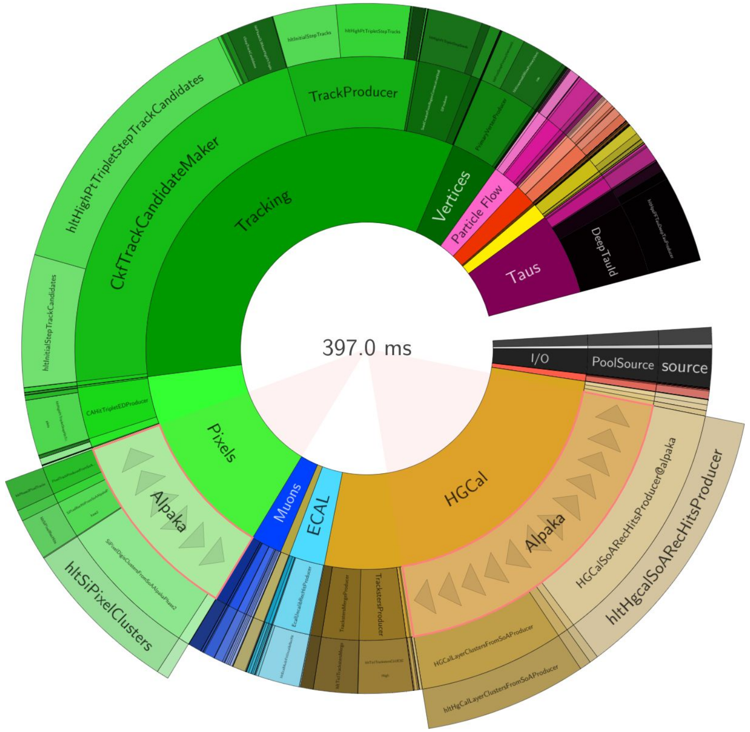
Performance of CLUE on GPU



All Legacy



Alpaka GPU



Zoomed Alpaka “GPU”

Performance of CLUE on GPU

Generic comments:

- Data transfer and conversions from/to legacy format are currently the bottlenecks
  - The more we port, the less we pay
- Experience with the current approach of portable SoA collection extremely positive
  - Could even use an external library w/o copying data around
- CLUE3D conceptually very similar to CLUE
  - Next candidate to extend the GPU processing chain
- Well advanced effort on RAW2DIGI+CalibratedRecHits on GPU [link to Pedro's talk]