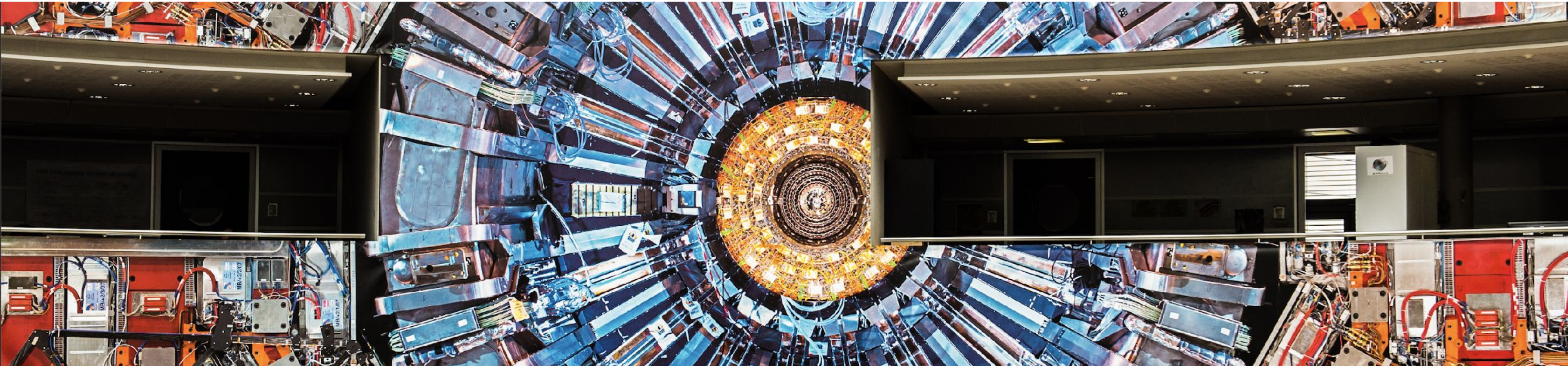
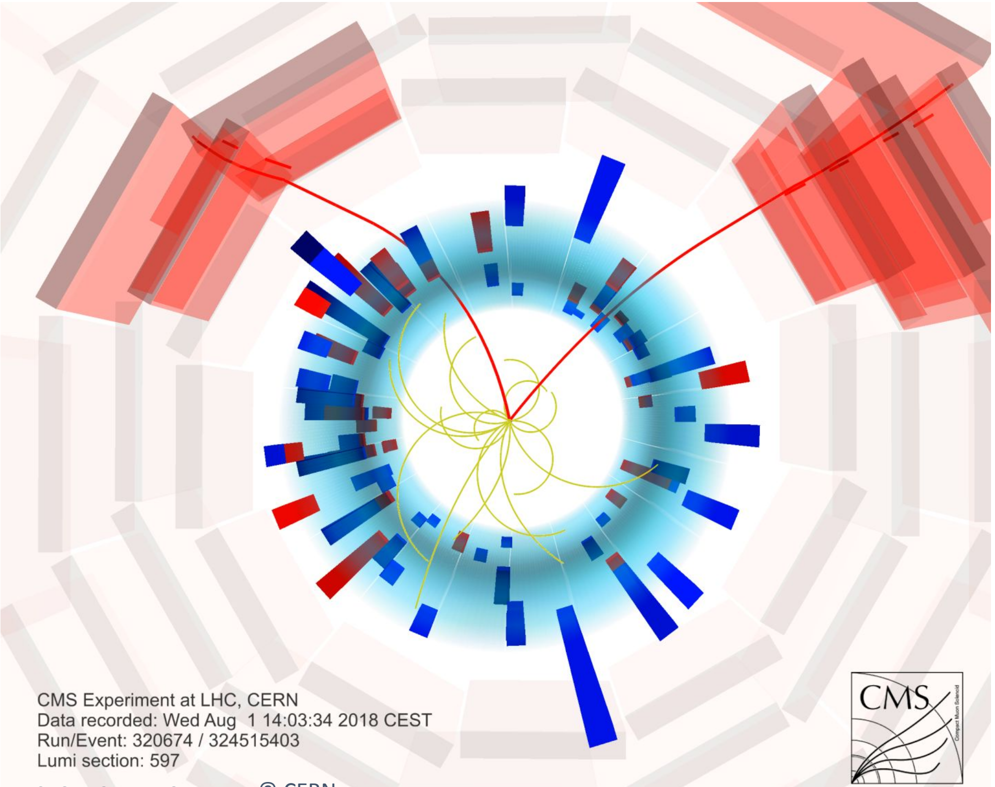
CBG Tier-2 Meeting

J. Letts (UCSD), D. Piparo (CERN) - September 19, 2022



This Talk

- The O&C area: structure, assets and activities
- Being a CMS Tier-2
- Challenges ahead: plans to tackle successfully the HL-LHC Challenge



CMS-PHO-EVENTS-2022-031 © CERN

Introduction

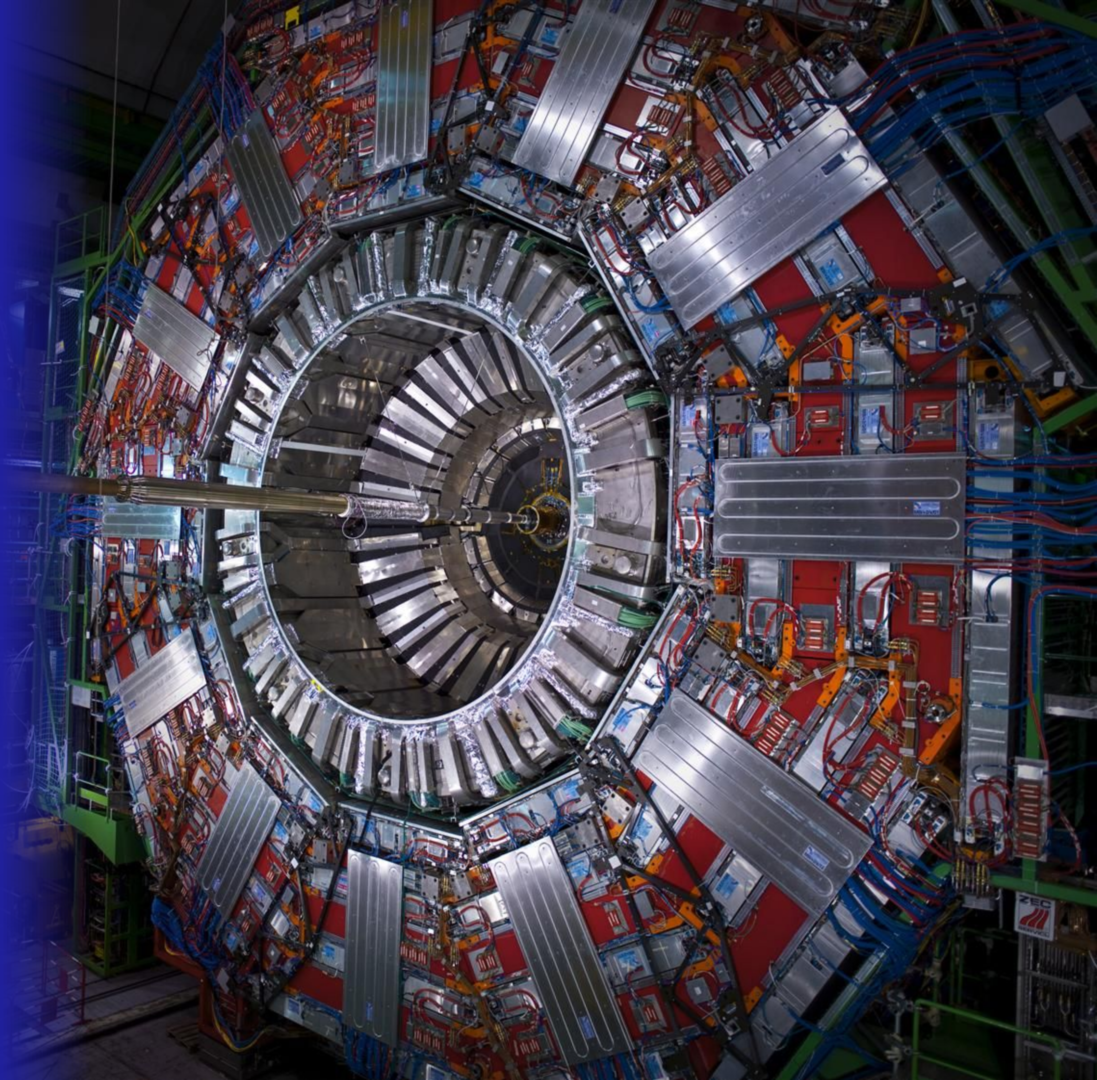


Image: © CERN

Offline Software and Computing

We deliver the datasets to enable the CMS Physics Programme and the software to produce, process, and analyse them

In other words, with software and computing, we enable the physics program of CMS.

Many **interesting activities at the bleeding edge of software and hardware technologies stem from this simple formulation.**

Our Groups

We are on Mattermost, and CMSTalk

Coordinators
D. Piparo, J. Letts

Core Software
S. Muzaffar, M. Kortelainen

Computing Operations
C. Paus, A. Wightman

Dyn. Res. Provisioning
D. Spiga, C. Wissing

Facility Services
G. Bagliesi, S. Lammel

Simulation
V. Ivantchenko, S. Bein

Workload/Data Mgt. Devel.
K. Lannon, D. Ciangottini

Reconstruction
C. Caputo, M. Nguyen

Resource Management
J. Flix, D. Lange

Monitoring & Analytics
F. Legger, B. K. Jashal

Offline Release Planning
A. Perrotta, S. Rappoccio

Upgrade Software
P. Srimanobhas, A. Di Florio

Submission Infra.
A. Perez-Calero, M. Mascheroni

Analysis Infra. & Support
S. Belforte, K. Ellis

Upgrade R&D and TDR
D. Elvira, F. Ferri

Web Services & Security
A. Pfeiffer, M. Imran

Generators *
S. Bhattacharya, M. Lu

Machine Learning *
G. Kasieczka, J. Ngadiuba

L1 Software **
C. Callol, E. Palencia Cortezon

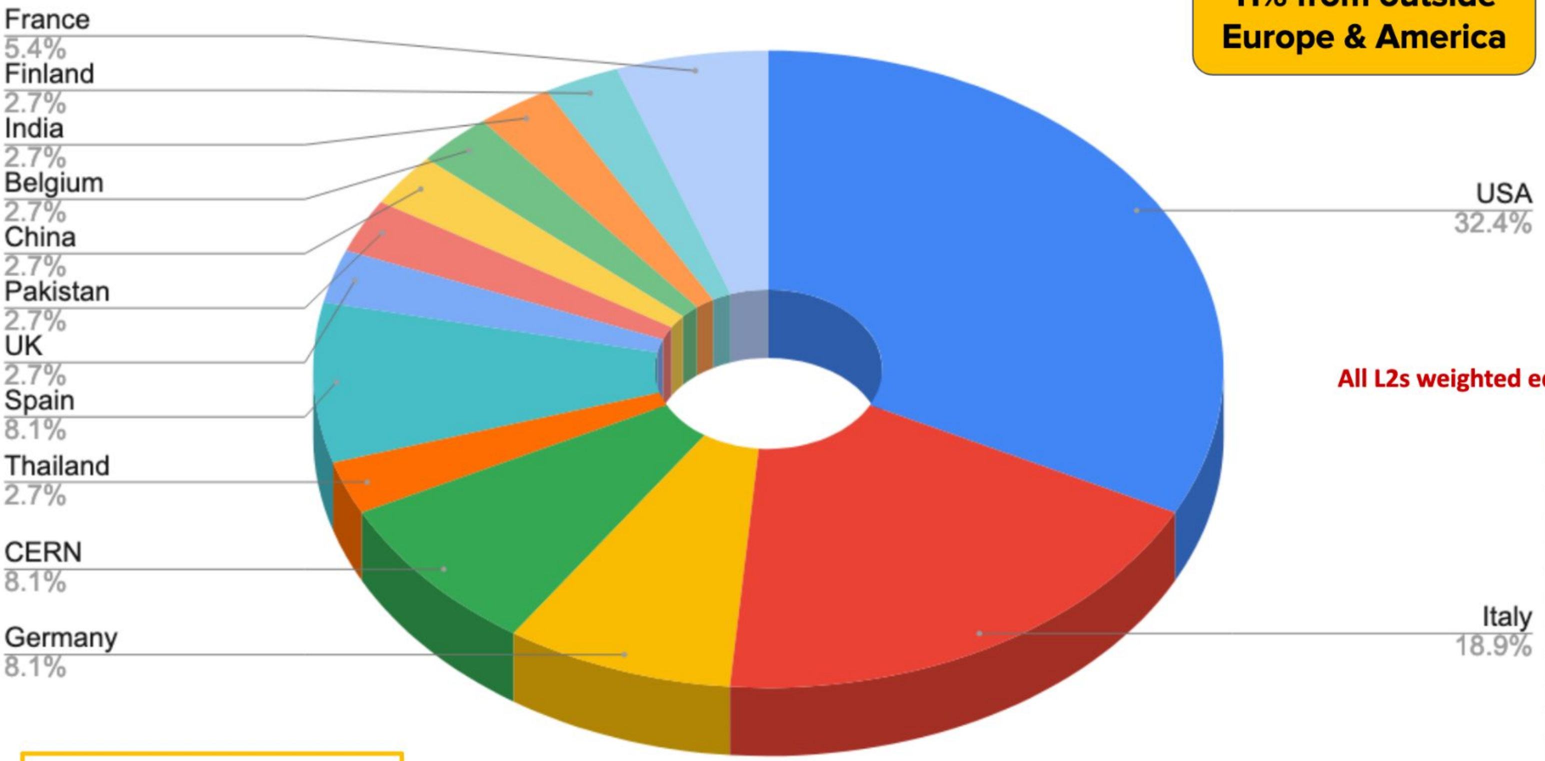
DPOA ***
K. Lassila-Perini, C. Lange (Dep.)

Computing Resources Board
J. Hernandez, K. Bloom

A very broad set of expertise

Our weekly meetings here
Join us for our O&C Week: 18-21 October !!

Management by Institute



11% from outside Europe & America

All L2s weighted equally

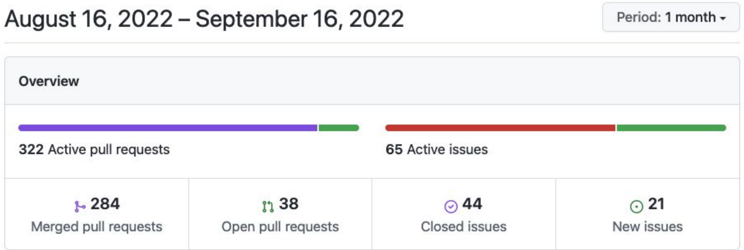
Country	L2s
USA	12
Italy	7
Germany	3
CERN	3
Thailand	1
Spain	3
UK	1
Pakistan	1
China	1
Belgium	1
India	1
Finland	1
France	2

L2 convenors as of Sep. 1st only

Our Software

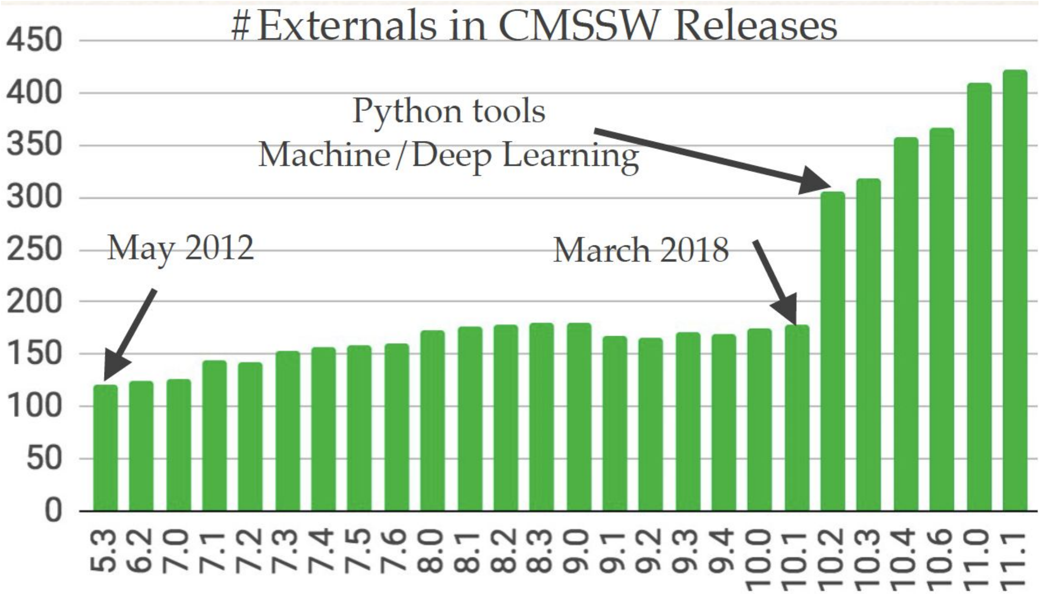
- **A crucial asset, built during many years**, condensing invaluable expertise
  - 1,100+ commits/month, 100+ committers/month
- 48% of it is C++, 29% Python (configuration but also HLT menus)
  - All this not including external 3rd party packages
  - **All algorithms and Framework written in C++ and CUDA!**
- Multithreaded: 4/8 core jobs
- Same codebase for High Level Trigger (HLT) and offline
  - Big advantage for CMS
- The CMS Software is on GitHub since 2012
  - Open source from the start

Language	LOC in CMSSW	Fraction in CMSSW
C++	4.9M	48%
Python	2.9M	29%
XML	1.5M	15%
Fortran	535k	5%



Detector Specific, but Incorporating 3rd Party Libraries

- Two broad categories of software components
  - Detector specific algorithms
  - External: all packages not owned by us, MC generators, compilers, ML libraries!



Our Innovations: Some Highlights

64-bit binaries in production

Invention and adoption of the VDT math library

Framework support for multithreading

HTCondor pools and GlideinWMS full adoption

Dynamo DDM in production

Multicore GlideinWMS pilots

Premixing for PU simulation in production

VecGeom in production

Beginning of NanoAOD adoption

20% of the HLT ported to CUDA

First CMSSW portability library evaluation

Geant4 10.7 integrated

30% analyses adopted NanoAOD

2010
2011
2012
2013
2014
2015
2016
2017
2018
2019
2020
2021
2022

Particle-flow adopted for reconstruction

First algorithm vectorised: vertexing

AAA and xRootD federation in production

Forward shower libraries for simulation

MiniAOD used for most analyses in production

Russian roulette in production

Unified (e.g. automatic tape handling, block-by-block processing) in production

First HLT algorithm running on GPU

Framework support for offload on accelerators

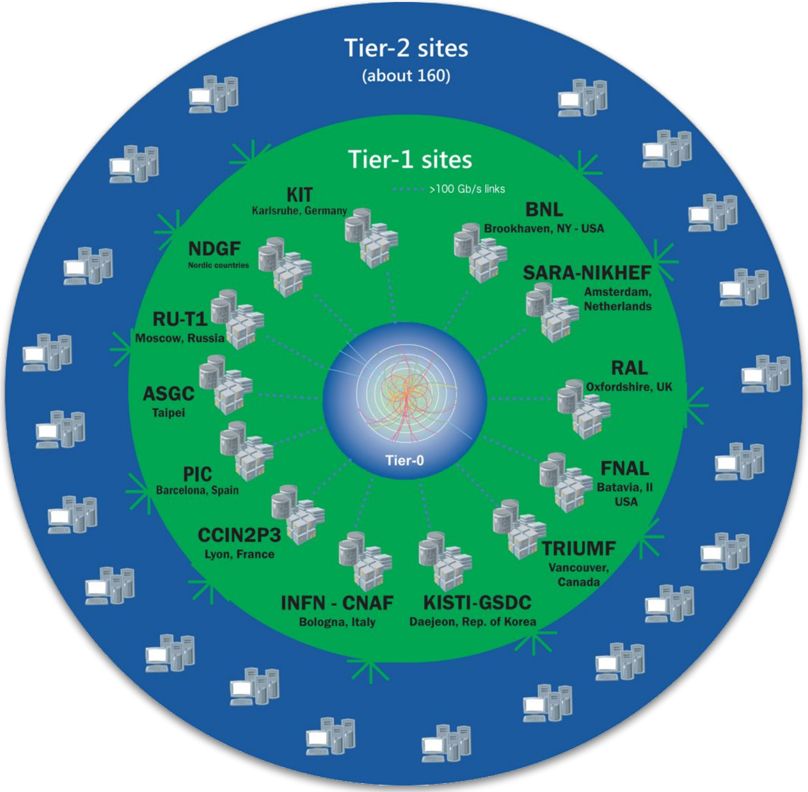
CMSSW offload to cloud accelerators demonstrated

Rucio and CTA in production

DD4hep integrated

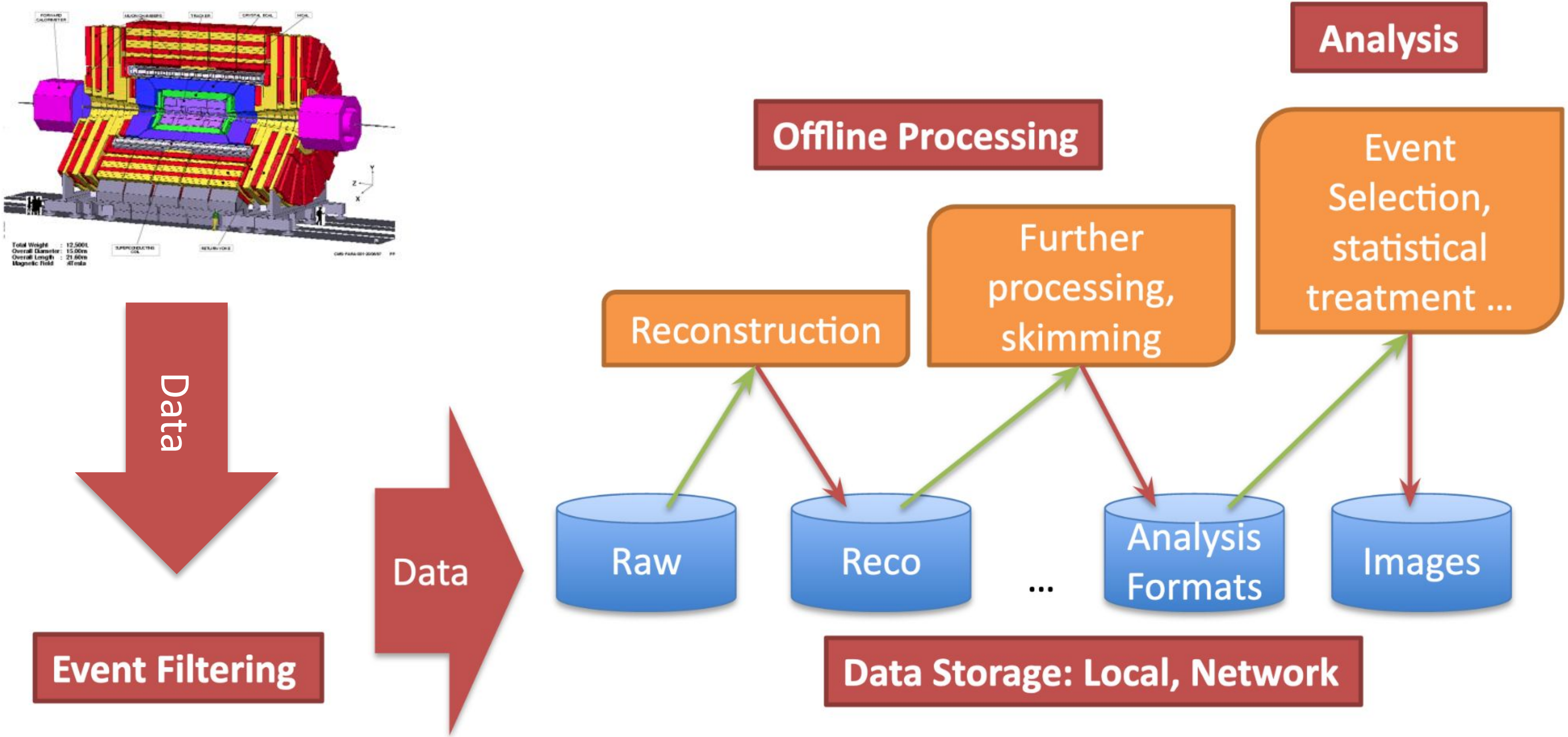
Central job scheduling on GPUs and POWER

The Worldwide LHC Computing Grid

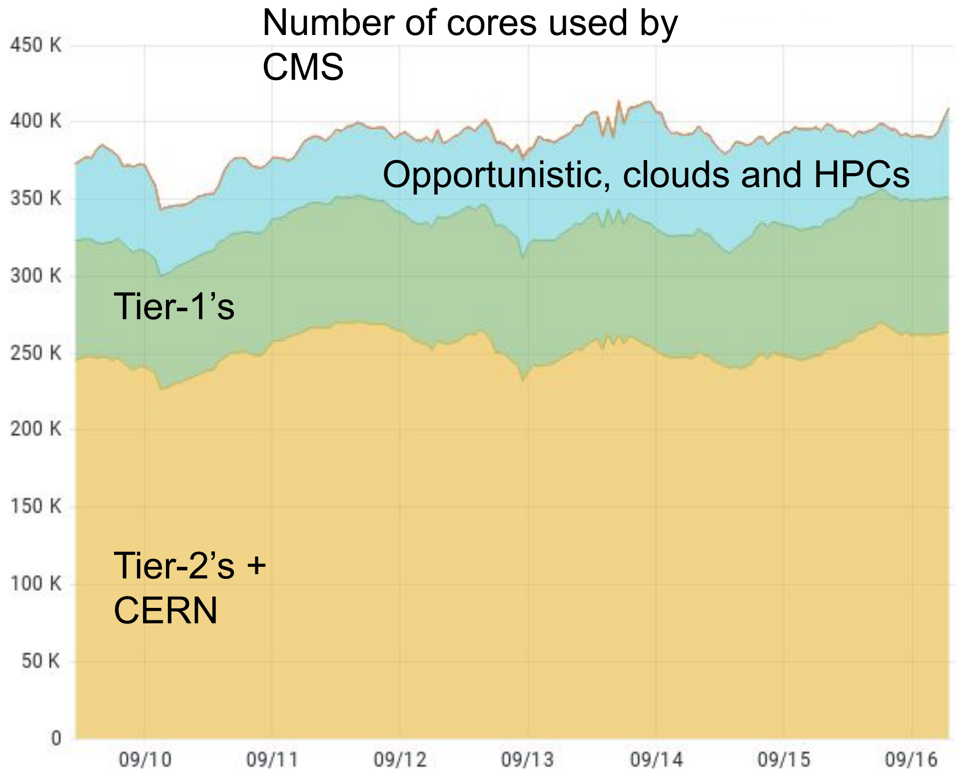


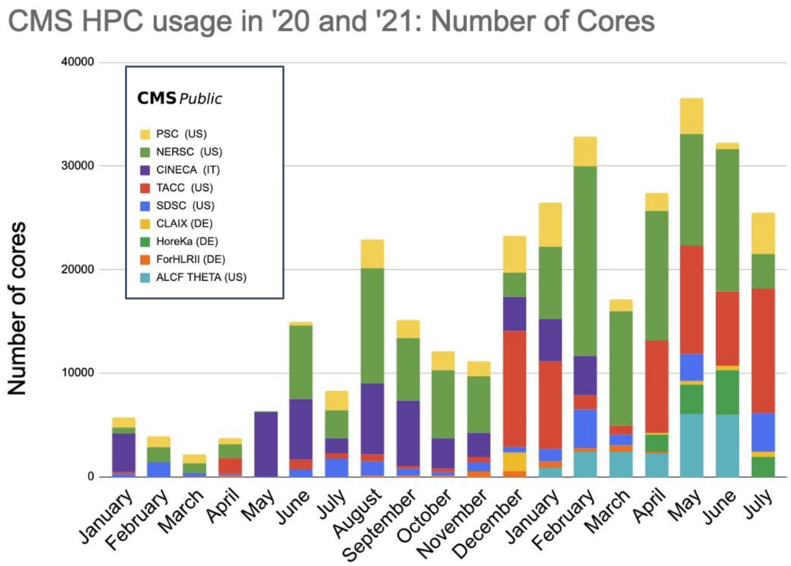
The growth of CMS in the Baltic region is an asset

HEP Data Processing, in a Nutshell



Our Computation



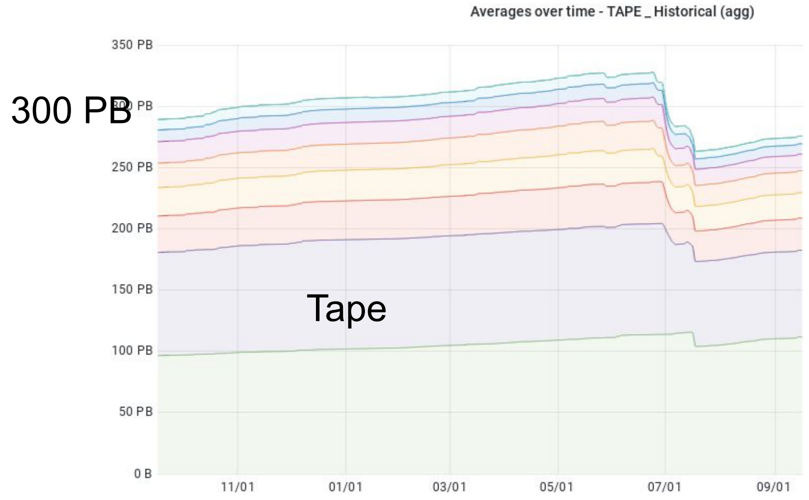


Cores at HPCs since 2020
HPCs: there to stay in our infrastructure, learning how to use them well!

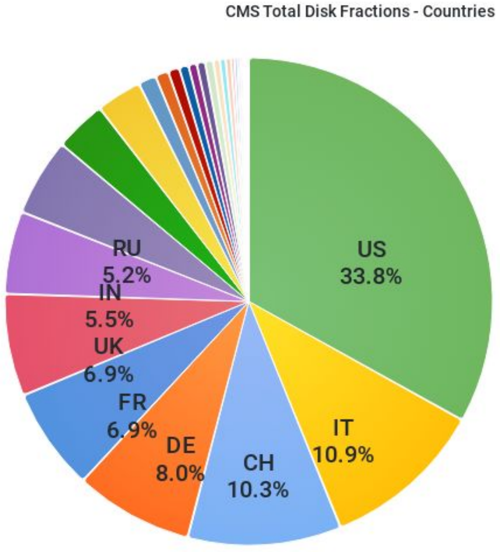
Our Data

RAW p-p Detector Data Sample:
58 PB at CERN, 44 PB at Tier-1's
Expect ~11 PB/year in Run 3 total

	max	current
Total(net)	328 PB	276 PB
T0_CH_CERN_Tape	116 PB	112 PB
T1_US_FNAL_Tape(net)	91.0 PB	70.7 PB
T1_IT_CNAF_Tape	34.5 PB	26.3 PB
T1_DE_KIT_Tape	27.3 PB	20.8 PB
T1_FR_CCIN2P3_Tape	23.8 PB	17.8 PB
T1_UK_RAL_Tape	19.1 PB	13.6 PB
T1_RU_JINR_Tape	11.7 PB	8.50 PB
T1_ES_PIC_Tape	9.13 PB	6.27 PB



	Value	Percent
US	58.1 PB	33.8%
IT	18.9 PB	10.9%
CH	17.8 PB	10.3%
DE	13.7 PB	8.0%
FR	11.9 PB	6.9%
UK	11.8 PB	6.9%
IN	9.5 PB	5.5%
RU	8.9 PB	5.2%
BE	5.9 PB	3.4%
ES	5.2 PB	3.0%
BR	2.0 PB	1.1%
KR	1.4 PB	0.8%
EE	1.2 PB	0.7%
HU	960.6 TB	0.6%
FI	841.7 TB	0.5%
UA	836.9 TB	0.5%
TR	818.1 TB	0.5%
TW	615.6 TB	0.4%
PL	535.1 TB	0.3%



Disk

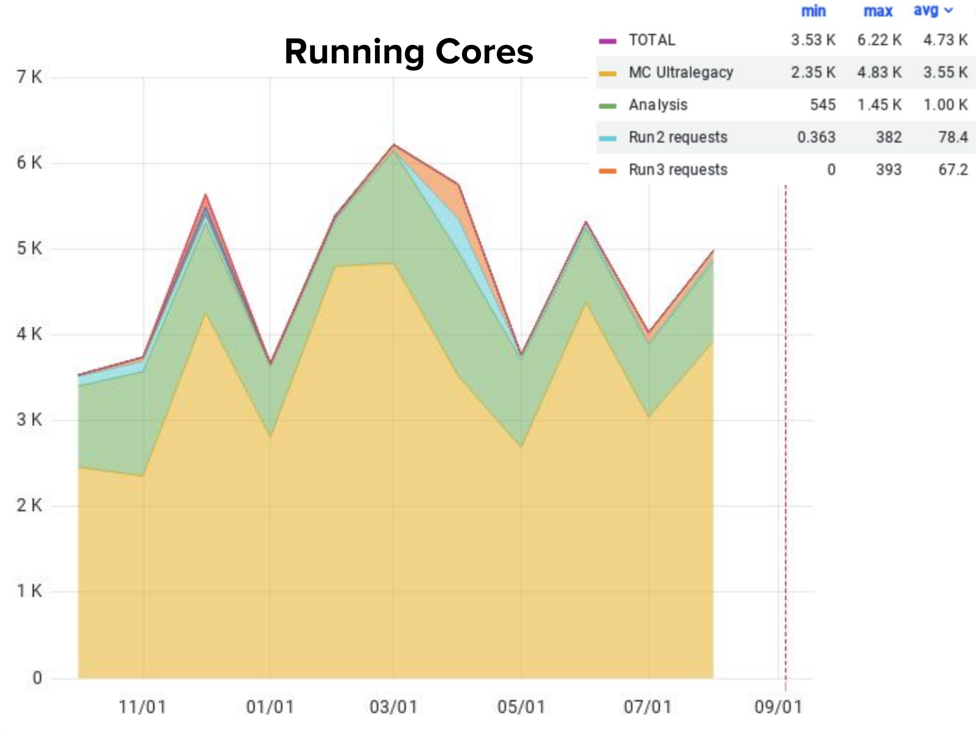
Being a CMS Tier-2

Image: © CERN

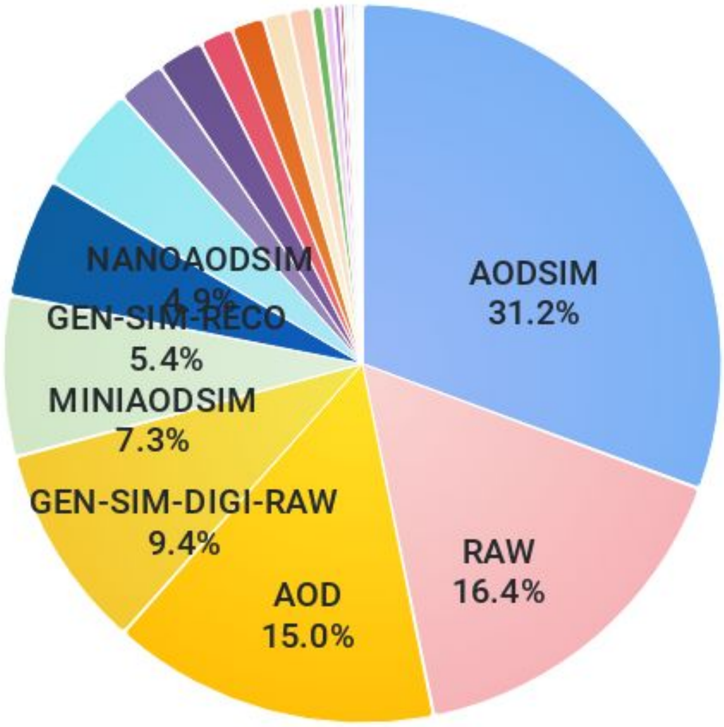
T2_EE_Estonia: CPU and Disk

- Highly reliable and powerful farm, since years

Running Cores



2 PB of Pledge Disk Storage



The Grid, Early 2000's

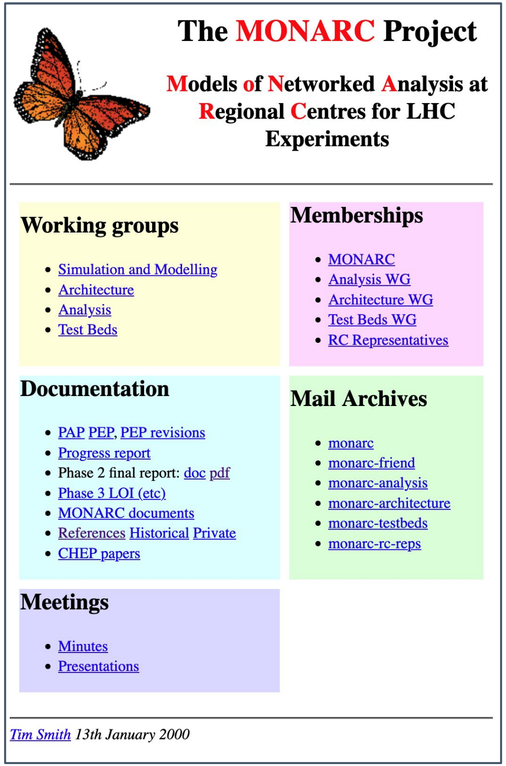
What does CMS need on a site plugged in the distributed infrastructure?

Not a unique answer; there is the historical one (MONARC):

- "You will install all the sites with the OP system we say"
- "You will give us WNs with 2 GB/core and 100 GB scratch disk, connected at 1 GBit/s"
- "You will have a managed disk with SRM protocol"
- "You will provide an experiment SW area"
- "You will have a Computing Element with this service release"
- "You will have X TB per Y CPUs"
- *Et cetera*: very standardised installation on Bare metal

We are past that. Today, CMS is much more flexible:

- "Give us the way to start a singularity container, and we will start our software"
- "Give us a remote network connections, and we will use it for sw and data"
- "If you have local storage, fine; otherwise a good network still works "



How can CMS be so Flexible?

The IT and sw technologies evolved around us during 2 decades. **We made of flexibility a priority of our computing model**, and:

- Because we have:
  - Reduced data formats, MT reducing memory, faster sw with a small footprint
  - Remote read capabilities like Xrootd, CVMFS, Squids
  - Virtualization and containerization offering new capabilities
- Because we need:
  - Some resources we need are not even “standard” for our “standard”: we start being able to exploit GPUs, high memory machines, fast SSDs for analyses
  - We cannot say “no thanks” to any remotely reasonable resource - we need to expand our resource base
  - It is anticipated that some Funding Agencies will want to unify HPC / GRID(“HTC”) infrastructures: we will need to adapt to new site configurations

Just Technical Work?

No. Plugged in the CMS and CERN scientific community, with many opportunities:

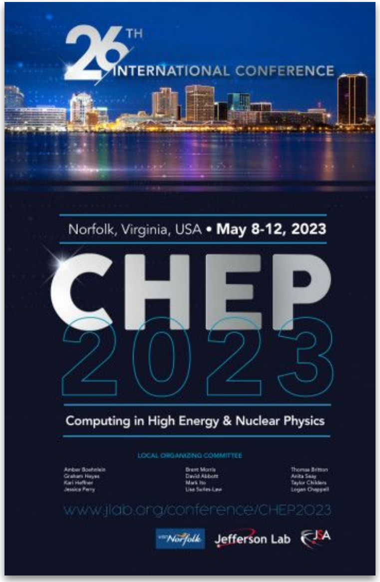
- Education, exchange of students/best practices, train personnel, collaboration with industry
- Publish, for example:

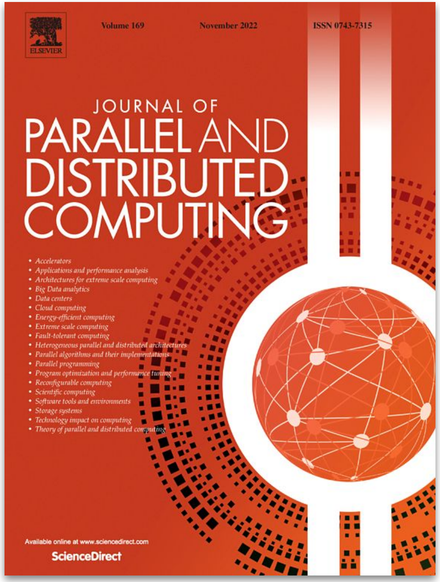
Conferences

- ACAT
- CHEP
- ICHEP
- CCGRID
- EuroPar
- ICCP
- IEEE Cluster
- ACM HPDC
- Supercomputing

Journals

- Journal of Parallel and Distributed Computing IF 4.5
- Computer Physics Communication IF 4.7
- Computing and Software for Big Science
- Journal of Computational Science
- Concurrency and Computation
- Journal of Grid Computing
- Distributed Computing
- The Journal of Supercomputing





Images from:
- https://www.jlab.org/conference/CHEP2023
- https://www.sciencedirect.com/journal/journal-of-parallel-and-distributed-computing

Challenges Ahead

(Highlights from the O&C pre-CDR document: CMS-NOTE-2022-008)

Image: © CERN

Timeline & Technical Challenges

The HL-LHC challenge: more complex events (140-200 PU vs 35 PU during Run 2) delivered to an ambitiously upgraded detector and recorded at a much higher rate (7.5 vs ~1.5 kHz)

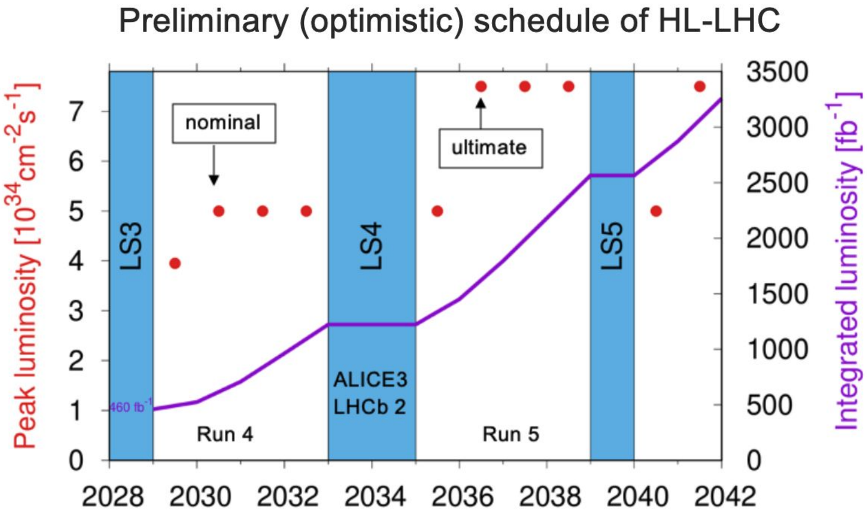
Latest schedule sees PU=200 only in Run 5, but we consider PU=200 to be the ultimate challenge.

These are the external conditions, what handles do we have to react?

We can affect:

- **Data format sizes (and their placement)**
- **Data processing and simulation time**
- ***How often* and *where* we execute workflows**

F. Gianotti, January 2022



The rest of this presentation will summarize the technical response to this challenge, taking the trigger rates as given from the recent DAQ/HLT TDR,

Unless otherwise stated, the results in our documentation refer to PU=200.

Framework for R&D

Software and computing-related R&D is tracked and coordinated centrally. Comprehensive R&D tables, which are updated regularly, summarize:

- Whether an R&D activity either **reduces resource requirements** or mitigates substantial **risks**, such as ability to use GPUs or HPCs (the *benefit*)
- Effort needed (the *cost*)
- Risks, dependencies, milestones, and decision points

Such R&D lines are generally carried out and managed in the wider collaboration.

In rare cases where a decision point excludes one or another project, we strive to make the decision **transparently** and by consensus.

Prioritization of R&D lines (if needed) is based on time to completion (e.g. benefits already in Run 3) and a cost-benefit analysis, weighted by the probability of success.

Three levels of confidence are used to build the ***Weighted-Probable Scenario***, i.e. the most probable outcome of our R&D program.

Not the final say:

- New initiatives may be started in the future
- Existing or planned R&D lines may (further) quantify their benefits

Probability of Success:
High = 100%
Medium = 50%
Low = 20%

Underpinning CMS' strategy for HL-LHC software & computing

Utilization of Accelerators and HPCs

Efficiently using all of the computing resources available to the experiment includes being able to take advantage of **accelerators** (e.g. GPUs), **HPCs**, as well as **non-x86** architectures (increasingly important at HPCs).

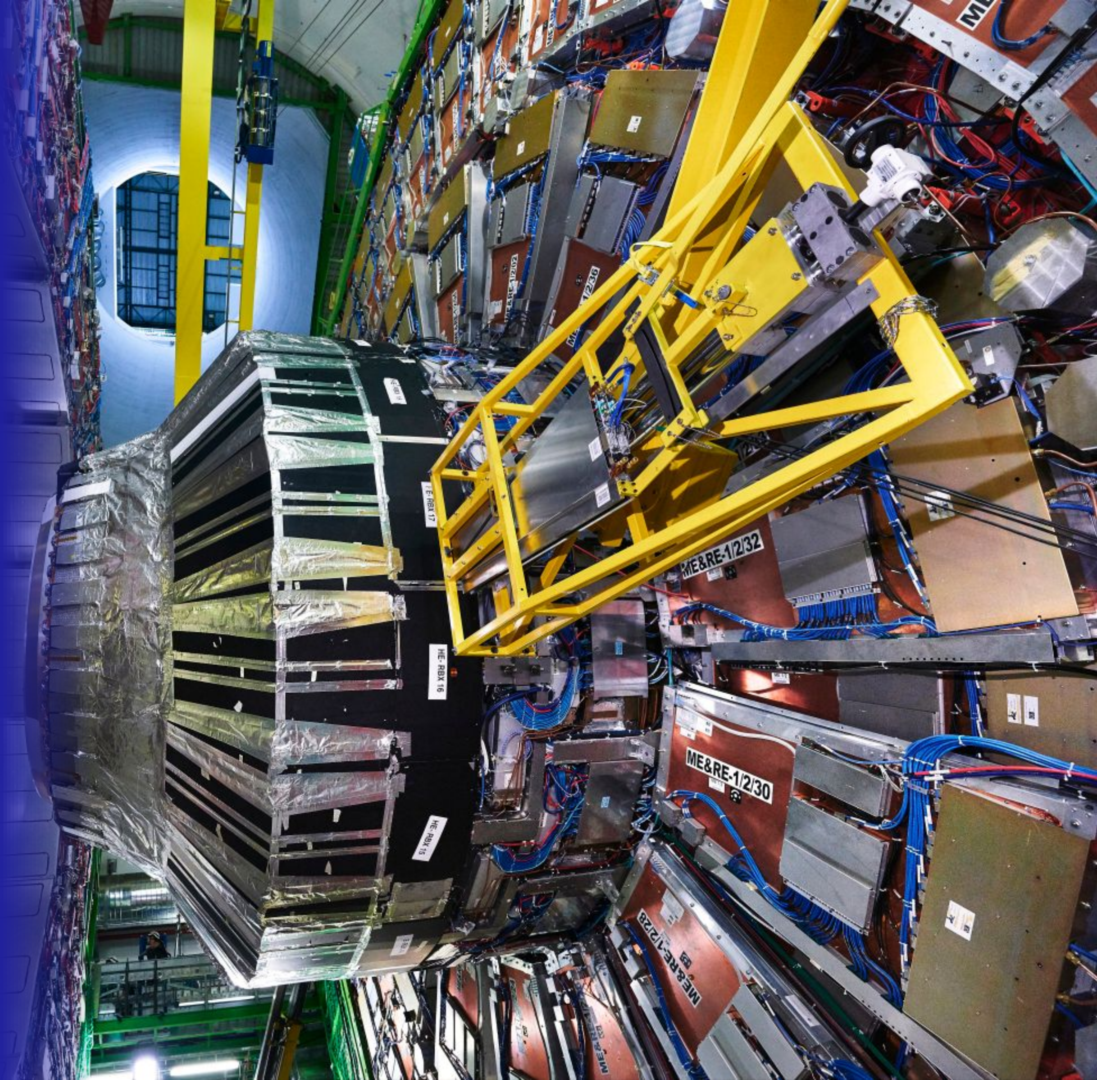
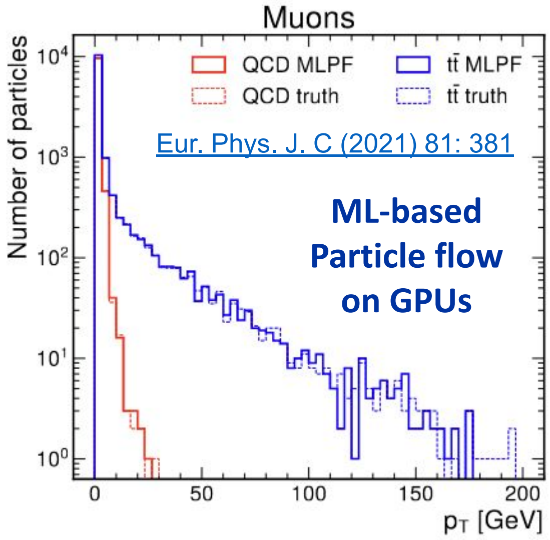


Image: © CERN

Usage of Accelerators (e.g. GPUs)

Short description	Impact	Probability of Success
Development and maintenance of CMSSW	critical	high
ML and algorithm offload on separate process	important	medium
Batching event data in the framework	important	high
Identification of the most adequate portability layer	important	high
Support accelerators in the submission infrastructure	critical	high
Benchmarking of heterogeneous platforms*	important	high

* In collaboration with the WLCG HEPScore Working Group



CMS has a solid plan to leverage GPUs both in the software framework (CMSSW) and the distributed infrastructure.

GPU algorithms used already at the HLT in Run 3!

Many Machine Learning R&D efforts (ex: MLPF, top right)

Making our computing model more flexible!

R&D efforts characterized by their impact (critical, important, or optional) *or* the potential percentage offload.

- More resilient to shifts in costs per unit of compute
- Allocations at HPCs may require use of GPUs

Short description	Impact as CPU offloaded Sim	Reco	Probability of Adoption
GPU based Geant4	100%	-	low
Pixel track	-	5%	high
CA for track building	-	10%	medium
Electron seeding	-	5%	medium
HGCAL TICL*	-	5%	high
HGCAL ML*	-	5%	medium
Particle flow	-	7%	high
Optimistic impact	100%	32%	
Weighted probable impact	20%	25%	

* 100% correlation accounted for in the summation of both overall impacts.

Performance Portability

- Performance portability library: one code base ➜ binaries for N architectures
  - E.g. for CPU, AMD-NVidia GPUs and/or other accelerators
- **Alpaka: CMS's performance portability solution for Run 3**
  - Good programming interfaces, **performance loss wrt native CUDA below 5%**
  - Investigation for HL-LHC continues: other promising options already available today
- Alpaka integrated in CMSSW 12_X (current release cycle). Next steps:
  - 26-30 September - Patatrack Hackaton about performance portability in person at CERN Idea Square. Reconnect with the community after the Pandemic
  - Q4 `22 - Current offloading framework ported to Alpaka
  - H1 `23 - Existing CUDA algorithms in Alpaka (HCAL, ECAL, Pixel Tracking)
  - H1 `23 - Partial offload of offline tracking + full primary vertexing on GPU
  - H1 `23 - HGCal reconstruction in Alpaka (from C++, no CUDA implementation)
- **More flexibility to be ready to run on GPUs if available to us**

A solid plan ahead, based on informed technical decisions

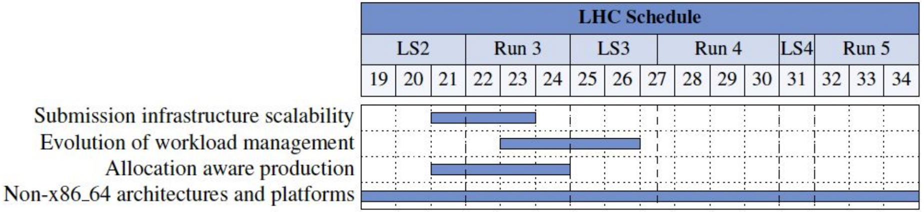
Documentation at http://alpaka-group.github.io/alpaka/

Integration of HPCs

HPCs are already part of the scientific computing infrastructure and will be in the future, nationally and internationally.

- Integration challenges: “Seamless” integration in existing WLCG sites has the least operational cost.
- Capacity has tripled year-over-year since 2019: Now ~10% of our total compute capacity (including opportunistic)
- Non-x86 architectures at HPCs:
  - CMSSW already built regularly on ARM and POWER9 archs. since several years.
  - Results of physics validation on POWER9 at the Marconi100 machine to be announced this week (INFN-CINECA)

Short description	Impact	Probability of Adoption/Success
Evolution of workload management system	critical	high
Submission infrastructure scalability	critical	high
Allocation aware production management	optional	high
Architectures and platforms	important	high



Minimizing Resource Requirements

Image: © CERN

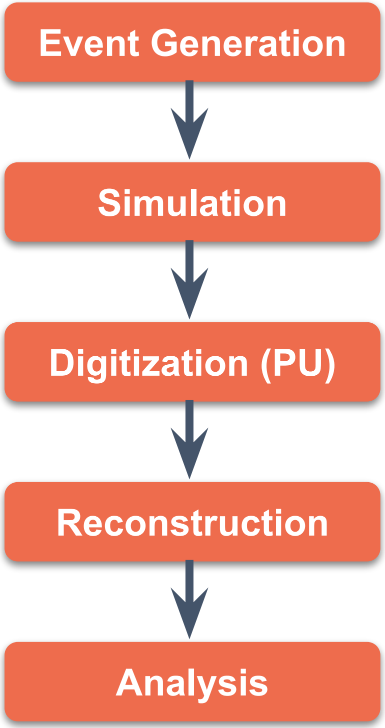
Minimizing Resource Requirements

Storage and compute resource requirements are ultimately driven by:

- Size of our data formats and their volume(s)
- Speeds of our data processing steps and how often we run them

Also drives network needs…

Tier	Event size [MB]	
	200 PU	140 PU
RAW	5.9	4.3
AOD	2	1.4
MiniAOD	0.25	0.18
NanoAOD	0.004	0.004



Minimizing Resource Requirements: Storage

There are several R&D lines which can reduce our requirements for tape and disk storage.

- RAW' - partial processing of RAW data (tracker strips)
  - Already will be used in Run 3 for Heavy Ions
  - Potential 15% reduction in RAW data size, if passes physics validation for proton-proton

- ROOT's RNTuple - new, efficient columnar storage for HEP
  - Works well for AOD, MiniAOD, NanoAOD - datasets with many columns containing collections
    - Integrated in CMSSW: we can write Nano in RNTuple format.
    - More work needed to expand scope of the integration
    - Potential 20% reduction in AOD, MiniAOD, NanoAOD event sizes
  - Not effective with RAW-events = blobs of binary output

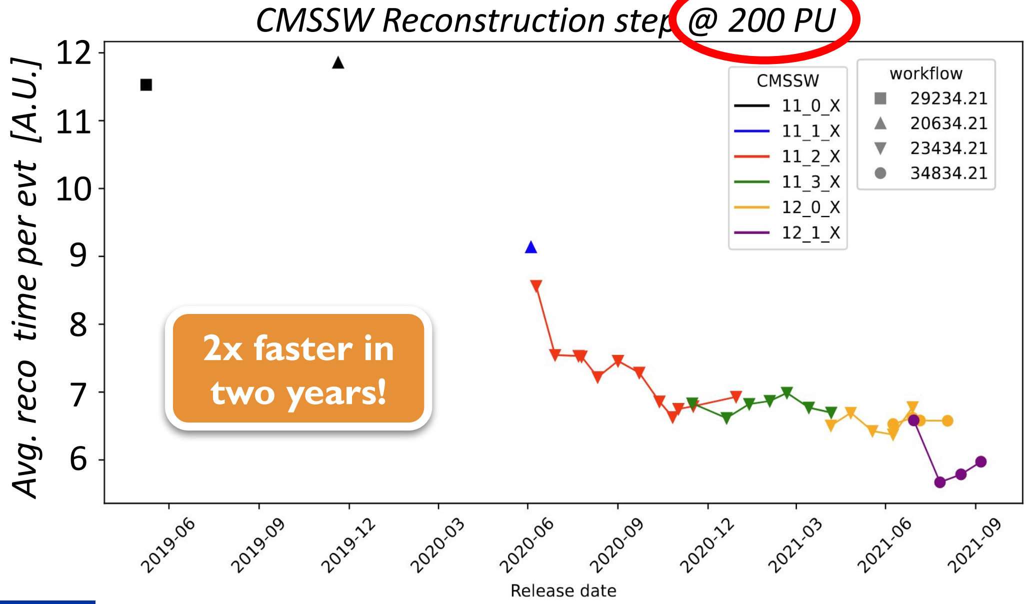
- Greater adoption of NanoAOD for analysis - smallest data format in CMS (4kB @ PU=200)
  - Goal that 50% of physics analyses adopt by end of Run 3 (currently 30%)
  - Allows us to keep less of the larger data formats on disk for analysis
  - Not so much an R&D activity, as a strategy to do analysis in the ab^{-1} era

Minimizing Resource Requirements: CPU for Reconstruction

Reconstruction: biggest CPU consumer in our processing chain, today and for HL-LHC.

Fortunately, there are many handles to reduce reconstruction time:

- Continuous optimization (see plot): refine algorithms, adapt thresholds, deploy new compiler versions and core libraries (e.g. memory allocators, math routines), vectorize: Expect 10% y/y improvement until Run 4 start, based on our experience.
- Tracking displacement and p_T cuts optimization: 30% improvement
- *mkFit* (vectorized and parallelized combinatorial Kalman filter based track trajectory building) 10% improvement, already in Run 3
- *TICL*-based HGCAL reconstruction (iterative clustering)



Minimizing Resource Requirements: CPU for Simulation

Two simulation approaches in CMS currently:

1. High-fidelity Geant4-based plus standard reconstruction
   - Approach used for the vast majority of the simulated events
2. Fast Monte Carlo Chain (in jargon “FastSim”): no G4, simplified geometry, fast simulation *and* reconstruction
   - 10x faster, accurate to within 10%, used mostly for parameter scans, e.g.
   - Plan to use Fast MC chain for a limited amount of events in Run 4/5 (see next slides)

Activities which aim to reduce CPU needs for G4-Based simulation:

- Geant4 incremental code performance improvement (external to CMS): 10% y/y, based on past history.
- Internal CMS improvements: 30% total reduction
  - Optimize geometry navigation
  - Adequate physics list for Phase-2
  - GFlash parameterizations in calorimeters per particle type for low-energy ranges

Total combined expected CPU time improvement for G4-based simulation 50% by Run 4 start

Minimizing Resource Requirements: CPU for Generators

The runtime performance of (N)NLO event generators needed for HL-LHC drives the CPU resource requirements for this step of the processing

Challenges:

- **Architecture of standalone generator code**
  - Improvements would reduce the cost of integration in the experiment's software stack
- **Thread-friendliness**
  - Needed to improve throughput and to reduce risk of encapsulating thread-unsafe code, as CMS increases the thread count per job in Run 4
- Code runtime **performance** and memory footprint
- **Negative weights**
  - Affects some approaches to NLO generators
  - A negatively-weighted event needs to be compensated by positively-weighted event(s)
  - Direct implication for both CPU and storage needs - Baseline scenario: Factor of 1.4x events for the same statistical power.
  - Generator community seems optimistic that this will be solved by the time of HL-LHC
    - A source of uncertainty for HEP but many recent breakthroughs

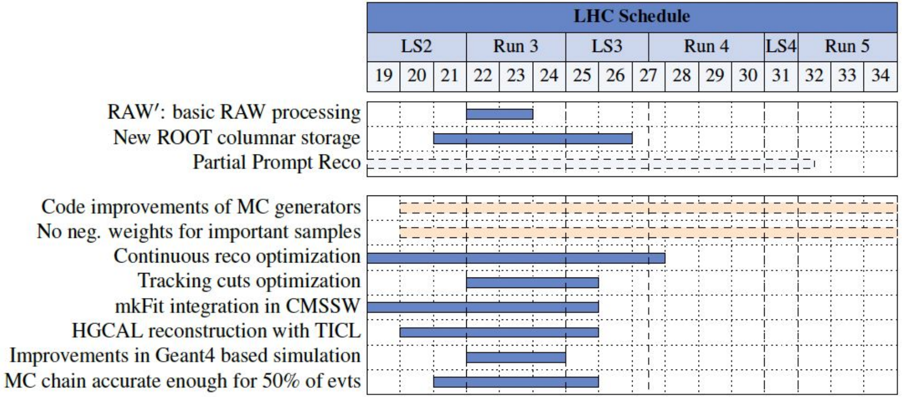
Summary of CPU and Storage Reductions

Short description	Impact per data tier RAW	AOD	MiniAOD	NanoAOD	Probability of Adoption
RAW′	-15%	-	-	-	medium
ROOT RNTuple	-	-20%	-20%	-20%	medium
Optimistic impact	-15%	-20%	-20%	-20%	
Weighted probable impact	-7.5%	-10%	-10%	-10%	

Short description	Impact per processing step Gen	Sim	Reco	Probability of Adoption
Generators code performance improvements	-10%	-	-	high
CMS and Geant4 improvements	-	-50%	-	high
Optimization of Reco code + Core libraries	-	-	-50%*	medium
Tracking cuts optimization	-	-	-30%	high
mkFit	-	-	-10%	high
Optimistic impact	-10%	-50%	-70%	
Weighted probable impact	-10%	-50%	-50%	

* -10% year-over-year improvement until Run 4.

Probability of Success:
High = 100%
Medium = 50%
Low = 20%



N.B. In the combinations, the (independent) reductions are multiplied and not summed.

CMS Computing Model: Resource Need Projections

Image: © CERN

Inputs to the CMS Computing Model

Common and CMS-specific inputs

LHC Run 4 and Run 5 parameters in input from the LHC Programme Coordination:

Yearly capacity evolution under flat budget scenario, starting from 2018 pledges:

CMS-specific parameters, also consistent with the DAQ & HLT TDR:

Parameter	**Run 4 ('27-'30)**	**Run 5 ('32-'34)**
	Common	
LHC Energy [TeV]	14	
Average PU	140	200
Integrated luminosity / year [fb^{-1}]	270 (135 in '27)	350
Livetime pp / year [$s/10^6$]	6 (3 in '27)	8
Livetime HI / year [$s/10^6$]	1.2	-
Yearly capacity evolution under flat budget for disk, CPU, and tape (hardware replacement included)	$+15 \pm 5\%$	
	CMS-Specific	
Prompt HLT Rate [kHz]	5	7.5
Collected events / year (10^9)	33	66
MC events / year (10^9)	79	100
CPU-GPU cost ratio per unit computation	2.8x	

Formula for the required number of simulation events from experience of previous LHC runs: **`9+N×0.2×L`**, where `L` is the recorded integrated luminosity in fb^{-1}, and `N=1.4` is a factor to take into account the effect of **negative weights** in event generation.

CMS R&D Scenarios

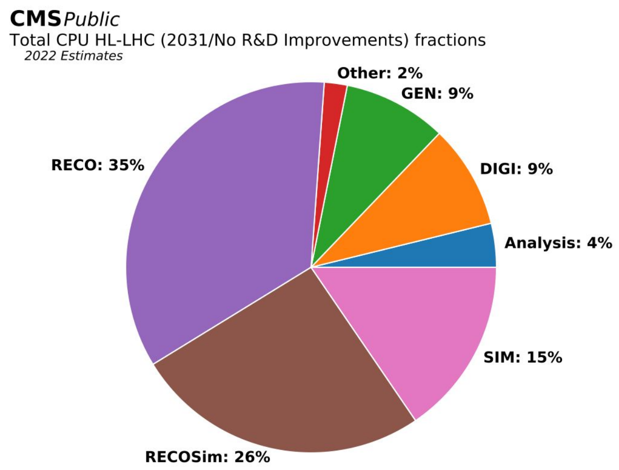
- Projected resources needs are formulated for two scenarios:
  - **Baseline:** Consider the computing model of today, no improvements from R&D incorporated, except that 50% of physics analyses use NanoAOD (today 30%).
  - **Weighted-Probable:** Incorporate into the computing model the improvements presented, weighted by their probability of success.
- **No consideration of improvement coming from GPUs incorporated in either scenario**
- **Partial prompt reconstruction is not considered in either scenario.**
- **Negative Weights problem assumed to be solved in the Weighted-Probable scenario only**

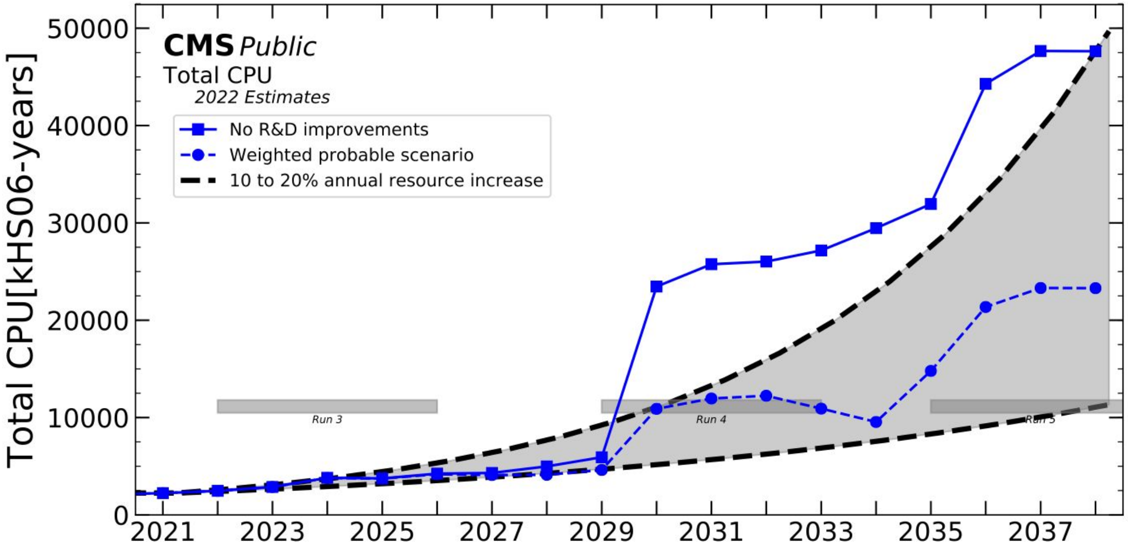
Short description	Variation wrt Baseline	
	Weighted Probable	Optimistic
Storage		
RAW event size	-7.5%	-15%
AOD/Mini/Nano size	-10%	-20%
CPU		
Fraction of Fast MC Chain events wrt to total number of MC events	10%	50%
Sim-Reco speedup of Fast MC Chain wrt regular processing	2x	10x
Reconstruction time per event	-50%	-70%
Simulation time per event	-50%	-50%
Generation time per event	-10%	-10%
Storage and CPU		
Additional MC events due to neg. weights	-100%	-100%

Resource Need Projections for HL-LHC: CPU

Main drivers: time-per-event of CMS processing steps, in particular reconstruction

- Tracking the largest consumer
- Minimum track p_T still to tune
- No consideration of GPUs





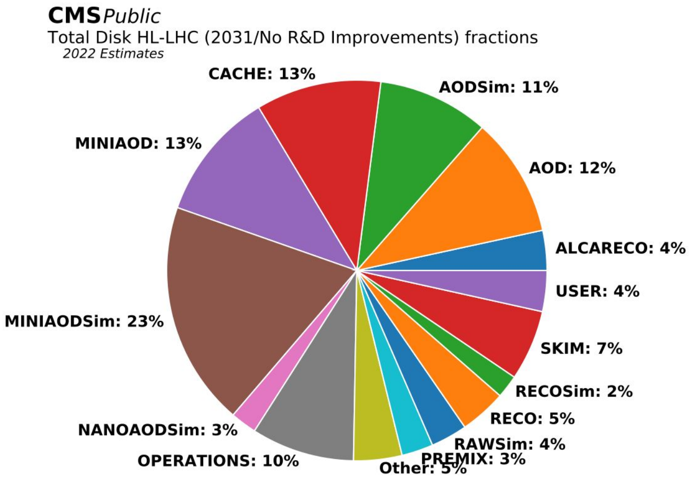
Weighted-probable scenario compatible with flat-budget capacity evolution.

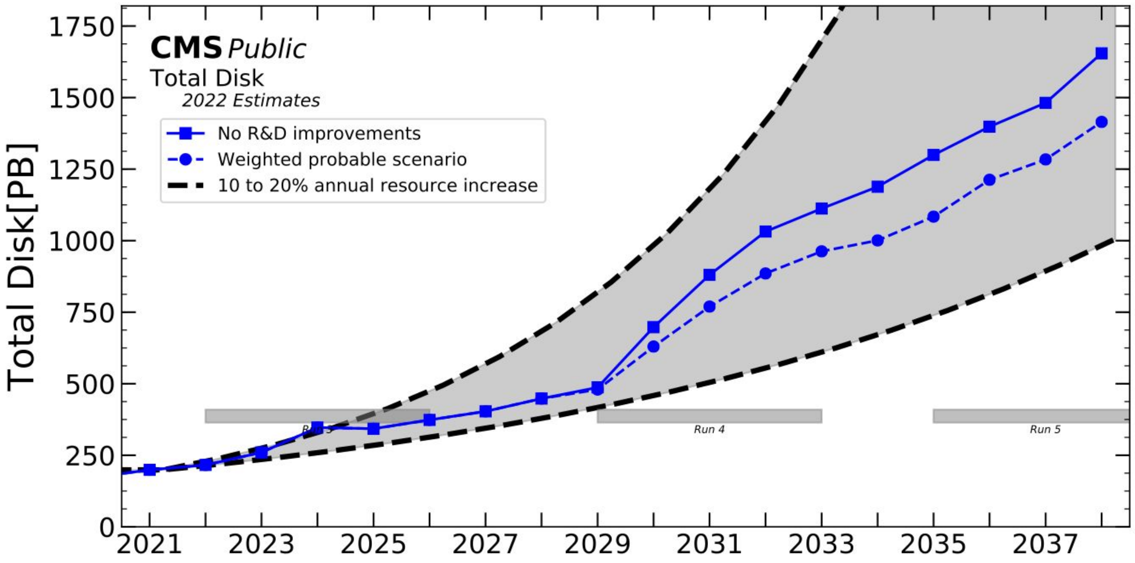
Motivated, curious physicists/developers wanted!

Resource Need Projections for HL-LHC: Disk

Monte Carlo samples are the top consumer of disk space.

N.B. Ratio of CPU/Disk is projected to be approximately the same in Run 4 as now: ~12 kHS06/PB



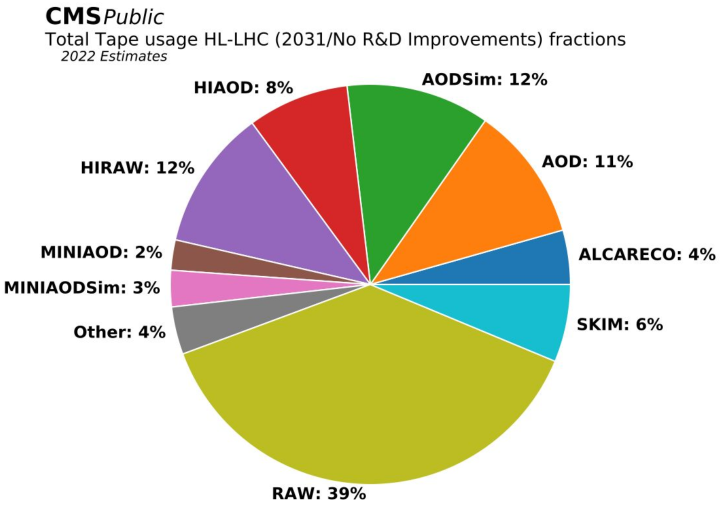


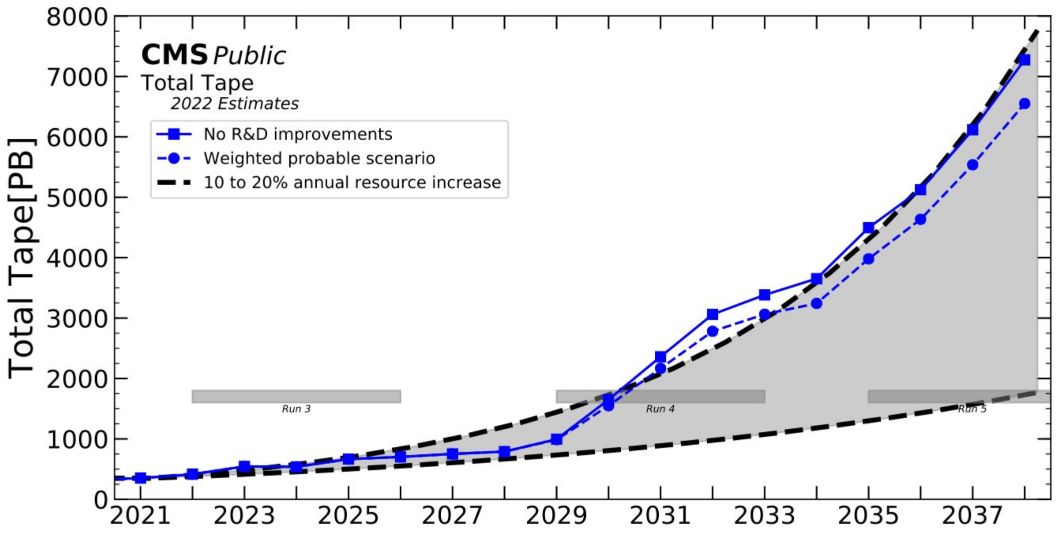
Baseline & Weighted-Probable Scenarios compatible with flat-budget capacity evolution.

Resource Need Projections for HL-LHC: Tape

Tape requirements are driven by the RAW event size and data volume.

- Critical to continue investigating the RAW event size, currently a **conservative estimate** to dimension the DAQ infrastructure.
- **Further reductions probable!**





Baseline & Weighted-Probable Scenarios close to flat-budget capacity, but not quite there

Uncertainties, Risks, and Future R&D

Many sources of uncertainty in making accurate predictions into the coming decade:

- **Flat-budget capacity evolution**: historical long-term trend is 15% year-over-year improvement - can be a threat or an opportunity.
  - Flexibility of using accelerators, HPCs mitigates this risk, for example
- LS3 / Run 4 schedule delays - one-sided opportunity for computing
- LHC outperforms and achieves full design luminosity earlier than foreseen
  - Risk mitigated by targeting R&D lines to handle PU=200 already in Run 4.
- Future availability of inexpensive tape as a medium for “cold” and archival storage
  - Common risk to all experiments and labs hosting tape libraries (especially CERN).

As mentioned earlier, there are still handles for further improvement outside of the R&D list in the documentation, e.g.:

- Adoption of NanoAOD flavors outside analysis, e.g. calibrations or generator studies (NanoAODGen)
- Fully exploiting ML, e.g. ML-driven creation of MC Nano datasets starting from NanoAODGen
- RAW event size reduction

Conclusions

Image: © CNAF

Conclusions and Summary

- We welcome Latvia in the family of CMS Tier-2 sites!
  - Expanding our presence in the Baltic region is an asset
- Many opportunities for collaboration, besides provision of resources. E.g.:
  - HPCs integration, detector sw for heterogeneous architectures, core sw, ML, innovative storage solutions, network management
  - Visibility on journals and at conferences
- Solid R&D plan ahead to face the HL-LHC challenge
  - Uncertainty on the level of resources that will be available
  - Need for motivated and creative physicists/developers/integrators