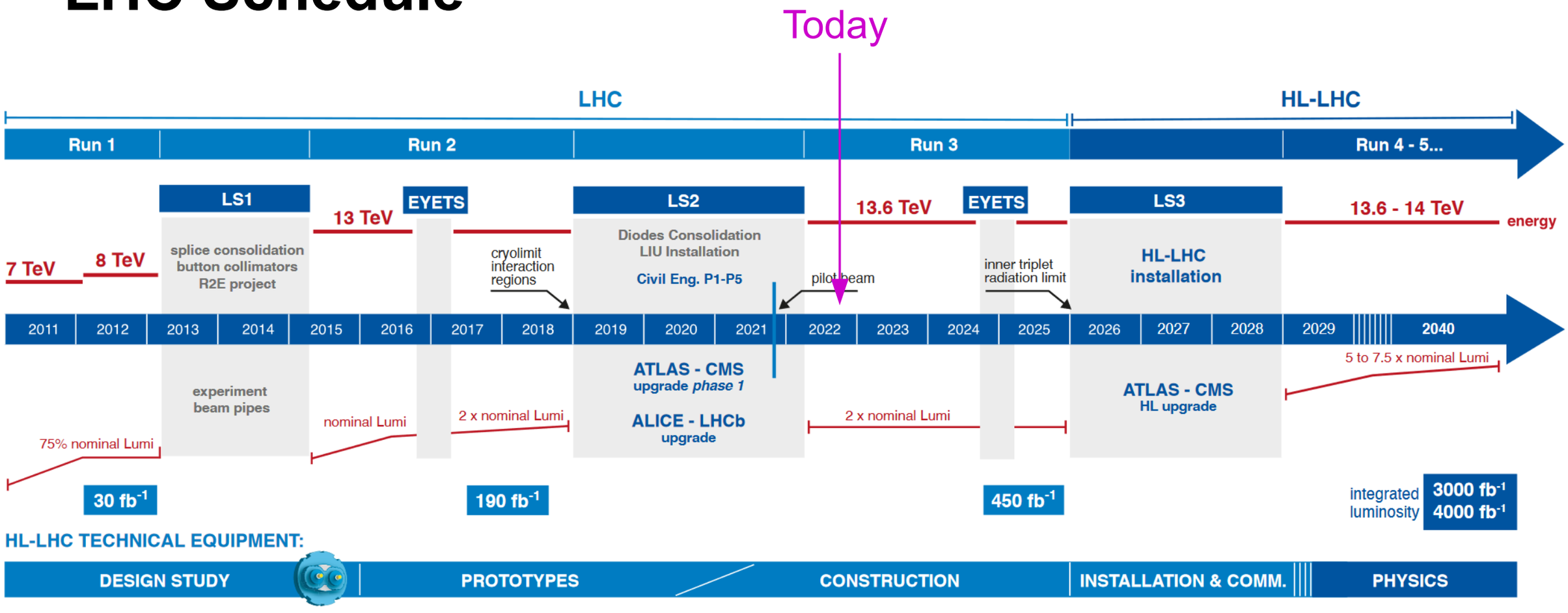
US ATLAS Overview
Plans and Resource Gaps for HL-LHC

Jana Schaarschmidt (Washington), Torre Wenaus (BNL),
Verena Martinez (UMass), Paolo Calafiura (LBNL)

IRIS HEP Retreat - 12. October 2022

Introduction

LHC Schedule



Run1: 25/fb usable data at 7-8 TeV
Run2: 140/fb usable data at 13 TeV
Run3: Expect to take ~250/fb of data at 13.6 TeV

} ~10% of the total LHC dataset

HL-LHC: 3-4/ab at 13.6-14 TeV

} ~90% of the total LHC dataset

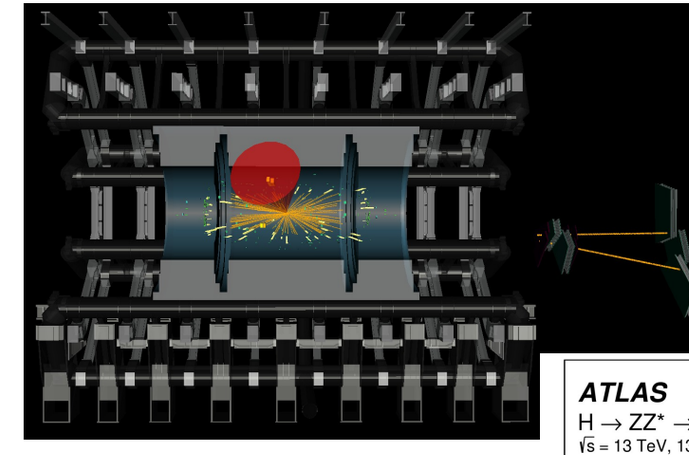
Computing Challenges for HL-LHC

Data processing challenges:

- 5-7x increase in luminosity (LHC upgrade)
- 4-5x increase in event size (new detectors)
- 10x increase in event rate (trigger upgrade)

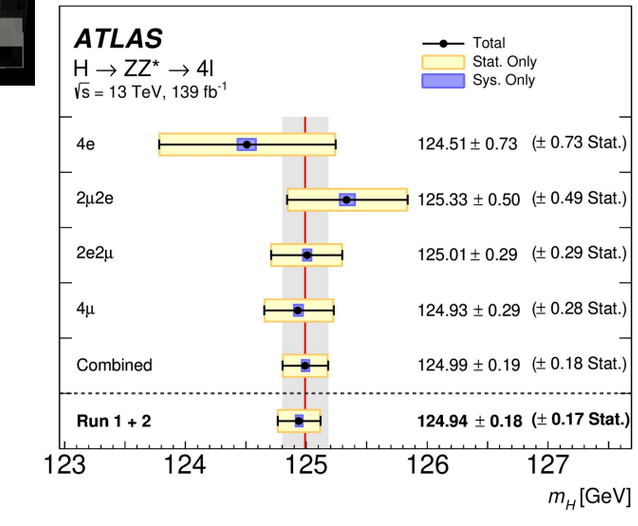
However, we don't expect comparable *funding upgrades* → resource gap!

	Run 4	Run 5
Integrated luminosity per year	< 270 fb^{-1}	< 350 fb^{-1}
Average pile-up	≤ 140	≤ 200
Proton collisions running time per year	6 x 10^6 seconds	8 x 10^6 seconds
Ion collisions running time per year (max.)	1.2 x 10^6 seconds	



Physics challenges:

- Higher precision measurements
- Explore unconventional (BSM) signatures
- More complicated/sophisticated analyses



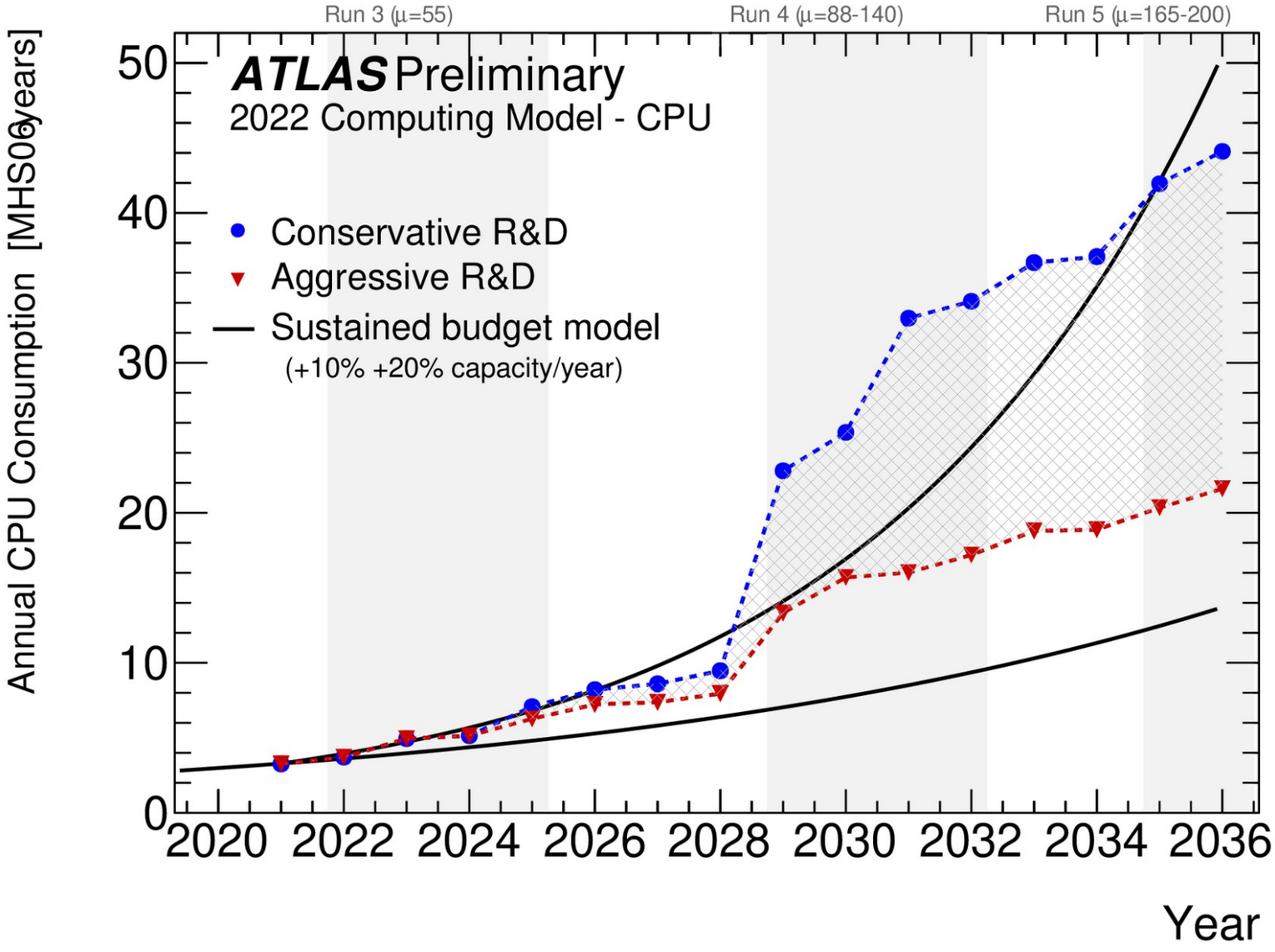
Opportunities:

- Open source software and public datasets
- Machine learning advances
- Evolving hardware, more concurrency, cloud resources, …

More information: HL-LHC CDR , HL-LHC Roadmap

Resource modelling

CPU Projections



Run4 requires aggressive R&D to stay within budget

Key elements:

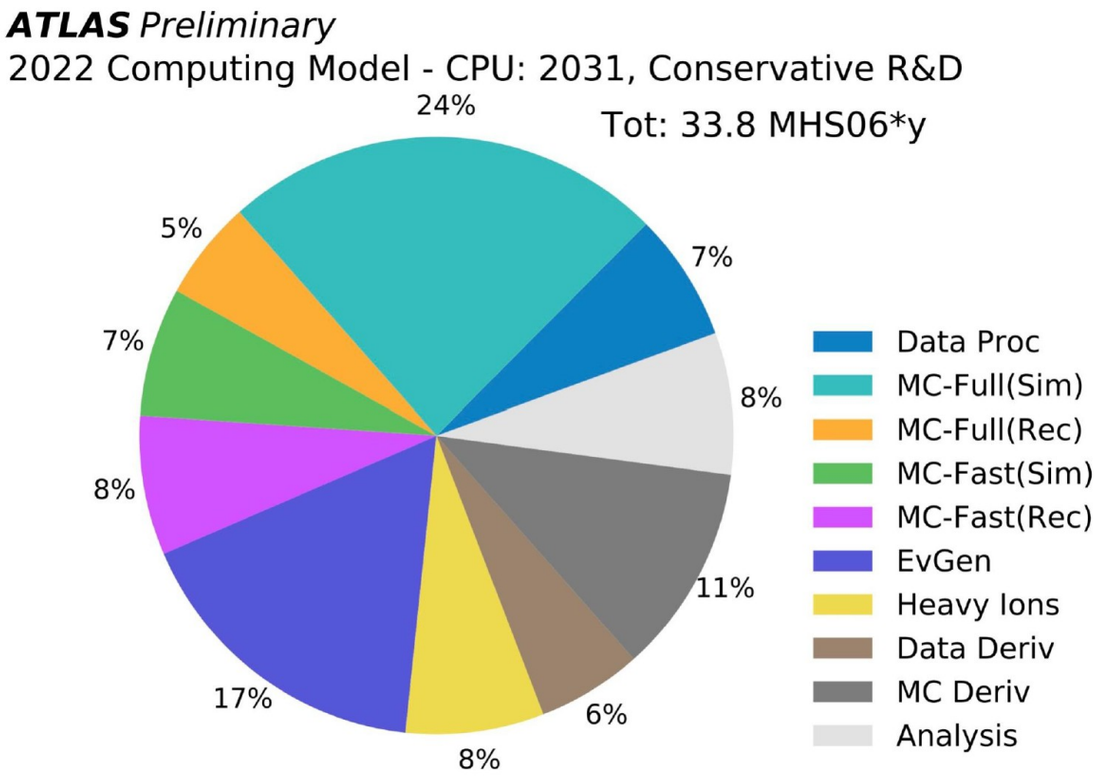
Fast simulation used for 70% (conservative) or 90% (aggressive) of MC

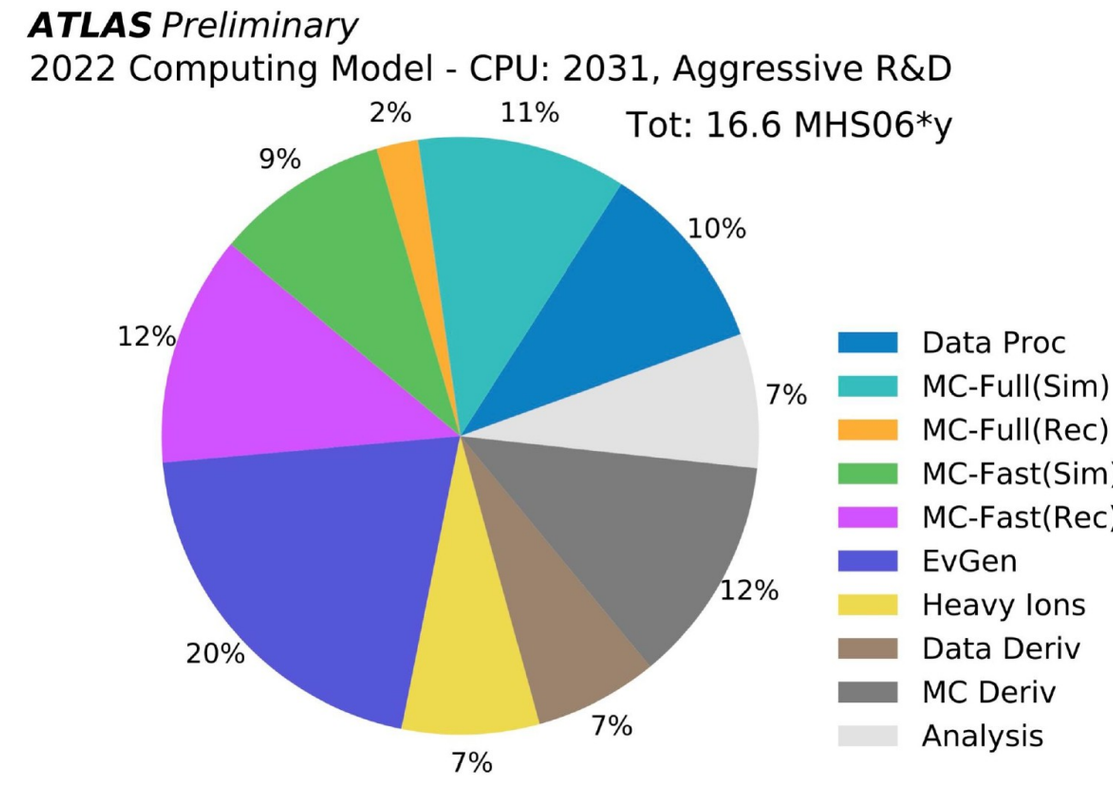
Faster event generation and reconstruction

„Aggressive“ also means new person power is needed!

GPUs not considered in this projection, but are expected to play an important role too. Impact of GPUs will be quantified in a later computing TDR.

CPU Projections





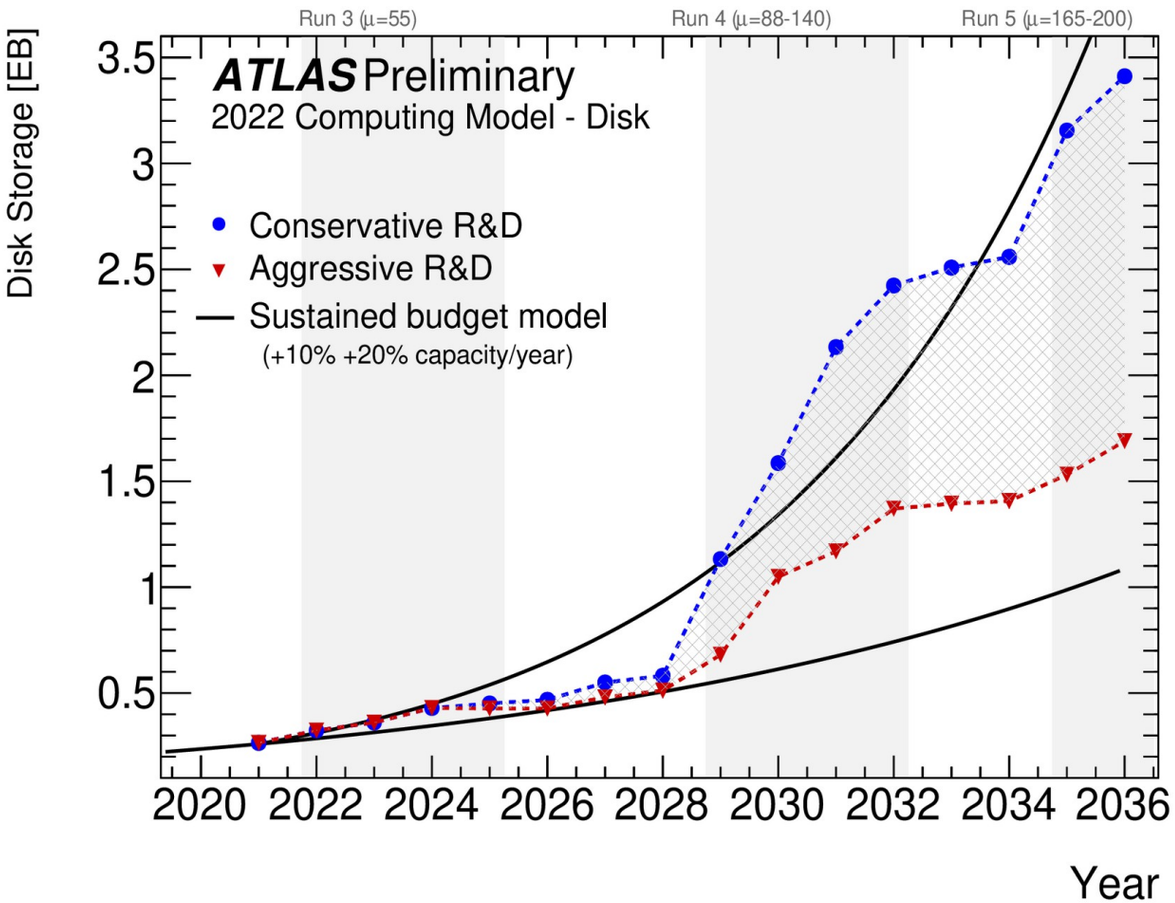
Good mix, but still dominated by FullSim fraction (24%) and EvGen production (17%)

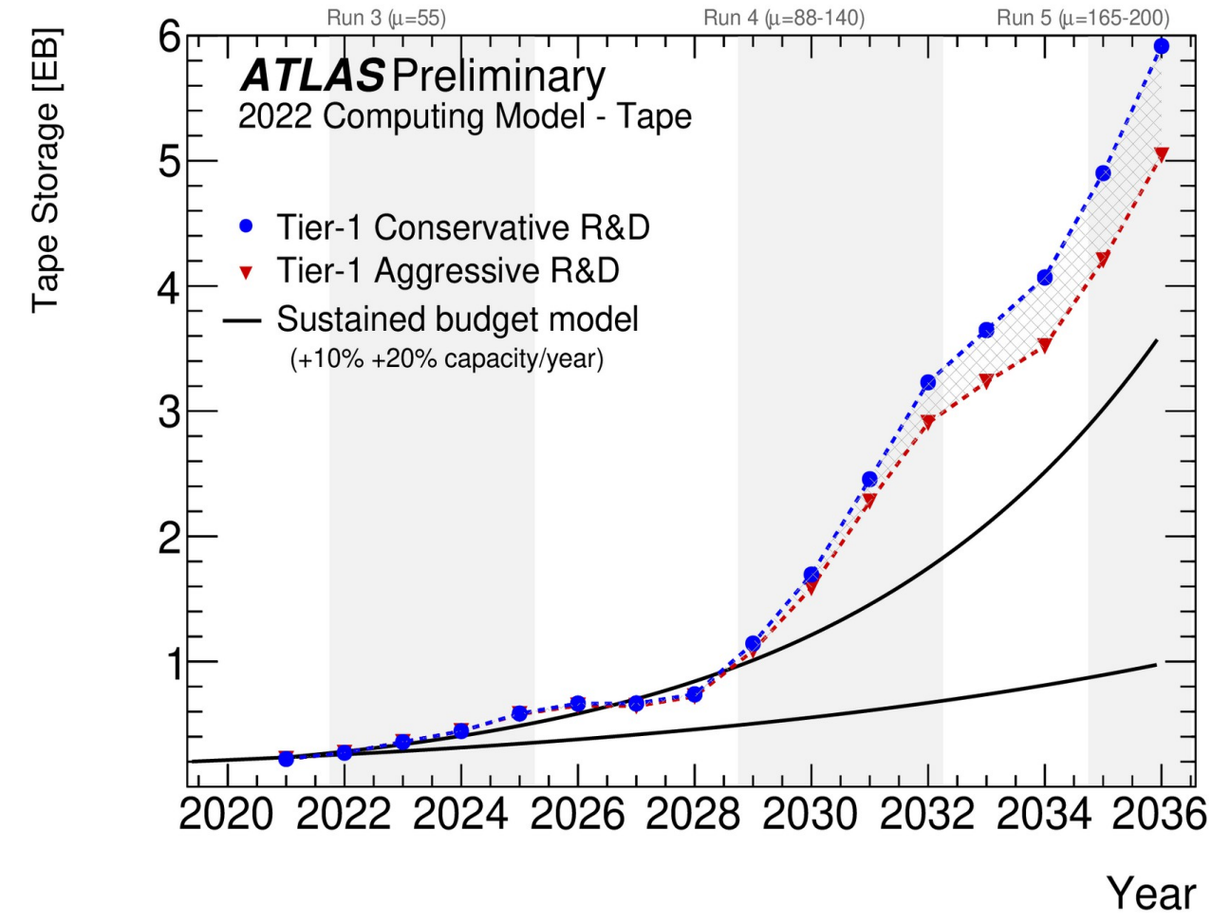
About a factor 2 overall speed-up

FullSim fraction significantly reduced

EvGen production now dominates the CPU

Disk and Tape Projections





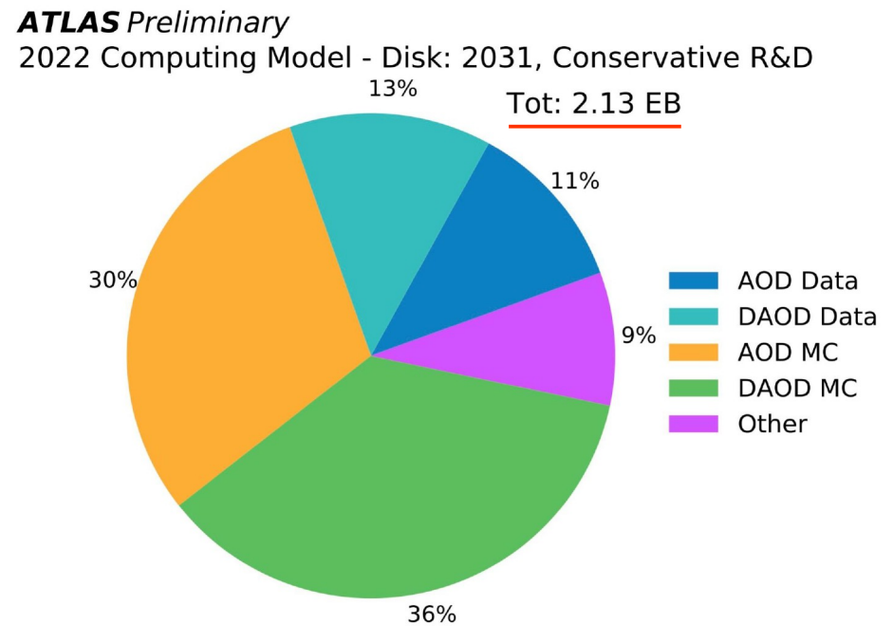
Tape is cheaper than disk, but data access is slower

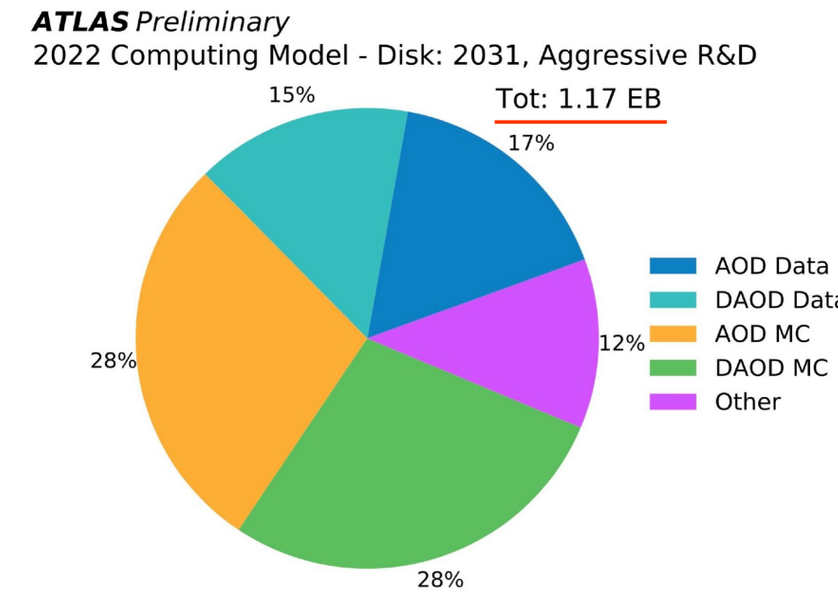
Tape storage budget has large uncertainties, but huge resouces needed for sure!
Ideas to reduce tape needs exist (eg. RAW compression, or not storing HITs), but impact unclear

Key elements: Multi-level data reduction resulting in small, commonly used analysis formats
„Aggressive“ reduction also through lossy compression

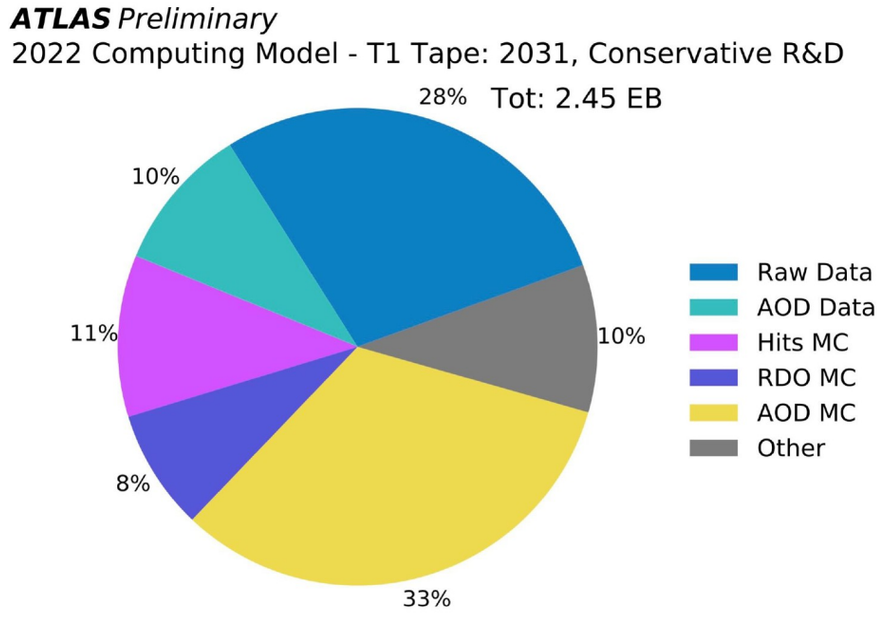
Disk and Tape Projections

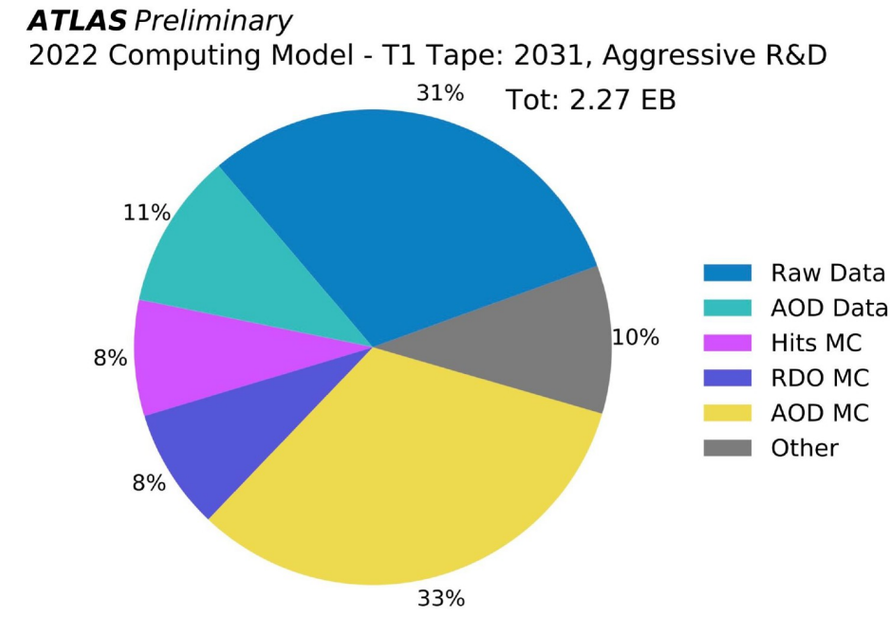
Disk





Tape



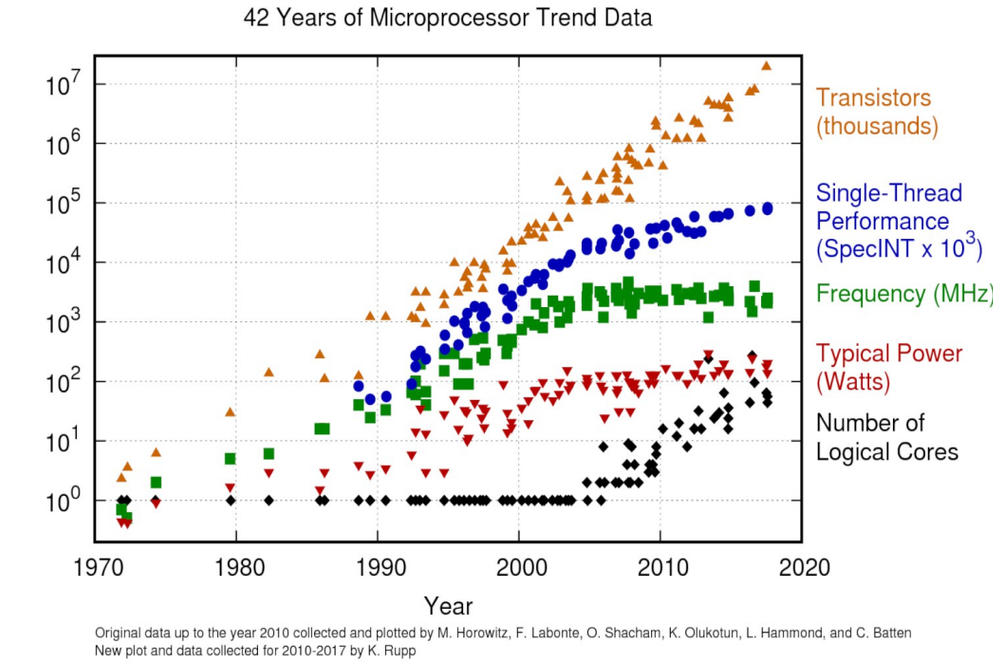


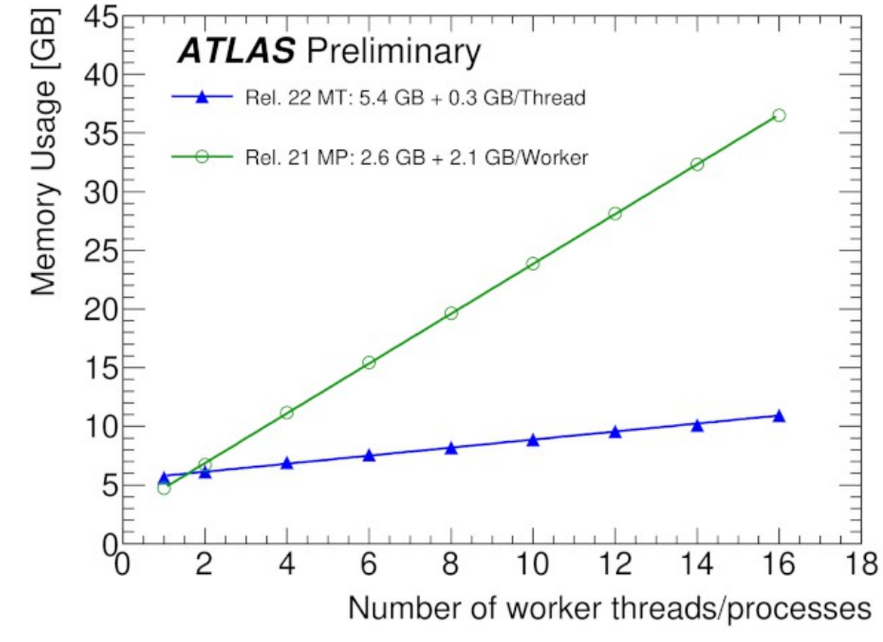
Fractions similar for both scenarios, but agressive R&D can reduce disk storage by almost factor 2

R&D Projects

Concurrency

- Trend in computer technology goes towards multi-cores, since clock speed has stalled due to thermal limitations
- Athena initially was not designed for multi-threading, but **AthenaMT** allows the parallel processing of data, while concurrently analysing multiple parts of an event at the same time (eg. tracking), made possible via careful algorithm scheduling
- Same event throughput with less memory achieved
- AthenaMT used for Run-3 already





Plans for Run-4:

- **Next-generation task scheduler** to maximize the event throughput (using threading model with fast context switches, task-based asynchronous programming model, computation offloading, distributed memory computing)

Databases

- ATLAS relies on **huge amounts of auxiliary data** (run configuration, calibration & alignment data, data quality, TDAQ, geometries, ...)

- Most of this data is organized using **Oracle**

- Work ongoing to consolidate Oracle database infrastructure,
  dependence on Oracle is to be reduced when possible, eg. by moving applications to CERN

- For run-3/4, migration of the **conditions database** from COOL to CREST ATL-SOFT-PROC-2021-023
  - based on REST api with JSON support, and with a HTTP query interface
  - relies more strongly on caches and requires less accesses from Athena grid jobs
  - access to conditions from CVMFS planned
  - Volume: 1-2 TB per data taking period

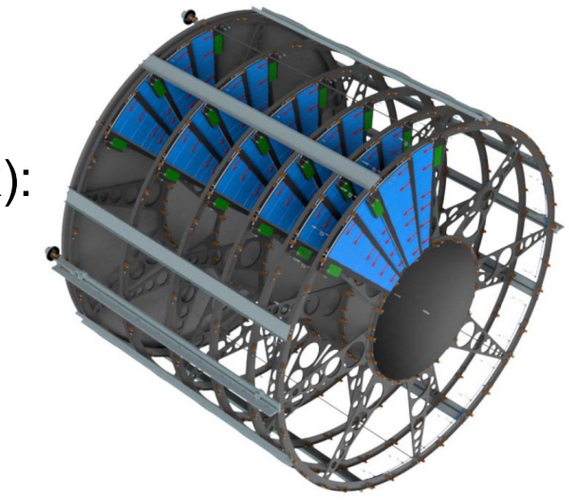
- **Other databases** need to evolve to be ready for run-4:
  - COMA (run database), AMI (ATLAS metadata interface), EventIndex
  - eg. reworking the tag system for AMI, and ensuring scalability of the EventIndex
  - EventIndex will increase from currently 35 TB to 1-2 PB in Run-4

Detector Description and GeoModel

- Current GeoModel service 20 years old, needs updates for run-4 detector design but also bring software improvements

- **Run-4 detector** will be significantly different from current ATLAS design (talk):
  - All-silicon Inner Tracker (Itk) will replace the ID
  - High-granularity timing detector (HGTD) for vertex timing information
  - Upgrades to muon system (NSW and barrel)

- Plans for HL-LHC are a **unified GeoModel of the whole detector** steered through XML-based database, for quick modifications



ITk strip endcap

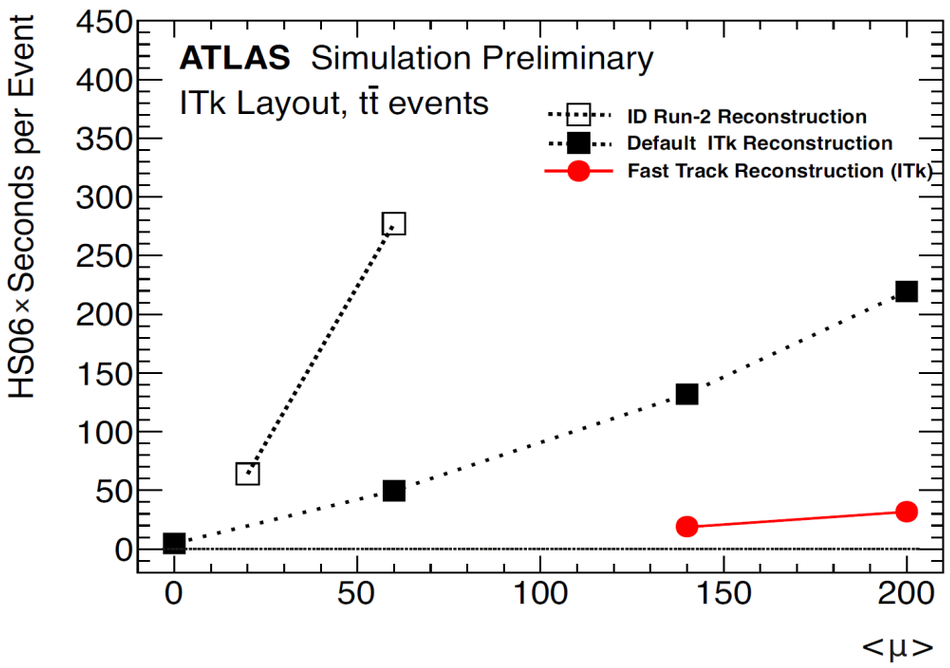
- Improved (3D) **visualization** system, for rapid feedback

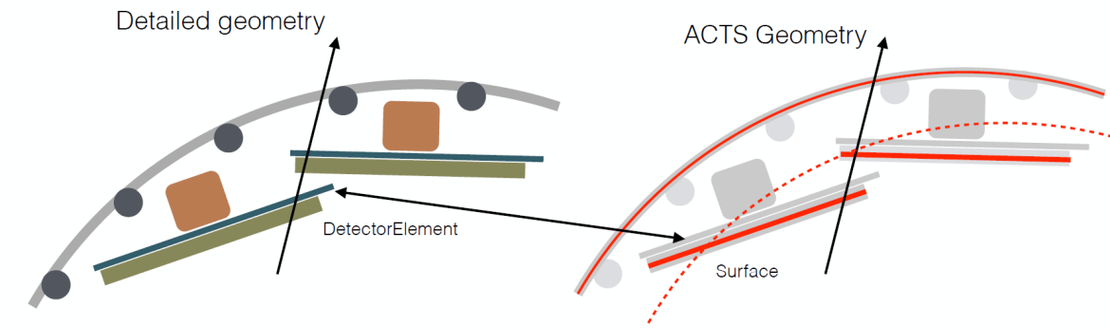
- **Validation** tools for detecting geometry clashes and anomalies, auto map generation, ...

- Will be integrated in Athena, strongly tied with the alignment system

Reconstruction and ACTS

- Object reconstruction **very challenging with on average 200 pile-up interactions** per event
- Phase-II tracker upgrade (ITk) designed to optimize performance and also minimize CPU needs
- **Fast track reconstruction** protoype in place that demonstrate large CPU gains with small performance losses ATL-PHYS-PUB-2019-041
- **ACTS** (A Common Track Reconstruction Software): Open source project to achieve CPU reduction and great performance for tracking and also particle flow, experiment-independent, supports MT. IRIS contributions crucial for the success of ACTS!
- **traccc:** Tracking chain on accelerators demonstrator





CPU needed to reconstruct tt̄ MC events:

$\langle\mu\rangle$	Tracking	Byte Stream Decoding	Cluster Finding	Space Points	Si Track Finding	Ambiguity Resolution	Total ITk
140	Run 2	$1.2^{(*)}$	17.1	6.0	41.1	58.2	124
140	fast	$1.2^{(*)}$	4.5	0.9	12.4	-	19.0
200	Run 2	$1.6^{(*)}$	26.3	8.6	85.8	92.0	214
200	fast	$1.6^{(*)}$	6.3	1.2	22.6	-	31.7

Fast Event Generation

- **MC event generation takes a huge fraction of ATLAS CPU resources**
- For High-Lumi, expect this fraction to increase because of higher statistics (more MC than data needed for most cases)
- **Theory advances** lead to more precise MC including higher orders, and often involve negative weights
- **Ongoing R&D:**
  - Optimisation of settings and choices for the generators and efficient event filtering
  - Weights to emulate different scales or PDFs etc. instead of regenerating samples, this can mean multi-dimensional reweighting methods using ML
  - Sharing evgens with CMS (problem: statistical correlations across otherwise independent experiments)
  - Usage of HPCs for high multiplicity events
  - New generators optimised to utilise GPUs (eg. **BlockGen**)

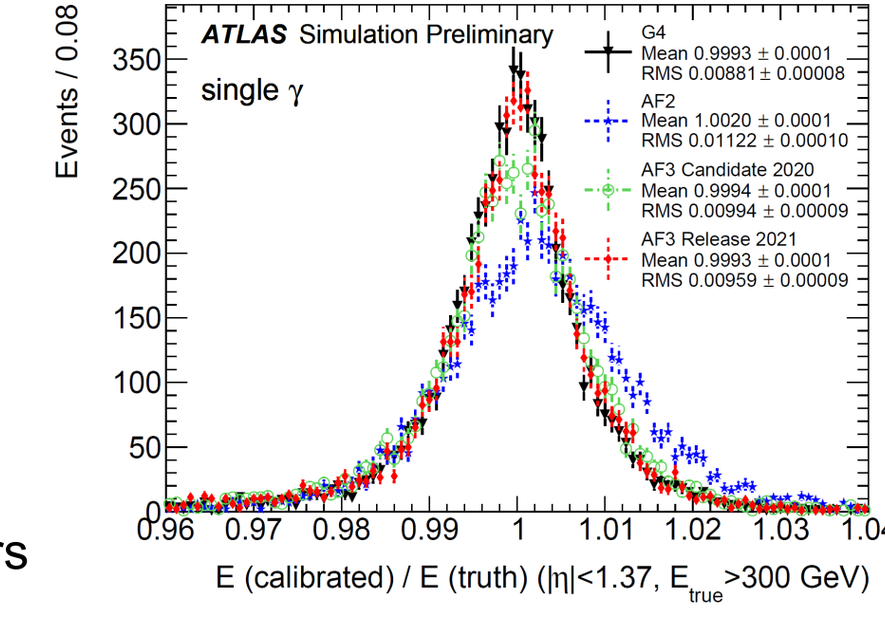
Fast Calorimeter Simulation (AF3)

- Fast Calorimeter Simulation (FastCaloSim) has a long tradition in ATLAS, an earlier version (AF2) used since run-1 for about 50% of all MC. It's **well motivated** since calorimeter simulation is most time-consuming part of full simulation (~80%)

- A year ago **AF3** was released, trained on latest Geant4 samples and using modern techniques, eg.
  - PCA for the longitudinal energy parametrization and modelling of layer-by-layer correlations
  - Lateral energy simulated including fluctuations which enables sub-structure modelling
  - GANs used for hadronic shower modelling in medium energy ranges

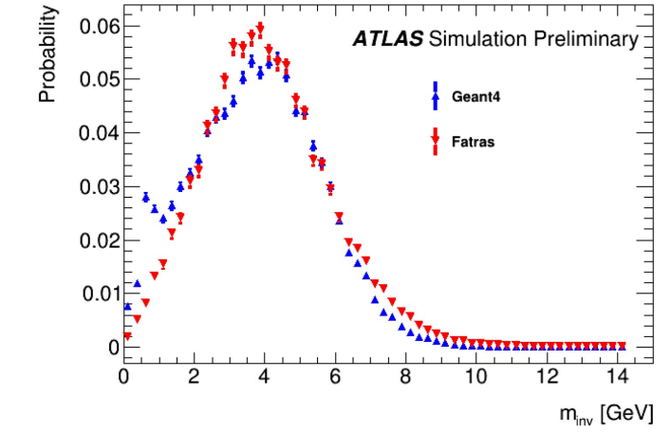
- AF3 is a paramount component for run-3 and beyond

- **Ongoing R&D:**
  - Retraining using latest Geant4 version
  - Small corrections for EM showers to increase precision
  - Voxelization optimisations for FastCaloGAN
  - Improved fluctuations modelling, eg. by using auto encoders
  - Porting to GPUs



Fast Chain

- FastChain means that many or all components of the MC production chain are replaced by faster tools
- Ideally it also is a way to generate samples in one step, ie. Event generation → reconstruction without the intermediate formats (ie. **FastChain transform**)



- **Components:**

  Event generation → Fast event generation

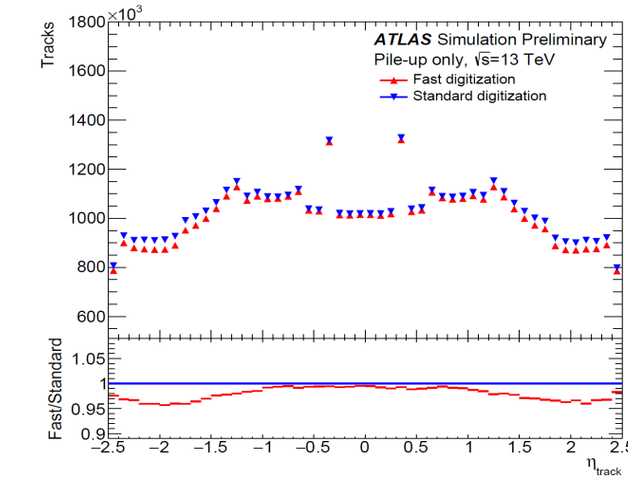
  Inner detector simulation → FATRAS (integrated into ACTS), could also be used for muon tracking

  Calorimeter simulation → FastCaloSim/FastCaloGAN

  Digitisation → Fast digitisation

  Reconstruction → Fast reconstruction

  Pile-up overlay → Track overlay (see next slide)



- Several configurations of fast+standard tools are discussed for HL-LHC
- **Full simulation can also be improved**, eg. by optimisations (such to reduce steps, done for run-3) or by exploiting parallelism (GeantV)

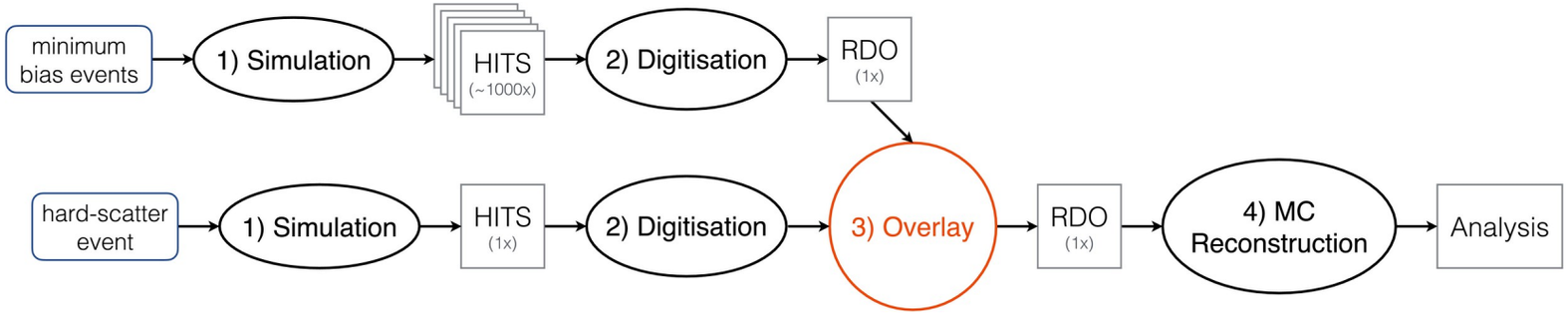
Pile Up Overlay

- **Pile-up effects** are emulated by simulating minimum bias samples, that are then merged with the hard scatter event. The digitisation happens for the combined PU+hard scatter event → Not efficient, digitisation is repeated over and over for the same PU events

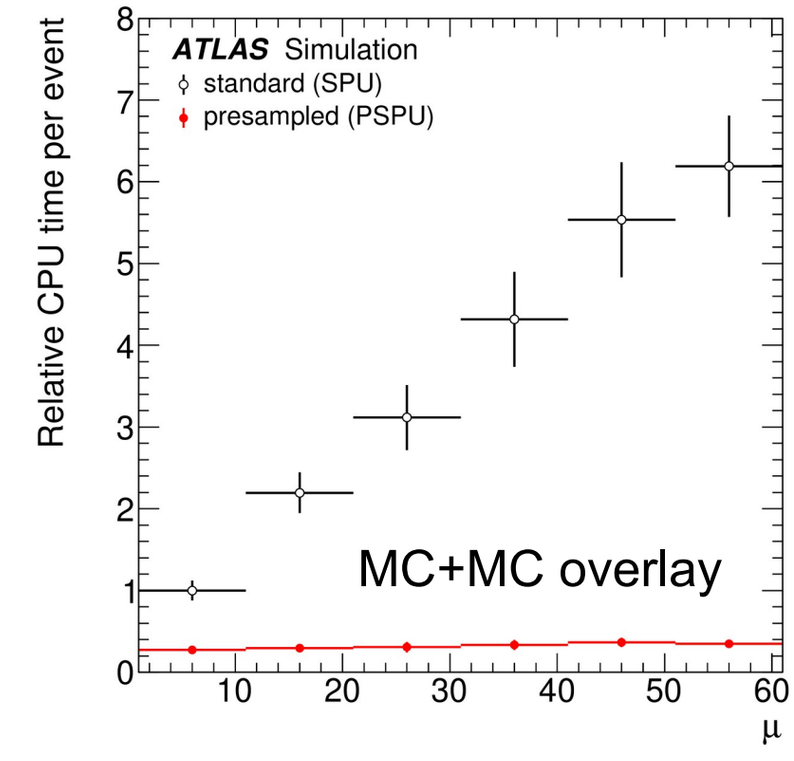
- **MC+MC Overlay** means that PU samples are digitised separately and then overlayed onto the hard scatter event. This avoids repeated digitisation of PU samples. In use for run-3.

- Could also use real **data** from zero-bias trigger stream for overlay

- For run-4, new idea is **track overlay:**
  - Reconstruction is done separately for the PU samples
  - Reconstructed tracks are merged with the track collection of the hard scatter
  - Avoids the repeated reconstruction of PU samples
  - Works well for topologies where hard scatter track reconstruction is not affected much by PU



SIMU-2020-01

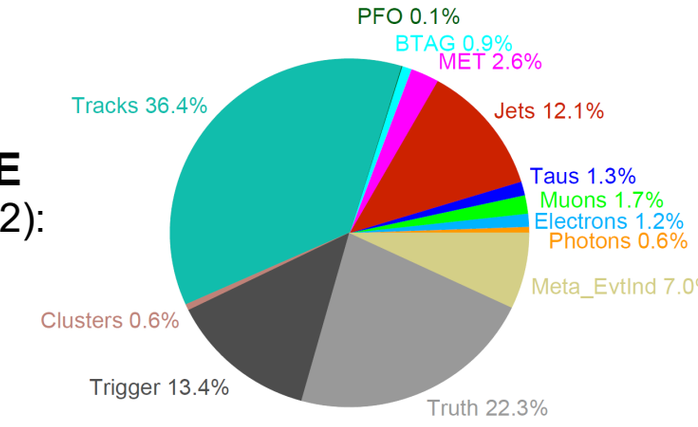


Analysis Model & PHYSLITE

DAOD_PHYSLITE is the future common derivation format, unskimmed, user-friendly, **containing already-calibrated objects for fast analysis**, to be analysed directly. A run-3 version is already is use.
Size/event: ~10 kB. To be used by ~80% of all physics analysis.
Total PHYSLITE size: ~1 PB / year

Special formats still needed for performance studies, non-standard analyses, B-physics etc.

PHYSLITE
$t\bar{t}$ MC (09/22):



Run-4 plans for PHYSLITE:

- **RdataFrame and RNtuple** (replacing TTree) and Root 7
- **Columnar analysis** (requires adapting CP tools and rethinking some analysis methods)
  Lots of amazing tools available, need to focus on integrating them into ATLAS analysis model
- **Parametrized and/or simplified systematics**
  Fast on-the-fly evaluation of variations, with no/little dependence on variables except the object 4-vector
- **Augmenting reduced formats for subset of events** (eg. friend trees)
  → to make common formats useable for non-standard analyses
- **Lossy compression** (storing floats with reduced precision) → can significantly reduce the file size

Vision: On-demand sample request, to avoid long-time storage in general. Relies on FastChain.

Distributed Computing

- Data processing strongly relies on distributed resources, WLCG operation will continue during HL-LHC
- ATLAS distributed computing infrastructure comprises eg. ProdSys, PanDa, Rucio, HammerCloud, … but resources need to evolve for HL-LHC (keywords: scalability, efficiency)
- **Some ongoing developments:**
  - Data Lake (concentration of disk resources at fewer but larger sites, and more caching)
  - Token-based authentification (replacing the current X.509 certificates)
  - HTTP-based protocol replacing GridFTP
  - Switch to new operating system (replacing CentOS 7)
  - Increase network bandwidth (also perform stress-tests to emulate HL-LHC conditions)
  - Integrate new resources, such as HPCs or cloud infrastructure

Data Carousel and iDDS

- „Sliding window approach“ to steer data processing, workload management and storage services with the aim to store data on lowest-cost media whenever possible (ie. tape) and make it accessable when needed

- Data processing is executed by staging and promptly processing slices of inputs onto faster storage, such that only the minimum required input data are available at any time

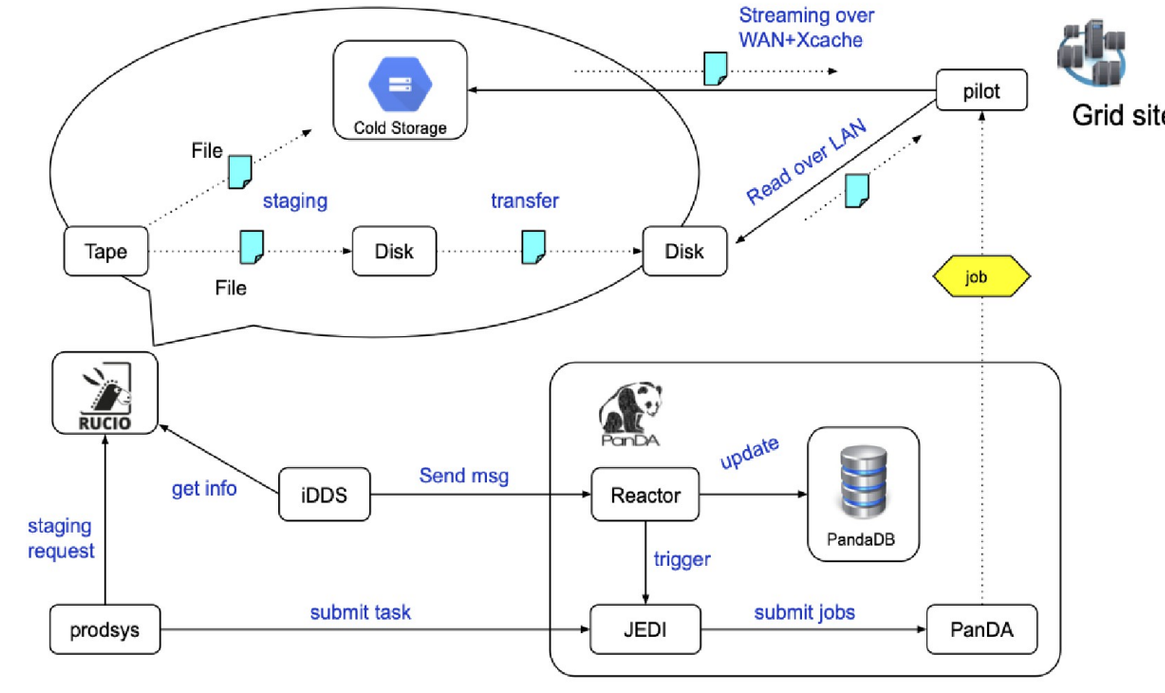
- Data carousel is introduced already for run-3 (ATL-SOFT-PROC-2021-012) for AOD storage

- **Plans for HL-LHC:**

  - Reconstructed data stored primarily on tape (with only very partial disk replicas), processed with the Data Carousel

  - Intelligent Data Delivery Service (iDDS) developed as part of PanDa to provide streaming services for data delivery for a wide range of workflows

    IRIS contributions are crucial for the success of iDDS.
    More details on iDDS and applications in the backup!



Trigger, DAQ and T0 Traffic

- TDAQ phase-II upgrade: https://cds.cern.ch/record/2285584
- Event filter (EF) farm to be built from CPUs with the option of additional accelerators (GPU, FPGA), can be used as a regular grid site during non-data taking periods
- Average EF output 10 kHz, but special **TLA (trigger-object level analysis) streams** have a higher rate
- Total bandwidth to Tier-0 expected to be ~45 GB/s
- „Spill-over“ workflow planned for run4 to redirect prompt processing to the grid in case of long LHC runs
- Projected T0 network traffic for Run-4:

Activity	EOS Read	EOS Write	CTA Read	CTA Write
DAQ: SFO → EOS	–	53 GB/s	–	–
Tier-0: Reconstruction	31 GB/s	10 GB/s	–	–
Tier-0: Merging	10 GB/s	10 GB/s	–	–
DDM: Export to Tier-1s	63 GB/s	–	–	–
DDM: EOS → CTA/EOS	63 GB/s	–	–	63 GB/s
CTA: Tape Backup	–	–	63 GB/s	63 GB/s
Total	**167 GB/s**	**73 GB/s**	**63 GB/s**	**(63 + 63) GB/s**

- Projected T0 CPU capacity:

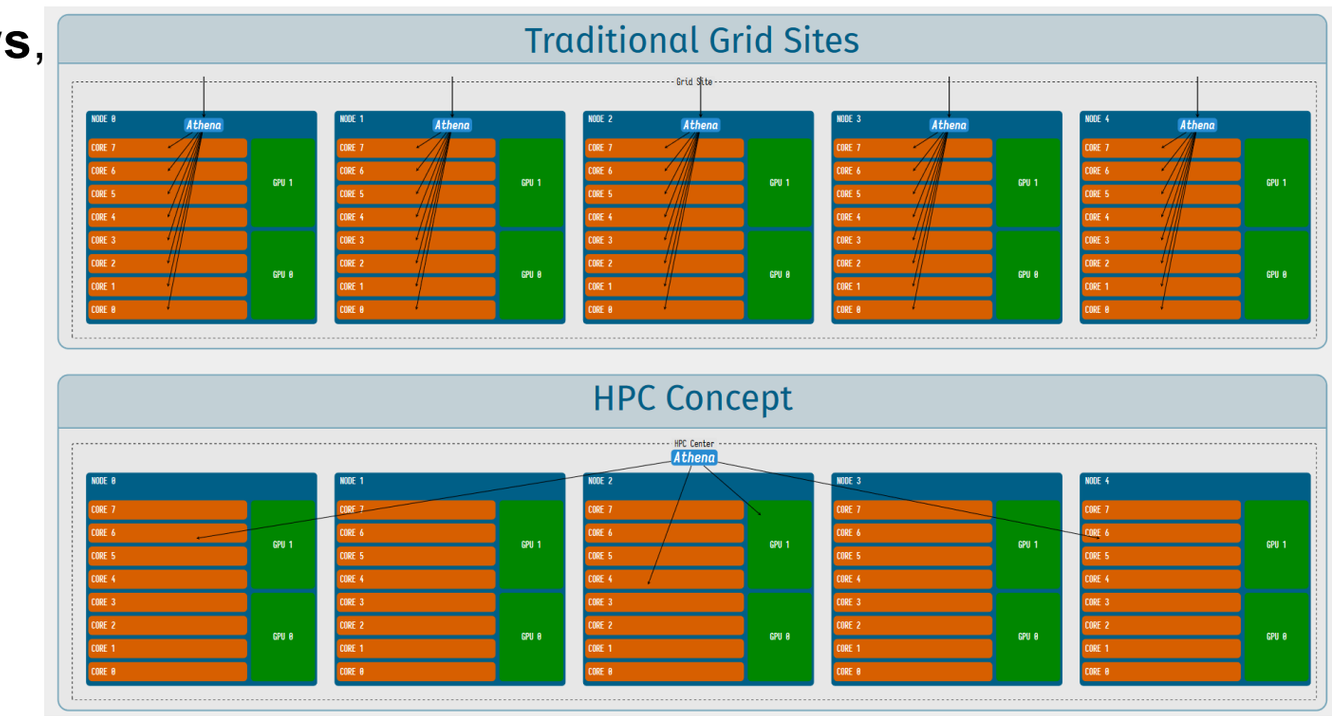
Pile-up	Reco. Time/Event	Required CPU Capacity	CPU Capacity wrt. Run 2
$\langle\mu\rangle = 140$	402 (215) HS06·s	3500 (1900) kHS06	8.2 (4.4)
$\langle\mu\rangle = 200$	584 (295) HS06·s	5100 (2600) kHS06	11.9 (6.0)

Infrastructure: Accelerators

- Accelerators are a relatively new resource that is **not yet widely used, eg. in production workflows**
- To understand how to efficiently use accelerators, dedicated forum in ATLAS
  HCAF (Heterogeneous Computing and Accelerator Forum)
- Interesting: **CPU/GPU hybrid workflows**
- **Ongoing R&D:**
  - GPU kernel scheduling for Gaudi
  - Integrating GPU hardware support in Athena
  - Development of heterogeneous applications to use as benchmark for core software,
    eg. pattern recognition algorithms that run on CPUs and GPUs
  - Prototyping accelerator-friendly event data model and geometry data model
  - Evaluate gains in GPU efficiency by batching (accumulate data from many events and offload them to GPU in a batch)
  - Decide on technology for algorithm parallelization (CUDA, Kokkos, Alpaka, ...)

Infrastructure: HPCs

- HPCs play already a role for grid computing and are **expected to become more important**
- Often HPC allocations are inconsistent with WLCG policies and operations (eg. HPCs can have more restrictive network access, or designed to run different payloads, or missing CVMFS support or database access)
- To overcome this, can use **interfaces** such as Harvester for US HPC sites
- ATLAS workflows and data management system need to be adapted to serve a variety of HPC platforms
- Possible approach: **containerised workflows**, that pack all necessary software into a container that is deployed at the HPC site
- Heterogeneous architectures require **new schedulers** that can run algorithms on the most appropriate resources and balance loads. Projects: Raythena, HPX



(Illustration: B. Stanislaus)

Infrastructure: Analysis Facilities

- **More complex physics analyses** based on modern techniques require dedicated computing resources, such as training complex machine learning algorithms,
  or exploiting python-based ecosystems for vectorized computations, visualizations etc.

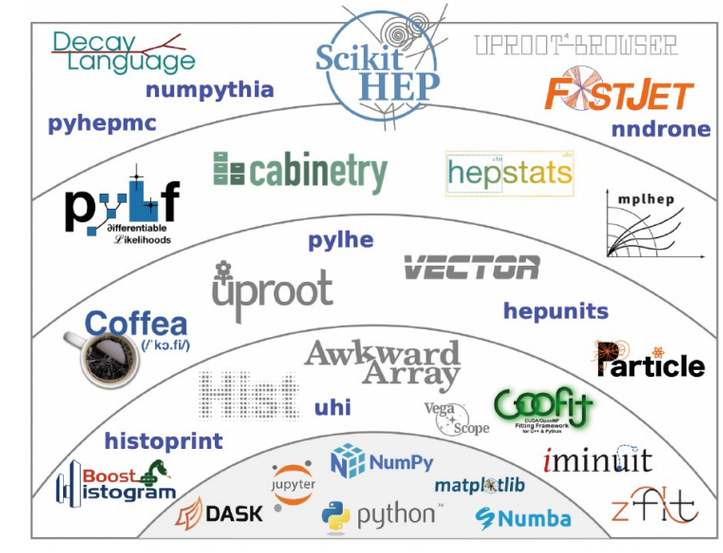
- Distributed computing system needs to evolve to support these workflows for **end-user analysis**

- R&D projects ongoing to identify new and useful tools that support these analysis workflows, also since advances in big data sometimes come from outside of HEP.
  Examples: scalable Jupyter notebooks, new data delivery services

- Analysis facilities are a new form of grid resource, focus on **interactivity, usability, user-support,** with federated access, that also store the data and MC samples needed for these studies
  (this becomes especially feasible when data/MC is stored in form of PHYSLITE)

- Deployed in various forms, for example integrated in large computing centres, as HPCs, or on the cloud

- Also can provide GPUs and specialised software for dedicated ML workflows



Summary

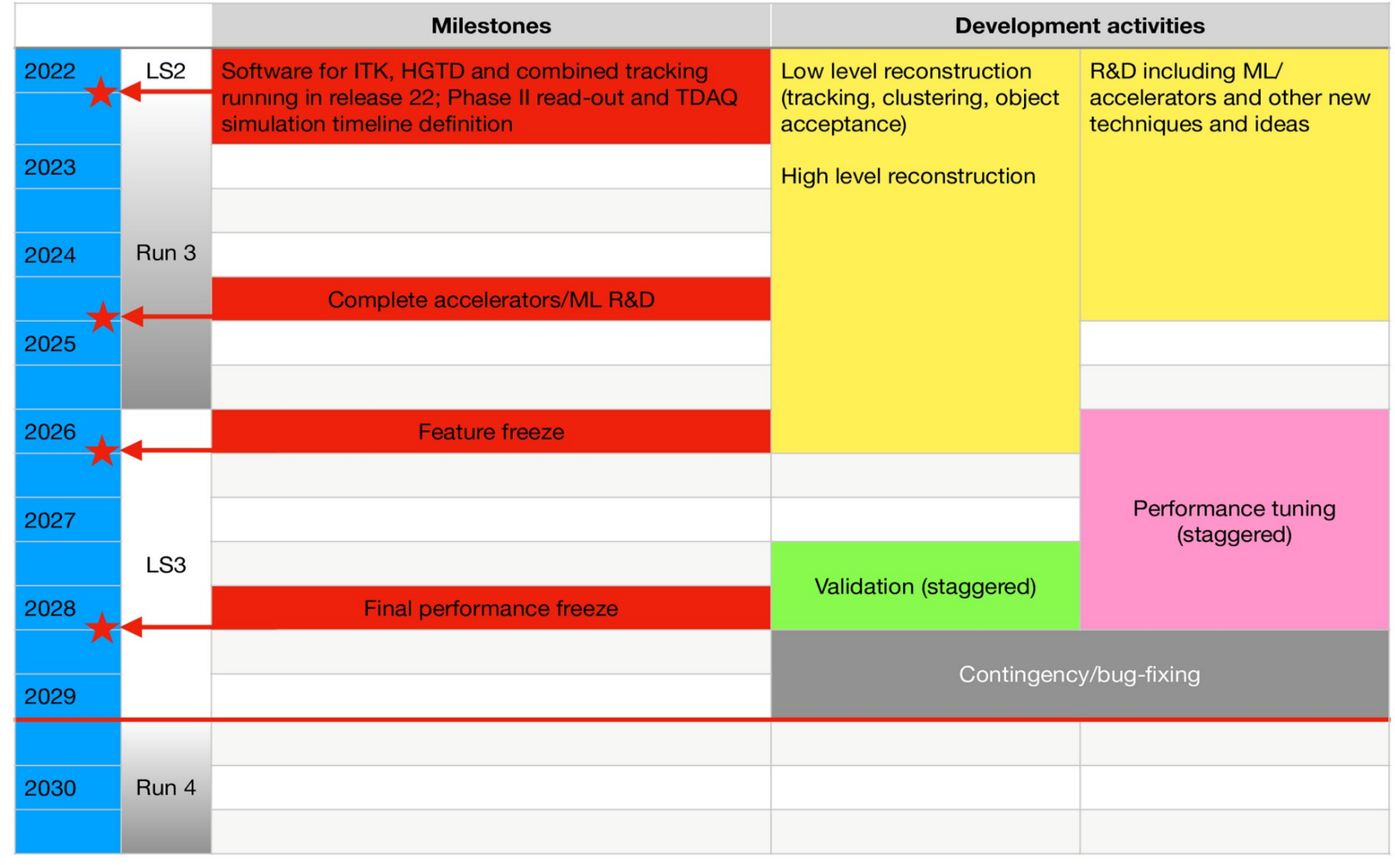
HL-LHC brings unprecedented physics opportunities and computing challenges

Critical for ATLAS physics program that software and computing is ready, to:

- process incoming data
- store it with sufficient redundancy
- produce enough MC
- provide efficient analysis infrastructure

All components need to be ready by 2028 to allow for one year of validation, bug-fixes and contingency

Many R&D projects ongoing, impact of those will become clear only in a few years.



Backup

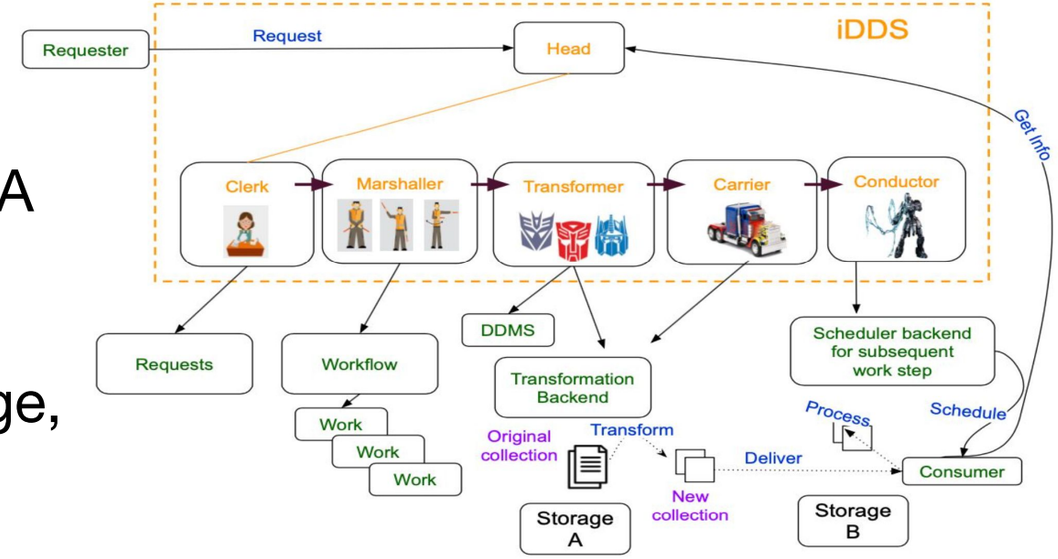
Intelligent Data Delivery Service (iDDS)

iDDS is an experiment-agnostic add-on to PanDA or other workload manager supporting granular data delivery and **orchestration of complex workflows** that are efficient in their use of storage, network and processing resources

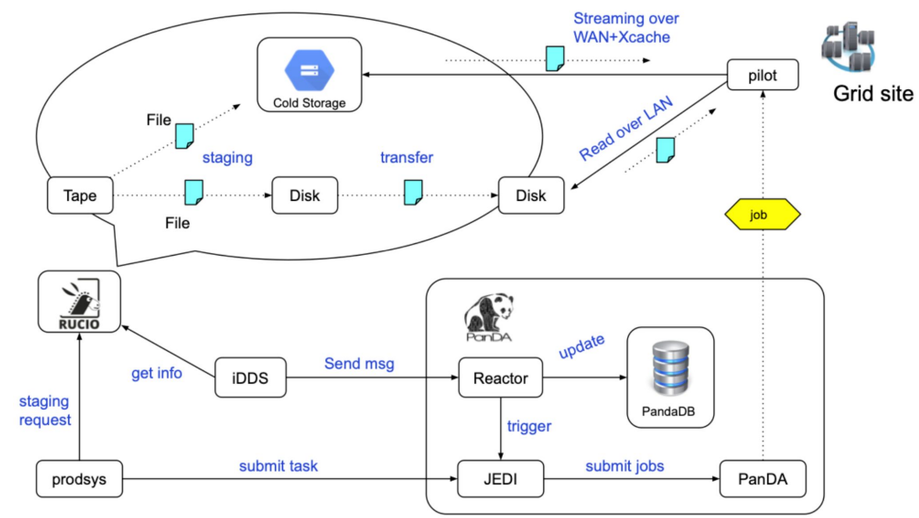
- A joint project with IRIS-HEP, project hosted by HSF
- Used by ATLAS, Rubin, sPHENIX

Used in a growing list of applications important for HL-LHC readiness and serving/scaling analysis

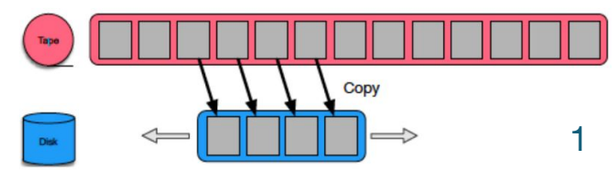
- **ATLAS Data Carousel** processes tape-resident data using a small disk storage footprint via a sliding window orchestrated by PanDA, iDDS and Rucio
  - In production for almost 2 years, reducing the storage needs of analysis object data, the dominant storage load for HL-LHC
  - Ongoing R&D to reduce the footprint and improve performance
- **Highly scalable ML services**
  - Enable analysts to run processing-intensive AI/ML applications on large scale, geographically distributed, heterogeneous resources
  - Shorten optimization and training latencies by orders of magnitude
- **Active learning services** drawing on the ML work



Intelligent Data Delivery Service (iDDS)

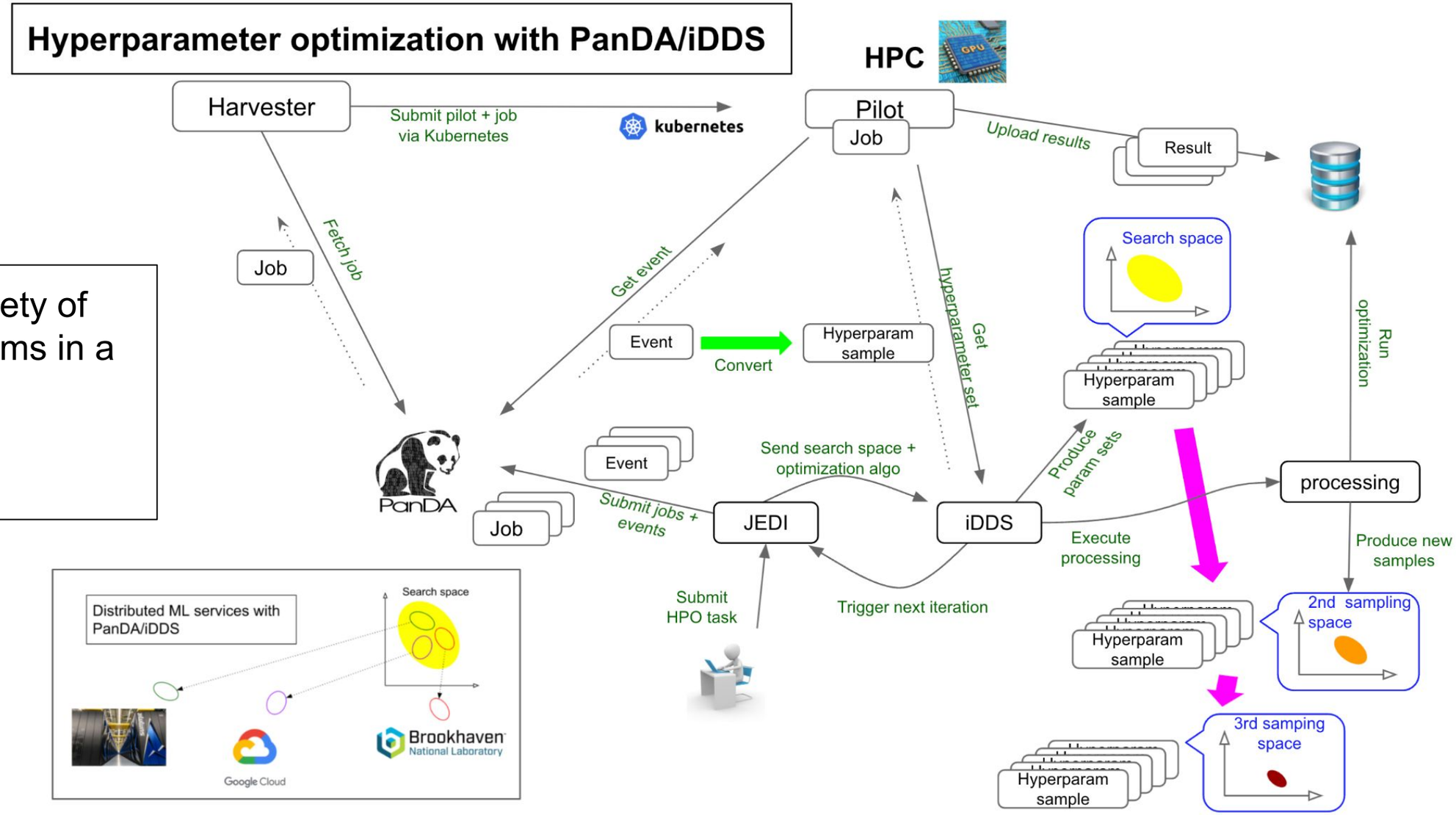


ATLAS Data Carousel using iDDS



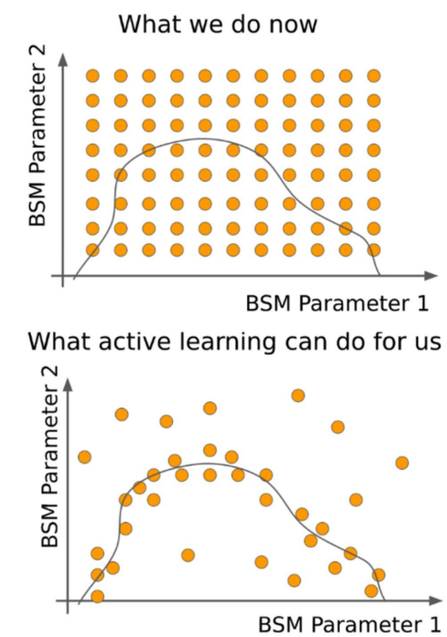
Distributed, scalable ML services

Leveraging a variety of distributed platforms in a coherent way for intensive AI/ML processing



Applying ML services via active learning in analysis

- The hyperparameter optimization (HPO) service we developed is in production use for FastCaloGAN, part of the production ATLAS fast simulation AtlFast3
- In working with the analysis community to find the next large scale application of the HPO service, what emerged was active learning, algorithmically a close relative
- The active learning technique we're using was developed by our ATLAS NYU colleagues
  - "Excursion Set Estimation using Sequential Entropy Reduction for Efficient Searches for New Physics at the LHC", Kyle Cranmer et al, ACAT 2019
  - in calculating an iso-contour surface f(x), conventional approach uses a grid with a sampling density not informed by the unknown f(x)
  - instead, use an iterative approach, using information about f(x) from previous evaluation cycles to sample parameter space more efficiently
  - find an iterative algorithm that suggests points to evaluate that help the most in finding the contour
  - Kyle and colleagues found a computationally efficient one
- In response to interest from analyzers (in particular that team), we adapted our ML hyperparameter optimization service to serve this similar iterative refinement algorithm
- The entire workflow from event generation -> simulation -> reconstruction -> derivation -> limit setting analysis and its iterative refinement loop is implemented and automated using PanDA and IDDS
  - It employs grid and REANA (Reproducible Research Data Analysis Platform) processing resources
- Modular and containerized
  - Analysts provide components specific to their analysis



Active Learning via iterative regression on a limit surface