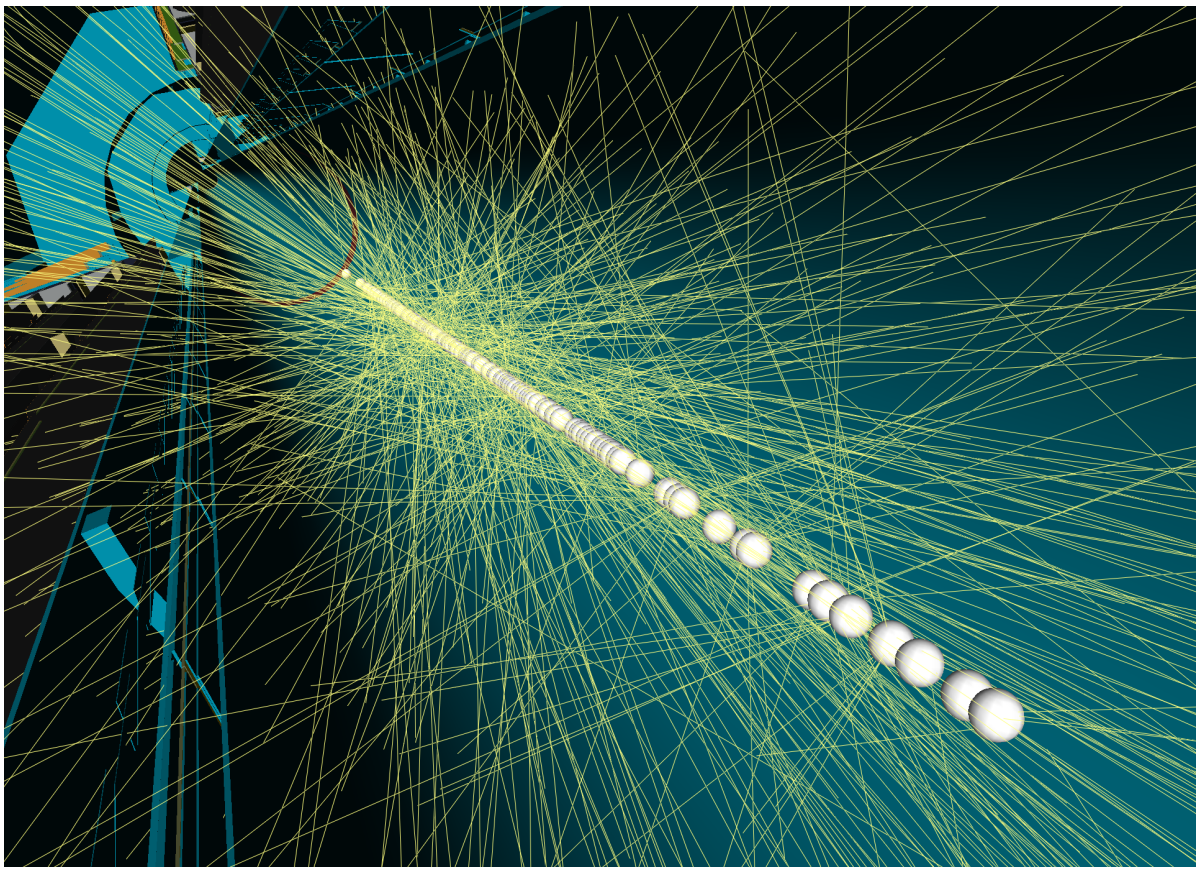
ATLAS HL-LHC Computing Conceptual Design Report



CERN-LHCC-2020-015 / LHCC-G-178
10/11/2020

Reference:

Created: 1st May 2020
Last modified: 2nd November 2020
Prepared by: The ATLAS Collaboration

Contents

1 Overview

The computing and software project in ATLAS exists to enable the Collaboration to fully exploit the LHC physics programme, while respecting the budgetary constraints associated with computing. This Conceptual Design Report (CDR) summarises the current approaches being taken and plans being developed within ATLAS to meet the challenges of the HL-LHC era. The HL-LHC will deliver unprecedentedly complex events, with up to 200 interactions per proton-proton bunch crossing. These events will be collected at a prodigious rate - ATLAS expects to record data at 10 kHz, which is approximately ten times more than during previous runs. By the end of Run 5 in 2034, the HL-LHC is expected to have delivered an integrated luminosity of up to 2500 fb^{-1}, five times more than all of the previous runs combined. As well as the challenges involved in collecting, storing, reconstructing and analysing such a colossal volume of data, simulated events ("Monte Carlo" or MC) will need to be produced in similar numbers. Taken together, the data and MC requirements of the HL-LHC physics programme are formidable, and if the computing costs are to be kept within feasible levels, very significant improvements will need to be made both in terms of compute and storage.

This document highlights areas where significant development effort, both within and outside of ATLAS, will be required in the years before the HL-LHC starts to operate. An extrapolation of the present computing model to HL-LHC conditions indicates a significant shortfall in disk space and computational capacity. This is true even if the investment from funding agencies continues at the rate established in previous years. This shortfall can be addressed through the development of performant and portable software, and more effective use of storage. Such development requires significant and sustained investment in people to carry out the work, but will avoid excessive computing costs in the future. In consequence the long-term sustainability of ATLAS depends on engaging individuals whose expertise is in software and computing. It also depends on physicists adapting their working practices in such a way as to use computing resources as efficiently as possible, without impacting the physics reach of the experiment. Sustainability over the next two decades will require that senior decision makers and funding agencies are able to invest in, and award research grants to, people who specialise in software and computing, whether they identify primarily as physicists or otherwise. Sustainability also requires that such individuals have realistic prospects of long term employment in the field.

Event Generator workflows are the first step in the production chain for MC events. They have been a large consumer of computing resources to date (approximately 10-15% of CPU cycles). Event generation is likely to become more prominent in Run 4, as many measurements and searches will require more accurate physics modelling, implying the use of Next to Leading Order (NLO) or NNLO event generators (§4). As well as software development to maximise computational efficiency, including through the use of accelerators, careful optimisations of physics choices, controlling the event generation as a function of a kinematic quantity of interest (thereby reducing the number of events needed), reducing negative event weights and the spread of weights, and even sharing generated events between collaborations, will all contribute to ensuring that event generation is sustainable in Runs 4 and 5. It should be noted that event generators are typically developed by small teams of theorists, whose primary concerns are not related to efficient utilization of computing resources, so there is plentiful scope for technical collaboration to speed up the software. Endeavours to increase collaboration on software across the field, especially through the HEP Software Foundation (HSF), are of crucial importance if applications such as event generation are to be scaled up to HL-LHC demands [1].

ATLAS currently expends around 40% of its CPU resources on detector simulation (§5). Approximately half of the events are produced with "full" simulation, using the GEANT4 physics library, with the rest coming from "fast" simulation, which uses a parameterised model of the calorimeter response and GEANT4 elsewhere. Full simulation uses around five times more computing resources than fast simulation, and ATLAS aims to use mostly fast simulation for Run 4 and beyond. Studies are ongoing to extend the fast simulation concept to all parts of the detector, including the inner tracker as well as the calorimeter. Novel simulation techniques made possible by deep learning may also prove highly effective at improving the precision of the fast simulation. Use of full simulation will remain unavoidable for certain applications, and to train the fast simulation. For this reason ATLAS will collaborate closely with the GEANT4 team to ensure the best possible physics fidelity for the lowest possible expenditure of resources. Finally, ATLAS must ensure that the computing costs of trigger simulation remain under control, especially as hardware-based tracking will be used in the HL-LHC trigger system.

Event reconstruction, which is fully under the responsibility of ATLAS, will benefit enormously from the

Phase II upgrade to the Inner Tracker, the ITk. As well as providing exceptional physics performance, it is also optimised to minimise CPU consumption of track reconstruction despite the much higher pile-up at the HL-LHC. At the same time, ATLAS is pursuing several promising avenues of software optimisation, which together with the ITk should ensure a sustainable reconstruction model in Run 4. These developments include improvements and optimisations to the classical algorithms that can further exploit the ITk design and provide very substantial CPU savings whilst maintaining acceptable physics performance (§6.1). ATLAS is also implementing new cross-experiment tracking software, which benefits from a modern design with a light-weight event data model. This is expected to yield further savings and efficiencies (§6.2). Novel approaches based on machine learning (ML), or new non-ML algorithms, may be able to improve the physics and computing performance further. Finally, the ATLAS offline software and High-Level Trigger (HLT) communities are investigating the possibility of using accelerators for aspects of the reconstruction and HLT, as part of the heterogeneous software programme (§6.6). All of these improvements require determined investment to maintain and increase the number of people available to work on such developments.

ATLAS is currently implementing a new analysis model (§8.1) which aims to reduce the disk footprint occupied by centrally produced data products used for analysis. It is based around two new formats, one with a maximum size of 50KB/event (`DAOD_PHYS`) and the other 10KB/event (`DAOD_PHYSLITE`). The first of these is primarily aimed at Run 3 analyses; it contains all of the variables needed to apply calibrations to reconstructed objects. This will replace most of the dedicated analysis formats currently in use during Run 2, and approximately halve the disk footprint of the analysis formats overall. The second smaller format - `DAOD_PHYSLITE` - contains precalibrated reconstructed quantities, and in consequence the variables needed to apply the calibrations do not need to be stored. The `DAOD_PHYSLITE` format will be used in Run 3 alongside its larger counterpart `DAOD_PHYS`, in preparation for it taking over as the main format in Run 4. The success of the new model in Runs 4 and 5 requires the adoption of this small format by most physicists, and this will require some physics groups to re-think how they perform their analyses. Another important piece of the new analysis model is the appropriate application of lossy compression, such that variables stored in reconstruction and analysis data formats are stored at a precision concomitant with the instrumental precision. A size reduction of approximately 10% could be made to all formats storing reconstructed data objects, should this be maximally utilised (§2.3).

Storage will present a significant problem for HL-LHC computing. Unlike CPU, the requirements for which will become approximately constant once the LHC reaches its design luminosity, storage needs will continue to increase during the lifetime of the HL-LHC. Furthermore, opportunistic computational resources exist, but not opportunistic storage. Significant optimisations can be made to fully exploit the available hardware - a broad range of ideas exist for optimising storage by breaking out of the disk/tape paradigm to a finer grained spectrum of storage cost-reliability-latency. In particular, staging AOD data from tape to a disk buffer when they are required for processing (data carousel, §10.4) can halve the total AOD volume permanently resident on disk. This activity is an investment in both people and hardware, since it involves optimising and tailoring data movement site by site, making the best possible use of continuously evolving infrastructure. Such an approach does impose some constraints on flexibility, in particular the frequency of processing campaigns. On-demand production is less practical if the input files have to be staged from tape before the jobs can start. These aspects are particularly relevant for the production of analysis formats. The expectations of physicists need to be managed, since they may experience a slower turnaround than previously and so need to plan their analysis format production schedules well in advance.

The backbones of ATLAS distributed data processing and management - Panda (§10.2) and Rucio(§10.3 - will need to be scaled up to HL-LHC workloads and volumes. Intelligent scheduling of jobs based on improved coupling between job characteristics and available resources, for instance CPU and accelerators, or I/O intensive jobs scheduled on nodes with storage with high I/O capability, is important to allow full exploitation of the heterogeneous resources that are likely to be provided in the coming years. Evolution of the WLCG infrastructure in this direction is anticipated. Intelligent automation coupled with new emerging technologies (such as containers and novel deployment tools) will increase the efficiency with which we will exploit Grid and opportunistic resources. Use of new storage paradigms, such as cloud storage, data lakes and caches, might also yield further benefits, especially for smaller sites. Caches are much simpler to deploy and to run than normal Grid storage, thus their use will save operational effort. R&D activities in this area are paramount, and are overseen by the Data Organisation and Management (DOMA) group within WLCG.

The general trend of the evolution of the IT industry is towards more heterogeneity in computational hardware. These changes are driven by the desire to continue decreasing the power and cost per operation. Market forces, driven particularly by advances in data science and machine learning, currently favour massively parallel architectures such as GP-GPUs. By the time of HL-LHC it is possible (or even likely) that hardware installed at dedicated WLCG sites will feature elements of these technologies as well. In the meanwhile computing architectures will continue to evolve. In order to make full use of all of the resources available in the future, it is clear that part of the software stacks used by HEP experiments must by then have evolved such that they can run efficiently on a range of massively parallel devices, whilst still being compatible with more familiar hardware. The recent transition by ATLAS to a multi-threaded framework is a necessary first step towards this aim (§2.1). Fledgling cross-experiment R&D efforts in portable parallelization strategies [2] hopefully will show the next steps.

To inform the discussion of the resources needed for HL-LHC computing, projections need to be made based on reasonable assumptions as to the activities that will be carried out by ATLAS, the actions it will take to minimise computing costs, and the provision from the funding agencies. For the purposes of this document, ATLAS has chosen to evaluate three scenarios, as follows:

- **Baseline**: ATLAS implements the new data formats foreseen by the Run 3 analysis model, the multi-threaded software framework AthenaMT, and updates to the tracking code, but otherwise continues in largely the same way as in Run 2. In particular the CPU time per event for event generation, detector simulation and reconstruction is assumed to remain at the level currently achieved by applying the current software to the Phase-II detector simulation, and the mixture of generators and simulation remains the same;
- **Conservative R&D**: the research and development activities currently under way for Run 3 are assumed to be successful, including the data carousel, fast track reconstruction, lossy compression, and most of the detector simulation is done with fast simulation;
- **Aggressive R&D**: ATLAS implements new developments that very significantly improve the speed or storage volumes of workflows that currently are heavy consumers of resources, for example, porting of high-precision generators to GPUs, sharing events with CMS, or speeding up the full simulation either by software efficiencies or porting parts of the code to GPUs. Almost universal adoption by the physics groups of `DAOD_PHYSLITE` and development of very high quality fast simulation that could replace full simulation in almost all cases, would also fall into this category.

The precise implementation of these scenarios in each part of the data processing chain is described in the relevant chapters of the document. This is then brought together to develop overall resources projections under the three scenarios.

2 Core Software

As ATLAS addresses data processing in the HL-LHC era, it must face two particular challenges:

- an order of magnitude increase of volumes of data due to increasing event sizes and rates;
- evolving architectures which are becoming increasingly heterogeneous.

The ATLAS core software must provide all necessary components and tools to allow the data processing applications to run efficiently in the face of these challenges.

2.1 Framework components

The ATLAS multi-threaded data processing framework, AthenaMT, uses the task scheduler from Gaudi, which in turn relies on the Intel Thread Building Blocks (TBB) library to map tasks to kernel threads. Although the basic functionality of the scheduler is already in place [3, 4], the scalability of the current solution — particularly over heterogeneous architectures — is limited by design. In order to overcome this limitation, we plan to design and implement a next-generation task scheduler in Gaudi, which will come with a set of advanced features designed to maximize event processing throughput. Such features include a hybrid threading model based on lightweight user-level threads with fast context switches, a task-based asynchronous programming model, support for computation offloading, and distributed memory computing.

The current trend towards power-efficient, heterogeneous computing architectures is projected to continue over the next decade. ATLAS software will likely run on systems including massively-parallel, domain-specific accelerators optimized for tasks such as tensor manipulation, dataflow algorithms and graph processing. This trend is currently seen in High-Performance Computing (HPC) systems, which provide large compute capacity via the use of accelerators. Given the size of the ATLAS software repository, and the number of available software developers, ATLAS cannot afford to rewrite its software stack for every new architecture. Furthermore, even if multiple versions of the code were available for each architecture, the effort to maintain and validate each new version would be onerous. Instead portability solutions that permit the same code to run on several different platforms must be sought. Accelerator hardware manufacturers also realize that to make large code bases portable, the current accelerator programming solutions will have to be standardised, and even be made part of a future C++ standard. Both NVidia and Intel are actively working with the C++ Standard Committee to try to make their programming interfaces (CUDA and DPC++ respectively) the standard inside C++. The ATLAS Core Software group will need to maintain an active relationship with actors within these corporations and the C++ Standards Committee itself to ensure that the final choice of software development platform for the ATLAS offline software is wise and well motivated. This will also require continuing discussions with other experiments on this topic, preferably through the HEP Software Foundation.

Future accelerator-centric platforms bring significant challenges to the ATLAS software and computing, since currently the workflows can run only on CPU-based systems. From the core software perspective one of the key problems is how to integrate accelerator programming models (e.g. CUDA, SYCL/DPC++) into the data processing framework, and how to efficiently schedule computations from multi-threaded applications to accelerator devices. In the longer term, research must be carried out to understand how a distributed, fine-grained workflow scheduling system can help to run hybrid multi-threaded/accelerated workflows on the combined resources comprising multiple experiment-owned CPU clusters and heterogeneous HPC centres. Such a scheduling system would ensure efficient and scalable execution of the hundreds of software components of a typical ATLAS workflow, assigning each one of them to the most appropriate resource. It should also be able to self-tune its schedules to different large-scale computing architectures to maximize the event processing throughput.

End-to-end workflows that are suited to accelerators are few and far between in HEP, but some are likely to be more amenable to such technologies than others. Preliminary studies [5] have shown that certain event generation packages are well suited for execution in the GPU environment, and programmes are under way to convert them. Workflows that spend significant fractions of their time undertaking ML tasks are also ideal for accelerators, and many ML packages have GPU back-ends that are transparent to the user. Individual tasks that are inherently parallel in nature, such as track seeding or calorimeter clustering may also function well on a GPU.

Even if just a few slow algorithms can be converted to use GPUs, significant gains in the total throughput can be realized. It should be noted that converting HEP data processing algorithms to run efficiently on GPUs can be a complicated task, as the inherent branching and memory access patterns of these types of algorithm are not well suited to GPU architectures. In order to simplify integration of user kernels with Athena, we must provide infrastructure to

- efficiently manage GPU kernel resources such as CUDA streams;
- manage GPU memory, possibly via custom allocators;
- prepare data for offloading from the CPU, and reconvert it when the kernel has completed;
- integrate kernel compilation into the build environment, via CMake directives for CUDA, DPC++, Kokkos[6], Alpaka[7], and other languages;
- validate results that are produced by the GPU, as bitwise comparisons with results from the CPU are impossible due to different code paths, levels of precision, and computational hardware.

Heterogeneous accelerated systems dominate high-performance computing today. New computing architectures will emerge over the next decade in the post-Moore era[8]. To be ready, ATLAS core software will focus on identifying and supporting portable, fine-grained parallelization solutions. The last point on validation is also particularly important. Significant time and effort will need to be invested in developing a strategy that can still validate software as being good for physics, when it may run on a wide variety of platforms, and potentially follow different code paths depending on which hardware it runs on.

2.2 Event Data Model

In order to make effective use of future computing resources, we will need to evolve the ATLAS xAOD columnar data model[9], which was developed for Run 2 and has proved to be successful. The current interfaces need to be streamlined to be able to better treat the data as arrays of structures. Simplified versions of the Event Data Model (EDM) classes may need to be defined for use on accelerators; the way the EDM classes are currently defined should be made more structured so that CPU and accelerator versions can be generated from the same definitions. Many of the variables used in the current data model are simple types, but some are not. Variables such as vectors will likely need to be migrated to a flat representation. It may also help to take more control over memory allocation, for example to allow storing all the data for a given collection of objects in a single contiguous region of memory. Changes along these lines should also make it easier to expose the data to Python as numpy[10] objects, allowing for better integration with the growing Python-based analysis ecosystem, and could also help with enabling access to the data from other compiled languages. For heterogeneous computing applications, one may need to deal with multiple representations of an object, for example on the CPU and on an accelerator, so these alternate versions might need to be represented explicitly in the transient event data store [11]. Finally, to run efficiently on accelerators with tens of thousands of processing elements, the transient data store may need to handle objects from multiple events at once.

2.3 I/O system

ATLAS uses a powerful and flexible infrastructure for reading and writing data objects using the Athena framework. While in recent years this infrastructure was in practice only used to read/write physical files using the ROOT I/O system, future data processing requirements may necessitate the addition of other I/O back-ends as well. Thanks to Gaudi's Transient-Persistent separation principle used for two decades throughout the ATLAS I/O system, the physics code will not be affected by the I/O back-end choice.

The wider parallelism of accelerators means that it may be expedient to process multiple events concurrently, allowing efficient offloading. Within the Athena framework event loop, this would require data collection across event contexts, which may offset the gains yielded by offloading. The I/O framework already deals with column-wise compressed data and may be a more efficient location for such data transfer. Tools in this area should be investigated and developed.

In addition to the challenges of providing CPU cycles for HL-LHC processing, ATLAS must also address the problem of storage. During Run 2, ATLAS has stored all data using lossless compression only and deployed an analysis model that, while being flexible and user-friendly, produced large data duplication; this resulted in the primary Analysis Object Data (AOD) taking up to 30% of total storage and the Derived AOD (DAOD) occupying another 40%. The situation will improve in Run 3 thanks to the new analysis model (§8.1). `DAOD_PHYS` and `DAOD_PHYSLITE` will reduce data duplication, thereby shrinking the storage footprint of analysis formats by up to half. Furthermore, the option of using lossy compression has been studied and implemented for AOD and DAOD, with the potential to save an additional 10-25% in storage. Other lossy compression techniques using deep learning are also under investigation [12]. For HL-LHC, these tools need to be developed even further to ensure that the available storage capacity is not exceeded.

2.4 Summary

To summarize, the main research activties within ATLAS related to core software include:

- Next generation of intra-node and inter-node task scheduling systems (**Aggressive R&D**);
- Evolution of the event data model (**Conservative R&D**);
- Storage optimization (**Conservative R&D**);
- Framework support for computation offloading to accelerator devices (**Conservative R&D**);
- Portable parallelization solutions (**Aggressive R&D**).

3 Detector Description

An accurate description of the ATLAS detector is crucial for the simulation and reconstruction of data. In preparation for Run 4, the current detector description (scheme and detectors), which has not changed

in its driving principles since its inception 20 years ago, will need to be updated to modern standards and functionality to meet present and future requirements, especially to ensure maintainability and the smooth integration of Phase-II upgrade detectors into the overall description.

3.1 GeoModel: Geometry kernel classes for detector description

The current class library providing primitives for detector description is called GeoModel [13]. It has been in service in ATLAS since 2003, and it depends only on the Eigen matrix algebra library [14]. It is supported by a heterogeneous set of database technologies and parsers. A new effort in detector description, targeting Run 4, is now under way. The goals include consolidating the various technologies in detector description, accurately describing the new detectors in Run 4, and improving the development tools for detector description. The defining objective is to dramatically shorten the detector description development cycle, by providing: a means to achieve a immediate modification of geometry description; tools for immediate visual feedback; and tools for rapid validation of geometries. These tools are largely independent of the Athena framework.

The Run 4 ATLAS detector (particularly the inner detector) will be radically different from its predecessor. This is an opportunity to simplify the code used to describe the detectors and streamline the development of the geometry designed for Run 4. The inner detector will be replaced with an all-silicon system called the Inner Tracker (ITk)[15, 16]. A silicon high-granularity timing detector (HGTD) [17] will be installed in front of the endcap calorimeter face to provide vertex timing information to the reconstruction. Additionally, the remaining portion of the New Small Wheels [18] will be installed. A substantial portion of the barrel muon system will be replaced with new chambers to improve trigger and tracking performance. Only the calorimeter systems remain largely unchanged; their upgrade programme consists only of improvements to the readout electronics.

3.2 The Streamlined Detector Description Workflow

The aim for Run 4 is to have a unified GeoModel of the whole detector, steered by an unified XML-based database and parsed by a single software package. The description should be free of geometry clashes, optimized for a performant detector simulation, and free of unnecessary dependencies which impact the portability[1]. A streamlined workflow is envisaged in which a developer creates or modifies a description of a piece of detector. Modifications to the geometry and visual feedback occur within a very short loop.

The programme of work includes the following items. Firstly, the geometry kernel classes will be reviewed, and the persistency model will be improved. A uniform system for accessing XML databases for feeding data to the geometry builders will be developed and employed in future detector description code. A visualization system (the "Geometry Explorer", `gmex`) is developed; it features extremely sophisticated visualization developed for the VP1 event display; together with the databases and the plugins, the system resembles a platform-independent CAD system for detector description. A tool suite consisting of automatic detection of geometry clashes and other anomalies, automatic generation of "geantino" maps, and auto-blending of volumes, is planned. Integration with Athena, including inter-operation with the alignment system, will be addressed. New detectors for Run 4 will be developed, and existing detector descriptions will be critically reviewed. Finally, all elements of the new infrastructure will be documented. Following this, and in some cases in parallel, existing detector description code will be reviewed, revised, and ported to the new system; the description of new detectors will be implemented using the new tools, and the geometry system will be put in a state of readiness for Run 4, and long-term maintenance.

4 Event Generators

During the course of LHC Run 2, the ATLAS experiment has devoted between 10% and 15% of its computing resources to Monte Carlo event generation, which translates to an approximate total of 70 billion events produced at an average speed of around 1000 HS06·s per event. Run 4 will see the need for high-statistics inclusive samples, and to efficiently populate the high jet-multiplicity phase space, as well as exclusive phase spaces explored by new-physics searches. At the same time, the best available

[1] Note here that a portable detector description may be the key to success in running simulation, at some future date, on novel platforms.

accuracy must be maintained, which is likely to be Next to Next to Leading Order (NNLO) in perturbative QCD for samples inclusive in additional radiation, and NLO for samples with high parton multiplicities; both interfaced to parton shower algorithms accurate to the leading-logarithm order and approximate to the next-to-leading logarithm orders. The need to compute virtual corrections, and the introduction of subtraction terms which are needed to take care of the divergences appearing at orders beyond the leading one, makes these configurations very slow to compute and typically introduces negatively weighted events. This calls for the development of strategies to reduce the computing resource usage of event generators without undermining the desired precision or the accuracy of the calculations.

4.1 Ongoing developments with an impact for Run 3 and Run 4

- A significant reduction in event generation computing resources can be obtained with a careful optimisation of the physics choices in an event generator. For the Run 2 ATLAS production of Sherpa NLO-merged V+jets events, the usage of a different clustering scale allowed for a speed-up of event generation by about a factor of two with no visible impact in modelling. Similarly, using an approximate colour treatment reduced the fraction of negatively weighted events from about 20% to about 10%.
- Monte Carlo event generators will naively produce most of the events proportionally to cross-section, and hence a lot of resources have to be spent on populating the often extreme regions of phase space that analyses are generally interested in. More sophisticated methods have now become available in most event generators which allow the event generation to be biased as a function of some kinematic quantity of interest, making the production overall more efficient. While this procedure typically makes the event generation slower, it allows for a significant reduction in the number of events that need to be generated, and subsequently simulated, reconstructed and stored.
- A large number of MC samples are typically produced to evaluate systematic variations. Most matrix-element generators now allow the computation of event weights using multiple scales and PDFs through a reweighting technique, avoiding the need to regenerate the samples with different scales and PDFs. Similar approaches are being adopted to compute variations of parton-shower parameters. For certain algorithmic variations that cannot be achieved through reweighting, workflows are being developed to save intermediate parton-level results, allowing them to be passed through alternative parton-shower or hadronisation models. Moreover, in a new technique developed during LS2 a multivariate regression has been employed to derive a multi-dimensional reweighting for algorithmic variations that traditionally would have had to be calculated explicitly. This approach would not avoid the need to generate the alternative MC sample, but would save resources otherwise spent on the simulation and storage of these alternative variation samples.
- The sharing of samples with other LHC experiments (mainly relevant for ATLAS and CMS) can potentially save up to a factor of two in CPU resources. This is being considered in particular for the most expensive part of the high-precision multi-parton calculations. This has the added benefit of an increased level of scrutiny of the expensive parton-level calculations, by multiple experiments as well as the generator developers, while leaving each individual experiment the freedom to choose specific parton showers with their preferred settings for the subsequent steps of the event generation chain. At the same time, the loss of statistical independence across the two experiments needs to be carefully considered. More recently, new workflows have been developed that achieve efficient generation of very high multiplicity parton-level events using High Performance Computing clusters [19]. This opens the possibility of producing these samples using joint HPC allocations by LHC collaborations.

4.2 Summary and R&D goals

The level of precision and accuracy required for a given Monte Carlo setup is ultimately driven by the physics choices made by the collaboration. The fraction of resources used for event generation varies with time, depending on the average complexity of the samples produced in accordance with the overall publication strategy of the collaboration. The total number of events that are generated, and then simulated per year is an important computing model parameter and is discussed in § 12.

Event generators will calculate higher order corrections, which in turn require more CPU. In the baseline scenario, it is assumed that the time/event used by event generators is similar to Run 2 and that compromises are made in the physics quality of the event generators. In the conservative R&D scenario, it is

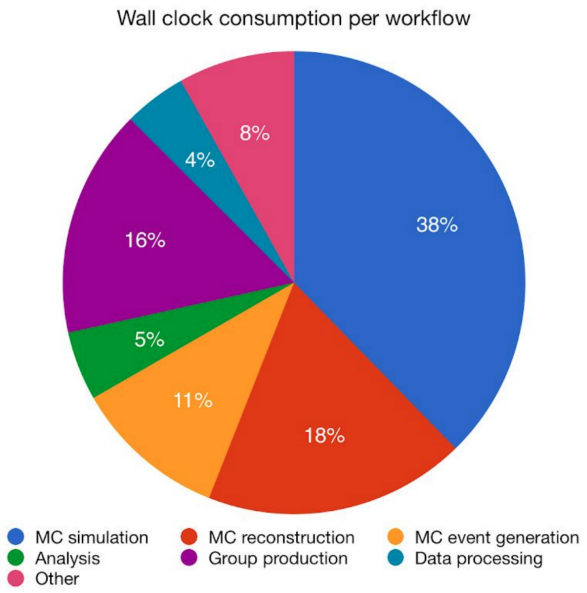


Figure 1: ATLAS CPU hours used by various activities in 2018

assumed that the CPU time needed per event for HL-LHC will be the same as in Run 2, such that a better physics description can be obtained with a similar CPU cost.

In the aggressive R&D scenario it is assumed that the total CPU time per event is reduced by a factor of 2, due to a combination of software improvements, sharing of events with CMS and physics choices. In this same scenario, it is also assumed that the total number of events generated is reduced by up to 30% by improving the treatment of events with negative weights and the sampling of the phase space. Finally, it is assumed that 10% fewer events will need to be simulated by further exploiting event-level reweighting techniques, thereby reducing the need for separate generator systematic variation samples. Reduction of the total number of events generated and simulated has an impact both on the CPU and the disk footprint. In this scenario, we also assume that some of the event generation is now done on GPUs.

5 Simulation and Digitisation

5.1 Introduction

ATLAS currently uses both full detector simulation and fast detector simulation for physics results. Full simulation is based on the GEANT4 toolkit. FastCaloSim [20] simulates calorimeter response to single particles using parametrizations, and is approximately ten times faster than GEANT4. FatRas [21] is a fast simulation of charged particle propagation in the ATLAS tracking detector with simplified material effects, based on the ACTS toolkit (§6.2). Each physics analysis group evaluates the requirements for statistical and systematic uncertainties for each physics sample that needs to be simulated. Based on these studies, physicists request some simulated samples with higher-accuracy full simulation, and others with lower-accuracy fast simulation. Reconstruction times for simulated events are currently similar for both simulation methods. On average, currently about 50% of all samples use full simulation. For the HL-LHC ATLAS aims to further reduce this fraction, with a consequent saving of CPU resources devoted to detector simulation at the HL-LHC.

5.2 Run 3 Detector Simulation Strategy

Detector simulation accounts for just under 40% of the CPU hours consumed by the ATLAS experiment, as shown in Figure 1. At the same time the physics reach of many analyses, including measurements in the Higgs sector, is limited by the available statistics of MC events. Reducing the amount of time spent on simulations is a priority for the HL-LHC R&D program. This problem is being tackled on many fronts by ATLAS and by the GEANT4 collaboration R&D projects.

Besides reducing the fraction of events simulated with GEANT4, ATLAS has an active R&D program aimed at optimizing the CPU requirements of GEANT4. Performance gains of 20% with no impact on

physics performance have been already demonstrated and another 20% should be achievable. The full simulation configuration for Run 3 will include the use of "Frozen Showers" in the forward region as in Runs 1 and 2.

Reducing the use of full simulation requires analysers to be confident in using samples produced using fast simulation techniques in their analyses. The agreement between full and fast simulation (and data) for key features of physics distributions has to be good enough that correction factors and uncertainties derived for full simulation can be used for fast simulation without application of additional corrections or uncertainties. In ATLAS this is mostly determined by the Combined Performance (CP) groups who are responsible for producing the calibrations which allow comparison between data and MC.

It should be noted that despite the goal of reducing the fraction of events simulated using GEANT4 in Run 3, GEANT4 will remain a critical part of our simulation strategy. It will be used to produce the samples used to tune parameterisations used by the fast simulation and will continued to be developed to give an improved description of interactions in the detector. A key improvement planned for Run 3 is to include energy deposits made directly by b- and c-hadrons, and τ leptons, which in Run 2 were dealt with solely by the generation step. This blurring of the line between generation and simulation will continue in Run 3 as GEANT4 works to include b-hadron interactions in their physics lists.

5.3 Run 3 Digitization Strategy

The detector digitization simulates the electronics and read-out chain of the ATLAS detector. For Run 3 ATLAS will combine hard-scatter and pile-up events at the detector digitization format (RDO) level. The approach will be to make pre-mixed pile-up RDO datasets by combining minimum bias events simulated by GEANT4 and digitizing them together in the absence of a hard-scatter event. Due to the integration time windows of the various sub-detectors, minimum bias events from a total of thirty-nine bunch crossings have to be included in the digitization ([-32,+6]), so for $\langle\mu\rangle = 40$ 1560 minimum bias events are required on average. The pre-mixed pile-up RDO datasets will then be stored on the grid. In the main production workflow hard-scatter events will be digitised and then "overlaid" on top of a pre-mixed pile-up RDO event (MC Overlay). The process of MC Overlay is considerably faster than, and with much reduced I/O requirements compared to, the process of pile-up digitization and scales much less steeply with pile-up luminosity. As long as each pre-mixed pile-up RDO event can be used more than once then there is a reduction in resource requirements over a whole campaign. A strategy for how best to store and access the O(PB) pre-mixed pile-up RDO datasets will be developed in the latter part of 2020. Another new issue for Run 3 is the variation of the beamspot shape at the same time as $\langle\mu\rangle$ is varying. The simplest way to deal with this is to have multiple (perhaps three) sub-campaigns for each data period, each simulated with a different beam spot shape. All but the last sub-campaign will look at beam conditions during the luminosity levelling regime, so $\langle\mu\rangle$ will be constant. The final sub-campaign will replicate the beam after levelling has finished and so will have a variable $\langle\mu\rangle$ value.

Development will be required to produce pre-mixed pile-up RDO events at $\langle\mu\rangle = 200$. Pre-mixed pile-up RDO events are currently produced using the pile-up digitization workflow used in Run 2. However, The memory requirements of this approach increase steeply with $\langle\mu\rangle$ - on average 7800 minimum bias events are required per hard-scatter event at $\langle\mu\rangle = 200$. Most likely this implementation will be used for Run 3, but it will not be suitable for the high $\langle\mu\rangle$ values expected in Run 4. A possible solution could be to increase the level of memory sharing between events in flight by running pile-up digitization using AthenaMT rather than serial Athena or AthenaMP. Achieving this requires the sharing of background minimum bias events between multiple threads, and a significant amount of code development and validation. While not critical, this new development would allow more efficient use of existing grid resources and increase the pool of usable resources.

5.4 Run 4 R&D

- **Baseline**: Events would be simulated full GEANT4 Simulation or fast simulation (primarily parametrized calorimeter response). Most likely the GEANT4 version would be updated at the start of Run 4. Digitization would continue to be done using MC+MC Overlay, but high memory queues would be required to produce the pre-mixed pile-up RDO files.
- **Conservative R&D**: Fast Simulation would be the default simulation method. Static compilation of Athena code with GEANT4 dependencies against GEANT4 will be implemented. AthenaMT

compatible pile-up digitization will allow grid resources to be used more efficiently for the pre-mixed pile-up RDO production step.

- **Aggressive R&D**: Substantial speed-ups in GEANT4 are found. FatRas would be available as an option to simulate particles in the inner tracker. Running simulation and digitization in a single Athena job and running EVNT to AOD in a single production step would be possible.

Run 4 analyses will need significantly more simulated events than in previous runs (see §12.2). ATLAS is undertaking a major simulation software R&D programme to speed up the MC production chain:

- A simulation based on the ACTS-based FatRas for the Inner Tracker and FastCaloSim for the calorimeter will make the simulation time small compared to the reconstruction time. Further speed-ups to the MC production workflow will require the reduction of the total digitization and reconstruction time.
- Using Trigger-like algorithms to filter events prior to reconstruction, so that events which will never be used in analyses are not reconstructed (or written out), could save CPU and disk space.
- Skipping reconstruction algorithms that are not needed for some MC sample production could also save CPU.
- Using MC generator information or simulation information augmented with parametrizations could speed up parts of digitization and reconstruction algorithms.
- Another idea involves using a strategy similar to MC Overlay, but reconstructing the pile-up tracks in separate job, to produce special RDO files containing pile-up tracks. These tracks could be copied through the overlay step. Reconstruction would then consist of running tracking for the hard-scatter, then combining the track collections and using the merged track collection as the input to the rest of the reconstruction.

Even if it is possible in principle to produce the required sample sizes, it may not be possible to store all the required AOD samples on disk. Not storing AOD formats for some simulated samples is a possible way to reduce the disk requirements.

$\langle\mu\rangle$	Full Simulation	GEANT4 + FastCaloSim V2	FatRas + FastCaloSim V2 + GEANT4	pile-up Digitization	MC Overlay
140	5684	1137	114	3317	183
200	5684	1137	114	4233	202

Table 1: Monte Carlo Chain CPU times in HS06 $\times$ seconds. Full Simulation refers to simulation using GEANT4 with Frozen Showers being used in the FCAL. GEANT4 + FastCaloSim V2 refers to simulation using FastCaloSim V2 in the calorimeter and GEANT4 elsewhere. FatRas + FastCaloSim V2 + GEANT4 refers to simulation using FatRas in the inner tracker, FastCaloSim V2 in the calorimeter and GEANT4 elsewhere. pile-up digitization refers to the approach of digitizing hard-scatter and minimum bias simulated hits together. MC Overlay refers to the approach of digitizing the hard-scatter and then combining the resulting digits with the digits from a pre-mixed pile-up event. The CPU requirements for the main production workflow are shown for two different amounts of pile-up.

Table 1 shows the CPU requirements for the simulation and digitization steps in different scenarios. The simulation times on the left hand-side of the table are for the hard-scatter only. In full simulation the time required to simulate particles in the calorimeters dominates. Using FastCaloSim V2 for these particles means that the inner tracker simulation time now dominates. Using FatRas to simulate particles in the inner tracker speeds up the simulation by a further factor of ten. The right hand side of Table 1 shows the huge speed-ups in digitization time that can be gained by pre-digitizing the pile-up backgrounds using the MC Overlay technique. This gain scales with $\langle\mu\rangle$. Clearly there is a cost for pre-digitizing the pile-up backgrounds, so overall gains are only made if the background files are used multiple times. Table 2 shows the CPU time required to simulate and digitize backgrounds and gives an estimate of the CPU cost per hard-scatter event. The re-use factors for minimum bias HITS are taken from Run 2 production and the re-use factors for pile-up RDOs are estimated from Run 2 feasibility studies.

Detector simulation is projected to continue to be one of the main consumers of CPU resources in Run 4 and beyond. ATLAS is eager to collaborate with international efforts to parallelize GEANT4 detector

	Production [HS06*s]	Re-use factor	$\langle\mu\rangle$	Events per Hard-scatter	HS06*s per Hard-scatter
Low pT minimum bias HITS	2120	1867200	140	5446	6
Low pT minimum bias HITS	2120	1867200	200	7780	9
High pT minimum bias HITS	5787	4800	140	14	17
High pT minimum bias HITS	5787	4800	200	20	24
Zerobias RDOs	3317	48	140	1	69
Zerobias RDOs	4233	48	200	1	88
Zerobias RDOs + ID Tracks	3441	48	140	1	72
Zerobias RDOs + ID Tracks	4447	48	200	1	93

Table 2: Cost of background sample production - taking into account re-use. Time/event (in HS06 $\times$ seconds) to simulate minimum and zero bias events to be used for the simulation of the pile-up. Re-use factors are the number of times these events are re-used later in the workflow and are based on re-use factors in the Run 2 Monte Carlo Campaign. The assumption here is that the size of the minimum bias HITS and zerobias RDO samples will scale with $\langle\mu\rangle$.

simulation. Preliminary evaluations and prototyping in the areas of GEANT4 Electromagnetic physics and particle transport indicate that a substantial speedup in GEANT4 event throughput is possible on a time scale of 5-10 years.

Finally a rich ATLAS-wide R&D program in collaboration with others is aimed at developing a fast, high-fidelity detector simulation using generative-adversarial neural network models. Depending on the fidelity achieved by these models in simulating tails in the detector response, a further step in reducing the overall dependence on the GEANT4 simulation may be possible.

5.5 Trigger simulation

The Run 4 trigger system is implemented in both hardware and software [22]. The software trigger is not "simulated" but instead the same reconstruction as done online during data-taking is run. However, unlike during data-taking all triggers are executed “unprescaled”. Furthermore, since simulation samples are generated before data-taking starts, the trigger menu contains additional triggers to ensure future data-taking conditions can be replicated. On average, the trigger reconstruction on simulation samples takes about 50% of the regular offline reconstruction time.

The hardware trigger is simulated using dedicated algorithms that strive to perform a bit-wise correct emulation of the trigger decision. The majority of the system consists of FPGA-based electronics that can be simulated on a regular CPU without much performance penalty.

The Hardware Track Trigger (HTT) on the other hand uses a custom associative memory chip to perform massively parallel pattern look-ups for finding track candidates. Regular ATLAS simulation and reconstruction will use the parameterised fast simulation (FastHTTSim), which applies smearing of offline/truth/HLT track parameters, with negligible need of resources. Simulation of the HTT is necessarily computationally expensive (since the very purpose of the HTT is to implement in hardware algorithms that are expensive to run in software). Such resource intensive full HTT simulation (HTTSim), will only be used for the following use-cases: performance studies at high pile-up in a limited ϕ slice (1/32); estimation of fake and dataflow rates with high-pile-up samples; parameterisation of the fast simulation using single-muon events. The generation of constants for the pattern-matching and track-fitting (HTTSimGen) requires high statistics samples of simulated single-muon events.

The required CPU resources for one HTT production campaign are summarised in Table 3. A new production will be required when data-taking conditions (e.g. tracker alignment, LHC interaction region) change significantly and is estimated to be about two campaigns per year. Additional campaigns will be needed during the commissioning phase of the project. HTT validation may also need additional samples to study tracking performance in dense environments.

The samples sizes and event processing times are conservative estimates with very large uncertainties derived from previous FTK studies. Further detailed studies are needed to get better estimates on samples

sizes and processing times. To reduce the required resources, alternative processing methods are under consideration, such as the usage of GPUs or the possibility of processing simulated data directly on the HTT hardware system at Point-1.

use case	sim. type	event type	events/cycle	HS06*s/event	HS06*s
ATLAS simulation	FastHTTSim	any	any	negligible	neglible
Performance at high-pile-up	HTTSim	min-bias	1 M	1.2 M/32	37500 M
Fake and dataflow at high-pile-up	HTTSim	min-bias	50 k	1.2 M	60000 M
Parameterization	HTTSim	single muons	50 M	100	5000 M
Generation of constants	HTTSimGen	single muons	1 B	135	135000 M

Table 3: Extrapolations of CPU resources for global HTT during regular data-taking per production campaign (see text). A CPU reduction of 20% per event is expected if only regional HTT is simulated.

6 Reconstruction

Dealing with the increased event complexity from unprecedented levels of pile-up, reaching an average of 140 simultaneous proton-proton collision ($\langle\mu\rangle$) during Run 4 and up to 200 in the subsequent runs, poses a challenge in particular to the offline event reconstruction and to the physics object identification. At the same time, it will be vital for the ATLAS Phase-II programme that the physics performance of the reconstructed objects will be preserved and wherever possible improved with respect to earlier runs, despite the increasing level of pile-up expected for Phase-II.

Detector	$\langle\mu\rangle$	inner tracking	muon spectrometer and calorimeter	combined reconstruction	monitoring	total
Run 2	90	1137	149	301	106	1693

Table 4: The CPU required in HS06 $\times$ seconds to reconstruct data 2018 events at an average of 90 pile-up from a dedicated high pile-up run when using the current software reconstruction software. Compared are the results for track reconstruction, muon and calorimeter reconstruction, combined reconstruction and monitoring using the Run 2 software release.

Estimates based using the current Run 2 software to reconstruct raw data from a dedicated run taken in 2018 with an average pile-up of 90, as shown in Table 4, indicate that for the current detector and software all aspects of event reconstruction require significant CPU resources to deal with the increased event complexity. The inner track reconstruction dominates because of its pronounced scaling with pile-up. ATLAS is undertaking a major detector and software upgrade programme to facilitate the required physics performance improvements and at the same time to help reduce the CPU requirements for event reconstruction:

- The design of the Phase-II tracker upgrade (ITk) [23] has been optimised not only for physics performance, but at the same time the design aims to minimise CPU for reconstruction at an average pile-up of 200. The five layer ITk Pixel Detector with its ring design will facilitate fast track seeding and finding approaches;
- ATLAS carried out a prototype study [24] to demonstrate that classical CPU based algorithmic approaches can exploit the ITk detector design for a fast track reconstruction to resolve the CPU problem, at the expense of some limited loss in physics performance (see §6.1 below for more details);
- ATLAS initiated the ACTS [25] open source project to develop the next generation tracking software in a common cross experiment project, with the aim to use ACTS to achieve both CPU reduction and excellent physics performance, for the Phase-II reconstruction. Moving to ACTS

will not only address the challenge of ITk reconstruction, but at the same time it will help to reduce the CPU for other aspects of reconstruction, such as particle flow or muon reconstruction;

- An equally important aspect of the ATLAS Phase-II reconstruction strategy is a strong programme of algorithm R&D to improve all aspects of event reconstruction, to further reduce the CPU needs and to maximise physics performance. This includes a broad set of R&D initiatives of applying ML and novel data science inspired algorithmic approaches to reconstruction, as well as R&D on exploiting accelerators (i.e. GPUs) for offline event reconstruction.

6.1 The Tracker Upgrade and the Fast Track Reconstruction Strategy for Phase-II

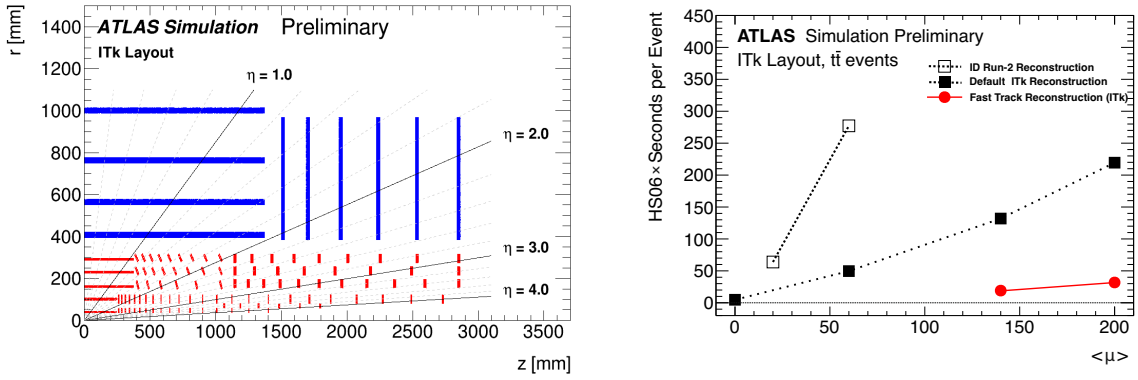


Figure 2: **Left:** A schematic layout of the "ITk Layout" with a five layer Pixel Detector (red) surrounded by the Strip Detector (blue). Only the positions of the active sensors are shown. **Right:** The CPU required in HS06 $\times$ seconds to reconstruct a $t\bar{t}$ event in the current ID and the ITk at different levels of $\langle\mu\rangle$. Standard Run 2 reconstruction was used for the current ID, while for the ITk results are shown using the Run 2 software (black line) and the fast track reconstruction (red). Figures taken from References [23] and [24].

Figure 2 illustrates the design of the ITk and the results of the fast track reconstruction study. The left plot shows a schematic $R-z$ view of the ITk detector layout that will consist of a five layer pixel system covering 8 units in η, surrounded by a four layer double sided strip detector with small stereo angle. The new tracking detector has been optimised for pile-up $\langle\mu\rangle = 200$, with an improved granularity, a constant hit coverage, a reduced material budget in the active tracking volume and a sensor placement that also aims at minimising CPU for pattern recognition. The right plot shows the CPU required for track reconstruction as a function of average pile-up for the current ID and the ITk. The Run 2 tracking software to reconstruct the current ID scales polynomially or worse in CPU utilization between $\langle\mu\rangle = 20$ and 60. Shown on the same plot is the CPU required by the Run 2 tracking code to reconstruct the ITk with an average event pile-up of up to 200. The ITk facilitates a significant reduction in the CPU to 124 and 214 HS06 $\times$ seconds, respectively, for an average pile-up of 140 and 200. At the same time the ITk will yield an improvement in physics performance compared to the current ID [23].

$\langle\mu\rangle$	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

$^{(*)}$ Scaled from Run-2, see text.

Table 5: The CPU required in HS06 $\times$ seconds to reconstruct $t\bar{t}$ Monte Carlo events with $\langle\mu\rangle = 140$ and 200 in the ITk. Listed are the results for the different reconstruction steps using the current Run 2 software and the fast ITk track reconstruction. An Intel Xeon E5-2620v2 was used with 2.1 GHz and six physical cores per CPU. The CPU time is multiplied with the HS06 factor of 17.8 for single thread running. The Table is taken from Reference [24].

ATLAS has undertaken a study [24] to demonstrate the possible CPU performance improvements

achievable by optimising the Run 2 track reconstruction techniques for Phase-II levels of pile-up. The two tracking algorithms requiring the largest fraction of CPU are the silicon track finding and the ambiguity resolution. The ambiguity resolution takes about 60% of the total CPU time to apply the final precise track fit and to handle the pixel cluster splitting in dense environments using a neural network [26]. For the purpose of this prototype study, the ambiguity resolution algorithm is omitted from the reconstruction chain. Instead, a tighter track selection and precise cluster calibrations are applied already at the stage of the silicon track finding to remove duplicate tracks and fakes. The five layer pixel detector covers the full range of $|\eta| < 4$ such that the seed finding could rely only on pixel hit combinations, leaving out the strip seed iteration. In addition, technical improvements were applied e.g. to the strip and pixel clustering and space point formation to better optimise the software for Phase-II levels of pile-up.

Table 5 summarises the CPU times for using both the Run 2 and the fast tracking code for reconstructing the ITk data. The fast version of silicon track finding is approximately eight times faster for $\langle\mu\rangle = 140$ and 200, respectively. The fast track finding is about a factor 1.8 faster for the $\langle\mu\rangle = 140$ sample, compared to the $\langle\mu\rangle = 200$ sample. Adding the CPU needs for the cluster finding and the space point finding algorithms, the total CPU requirement for the fast ITk track reconstruction becomes 19 and 31.7 HS06 $\times$ seconds for $\langle\mu\rangle = 140$ and 200, respectively. The right plot of Figure 2 shows the result for the fast reconstruction in red.

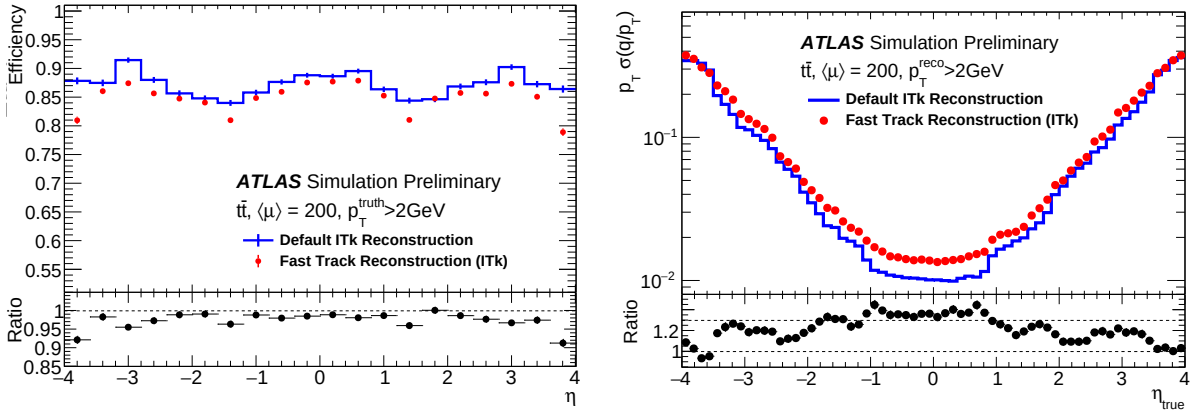


Figure 3: **Left:** Tracking efficiency as a function of η for the fast and the default ITk reconstruction. **Right:** Relative resolution of the inverse transverse momentum q/p_T as a function of η_{true}. Samples of $t\bar{t}$ events are used with $\langle\mu\rangle = 200$. A p_T cut of 2 GeV is used for the generated particles, to avoid turn-on effects. The ratio is given by the efficiency for the fast reconstruction divided by the efficiency for the default reconstruction.

This is a preliminary study and the fast ITk track reconstruction is not fully optimised; as a result there is some loss in physics performance due to approximations made. Figure 3 shows the loss in track reconstruction efficiency and in momentum resolution when comparing the fast and the default reconstruction on a sample of $t\bar{t}$ events with an average pile-up of 200. A more detailed assessment of the tracking performance can be found in Reference [24]. This study demonstrates that it is possible to address the ITk track reconstruction problem for Phase-II levels of pile-up using current tracking techniques on a classical CPU and that further developments are needed to address the expected performance deficits of the current fast reconstruction prototype.

6.2 The ATLAS Reconstruction Software Upgrade using ACTS

The goal of the ATLAS Phase-II software upgrade programme is not only to achieve the ultimate physics performance, but at the same time to modernise the software technology and to make best use of upcoming and future processing technologies. Technical performance improvements are expected in the area of event data model, data structures and data locality, in mathematical library optimisation and in the algorithms used for reconstruction.

ATLAS has launched the ACTS project [25] as an open source tracking software project within the HEP community at large. The ACTS tracking software suite has been designed from the start for multi-threaded event processing and having data locality in mind. It significantly simplifies the technical overheads with respect to the current ATLAS tracking software and is aimed at a better utilisation of vector processing

capabilities of modern CPUs. ATLAS is currently in the process of integrating the first functional components of the ACTS suite for the Run 3 event reconstruction.

Replacing the current suite of tracking tools used in the current Run 2/3 reconstruction with ACTS will result in significant CPU gains across the application, from the Inner Detector and Muon Spectrometer tracking, to all combined reconstruction using charged particles. An important deliverable of the ACTS projects is a fast combinatorial Kalman filter, that will be used in the fast silicon track finding to recover the full physics performance. The same Kalman filter will be used in the Ambiguity Resolution together with the Neural Network cluster splitting to recover from physics performance losses of the fast ITk track reconstruction prototype in the core of dense jets.

The Muon Spectrometer and combined muon reconstruction will benefit from the improved technical performance of the ACTS tracking tools. The new ACTS Gaussian Sum Filter (GSF) implementation is foreseen to replace the current software in the combined electron reconstruction and the ACTS extrapolation code will be used for the particle flow algorithm combining calorimeter and tracking information for jet finding and missing energy reconstruction. Primary vertex reconstruction, conversion finding, τ reconstruction and b-tagging will be based on the ACTS vertexing package. The reconstruction of the Phase-II High Granularity Timing Detector (HGTD) will use the ACTS software for associating timing hits to forward tracks.

6.3 Optimising Reconstruction for Phase-II Levels of Pile-up

The physics performance of all object reconstruction and identification at Phase-II needs to match and where possible exceed the current Run 2 performance, if ATLAS is to reach its physics goals. All algorithms in the reconstruction chain need to be fully optimised for high pile-up, taking into account the CPU resources limitations. Technical software improvements, such as the introduction of ACTS for the full reconstruction chain, are essential to meet the goals.

ATLAS is running an ambitious development programme for improved and novel algorithmic approaches for all detector and combined reconstruction aspects. Examples are optimised strategies for combined muon reconstruction that improve the acceptance for soft muons or handle more gracefully events with hadronic showers leaking into the Muon Spectrometer.

Calorimeter reconstruction is expected to be unaffected by the high luminosity, both in terms of CPU time consumption and memory usage and both for data and Monte Carlo simulations. This is due to the particular reconstruction algorithms for this detector system, which are described in detail in Ref. [27]. As for Run 1 and Run 2, they start from the whole (fixed size) list of individual signals and yield a stable average number of final signal objects (topological cell clusters), rather independent of the luminosity. While the existing calorimeter reconstruction algorithms are not expected to change significantly, possible reconstruction performance improvements and computing resources reductions are under study. In particular, these include the formation and calibration of the topological cell clusters using ML techniques (see Reference [27]).

New particle flow [28] objects will be introduced that support a more efficient extraction of overlapping and shared signals in complex Phase-II events, thereby allowing the efficient extraction of an unambiguous event representation. Improved techniques for pile-up rejection will be developed for all combined reconstruction, exploiting the detailed particle flow information, the improved tracking performance with the ITk and the timing information from the HGTD in the forward region.

6.4 Streamlining Reconstruction for Unconventional Signatures of New Physics

One of the primary goals of the (HL-)LHC physics programme is the search for New Physics. Several models for New Physics give rise to experimental signatures involving meta-stable particles. A possible signature is the decay of charged particles within the tracking system, leading to signatures of “disappearing tracks”. Meta-stable heavy particles, as predicted in some R-parity violating SUSY models and others, may give rise to displaced production vertices for charged particles that tend to have large impact parameters and thus are only measured in the outer layers of the tracking system.

During Run 2 ATLAS used a “pixel tracklet” reconstruction step, which was run after the primary track finding to identify candidates for disappearing tracks. A set of filters applied to the main physics stream was used to pre-select candidate events for CPU intensive reconstruction of tracks from displaced vertices. A significant fraction of the total CPU for data reconstruction was spent on the processing of the

additional stream of selected displaced vertex candidate events, which also required significant additional data storage capacities. ATLAS aims to simplify the reconstruction strategy for such unconventional signatures for Phase-II. It is planned to integrate the pixel tracklets as a second track selection strategy in a unified single path silicon track finding step, minimising the additional CPU resources needed. Due to the significant differences between the detector signatures of primary charged particles and those for displaced tracks, a second path of the silicon track finding will be required. To limit the amount of CPU resources needed, this second path will only be applied to events pre-selected by dedicated filters for such signatures.

Integrating the track reconstruction for unconventional signatures into the data processing chain will increase the CPU requirement for Tier-0, but the total amount of CPU required for reconstruction will be significantly reduced and the data processing model will be simplified, also reducing the amount of storage needed during Phase-II.

6.5 Prospects for the technical Performance of the Phase-II Reconstruction

Two scenarios for the technical performance of the Phase-II reconstruction were studied. The results of the current Run 2 based software for Phase-II reconstruction are used as a baseline. Future improvements from the fast ITk reconstruction, the introduction of ACTS, the improvements in muon and calorimeter software, as well as in combined reconstruction, are expected to lead to significantly lower CPU resources requirements as a result of the Phase-II reconstruction software upgrade programme.

$\langle\mu\rangle$	primary tracking	unconventional signatures	calorimeter, muon spectr.	combined recon.	monitoring	total recon.
140	124 (35)	- (25)	157 (85)	51 (35)	70 (35)	402 (215)
200	214 (50)	- (30)	176 (95)	94 (70)	100 (50)	584 (295)

Table 6: Prospects for CPU required in HS06 $\times$ seconds for reconstruction at different levels of average pile-up for Phase-II. Shown are the numbers for a $t\bar{t}$ Monte Carlo sample as measured for the current Run 2 based reconstruction software. In brackets the goals of the software upgrade programme are shown exploiting the Fast Track Reconstruction, ACTS and further technical and algorithmic optimisation of all aspects of the reconstruction software chain. Results for MC truth processing and framework overheads for I/O (etc.) are not included.

Table 6 summarises the technical CPU requirements for the baseline scenario using the current Run 2 based reconstruction software and, in brackets, when extrapolating the expected software and algorithmic improvements from the Phase-II software upgrade programme. The latter are used for the conservative R&D scenario and also include the CPU for the reconstruction of unconventional signatures in the normal processing chain. The budget for offline monitoring on real data for Phase-II for the baseline scenario is assumed to be 100 HS06 $\times$ seconds and 50 for the conservative R&D scenario, for running at pile-up of 200. At pile-up of 140, 70 and 35 is assumed, respectively.

The software upgrades, if successful, may lead to more than a factor two speed improvement, compared to the current Run 2 software, also incorporating the processing of events for unconventional signatures. Track reconstruction will still account for a significant fraction of the overall reconstruction time, with a larger number of other reconstruction steps contributing at a similar level to the total CPU needed.

While the baseline and conservative R&D scenarios are based on an evolution of the current reconstruction software, this is no longer the case for an aggressive R&D scenario. In such a scenario the innovative algorithm R&D, ML techniques and the extensive use of accelerators may lead to additional significant resources reductions that will only be quantifiable once more elaborate prototypes become available. For the purpose of this document, the estimated resource needs for the aggressive R&D scenario for reconstruction are therefore based on the conservative R&D estimates.

6.6 Algorithm R&D, machine learning and accelerators

The offline reconstruction will need to be adapted to the rapid evolution in software and processing technologies. Over the past years ML and other Data Science inspired techniques have been developed that promise to boost the physics performance for many aspects of event reconstruction. Techniques like similarity hashing, metric learning or graph networks are being investigated as alternatives [29]

to the classical track reconstruction techniques. For Phase-II such novel techniques may be applied in the offline reconstruction or trigger context, with the goal of improving the technical CPU performance without sacrificing physics performance. It is therefore vital that ATLAS continues to further invest in such R&D.

Most current and next generation HPC systems will provide the majority of their computing power in form of GPU co-processors. Online reconstruction for trigger processing can potentially benefit from deploying GPUs in the trigger farm of the experiment. While all GPUs offer large parallel processing capabilities, the model for programming those devices significantly differs from classical X86 processors. Supporting a heterogeneous set of accelerator technologies is therefore also a software development challenge. ATLAS is currently exploring different technologies (see §2). The aim is to develop tools for efficient offloading of algorithmic code onto different accelerators using the same code base and to minimise the need of vendor specific software development for the applications itself.

The model of using GPUs for event reconstruction depends on the sharing of workloads in the application. Offloading a few algorithmic kernels, that otherwise would require a large fraction of the overall CPU budget, is not possible for the ATLAS Phase-II event reconstruction. Packages such as ACTS and new ATLAS algorithmic code developed for Phase-II are being designed from the start to better support parallel processing. Memory models required for efficient processing on GPUs will also help improving data locality for X86 processing. Novel ML and Data Science inspired algorithmic approaches for event reconstruction are also evaluated for the ability to exploit GPUs. The exact model for a more fine-grained offloading of algorithmic workloads onto accelerators is the subject of intensive R&D. The ATLAS Phase-II computing model extrapolations for offline reconstruction are therefore based on X86 processing and no assumptions are made on additional gains from particular accelerator and related software technologies.

7 Visualization

Interactive visualization is a key tool in High Energy Physics experiments [30]. Not only does interactively visualizing data from particle collisions help in understanding the physics involved in the interactions between fundamental particles; it is also a necessary tool for a number of different tasks in the HEP workflow, from detector development to simulation, reconstruction, physics analysis, and outreach [31].

In Phase-2, as discussed in §6, the foreseen increase in the number of simultaneous proton-proton collisions will yield a related increase in the event complexity. The number of physics objects to reconstruct will increase enormously and, with that, the number of objects to be visualized. Thousands of superimposed tracks and hundreds of very close vertices will be visible in any given event, besides the other objects. The development of new visualization software techniques, alongside the upgrade to modern technologies, will be vital to properly show such a large number of objects and to correctly interact with them. Due to the many superimposed objects that clutter the view and the difficulty of correctly selecting them, in the absence of new developments it will be difficult or impossible for physicists to visually investigate collision data effectively to debug their collection, reconstruction, and analysis.

A certain number of software and technology updates are already foreseen during Run 3 . These include the improved display of information on secondary and displaced vertices—to support the reconstruction and the analysis of New Physics signatures, as explained in §6.4—and the availability of 3D views in the "Online Event Display", the suite of interactive data visualization tools used in the control room of the ATLAS experiment to check the ongoing data taking.

A large number of new developments are foreseen for Run 4 in the context of detector description visualization. Work is currently ongoing to create visualization tools tailored to the detector geometry. The vision is to have standalone, interactive tools to inspect, query, and explore the geometry information tree and to handle geometry volumes. The new tools will play a key role in the implementation and debugging of the new detector description, letting developers of sub-systems implement new geometries in an efficient workflow with fast visual feedback.

The graphical rendering of physics objects will also be significant for visualising HL-LHC events. In other fields of research, such as studies of the brain and other medical applications, visualization techniques have evolved to let scientists effectively visualize objects in very dense environments. The visualization of those objects shares many of the challenges that must be addressed to dynamically and selectively visualize tracks and vertices in HL-LHC events. Development work will be needed to replace the graphics

engines used in our current visualization tools with more modern incarnations and to develop new rendering techniques to visualize physics objects more efficiently in highly-populated events.

Following the recommendations from HSF [32], we will also explore decoupling the visualization of the experimental data from the experiment-specific data retrieval and handling. This server-client approach will offer more flexibility to exploit new technologies like web-based 3D rendering, or Virtual and Augmented Reality. It will also facilitate the co-development of common visualization packages across experiments. Major design and development work will be needed to develop a client-server pipeline. In return, this approach will offer major returns in terms of future visualization software development and maintenance effort.

8 Analysis model

8.1 Introduction

While the bulk of computational resources at the LHC are used for preparing real data and simulated events, most individuals in the collaborations spend the majority of their time on the analysis of the data collected by the experiment, to produce physics results. It is imperative that the analysis model is conducive to a streamlined workflow which can be used efficiently by physics analysts. In light of the data volumes expected at the HL-LHC, an evolution of the current data reduction workflow [33] that consolidates the production of calibrated event data for analysis is important. In the baseline analysis model, ATLAS sees the full deployment and adoption of a new analysis model which will be introduced during Run 3. In addition, on the timescale of the HL-LHC, the tool-kit accessible to physicists will broaden as industry tools for large scale data analysis are incorporated into the HEP ecosystem. The analysis model must serve both traditional and modern analysis tools. While these are not 'baseline', the early adoption of industry-standard tools should be encouraged and should replace traditional tools during Run 4.

8.2 Analysis Data Formats

In Run 2 a centralized data reduction system (the Derivation Framework [33]) was introduced. This allowed individual analysis teams to define formats (referred to as Derived AODs, DAODs, or just 'derivations') tailored for their specific analysis. These formats (of which over 80 are currently defined) are derived from the output of the reconstruction by removing unwanted variables (slimming), objects (thinning) and whole events (skimming), and adding new variables or objects as required by the individual analysis teams. The per-event content of the retained objects is standardised to ensure that all variables required for object calibration and the assessment of instrumental uncertainties are stored without individual analysts needing to know the full list of variables required. While highly successful in terms of utility for analysts, a significant overlap in the output formats produced by the various analysis groups is observed, especially for Monte Carlo. This leads to a heavy disk footprint for the analysis formats.

To address this, ATLAS assembled an Analysis Model Study Group for Run 3 (AMSG3) in late 2018, which delivered its final recommendations in mid 2019 [34]. A key component of the Run 3 model is the introduction of two new common unskimmed data formats, `DAOD_PHYS` ($\sim$ 50kb/event) and `DAOD_PHYSLITE` ($\sim$ 10 kb/event). In common with the other analysis formats, the `DAOD_PHYS` format is designed to include sufficient event data to perform final object calibration and systematics, with those object calibrations being applied at the user analysis step. This format will replace the majority of the existing DAODs, thereby yielding sizeable savings by reducing the overlaps between the formats. Centralised skimming (event removal) will still be offered to individual analysis teams who require only a small fraction of the events for their analysis. The light-weight `DAOD_PHYSLITE` format will be a centrally produced data format in which all object calibrations are already applied, thereby permitting many variables to be dropped. In the current computing model, a large amount of disk space, and significant human time, is typically taken up by processing and storing systematic variations on the calibrated objects. In order to access systematic variations in the `DAOD_PHYSLITE` event data it will be required that they can be retrieved using standard interfaces based on (for example) particle object interfaces.

The goal of these data formats is to cover the needs of up to 80% of ATLAS analyses. The remaining analyses, principally those looking at long-lived particles and exclusive B-hadron decays (whose study requires detailed inner tracking information), and also those analyses establishing common calibrations

and systematics, will continue to be served with custom Run 2 style DAODs and raw data filters. For these exceptions, every effort must be made to reduce the number of stored events. The DAOD_PHYS format is intended to be the primary format for Run 3, with DAOD_PHYSLITE acting as a useful means of running fast analysis. Run 4 should see the large majority of analyses making full use of DAOD_PHYSLITE. Since this is a new format, with which ATLAS currently has no experience, the degree to which it is adopted by physicists is one of the parameters of the resources usage modelling.

8.2.1 Analysis storage needs

Assuming 2×10^{11} simulated events per year, the expected size of DAOD_PHYS will be 10 PB per year, while for DAOD_PHYSLITE 2PB are projected. For data, 7×10^{10} events are expected annually and so the sizes of DAOD_PHYS and DAOD_PHYSLITE will be 2 PB and 0.5 PB respectively. Since 80% or more of analyses are expected to use these formats, the AODs (200 PB per year for MC and 35 PB for data) need not be available on disk but rather served from tape using a data carousel approach (§10.4). Additionally, it is planned that DAOD_PHYSLITE can be derived from the already-reduced DAOD_PHYS format rather than from AOD. Under this workflow the AODs will only need to be accessed when calibration algorithms and recommendations are renewed, which will imply that the analysis formats must be re-made. A yet more aggressive option would be to forgo persistent storage of most AODs and instead rely on the ability to re-generate AODs on-demand. Fast simulation and reconstruction would be mandatory for such an approach. A similar strategy was successfully pursued for the much larger unabridged reconstruction output format known as Event Summary Data (ESD) before Run 2.

8.2.2 Potential future improvements

The DAOD_PHYSLITE format should enable columnar data access and enable the collaboration to provide analysts with an EDM on which analysis tools can rely. The xAOD event model used for all current and envisaged analysis formats, which comprises jagged data containers and light-weight "interface EDM classes", is very similar to patterns that are emerging in new HEP data analysis tools. Columnar data may be served to the analyst either over the network or from on-disk columnar data formats such as ROOT and Apache Parquet [35].

8.3 Analysis workflows

Broadly speaking there are three paths an analyst can take to go from the common derived data to plots, tables and statistical analysis. The path will depend on the type of data the physics analysis needs.

1. DAOD_PHYSLITE - This format will be easily readable without a collaboration-specific framework. A light-weight event data model library using common tools will be provided to aid in its use. For example, this library will make it easy to derive systematic variations. Development work is required to get to baseline usage.
2. DAOD_PHYS - In common with the Run 2 AOD and DAODs, this is an xAOD format. In that sense, the tools and frameworks are already currently available. CP tools, which apply the object calibrations, should continue to be maintained as they are for Run 2. Furthermore, those CP tools will be used in the production of DAOD_PHYSLITE.
3. DRAW and Run 2 style DAODs - A small number of analyses will not be able to use either the DAOD_PHYS or DAOD_PHYSLITE format. Long-lived particle searches are an obvious example. Evaluation of the parameters needed to set calibrations is also likely to require dedicated DAODs, as in Run 2. The workflow for these analyses will look very similar Run 2: a derivation will be produced, and analysts will run on that derivation.

ATLAS will have to supply the resources to run on these centrally produced datasets. The model for DRAW/DAODs is exactly the one used in Run 2: the physics analyst will submit a grid job to process the datasets and then download the resulting datasets locally. These are expensive jobs as calibrations have to be applied and systematic errors derived - which frequently takes about 1 second per event. The DAOD_PHYS dataset will have as size of around 10 PB per LHC running year, and will have to be produced according to a planned and organised schedule rather than on-demand. Such production runs will be restricted to a few times per year. The remnant Run 2 style DAODs will also need to be made in the shadow of these major production campaigns, which will require some changes to the working practices of the physics groups. DAOD_PHYSLITE, at about 2 PB/year, is small enough not to require any special

approach to run. The same is true for skims of DAOD_PHYS. It is assumed that physicists will submit jobs on the grid as required to run against DAOD_PHYS, DAOD_PHYSLITE and skims thereof. The experiment will need to maintain enough copies of these formats to support quick and efficient access. If this is not the case, analysts are likely to create and save new derivations, nullifying the potential disk space savings.

A crucial element of analyses is the interpretation of results using suitable statistics software. In Run 2 this often required format conversions to standalone n-tuple formats. The software making these formats is developed independently from the xAOD-based formats by individual teams and often incurs significant storage cost for the analysis teams. Minimizing the overhead of using a light-weight event data model with DAOD_PHYSLITE and enabling its 'framework-less' processing should reduce the need for such format transformations.

8.3.1 Potential future improvements

The workflow outlined above corresponds to the traditional practices followed by ATLAS analysts during Run 2. There are several developments in the HEP software ecosystem that could improve the efficiency with which ATLAS accesses and reduces the data: the new RDataFrame and RNTuple data structures from ROOT, tools from the Python ecosystem, and distributed analysis and modern high-performance distributed tools. Finally there is also the possibility to use co-processors to speed up analysis (for example GPU or FPGA).

RDataFrame and RNTuple A production release of the latest version of ROOT - ROOT7 [36] - will soon be available for the community. Two components within this major release are of particular interest to ATLAS analysts: RDataFrame and RNTuple. RDataFrame [37] is a declarative analysis tool optimized to process large amounts of ROOT data with a transparent API. It supports multi-threaded running and simple selection semantics, along with virtual columns and other features commonly needed during data reduction. It is expected that RDataFrame will be one of the main methods by which analysts will inspect DAOD_PHYSLITE within the ROOT environment. Development work by ATLAS must be done to make sure that the DAOD_PHYSLITE libraries can be used in the RDataFrame multi-threaded environment. ATLAS should also track Analysis Facilities developments based on RDataFrame.

RNTuple is a new high speed data format and is the next generation of the well known TTree and TFile. This is a potential new file format for DAOD_PHYSLITE and ATLAS should allocate time to investigate its performance characteristics such as speed and storage space on several analysis platforms commonly used by ATLAS.

The Python Ecosystem as an Analysis Platform In the wider domain of data intensive physical sciences and ML, the use of the Python programming language as the primary entry point for data analysis has increased significantly and a collection of tools, commonly referred to as the 'scientific Python ecosystem', has emerged. The core packages of this ecosystem are numerical libraries for vectorized array computation, NumPy[10], algorithms (SciPy [38]), Visualization (Matplotlib [39]) and others. Seamless integration with this 'ecosystem' is highly desirable, to enable the collaboration to capitalize on distributed computing for data analysis developed outside of HEP. An important connecting component here is the uproot[40] library, which provides read and write capabilities for the ROOT format.

Fast Serving of Data to the analyst Data storage in the HL-LHC era may evolve towards a Data Lake managed by Rucio, as described in §10.5.3. To improve access, tools that perform basic transformations near the data, such as intelligent Data Delivery Service (iDDS) [41] and ServiceX [42] projects, are being developed. The tools, working together, with high bandwidth access to the data, could skim and apply basic transformations. For example, the analyst could request all events with 2 clean jets and jet $p_T > 300$ GeV, and have it delivered in close to real time. It is assumed that the DAOD_PHYSLITE format would be main source of data for such operations. The nominal access pattern for this sort of data is to submit a grid job: this will run on a dedicated cluster optimized to access the data and deliver only what is needed. Output formats include ROOT based files as well as columnar data (in the Apache Arrow format [43], for example). Groups in ATLAS are working on these tools, and support from ATLAS will be required if they are to be integrated into operations.

Distributed interactive analysis, out-of-core Data Frames While ultimately the majority of analysis computation will be performed in a non-interactive manner, interactive analysis remains crucial during the development of an analysis. At the HL-LHC, such interactive analysis will necessarily require processing data beyond the memory capacity of individual compute nodes. In recent years, the approach of interactive analysis interfaces, such as Jupyter Notebooks, connected to horizontally scalable compute clusters such as Spark [44], Dask [45] or Ray [46] or batch systems has proven to be promising. In the current analysis landscape, most frameworks are only interfaced with batch systems in order to schedule scale-out workload.

For future developments, integration with industry-standard tools and environments appears to be the most promising direction, either through direct integration by ROOT via RDataFrame or RNtuple processing, or by leveraging packages in the HEP Python ecosystem discussed above. Any interactive scale-out system should seamlessly integrate with the data-access infrastructure (iDDS, etc) provided by the experiment.

8.3.2 Analysis Facilities

As discussed above, ATLAS relies on its central production and data management systems to produce and deliver DAOD datasets to the analysis groups. The final steps of an analysis, producing analysis n-tuples and from them the physics distributions, statistical workspaces and final results, are left to groups and individuals. Traditionally the grid resources have been used for bulk user analysis such as dataset preselections. For iterative, though largely batch-based analysis, analysts utilise the data managment system to transfer the relevant datasets to local batch clusters or large facilities at labs such as LXPLUS (CERN), NAF (DESY) or the shared Tier-3 at BNL. These resources provide the user with faster turnaround times, stable storage and fine-grained control over workload submissions crucial for later stages of the analysis. Explorative, interactive processing using (for example) Jupyter Notebooks is mostly constrained to single-node analysis, statistical analysis and result visualization.

For the HL-LHC, it is desirable that users have access both to resources that support running both asynchronous, batch-like workloads as well synchronous interactive, but horizontally scalable workloads as described in the previous sections. Thus there is an opportunity to bridge both the gap between batch on the grid and on shared facilities as well as between non-interactive and interactive analysis by expanding on the grid and local batch/Tier-3 concepts. Resources that can provide such integrated solutions are referred to as 'Analysis Facilities'. Analysis facilities focus on interactivity, usability, and strong user support. They are more defined by the set of applications that they offer rather then the resources on which they run. As such, analysis facilities may be integrated into large existing resources such as Tier-1s or be fully virtualized and deployed on private or commercial cloud providers. Here first experiences have been gained with the CSU Virtual Tier-3 [47]. Typical applications available on analysis facilities may include classic batch systems, Jupyter Hub [48] deployments for notebook computing, or dataframe processing clusters such as Dask, Ray or Spark.

To ensure equitable access, such facilities should be integrated into the federated computing infrastructure of the collaboration. As with classic grid use, analysts should be able to move their analysis work seamlessly between such facilities and resources providers should be able to easily provide facility services. This is enabled by both using a collaboration-wide identity system and common applications such as Jupyter Hubs.

See §10.5.2 for the distributed computing aspects of analysis facilities.

8.4 Integration with machine learning

The use of advanced ML techniques has steadily increased in recent years and these techniques are expected to become an integral part of many analyses of LHC collision data. It is therefore important to ensure that the use of these techniques is smoothly integrated into the overall analysis model.

Currently ML workloads and standard analysis work are often developed independently of each other, primarily due to the dissimilarity of the software stacks. Standard analysis work is often conducted within the centrally provided Analysis Releases, while ML workloads, such as training and optimization, is performed using industry ML frameworks primarily using Python-based interfaces. Most often, inputs are not read in ROOT-based formats but rather preprocessed to produce specialized formats based on data structures such as HDF5.

A more integrated workflow is achievable through two key technical capabilities:

- Ensure ATLAS analysis releases do not require many system dependencies and are easily installable alongside other software (e.g. sharing dependencies such as language runtimes)
- Streamline format conversions within python through libraries such as uproot, which can convert ROOT-based xAOD files directly to python-based arrays.
- Evolve the xAOD format toward a more ML-friendly structure, both by arranging data in a more columnar format and by investigating backends beyond ROOT.

The paradigm of Deep Learning is evolving into a more general notion of numeric programs that are end-to-end optimizable through the use of auto-differentiation techniques. Here, projects such as pyhf[49] use ML libraries not for standard neural network training but rather to compute traditional HEP targets such as profile likelihoods leading to significantly higher performances for (e.g.) maximum likelihood fitting or limit calculation. Such differentiable analysis modules will also allow more direct integration of systematic uncertainties into the ML training, leading to more robust classifiers that are well-aligned with the final science objective.

Enabling the seamless use of supported ML frameworks within analysis code will similarly allow an easy utilisation of hardware accelerators such as GPUs or TPUs as the frameworks already provide the necessary low-level code to address such accelerators and no detailed accelerator know-how is required on the part of the analyst.

8.5 Analysis Preservation, Reusability and Data Products

In Run-2 ATLAS has made significant progress in the adoption of industry tools to ease analysis development, deployment and preservation. In particular, wide adoption of centrally built container images for Analysis Releases have helped the adoption of continuous integration and deployment techniques. A major use-case for this is the preservation of data analysis pipelines for later reuse in combined summary analysis such an ATLAS-wide assessment of SUSY [50] using the RECAST framework [51]. As the LHC experiments enter an era of high-precision measurements such combinations of multiple disparate analyses will be crucial to maximally exploit the physics potential of the HL-LHC.

As stated earlier, it is expected that the use of industry tools will increase in Run 3. At the timeline of the HL-LHC, the analysis model should allow a seamless deployment and sharing of data analysis pipelines within the collaboration, such that a combined analysis can be performed independently of the original analysis authors.

As the analyses on the full HL-LHC dataset will form the legacy measurement of the LHC at the final centre-of-mass energy, the preparation of suitable data products for measurements and searches is a priority. A common HEP-wide public data repository such as HepData is thus crucial and widely used within ATLAS today. As a baseline all tabular data and auxiliary material is provided on HepData and more recently ATLAS has begun publishing the full statistical model (the 'likelihood'), including a full list of systematic uncertainties, which enables external researches to correctly incorporate ATLAS results in combination such as global fits.

8.6 Non-standard workflows

ATLAS has been employing non-standard data taking and analysis workflows to increase the amount of data that can be made available beyond what is foreseen in the baseline scenario. These workflows target specific physics use cases that would not otherwise be covered, without significantly adding to the overall computing costs. Some of the physics motivations for the use of these techniques are outlined in the Trigger and Data Acquisition (TDAQ) TDR [22]. Non-standard workflows are designed to overcome storage and CPU limitations that prevent data accepted by the first level triggers (and often also processed by the high level trigger) to be stored and processed. These workflows use data formats where only subsets of the full event information are saved in dedicated data streams, and/or data reduction is performed online in the trigger computing farm. This enables a variety of physics analyses that would otherwise be unfeasible using conventional triggers, due to the combination of large (raw and analysis-level data) event sizes and high data taking rates. [2]

[2] Efforts on developing HL-LHC non-standard workflows have been encouraged by the HEP Software Foundation [52, 53, 54].

One non-standard data taking technique deployed in ATLAS is called Trigger-object Level Analysis (TLA) [55] [3]. In TLA, physics objects (such jets, photons, electrons and so forth) are reconstructed in real-time within the HLT. Only this reduced data, with an event size corresponding to less than 1% of the total event in the case of jets, is saved for offline analysis [55, 58] at much higher rates than in traditional analysis formats [4]. So far, this technique has been successfully used to search for low-rate dijet resonance signals with high-rate backgrounds [55]. With the expanded capabilities offered by the TDAQ upgrade, the TLA technique will allow searches for new particles at the HL-LHC to retain the low analysis thresholds that allow the probing of rare processes at the electroweak scale. Availability of tracking information at the trigger [22] is needed to mitigate the effects of pile-up at the HLT if this approach is to be effective. The TLA approach motivates work towards the continued improvement of the reconstruction and calibration of trigger objects [58]. This is in line with the ATLAS strategy of minimizing differences between online and offline physics object quality, as it benefits bandwidth and storage optimization.

Another technique is Partial Event Building (PEB), where only a subset of raw detector information is recorded in selected regions of interest, for instance near to objects reconstructed directly in the high level trigger. This technique can be useful for processes where new particles leave complex yet localized non-standard signals in the detector (such as jets from dark sector cascades), and where the distinctive features of the signals would be too time-consuming to be reconstructed in the HLT. For example, if sufficient tracking information is available, high-precision $b-$tagging with offline calibrations can be done using the PEB technique for only for the jets of interest. In Run-2, PEB was restricted to calibration and B-physics [59] and in the 2018 heavy ion run, and there are plans to extend this to more use cases and physics objects in Run 3 in combination with the TLA technique. The use of TLA+PEB will require multiple working points, to be optimized depending on the application and detector subset desired. The event size ranges from that of the format used for the Run 2 trigger-level analysis of dijets (factor of ≈ 200 smaller), to full-detector information in the regions of the detector corresponding to prominent objects (factors of $\approx 2-4$ smaller).

So far, the TLA-only workflow has been running in the shadow of traditional workflow, both in terms of CPU and storage. It is expected that this program will grow, especially for non-standard signatures. The addition of more complex objects to be reconstructed at the HLT as well as the addition of information in the reduced event data format will expand analysis prospects at a fraction of the cost of traditional workflows, but still will not be completely cost-free. For this reason, the physics opportunities and corresponding needs will have to be considered in the resource planning of the overall ATLAS physics program.

In preparation for Run 3, prototypes of more generic TLA streams are being developed and the implementation of PEB for selected signatures beyond muons (e.g. jets in dark sectors) is being carried out. The main challenge for PEB will be the reconstruction of physics objects using only selected regions of the detector. At the HL-LHC, it will be crucial to have fast and flexible tracking algorithms implemented at the HLT in order to face the harsher pile-up conditions. A much higher-rate version of TLA could also be implemented by reading out selected physics quantities from the upgraded framework of the low-level (Global) trigger directly into analysis-level histograms. While this does not impact the computing resources, the main challenge for the success of this scenario is the calibration of analysis-quality physics objects with a much more limited set of information, and it will require further studies that benefit from the experience gained with the calibration of TLA objects.

9 Tier-0 and HLT farms, CERN infrastructure

9.1 Trigger and DAQ

The Phase-II upgrade of the ATLAS Trigger/DAQ (TDAQ) system and its Event Filter (EF) is described in detail in Ref. [22]. The EF system processes the events in almost real-time, takes the final trigger decision and assigns events to data streams. The EF farm will be built from commodity CPU servers with the optional addition of accelerators (GPU, FPGA) if they are found to be cost effective. The Dataflow system stores the accepted events in files and publishes them to the offline system. It is able to buffer

[3] This technique is called Turbo stream by the LHCb Collaboration [56], Data Scouting by the CMS Collaboration [57].

[4] Care must still be taken that the trigger-level reconstruction does not exceed available CPU resources, and that adding more HLT information does not lead to a significant increase in event size.

events for up to 48 hours if needed. During non data-taking periods the EF farm acts as a standard Grid site and can be used to run e.g. simulation jobs ("Sim@P1").

Based on the current trigger menu draft ([22] Table 6.4) an average EF output rate of 10 kHz at $\mathscr{L} = 7.5 \times 10^{34}\,\mathrm{cm}^{-2}\mathrm{s}^{-1}$, corresponding to approximately 200 inelastic proton-proton collisions per bunch crossing, is expected. This rate includes the main physics stream(s) but does not include detector calibration or Trigger-object Level Analysis (TLA) streams. TLA streams typically have a very high rate but with very small event sizes of the order of 10 kB. Based on data from Run-2, those additional streams contribute about 20% of the total bandwidth and storage at the Tier-0.

The estimated event size for each detector system at the ultimate Phase-II luminosity is listed in Table 7 resulting in a total average event size of 4.4 MB. This is a slight reduction compared to the 5.2 MB listed in the TDAQ TDR ([22], Table 3.5) due to the reduction in the Pixel event size in the latest iteration of the readout chip. The event size of the new High-Granularity Timing Detector (HGTD) has been taken from Figure 4.26 in Ref. [60]. Allowing additional bandwidth for calibration and TLA streams, the total bandwidth to Tier-0 is approximately 45 GB/s.

Detector	Pixel	Strip	HGTD	LAr	Tile	Muon	TDAQ	Total
MB/event	1.4	0.5	0.2	0.7	0.2	0.8	0.6	4.4

Table 7: Expected average event size at $\langle\mu\rangle$=200. Forward detectors are not listed since the associated event size is negligible.

9.2 Tier-0

One of the basic functions of the Tier-0 is the recording and archival of raw data, and its export to Tier-1 centres. We assume that this will remain the case in Run 4. In order to achieve that, the Tier-0 must be endowed with sufficient network bandwidth and storage capacity (both disk and tape). Estimates on the expected data volumes are presented in §12.3. Here we complement them with estimates on the required Tier-0 network capacity for Run 4 (cf. §9.2.1).

Performing the processing of data at Tier-0 in Run 4, in the way it was done in Runs 1 and 2, and will be done in Run 3, will require a substantial upgrade of the Tier-0, in the most optimistic scenarios by a factor of 4. §9.2.2 gives estimates on the required CPU capacity, depending on the reconstruction timing estimates of §6. Depending on available processing capacity at Tier-0, various processing scenarios are discussed in §9.2.3.

9.2.1 Tier-0 Network Traffic

To estimate the necessary Tier-0 network capacity in Run 4, we make the following assumptions:

- RAW data recording: 10 kHz physics rate, $\sim$ 4.4 MB full physics event size (estimate for $\langle\mu\rangle$=200, cf. Table 7 above), 20% of bandwidth occupied by non-physics streams (calibration, TLA, etc.);
- 70% LHC time in stable beams;
- Raw data backed up to tape and exported to Tier-1s as fast as possible (“live”);
- Tier-0 processing spread over fill and inter-fill periods ($\Rightarrow$ scale factor 0.7 applied on data rates);
- Reconstruction outputs: AOD of 700 kB/event, total data volume of other products (DRAW, performance DESD, DAOD, etc.) not more than that of AOD;
- Merging of reconstruction outputs in a separate, subsequent step.

Table 8 shows the estimates on the required Tier-0 network traffic and bandwidth. SFO (Sub-Farm Output) is the online DAQ service that transfers the raw data from the detector to the CERN computing centre. EOS is the disk storage service at CERN [61]. CTA refers to the CERN Tape Archive, the service that manages data recording and retrieving data from tape. It replaces the old CASTOR system. DDM refers to the Distributed Data Management system (§10.3).

The figures do not contain contingencies. Extra capacity of 30–50% would cover all conceivable circumstances. On the other hand, a fraction of 70% in stable beams is a very optimistic assumption that already leaves some headroom.

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**

Table 8: Estimated Tier-0 network traffic for Run 4.

9.2.2 Tier-0 CPU Capacity

To estimate the necessary Tier-0 CPU capacity in Run 4, we make the following assumptions:

- 10 kHz physics rate;
- 70% LHC time in stable beams (⇒ 7 kHz effective reconstruction rate required);
- 80% of CPU used for physics processing, 20% for everything else (calibration/alignment, Data Quality Monitoring (DQM), merging, etc.);
- Reconstruction timing estimates according to §6, Table 6;

As a reference, the Tier-0 cluster capacity at the end of Run 2 (2018) was 430 kHS06. Table 9 shows the estimates on the required Tier-0 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)

Table 9: Estimated Tier-0 CPU capacity for Run 4, for the "Baseline" and "Conservative R&D" (in parenthesis) scenarios, respectively (see Table 6).

9.2.3 Considerations on Tier-0 Workflows in Run 4

Prompt processing for calibration, alignment, beam-spot determination, DQM and such like, has to be accomplished within a time window of 48 hours (the so-called “calibration loop”) to provide fast feedback and turnaround. It relies heavily on CERN-based infrastructure, such as live Detector Control System (DCS) information from local conditions databases. The Tier-0 is and will remain the most suitable resource to run prompt processing.

In Run 2, O(10%) of CPU resources were used for calibration- and alignment-related processing. One may expect that the total capacity needed to perform all the calibration processing will remain approximately the same. In any case it will not have to scale up 1:1 proportionally to the physics rate.

In Run 3, it is foreseen that the Tier-0 will have sufficient capacity to perform all the bulk processing (physics_Main stream), as with previous runs. Bulk processing is launched after the 48-hours’ calibration loop. Launching the reconstruction for newly recorded data within a short time following the end of the run, and executing this reconstruction on a central facility at CERN, have to date been strong requirements by the Data Preparation and Physics communities. The advantage is the fast availability of outputs that can be directly used for physics analysis. Accompanying DQM and other Combined Performance (CP) processing, which relies on the full available statistics and often makes use of the local CERN infrastructure, can be performed promptly as well. For instance, DQM results are used for run-by-run sign-off and the compilation of Good Run Lists.

Running first-pass bulk processing fully at the Tier-0 has proven to work well and reliably, and it also has considerable benefits as described above. The proposal is to maintain this processing model in Run 4 and to equip the Tier-0 with the corresponding CPU capacity.

Since Run 2, the Tier-0 batch farm has been configured in a way that its CPU resources can be shared between Tier-0 and grid processing. In periods of data taking and processing, Tier-0 runs with priority, otherwise the grid takes over the resources. This ensures that the sizeable capacity of the Tier-0 is used efficiently at all times. After a large upgrade of the Tier-0 CPU capacity for Run 4 there would be no risk that resources would run idle and be wasted in periods of no data taking – they would just be used by the grid.

A workflow has been commissioned and tested in Run 2 where prompt processing is outsourced to the grid, a mechanism known as 'spill-over'. This is in place to handle situations where the prompt processing requirements exceeds the available CPU resources at the Tier-0. This might occur if the LHC were to perform much better than expected with long stable-beam periods.

Spill-over was demonstrated to work satisfactorily when tested on the physics_Main stream for a limited number of runs, successfully passing validation. It has also been used in routine operation to process certain special heavy ion data streams. Tasks were defined at Tier-0 and injected through a python API into the grid production system. Reconstruction and merging were run on the grid, outputs (AOD, HIST) were shipped back to CERN for further special processing at Tier-0 that could not be done on the Grid (e.g. DQM).

Development work on easier interaction between the Tier-0 and Grid Production Systems, and a better integration of the respective monitoring tools and interfaces, is already planned for Run 3, in order to fully and efficiently exploit the potential of the spill-over workflow. Spill-over will be ready to use for Run 4 data processing.

10 Evolution of distributed computing

It is anticipated that distributed sites will continue to provide the bulk of resources used for data processing, simulations and analysis in the HL-LHC era. This infrastructure is expected to be provided via WLCG mechanisms based on bilateral agreements (MoU) between WLCG and contributing funding agencies that cover the entire duration of the LHC program, including HL-LHC. ATLAS has a sophisticated distributed computing system (ADC) that optimally makes available hundreds of clusters and associated storage at distributed WLCG sites. ADC provides a fully integrated system for all workflows and for all users. The automated tools developed by ADC include ProdSys, PanDA, Rucio, HammerCloud, and others. These tools will continue to be upgraded and evolve as needed. Many of the upgrade plans require tighter future integration among tools to improve efficiency in the more complex environment of the HL-LHC. This section describes examples of long range R&D plans which are needed to transition to the new challenges at the HL-LHC. The metrics used to evaluate progress with the R&D projects will include: scalability for HL-LHC, improved efficiency in resource usage (CPU, storage and network), and improved operational efficiency. Some of the R&D projects described here will require additional effort or redirection of effort.

10.1 Evolution of WLCG

WLCG aims to accommodate many scientific endeavours in the future, and already DUNE [62] and Belle-II [63] have joined the infrastructure. More projects are expected to join soon, most notably, those that take part in the European ESCAPE [64] cluster of ESFRI activities. The overarching challenge for all these projects is distributed data handling, so WLCG focuses on R&D in the areas of Data Organisation, Management and Access (DOMA), which is expected to have relevance to the data infrastructure and services.

In particular, the Data Lake approach (see §10.5.3), under development and evaluation by current R&D projects (DOMA, ESCAPE, IRIS-HEP, and others) foresees consolidating storage at a few federated regional centres, with multi-level caching (using for example Xcache) providing a content delivery network down to the CPUs. This approach will reduce storage operation cost but also bring implications on data placement, delivery and caching, which will need to be properly accommodated by ATLAS tools and services. Another area in DOMA aiming to reduce storage cost (and thereby provide more capacity) is the Quality of Service (QoS) R&D, which defines different categories of storage based on

latency, reliability and performance. ATLAS systems must be able to efficiently use and manage the transition between different QoS to maximise the trade-off between storage cost and capability. DOMA will also modernise many of the legacy tools and paradigms created at the beginning of Grid computing, introducing new third-party transfer data transfer protocols and token-based authentication.

In §12.3 the resource challenges facing ATLAS in the HL-LHC era are shown. Many R&D activities have already started to address the CPU challenge – the CPU deficit should be substantially reduced through these efforts described in other sections. However, the storage and network challenges are formidable, and will be partly addressed through the WLCG DOMA project. Later in this section a few additional R&D projects to manage the storage shortfall are described, such as Data Carousel and iDDS, which were both started by ATLAS and becoming cross experiment.

10.2 PanDA and ProdSys

PanDA is the ATLAS workload management system used to execute all scientific workflows for all users on widely distributed heterogeneous resources. It is tightly integrated with the ATLAS distributed data management system, Rucio. For the HL-LHC, the PanDA development team has started multiple R&D efforts in collaboration with the rest of the ATLAS Distributed Computing community: the Data Carousel project, the ProdSys system, HPC and cloud orchestration, and the intelligent Data Delivery Service are prominent examples described separately. In addition, the PanDA team is working on global shares and unified queues, edge services through Harvester, MPI services, event service, SciTokens authentication, active network management, operational intelligence, anomaly detection, and ML techniques. More effort will be needed for these R&D topics in the next years. The overall goal is to meet the HL-LHC scale and resource capacity challenges without sacrificing the flexible ADC system.

ProdSys [65] is the primary interface between users and PanDA, providing a task management interface. New workflows are regularly developed in ProdSys. Monitoring systems are evolved to meet new requirements. The most disruptive challenge facing ProdSys is the growing importance of ML techniques, especially for distributed training. New schemas are being developed for non-traditional HEP use cases that are not collision event based. As new analysis models emerge for HL-LHC, we expect major additions to ProdSys capabilities. R&D efforts in ProdSys for the HL-LHC will be primarily focused on supporting non-traditional workflows.

10.3 Rucio

Rucio [66] is an open-source software framework developed principally by ATLAS that provides scientific collaborations with the functionality to organize, manage, monitor, and access their distributed data at scale. Rucio is the fundamental system in ATLAS for data management. For ATLAS the system manages more than 500 petabytes spread on 130 data centres with 600 storage locations, with daily data access of over 12PB, as well as over 2PB of data transfers, deletion, and recovery. The success of Rucio in the ATLAS experiment has led to it being adopted as the data management system by the CMS, DUNE, XENON, and AMS experiments and it is under evaluation by several other established and upcoming experiments such as Belle II, LIGO/Virgo/KAGRA, and SKA. Consequently, Rucio is becoming established worldwide as the community solution for scientific data management.

Future developments, specifically aimed at the HL-LHC era, aim to further improve the flexibility of the system, to ensure that the system can continue to operate efficiently at the requirements of HL-LHC data rates. These developments are ongoing and especially target improvements of the client interfaces, rule engine, deletion, and transfer components of the system. Further developments are planned to better support storage of HPC centres, evolve the metadata component to natively support future workflows such as ML, as well as to introduce the functionality of storage Quality of Service (QoS) to support a wider variety of storage. Rucio already supports numerous database backends, such as Oracle, MySQL, and PostgreSQL, and continuous improvements are introduced whenever new database features become available. Additionally, as more experiments and communities at similar data rates as the HL-LHC will come online in the next years, another development will be orchestration of dataflows across multiple experiments via cooperating Rucio instances, as well as direct integration with the network layer for dynamic provisioning of bandwidth.

10.4 Data Carousel and iDDS

The cost of disk storage will be prohibitive for current workflow patterns at HL-LHC volumes. Better use of cheaper media with appropriate QoS must be made – today this means magnetic tapes, but the method does not exclude other approaches, for example spun-down hard disks or erasure-coded file systems. The Data Carousel is a sliding window approach to orchestrate data processing across workload management, data management, and storage services with the majority of data resident on lowest-cost, and thus typically high latency tape storage. 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. This project should demonstrate that this is the natural way to dramatically reduce storage costs. The first phase of the project was started in the autumn of 2018 and was related to I/O tests of the sites archiving systems. The second phase requires a tight integration of the workload and data management systems and will be used already in Run 3 for reprocessing of RAW data and tentatively DAOD production, minimizing the disk space needed by the AOD. At HL-LHC reconstructed data will be stored primarily on tape with only very partial disk replicas and they will be processed mainly using the Data Carousel. This will require further developments such as the orchestration, through the intelligent Data Delivery Service (iDDS), of the activation of the jobs as soon as each file is successfully recalled from tape. The iDDS system in PanDA is being developed to provide streaming services for data delivery for a wide range of workflows. Data Carousel is an important and first proof of concept demonstration of iDDS for ATLAS.

10.5 Infrastructure

10.5.1 Facilities and operations

While the rigid hierarchical distinction between Tier 1 and Tier 2 sites already disappeared in ATLAS during Runs 1 and 2, these sites will continue to be the backbone of the computing infrastructure for ATLAS at the HL-LHC. Tier 1 sites are expected to continue providing a custodial service for primary data through tape storage. A more flexible classification of sites based on capabilities is emerging: sites will be exploited based on infrastructure and workload capabilities irrespective of their Tier label. HPC, Cloud and other opportunistic resources are expected to extend the capabilities of ATLAS managed Tier 1 and Tier 2 sites, but not replace their custodial roles or provide the wide array of time-critical processing and storage services.

10.5.2 Analysis Facilities

During Runs 1 and 2, WLCG Tier 1 and 2 sites provided distributed processing capability for ATLAS analysis users, at the scale of a million analysis jobs executed per day. However, these facilities are not well suited for the final interactive stages of user analysis, for example visualizations, plot generation, re-weighting, systematic studies, limit testing, etc. Local sites (often called Tier 3 sites), funded and managed outside the WLCG framework, often provided the necessary resources for end-user analysis. Many of these sites are connected through ADC tools like PanDA and Rucio. The future of these analysis facilities (also discussed in §8.3.2) depends on the success of new analysis tools and analysis models. R&D work is needed to determine the size, scale and cost of analysis facilities. Metrics of reduced cost, increased efficiency and scalability need to be demonstrated. This R&D work is dependent on the analysis tools R&D described earlier in the document, and will require additional effort to evolve the future analysis facilities.

10.5.3 Data Lakes

One of the directions we are actively pursuing for grid storage provided by WLCG resources is to extend the concentration of disk resources at fewer and larger, possibly federated, sites. Smaller cache-like storage would serve the more widely distributed computing resources. The reasoning is that the provisioning and operation of robust storage requires significantly more manpower than needed for a compute cluster with a cache of secondary replicas. In principle, distributed operations are simplified in this scenario, although separation of storage and computing raised their own reliability and operational challenges in the past. Results from a WLCG survey suggest the staffing requirement is perhaps not a driving factor for a typical Tier 2. Also local and regional funding tends to work against concentration of resources. Nevertheless, some sites or funding agencies have already chosen to connect computing

facilities to remote storage through cache layer. ATLAS needs to prepare for a significant fraction of remote accesses.

10.5.4 Network

The exponential growth of essentially free research network bandwidth has thus far enabled an abstraction of the ATLAS data transfer activities from the underlying network. The major developments likely to affect ATLAS are packet marking, new cost models and programmable WAN links.

Packets traveling on dedicated WLCG sections of the network will be marked by the application, in order to attribute the activity to a particular Virtual Organization (VO). This capability is especially important because it is not always clear what the impact to wide-area networking is when making changes to complex, global infrastructure. Being able to identify owners and types of traffic flows anywhere in the network makes it possible to identify the root cause for significant changes in network traffic. Another consideration is that the cost for a particular bandwidth, for example transatlantic, may become more direct, rather than hidden in general research network budgets. In addition, usage of commercial cloud storage egress incurs very clear costs for the experiment. Programmable WAN links offer the potential to be able to boost bandwidth between sites on demand.

Rucio has a matrix of connectivity and bandwidth between sites, with information taken from the File Transfer Service (FTS) and manual input. This will be augmented with information about cost and potential programmable links. PanDA can use this to influence job placement, but will need improvements to optimize for speed or cost.

Much of this work is being discussed in twice yearly LHCONE/LHCOPN meetings and has been reported on by the HEPiX Network Function Virtualization working group in their phase I report [67].

It is important to note that ATLAS will need to work more closely with both the national research and education networks, the various networking research efforts, and the other experiments to effectively prepare for the HL-LHC era. In early 2020 there are efforts underway to create a networking technical working group to discuss, document and prototype capabilities identified as being important for the LHC experiments. ATLAS also needs to be ready to take advantage of international scale network testbeds capable of providing a geographical footprint, advanced services and capacities relevant for the HL-LHC. Testbeds like that will be critical for testing network capabilities and services to evaluate their impact for ATLAS in the context of HL-LHC.

10.5.5 High Performance Computing resources

Historically, some significant resources were pledged to ATLAS through allocation or dedicated partitions at generic public High-Performance Computing facilities. One notable example is the NDGF Tier 1 [68] where all computational power is provided via national research data centres shared with other scientists. It is anticipated that in future more resource providers will use this model, and a larger part of the pledged resources will come through non-dedicated shared facilities. While they will still be a part of the WLCG infrastructure, ATLAS will have to cope with the necessity to adapt to such resources in terms of e.g. processor architectures, operating systems, file systems, quotas etc.

At the same time, for the last four years large HPC facilities, for example the Leadership Class Facilities funded by the U.S. Department of Energy, have allocated to ATLAS significant CPU resources[5], and are also allowing ATLAS payloads to run opportunistically in "backfill mode" [69]. With an increased investment worldwide into exascale computing, it is safe to assume that HPC centres represent a significant opportunity for ATLAS computing. They also represent a significant challenge: most ATLAS HPC allocations are outside of the service levels as defined in the WLCG MoU [70] and not subject to WLCG policies and operational procedures in general. Adapting ATLAS workflow and data management systems to a variety of HPC software platforms and system policies has been a major challenge for ADC. An even more significant challenge is to run ATLAS software efficiently on an ever-increasing range of parallel computing architectures (§2). ATLAS is approaching these challenges by focusing the porting efforts on the most CPU-intensive workloads (currently detector simulation). Crucially, ATLAS is also engaging HPC centres and parallelization experts worldwide. Besides enlisting their help in application porting, the long-term goal is to increase the HPC community awareness of the unique performance characteristics and policy requirements of HEP workflows.

[5] roughly corresponding to a large Tier 2 centre

10.6 Environmental impact of ATLAS computing

As concerns grow about the impact of anthropogenic greenhouse gas emissions on the climate, it is reasonable to ask about this topic in the context of ATLAS computing. Obtaining a meaningfully precise estimate of the tonnes of CO_2-equivalent emissions per unit of computing or storage used by ATLAS is difficult. This is partly because many of the data centres that provide ATLAS with computing services are unable to factor out the fraction of their power consumption due to ATLAS (or depending on the internal accounting, even the general computing usage). Secondly, a large fraction of the power consumed by data centres hosting ATLAS resources is provided by grid companies that do not reveal how much carbon is released per kWh of electricity supplied.

Rather than quoting a hopelessly inaccurate figure, ATLAS has instead compiled a list of good environmental practices undertaken at various data centres, and publicised them within the community in the hope of inspiring other sites to follow. The measures reported range from the installation of a large wind turbine on a hillside near a British university for direct provision of power to its data centre, to a wide range of techniques to improve the efficiency of cooling, notably in several German and Canadian sites. Significant numbers of the ATLAS sites are in countries with low-carbon electricity (especially France, Switzerland and the Nordic countries) and in the US and Canada there is a concerted push to preferentially use low-carbon sources such as hydro-electric or nuclear power. Most data centres reported investing in more energy-efficient equipment, and also the use of sustainable recycling services when old equipment is removed.

10.7 Summary

The various R&D activities described in the section above can be expressed in the following broad categories.

Baseline: Consolidation of smaller sites to reduce operational load through DOMA projects. Evolution of PanDA, ProdSys and Rucio to enable new workflows at the HL-LHC, for example ML. SciTokens, non-gridFTP TPC, event service, network monitoring, operational intelligence.

Conservative R&D: PanDA, ProdSys and Rucio scaling for HL-LHC, intelligent data delivery service (iDDS), data carousel, active network management, integration of HPC, cloud and alternative architectures, basic QoS capabilities.

Aggressive R&D: Analysis facilities, advanced QoS capabilities.

11 Evolution of databases

11.1 Database technologies

The processing of the experimental data collected by ATLAS requires a wide variety of auxiliary information from many systems: the Detector Control Systems (DCS); Trigger and Data Acquisition (TDAQ); data quality, the LHC itself and ATLAS sub-detectors' calibrations and configurations; the detectors' geometries and metadata associated with file storage and processing. Databases are used to manage this structured data which can be inserted, updated, and deleted in a centralized repository and queried by a variety of specialized clients in an efficient way from a variety of platforms (online, offline, and on the world-wide computing grid). Moreover, database storage is the foundation of all monitoring and analytics interfaces, which are used to observe current operations and past history to look for problems as well as procure improvements. The distributed computing workflow and data management tools described in §10 also need to store fine-grained information on the status of each job being planned, executed or completed, as well as for each file being stored or transferred to and from ATLAS sites.

Production data is stored in relational databases and a combination of normalization and storage optimization techniques are used to minimise the storage volume. A relational database storage management system (RDBMS) is used to perform atomic data insertion and manipulation quickly and accurately, provide security protocols appropriate to requirements, facilitate data completeness and integrity checks, and furnish mechanisms for recovery (backup) in cases of infrastructure or human failures. ATLAS uses an Oracle RDBMS for these purposes, a commercial product and industry standard, for its reliability and robust feature set which has proven to satisfy all the requirements above for our largest scale applications

(5-10 TB scale). Other relational database platforms exist, however they lag behind in available features and some are known to fail scalability requirements. ATLAS will work with CERN-IT in evaluating different solutions in the coming decade. In the end, the most valuable assets in the database area are the data itself and expert development time, which are optimized by using a RDBMS with the features needed to streamline services.

The importance of analytics tools increased during Run 2, and were instrumental in improving the efficiency of the ATLAS workflow and data management. Most analytics tools are based on Elasticsearch [71], a technology which is expected to scale to Run 4 rates.

11.2 Conditions database

The ATLAS Conditions database is based on the LCG package COOL [72] developed in advance of LHC Run 1. The ATLAS implementation exploited much of the flexibility of COOL to develop a sophisticated data model. This resulted in a very large number of diversely structured underlying Oracle database tables (O(10^4), many of which are not cache friendly when accessed via the COOL API. The efforts to migrate ATLAS Conditions towards a simpler data model and to manage them via a dedicated REST API are ongoing in collaboration with colleagues from CMS. The usage of modern technologies and the simplicity of the approach in the data model (O(10) tables) allows to have a server implementation (the CREST project [73]) which disentangles the client from the underlying storage solution. At the client level, Athena needs to be flexible enough to profit from a pure REST client in C++.

The ultimate goal is to structure the data such that client requests are better aligned with the caching infrastructure (based on Frontier [74]) which allows HTTP access and caching of previously requested data via Squid proxy. The caching performance is determined by the capability of the client to reproduce exactly the same SQL request for the same data loaded. This capability is highly improved by the CREST data model approach. The volume of the Conditions data will increase with the number of the detector channels but not with the luminosity, to about 1-2 TB of total per data taking period.

The time series data coming from the Detector Control System (DCS) are stored in relational tables. Solutions are being explored by the detector sub-systems to reorganise and reduce data with the goal to lower the total volume to the minimum that is needed for offline processing.

11.3 Metadata and integration of different sources

All data in ATLAS contains metadata which relates that data to other aspects of the experiment. For example, the 6 digit run number is a metadata field ubiquitous in any data recorded about the set of events recorded during that ATLAS run. A multitude of such key fields exists which allow cross referencing from one system to another to locate related data unambiguously for data processing or monitoring, etc. In addition to system-specific data, ATLAS has three dedicated metadata repositories AMI, COMA, and EventIndex, which collect information at the dataset, run, and event levels, respectively. Each provides a distinct base of services and each has grown in scope as new use cases arise which can exploit the collected information in new ways. The success of these systems depend inherently on uniform standards for key metadata fields as well as access to data from the component systems upon which it depends.

ATLAS data files contain some 'in-file' metadata which provides basic information about the constituent data. With the advent of parallelism and AthenaMP the handling of the in-file metadata is causing inefficiencies, and development is ongoing to record this information file-wise in a centralized repository which could then be used to understand a file content remotely.

11.4 Evolution of the EventIndex and other event-level metadata

The EventIndex provided, throughout Run 2, cataloguing services at the event level for all simulated and real data. The main initial use case, event picking, will always retain a primary importance, and all current use cases related to production consistency checks, trigger and derivation dataset overlap counting will always be covered. It is nevertheless possible to imagine different ways to store event data, or event parts, and read only the needed information from a small number of events. In this case the event catalogue will have to keep track of all event parts and provide pointers to the files, and within the files, that contain them to the processes analysing these events. The EventIndex currently stores 35 TB of data in Hadoop [75], with an expected increase to 1-2 PB for Run 4, which is achievable based on the expected technology scaling. The open source structured storage technologies evolve very quickly. The

modular architecture of the EventIndex allows the replacement of single components in order to achieve increased performance and functionality. For instance the replacement of the core storage system based on HBase [76] is currently (early 2020) under development for Run 3 but it is targeted at supporting the Run 4 data rates. The concept of virtual dataset (VDS) may become more relevant in the future, given the foreseen restriction on the amount of disk storage that will be available relative to the amount of available data. A VDS would be a list of events, selected by online or offline triggers, which are not copied to additional files but are kept only in the list. An intelligent delivery system as discussed in §10.4 would retrieve single events, or event parts, using pointers provided by the central catalogue. In order to cover these additional use cases, R&D is necessary during the Run 3 period, so as to have at least a functional prototype by the start of LS3 and a working system ready for Run 4.

12 Resource estimates

A projection of computing resources needed by ATLAS for HL-LHC is provided to guide computing centres and funding agencies on the likely requests during Runs 4 and 5. Current projections show that the majority of resources will be needed to produce simulated events, from physics modelling (event generation) to detector simulation, reconstruction and production of analysis formats. There will be an increase in the total number of events simulated per year, along with the integrated luminosity. Improvements over time in physics modelling, simulation, reconstruction and data quality imply that both data and simulated samples will need to be to be re-produced periodically to take advantage of such developments.

High-precision physics analyses, such as Higgs pair searches or the measurement of $\sin^2\theta_W$, will need simulated events to be produced in large numbers. Providing these events will put significant pressure on computing resources, limiting the accuracy of the physics results that can be driven by Monte Carlo statistics. Improvements in the software, as well as further increases in the available resources, are needed for ATLAS to achieve its full physics potential. In this section the resources needed under the three different scenarios described in §1 are examined. All three scenarios assume similar performance in terms of physics output, and demonstrate that aggressive software improvements are needed along with an increase of computing resources on the time scale of the HL-LHC.

12.1 ATLAS Computing resources model and LHC parameters

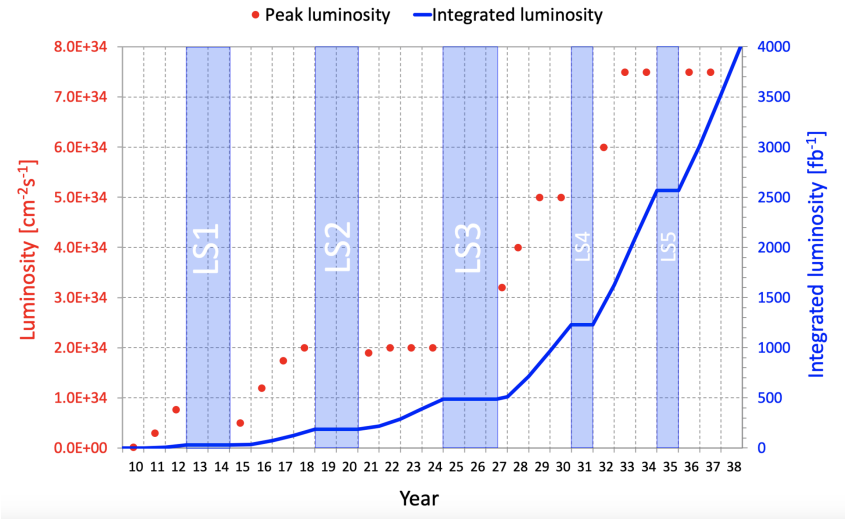


Figure 4: Schedule and luminosity forecast for the LHC.

The projections are calculated using a model implemented in a set of python scripts, which are used to model the resources needed as a function of time. The model has a one-year granularity and different scenarios are modelled by using different sets of input parameters. The model depends on approximately sixty such parameters, which can be categorised as follows:

- **Run parameters** are used to estimate the number of events recorded. Figure 4 shows the latest LHC schedule and estimated luminosity profile which is used in the projections. The main parameters of this type are summarised in the table 10 for three representative years of Runs 3, 4 and 5. A nominal "production" year of proton-proton collisions is assumed to consist of 161 days of data recording. Commissioning years at the beginning of the runs are shorter. The LHC live cycle fraction parameter is used to estimate the effective LHC beam time; the live fraction cycle is based on experience from previous years.

Parameter	unit	2023 Run3	2029 Run 4	2033 Run5	2028 LHCC c.s.
Interaction/crossing	max μ	55	140	200	200
Integrated luminosity	$fb^{-1}y^{-1}$	100	300	450	450
LHC ready for physics	10^6 s	7	7	7	7
Rate	kHz	1.4	10	10	10
Recorded events	10^9	10	70	70	70

Table 10: LHC and ATLAS run parameters. The last column represents a common scenario to be used by ATLAS and CMS, nominally in 2028, for comparisons in the context of the LHCC review.

- **Computing parameters** such as the event sizes for the various formats. The event size for the raw data format is shown in the table 7. For simulation, the average size (in MB/event) is estimated using certain benchmark processes (such as $t\bar{t}$ events), and expected levels of pile-up according to the LHC luminosity. The sizes thus obtained are then re-scaled to describe a realistic set of processes that may be produced during a typical simulation campaign. Other computing parameters include the average CPU time per event for the various workflows. Many of these are discussed in earlier chapters and are reported in tables 1, 2 and 4. The CPU time parameters are estimated using benchmark samples and configurations.
- **Operational parameters** capture the choreography of the data and MC production and reprocessing campaigns. Such parameters include the number of production and reproduction campaigns for data and MC. Typically, at the end of the year, ATLAS runs data reprocessing campaigns to calibrate the data with updated conditions, and improved alignment constants and detector quality flags. Re-reconstruction of the Monte Carlo is also possible should it be necessary. The operational model includes a certain number of processings (reconstruction, derivation) of data recorded (and MC produced) in each year, as well as reprocessing of data and MC from the previous years. The number of concurrently stored versions of a given format type for a given sample, and the number of copies stored on disk and tape, are also considered by the model. To model the data carousel, assumptions are made on the fraction of samples kept on disk or tape. Another example of operations modelling is the use of Tier-0 resources. These are defined by the capacity required for the prompt data processing as described in §9.2.2 but operationally these are made available for all other workflows when there is no data taking. Other operational parameters which have an effect on resources usage relate to the number of replicas of a given sample. These additional replicas are typically made to ensure smooth day-to-day operations.

12.2 Discussion on number of MC events needed

The best possible estimate of the number of MC events needed for the analysis of data from future runs would require detailed knowledge of the LHC operation parameters during that run, and the physics program. This would enable a reasonably precise estimate of the number of MC events required per inverse femtobarn of data collected. At the time of this writing, the physics programmes and trigger menus for Run 4 and beyond are not defined, so a more approximate estimate must be used for the resources projections.

During Run 2, the number of simulated events was between two and three times the number of data events. For the **conservative R&D** model we follow this pattern and make the following assumptions:

1. In each year during which data is taken, the number of MC events is assumed to be 2.5 times the number of data events collected;

2. In each year the production of the preceding three years is re-reconstructed from simulated hits using the latest and best tuned reconstruction and detector conditions and quality flags, and analysis formats produced from the newly reconstructed files. The old reconstructed MC and its derived data products are deleted once the new events are ready;
3. During the inter-run years when no data is taken, MC for the next run is produced in advance, to ensure that one year's worth of expected data is ready before data taking begins;
4. Every six year a full campaign of reproduction is done, including re-generation using the latest and most precise physics modelling. This reprocessing is assumed to be spread out across 1-2 years. Again, the old events are deleted once the new events are ready;
5. The fraction of events produced with fast simulation is assumed to be 75% throughout runs 4 and 5.

For the **aggressive** scenario, rather than producing 2.5 times as many MC events as data, we instead assume that MC and data events are produced in a ratio of 2:1 beyond Run 3. Fully 90% of the events are assumed to be produced with fast simulation, with fully simulated events mainly being used to tune the fast simulation. Other assumptions are the same as described above.

12.3 Projections for the three scenarios

The three scenarios laid down in §1 are used to define the input parameters to the computing model, which performs the extrapolation to HL-LHC.

Figure 5 shows the estimated resources needs in the various scenarios for each year, following the present understanding of the LHC parameters as in table 10. The blue circles show the **baseline** scenario, which will be the model for Run 3. The blue up-pointing and down-pointing triangles show respectively the **conservative R&D** and **aggressive R&D** scenarios. The red triangles represent the resources needed for the **conservative R&D** scenario in the LHCC common scenario described in 12.3.1. The lines forecast a 10% and 20% increase in resources capacity (due to technologies or budget improvements) starting from the resources required for 2021.

The impact of some of the activities designated as being part of the aggressive scenario in the document (for example ML models for fast simulation and reconstruction) is not known at this time, and has not been included. Further, it is assumed that some of the projects classed as being under the conservative scenario, including the parameterised fast detector simulation, will come to fruition over the course of Run 3.

The storage needed at the HL-LHC will increase due to the growth in the numbers of recorded and simulated events. An increase of high-latency, but cheaper than disk, storage, such as tape, will be needed to accommodate larger events from a detector with more active channels and a larger number of events. More tape resources will also be needed to store data formats that are rarely accessed (only a few times a year) by using the data carousel techniques described in §10.4. The overall usage of the data carousel will depend on the total throughput of the tape infrastructure (or some other future technology) across the WLCG.

Low-latency storage, such as disk, will be used primarily by analysis formats such as DAODs, since these are frequently accessed by physicists requiring almost immediate access. Depending on the performance of the data carousel, a fraction of the AODs used to produce DAODs will need to be stored on disk, alongside the event generator outputs (EVNT) which are used as input to the simulation, as well as the pile-up overlay events. Transient formats produced within a given production workflow but not permanently stored, and secondary copies that are needed to guarantee efficient operation of the data management system, will also need to be disk-resident. In the baseline scenario, one copy of the latest AOD version and half a copy of the previous AOD version is kept on disk (as in Run 2), while in the conservative and aggressive scenarios, only partial copies of the AOD are kept on disk (70% and 50% respectively). The DAODs constitutes the majority of the disk space: in the baseline and conservative scenarios two replicas of DAODs will be stored for detector performance studies (about 30% of the AOD size), along with four replicas of `DAOD_PHYS` and `DAOD_PHYSLITE` for fast and effective analysis access. In the aggressive scenario ATLAS will reduce the DAODs for performance studies to a single replica, and reduce to two the replicas on disk of `DAOD_PHYS`.

Re-reconstruction of data events will take place once for the bulk of the data to ensure the latest calibration and alignment constants are used. Special samples will be reprocessed more often. Monte Carlo events

will be re-reconstucted at the same time as the data to ensure consistent software is used for both data and Monte Carlo. It is assumed that after some six years, there will have been sufficient improvements in the quality of the physics modelling and simulation to warrant fully re-generating, re-simulating and re-reconstructing all MC samples relevant to ongoing and planned physics analyses. DAODs are assumed to be rebuilt every four to six months to account for new object reconstruction calibrations. Each version is assumed to be kept on disk for two years to ensure physics analysis can use a consistent version throughout the publication process.

12.3.1 Estimates for the LHCC common scenario

In the context of the LHCC HL-LHC Computing review, ATLAS and CMS agreed to provide resource estimates based on a jointly-defined scenario that assumes an instantaneous luminosity of $7.5 10^{34} cm^{-2} s^{-1}$ during 2028 in Run 4. This scenario is summarized in the last column of table 10.

Both table 11 and figure 5 show the results for this scenario. The three ATLAS computing scenarios and the forecast of +10% and +20% resources capacity increases per year are shown in the table. The red open triangles shown in Figure 5 represent the resources needed under the ATLAS computing conservative R&D scenario. The projected results for Run 4 are very similar to the ATLAS conservative R&D scenario for Run 5, except for the tape requirements. These are lower due to smaller amount of data recorded up to 2028.

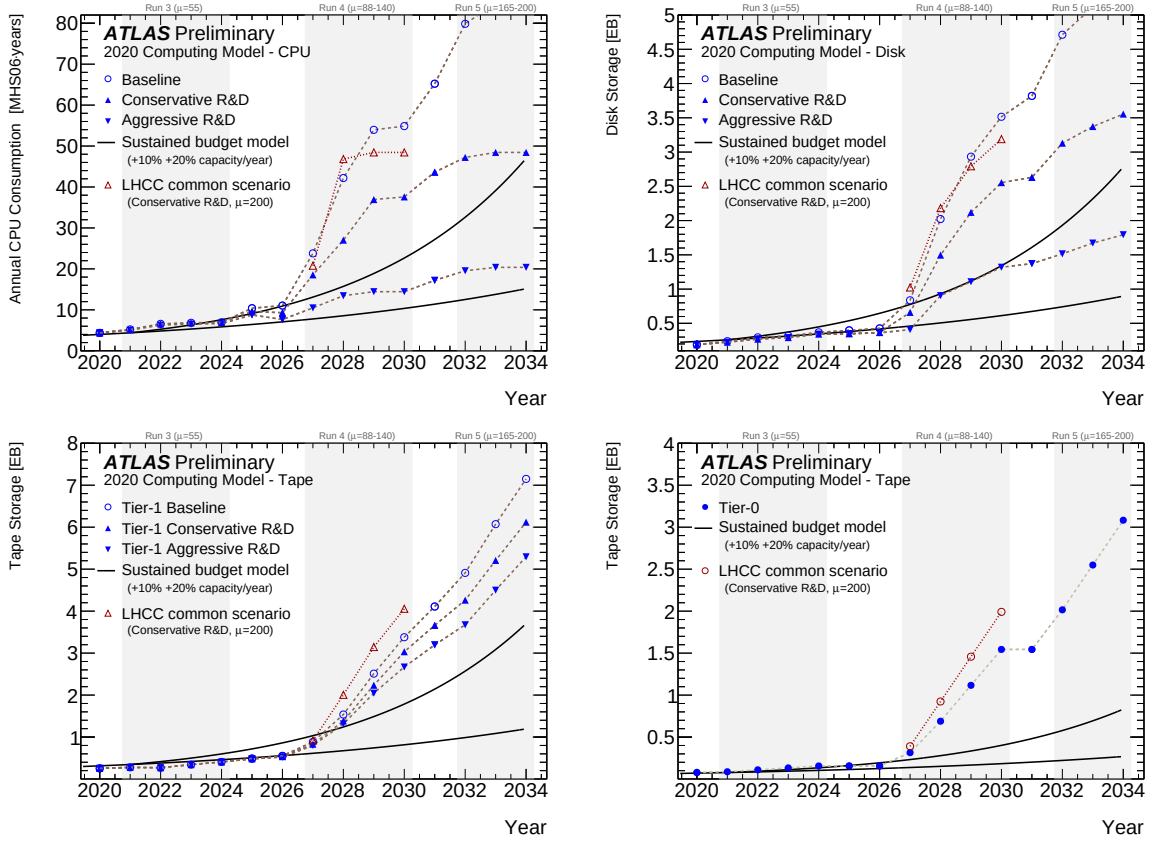


Figure 5: Estimated CPU, disk and tape (at the Tier-1 and Tier-0) resources needed for the years 2020 to 2034 under the different scenarios described in the text. The solid lines indicate annual improvements of 10% and 20% in the capacity of new hardware for a given cost, assuming a sustained level of annual investment. The blue dots with the brown dashed lines represent the three ATLAS scenarios following the current LHC schedule. The red open triangles indicate the Conservative R&D scenario under an assumption of the LHC reaching $\langle \mu \rangle = 200$ in 2028

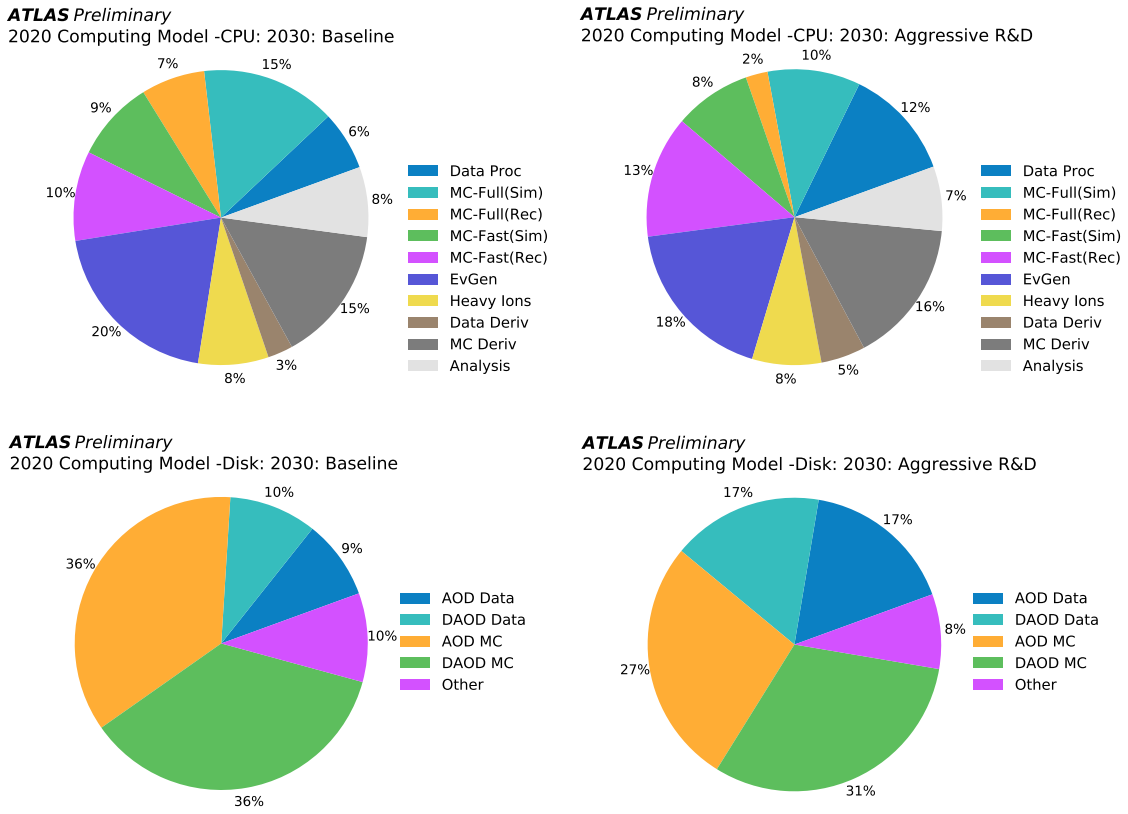


Figure 6: Breakdown of projected CPU and disk resources usage in 2030 for the Baseline R&D and Aggressive R&D scenarios.

Resource usage in 2028 (LHCC common scenario)	CPU [MHS06· y]	Disk [PB]	Tape Tier-1 [PB]	Tape Tier-0 [PB]
Baseline	83	3510	2370	925
Conservative R&D	47	2180	2000	924
Aggressive R&D	20	1030	1760	924
Sust. budget model +20%	16	930	1240	280
Sust. budget model +10%	9	510	674	150

Table 11: Resource estimates under the jointly ATLAS and CMS assumptions (as from table 10) during 2028 for the three ATLAS computing scenarios.

13 Timeline and high level milestones for HL-LHC R&D

This section provides a timeline and a set of high-level milestones (Table 12) common to all new developments targeting Run 4, with the hope to inform the detailed planning of Run 4 activities. The timeline follows the LHC schedule of Fig. 4 and it assumes that the ATLAS HL-LHC Computing Technical Design Report (TDR) will be completed in 2023.

Run 4 activities can be organized in four phases:

Phase 1: LS2 (now to 2021) The goal is to complete before Run 3 all "Baseline" developments identified in this document including multi-threaded athena, the new DAOD data formats for Run 3 analysis model, and updates to fast calorimeter simulation and tracking code. As a stretch goal, or if Run 3 start is postponed, move towards the validation of well-advanced "Conservative R&D" projects, for example, Data Carousel, ACTS, and fast detector simulation.

Phase 2: Computing TDR (now to 2023) To inform the contents of the TDR, complete exploratory R&D projects pursuing all R&D directions listed in this document. This will happen in collaboration with HSF and cross-experiment research projects. The TDR will prioritize these R&D projects and select those to pursue during Phase 3.

Phase 3: Complete R&D Program (2023-2024) The goal is to develop the first fully functional version of all R&D projects endorsed by the TDR as high-priority . Decide which projects move to Phase 4 and which are targeting Run 5. Train ATLAS developers community to enable them to contribute to Run 4 development.

Phase 4: LS3 (2025-2027) Focus shifts from R&D to development. ATLAS physics community becomes involved with high priority in code development, and later validation. R&D targeting Run 5 continues with lower priority.

Phase 1 and Phase 4 rely crucially on the contributions of the wide physics community, therefore they have been scheduled during LS2 and LS3 respectively. The R&D work of Phase 2 and Phase 3 will be performed mainly by the core group and by cross-experiment R&D projects in the framework of HSF and WLCG.

ID	Year	Milestone
P1.1	2020	CDR released: identifies HL-LHC R&D needs, and the projects attempting to address them.
P1.2	2020	Run 3 deployment: includes all Baseline updates and releases and potentially some Conservative R&D.
P1.3	2021	Run 3 starts: R&D focus shifts to HL-LHC R&D.
P2.1	2021	Run 4 R&D plans: all R&D projects targeting Run 4 provide a program of work to 2024, including risks and effort estimates.
P2.2	2022	Run 4 R&D demonstrators: all R&D projects targeting Run 4 provide proof-of-concept demonstrators
P2.3	2023	TDR released: prioritizes Run 4 Conservative R&D projects. Endorses a limited amount of Aggressive R&D projects for Phase 3
P3.1	2024	Run 4 projects approval: go/no-go decision for R&D projects targeting Run 4. Every project must provide functionally complete prototypes.
P3.2	2025	Run 4 development planning: each Run 4 project provides a WBS inclusive of effort and risk estimates. It also provides training and documentation tailored for the target developers community.
P4.1	2025	First Run 4 deployment: includes production-quality implementations of all new developments to be included in Run 4. Not all physics code migrated yet.
P4.2	2026	Second Run 4 deployment: includes production version of all Run 4 code and integration. Validation starts.
P4.3	2026	Run 4 dress rehearsal: test software and ADC readiness for data taking.
P4.4	2027	Run 4 starts: R&D focus shifts to Run 5.

Table 12: High-level milestones for HL-LHC Computing and Software R&D program

14 Conclusions

The preceding chapters of this document have described the research and development plans of the ATLAS computing and software community for the next years, in preparation for the HL-LHC era. Projections of the estimated resources required for this period have been presented under three scenarios of increasing ambition. The most ambitious development plans imply the least use of computational and storage resources, such that the projections indicate that ATLAS could successfully execute its physics programme with only modest increases in disk, tape and CPU. However, delivery of these plans requires

more investment in human resources so that we can engage in and complete on time the necessary research and development projects.

Storage remains the most difficult of the HL-LHC challenges. ATLAS will face this by reducing the storage footprint of the data formats, in particular creating new smaller derived data types (DAOD_PHYS and DAOD_PHYSLITE). It will also invest in novel approaches including the Data Carousel. This will enable cheaper storage technologies such as tape to be used in a transparent, efficient and more automated manner, whilst ensuring the most regularly used and critical data is placed on higher performance storage technologies such as disk. ATLAS will continue to engage with the distributed computing community to develop new schemes for data organization, such as data lakes, which might optimize the overall storage costs while keeping the performance at the standards needed to perform efficient data analysis.

Discussions involving HL-LHC computing tend to focus on the projected shortfalls in computational capacity. Here a variety of solutions must be applied. First amongst them is ensuring that CPU-intensive precise calculations are only deployed in physics modelling and detector simulation where the physics demands it, using lower fidelity computations (for instance, fast simulation) where this does not unduly impact the competitiveness of the results. Exhaustive optimisation of the software running event generation, simulation and reconstruction is also of great importance, whether this be done by code optimisation or by re-configuring workflows or run-time settings. Finally, making use of computational accelerators, such as GP-GPUs, will become increasingly important with time, especially as they work their way into ordinary commodity hardware of the type installed in WLCG sites. Significant levels of effort will need to be deployed in this area to ensure that the software is capable of offloading work to such accelerators.

User-level analysis brings these two aspects together, as analysts must be able to aggregate small portions of large volumes of data, with as little latency as possible. Research into new methods of serving physics analysis is heading in several highly promising directions, both within and beyond ATLAS, and the level of collaboration is encouraging, especially at the level of the HSF. An increasing tendency for analysts in particle physics to reach for tools used more widely in industry, for example the Python-based environments, will also help in this regard.

The role of ML may prove to be decisive in some or all of the above. Its role in physics analysis has been cemented by many years of successes including fundamental discoveries related to top and Higgs physics. Such successes will be built upon as more sophisticated models are deployed. Early results from fast simulation and from jet reconstruction and flavour-tagging are also very promising. It remains to be seen whether ML can have a similar impact on inner tracking reconstruction, and explorations of its potential in physics modelling are at an early stage, but it is clear that this area may unlock significant advances in experimental particle physics.

Neither ATLAS, nor any of the LHC experiments, are alone in these endeavours. The four experiments are joining forces with other communities, both within and outside of the particle physics field. The HEP Software Foundation is proving to be particularly important in this regard, especially for interactions with theorists and analysis software experts. Fruitful collaboration through the WLCG has been maintained for many years. Interactions with industrial partners such as Google, Intel, Amazon and many others, both directly and through channels such as OpenLab, are also proving to be highly productive, as is interest in our activities from the academic computer science community, and support received from the national funding agencies. With such a wide range of potential collaborations and partners, it is essential that such activities are streamlined and well organised, to ensure that each experiment can gain maximum benefit from external interactions.

The days when software and computing grew up organically around a new detector are long passed. The computing and software elements of an LHC experiment are of equivalent complexity to the hardware itself, and this will be intensified by the HL-LHC data. Dedicated human resources, committed to HL-LHC computing over a long period, will be needed if the HL-LHC is to be a success. Skilled software developers will only be attracted to particle physics research if there is a realistic chance of long term and stable employment. Appropriate levels of investment in human resources, sustained for a long period, are therefore essential.

15 Bibliography

[1] The HSF Physics Event Generator WG, *Challenges in Monte Carlo event generator software for High-Luminosity LHC*, 2020, arXiv: 2004.13687 (cit. on p. 1).

[2] *HSF WLCG Virtual Workshop on New Architectures, Portability, and Sustainability*, 2020, URL: https://indico.cern.ch/event/908146 (cit. on p. 3).

[3] Illya Shapoval et al., "Graph-based decision making for task scheduling in concurrent Gaudi", *2015 IEEE Nuclear Science Symposium and Medical Imaging Conference (NSS/MIC)*, 2015 1 (cit. on p. 3).

[4] Shapoval, Illya, "Adaptive scheduling applied to non-deterministic networks of heterogeneous tasks for peak throughput in concurrent Gaudi", PhD thesis: CERN/UNIFE, 2016, URL: http://cds.cern.ch/record/2149420 (cit. on p. 3).

[5] Junichi Kanzaki, *MadGraph on GPU*, 2019, URL: https://indico.cern.ch/event/759388/contributions/3303060/attachments/1814389/2965195/MadGraph_on_GPU.pdf (cit. on p. 4).

[6] H. Carter Edwards, Christian R. Trott and Daniel Sunderland, *Kokkos: Enabling manycore performance portability through polymorphic memory access patterns*, Journal of Parallel and Distributed Computing **74** (2014) 3202, Domain-Specific Languages and High-Level Frameworks for High-Performance Computing (cit. on p. 4).

[7] Erik Zenker et al., "Alpaka - An Abstraction Library for Parallel Kernel Acceleration", 2016 (cit. on p. 4).

[8] John L. Hennessy and David A. Patterson, *A New Golden Age for Computer Architecture*, Commun. ACM **62** (2019) 48 (cit. on p. 4).

[9] Andy Buckley et al., *Implementation of the ATLAS Run 2 event data model*, Journal of Physics: Conference Series **664** (2015) 072045 (cit. on pp. 5, 44).

[10] Stéfan van der Walt, S. Chris Colbert and Gaël Varoquaux, *The NumPy Array: A Structure for Efficient Numerical Computation*, Computing in Science & Engineering **13** (2011) 22 (cit. on pp. 5, 20).

[11] Paolo Calafiura, Charles Leggett, David Quarrie, Hong Ma and Srini Rajagopalan, "The StoreGate: a Data Model for the Atlas Software Architecture", *PROCEEDINGS OF CHEP 2001*, 2003 522, arXiv: cs/0306089 [cs] (cit. on pp. 5, 44).

[12] Eric Wulff, *Deep Autoencoders for Compression in High Energy Physics*, Lund University Master's Thesis Student Paper, 2020 (cit. on p. 5).

[13] J. Boudreau and V. Tsulaia, *The GeoModel Toolkit for Detector Description*, 2005 (cit. on p. 6).

[14] Gaël Guennebaud, Benoît Jacob et al., *Eigen v3*, http://eigen.tuxfamily.org, 2010 (cit. on p. 6).

[15] ATLAS Collaboration, *Technical Design Report for the ATLAS Inner Tracker Pixel Detector*, 2017, URL: https://cds.cern.ch/record/2285585 (cit. on p. 6).

[16] ATLAS Collaboration, *Technical Design Report for the ATLAS Inner Tracker Strip Detector*, 2017, URL: https://cds.cern.ch/record/2257755 (cit. on p. 6).

[17] ATLAS Collaboration, *Technical Proposal: A High-Granularity Timing Detector for the ATLAS Phase-II Upgrade*, 2018, URL: https://cds.cern.ch/record/2623663 (cit. on p. 6).

[18] T Kawamoto et al., *New Small Wheel Technical Design Report*, ATLAS New Small Wheel Technical Design Report, 2013, URL: https://cds.cern.ch/record/1552862 (cit. on p. 6).

[19] Stefan Hoeche, Stefan Prestel and Holger Schulz, *Simulation of vector boson plus many jet final states at the high luminosity LHC*, Phys. Rev. D **100** (2019), arXiv: 1905.05120 [hep-ph] (cit. on p. 7).

[20] J. Schaarschmidt on behalf of the ATLAS Collaboration, *The new ATLAS Fast Calorimeter Simulation*, Journal of Physics: Conference Series **898** (2017) 042006 (cit. on p. 8).

[21] *Fast track simulation extension for the ACTS project*, URL: https://gitlab.cern.ch/acts/acts-fatras (cit. on p. 8).

[22] ATLAS Collaboration, *Technical Design Report for the Phase-II Upgrade of the ATLAS TDAQ System*, 2017, URL: https://cds.cern.ch/record/2285584 (cit. on pp. 11, 22–24).

[23] *Expected Tracking Performance of the ATLAS Inner Tracker at the HL-LHC*, 2019, URL: https://cds.cern.ch/record/2669540 (cit. on pp. 12, 13).

[24] *Fast Track Reconstruction for HL-LHC*, tech. rep. ATL-PHYS-PUB-2019-041, CERN, 2019, URL: https://cds.cern.ch/record/2693670 (cit. on pp. 12–14).

[25] *acts GitLab@CERN*, URL: https://gitlab.cern.ch/acts/acts-core (cit. on pp. 12, 14).

[26] ATLAS Collaboration, *A neural network clustering algorithm for the ATLAS silicon pixel detector*, JINST **9** (2014) P09009, arXiv: 1406.7690 [hep-ex] (cit. on p. 14).

[27] ATLAS Collaboration, *Topological cell clustering in the ATLAS calorimeters and its performance in LHC Run 1*, Eur. Phys. J. C **77** (2017) 490, arXiv: 1603.02934 [hep-ex] (cit. on p. 15).

[28] ATLAS Collaboration, *Jet reconstruction and performance using particle flow with the ATLAS Detector*, Eur. Phys. J. C **77** (2017) 466, arXiv: 1703.10485 [hep-ex] (cit. on p. 15).

[29] *Learning to Discover : Advanced Pattern Recognition*, URL: https://indico.cern.ch/event/847626/ (cit. on p. 16).

[30] R M Bianchi et al., *Event visualization in ATLAS*, J. Phys.: Conf. Ser. **898** (2017), (on behalf of the ATLAS Collaboration) (cit. on p. 17).

[31] *ATLAS Public Event Displays*, https://twiki.cern.ch/twiki/bin/view/AtlasPublic/EventDisplayRun2Physics (cit. on p. 17).

[32] M. Bellis et al., *HEP Software Foundation Community White Paper Working Group — Visualization*, 2018, arXiv: 1811.10309 [hep-ex] (cit. on p. 18).

[33] ATLAS Collaboration, *A new petabyte-scale data derivation framework for ATLAS*, J.Phys.Conf.Ser. **664** (2015), URL: https://doi.org/10.1088/1742-6596/664/7/072007 (cit. on p. 18).

[34] Johannes Elmsheuser et al., *Evolution of the ATLAS analysis model for Run-3 and prospects for HL-LHC*, 2020, URL: https://cds.cern.ch/record/2708664 (cit. on p. 18).

[35] Apache Software Foundation, *Parquet*, URL: https://parquet.apache.org (cit. on p. 19).

[36] ROOT Team, *Software Challenges For HL-LHC Data Analysis*, 2020, arXiv: 2004.07675 (cit. on p. 20).

[37] Danilo Piparo et al., *RDataFrame: Easy Parallel ROOT Analysis at 100 Threads*, EPJ Web of Conferences **214** (2019) 06029 (cit. on p. 20).

[38] Pauli Virtanen et al., *SciPy 1.0: Fundamental Algorithms for Scientific Computing in Python*, Nature Methods **17** (2020) 261 (cit. on p. 20).

[39] J. D. Hunter, *Matplotlib: A 2D Graphics Environment*, Computing in Science Engineering **9** (2007) 90 (cit. on p. 20).

[40] *uproot*, URL: https://github.com/scikit-hep/uproot (cit. on p. 20).

[41] *intelligent Data Delivery Service*, URL: https://idds.cern.ch (cit. on p. 20).

[42] *IRIS-HEP ServiceX*, URL: https://iris-hep.org/projects/servicex.html (cit. on p. 20).

[43] Apache Software Foundation, *Apache Arrow*, URL: https://arrow.apache.org (cit. on p. 20).

[44] Apache Software Foundation, *Apache Spark*, URL: https://spark.apache.org (cit. on p. 21).

[45] Dask, *Scalable analytics in python*, URL: https://dask.org/ (cit. on p. 21).

[46] Ray, *Fast and simple distributed computing*, URL: https://ray.io/ (cit. on p. 21).

[47] CSU, *CSU AWS Tier 3*, 2020, URL: http://aws-csufresno-atlas.education/ (cit. on p. 21).

[48] Jupyter, *Jupyter Hub*, URL: https://jupyter.org/hub (cit. on p. 21).

[49] Heinrich, Lukas and Feickert, Matthew and Stark, Giordon, *pyhf: v0.4.1*, version 0.4.1, URL: https://github.com/scikit-hep/pyhf (cit. on p. 22).

[50] ATLAS Collaboration, *Summary of the ATLAS experiment's sensitivity to supersymmetry after LHC Run 1 — interpreted in the phenomenological MSSM*, JHEP **10** (2015) 134, arXiv: 1508.06608 [hep-ex] (cit. on p. 22).

[51] *The RECAST framework*, URL: https://iris-hep.org/projects/recast.html (cit. on p. 22).

[52] Johannes Albrecht et al., *A Roadmap for HEP Software and Computing R&D for the 2020s*, Comput. Softw. Big Sci. **3** (2019) 7, arXiv: 1712.06982 [physics.comp-ph] (cit. on p. 22).

[53] Johannes Albrecht et al., *HEP Community White Paper on Software trigger and event reconstruction: Executive Summary*, 2018, arXiv: 1802.08640 [physics.comp-ph] (cit. on p. 22).

[54] Johannes Albrecht et al., *HEP Community White Paper on Software trigger and event reconstruction*, 2018, arXiv: 1802.08638 [physics.comp-ph] (cit. on p. 22).

[55] ATLAS Collaboration, *Search for Low-Mass Dijet Resonances Using Trigger-Level Jets with the ATLAS Detector in pp Collisions at* $\sqrt{s} = 13\,TeV$, Phys. Rev. Lett. **121** (2018) 081801, arXiv: 1804.03496 [hep-ex] (cit. on p. 23).

[56] LHCb Collaboration, *Tesla : an application for real-time data analysis in High Energy Physics*, Comput. Phys. Commun. **208** (2016) 35, arXiv: 1604.05596 [physics.ins-det] (cit. on p. 23).

[57] CMS Collaboration, *Search for narrow resonances in dijet final states at* $\sqrt{(s)} = 8\ TeV$ *with the novel CMS technique of data scouting*, Phys. Rev. Lett. **117** (2016) 031802, arXiv: 1604.08907 [hep-ex] (cit. on p. 23).

[58] ATLAS Collaboration, *Trigger-object Level Analysis with the ATLAS detector at the Large Hadron Collider: summary and perspectives*, 2017, URL: https://cds.cern.ch/record/2295739 (cit. on p. 23).

[59] ATLAS Collaboration, *Performance of the ATLAS trigger system in 2015*, Eur. Phys. J. C **77** (2017) 317, arXiv: 1611.09661 [hep-ex] (cit. on p. 23).

[60] ATLAS Collaboration, *Technical Proposal: A High-Granularity Timing Detector for the ATLAS Phase-II Upgrade*, 2018, URL: http://cds.cern.ch/record/2623663 (cit. on p. 24).

[61] Andreas J Peters and Lukasz Janyst, *Exabyte Scale Storage at CERN*, Journal of Physics: Conference Series **331** (2011) 052015 (cit. on p. 24).

[62] *Deep Underground Neutrino Experiment*, URL: http://www.dunescience.org/ (cit. on p. 26).

[63] *Belle II*, URL: https://www.belle2.org/ (cit. on p. 26).

[64] *ESCAPE – European Science Cluster of Astronomy & Particle physics ESFRI research infrastructures*, URL: https://projectescape.eu (cit. on p. 26).

[65] F H Barreiro et al., *The ATLAS Production System Evolution: New Data Processing and Analysis Paradigm for the LHC Run2 and High-Luminosity*, Journal of Physics: Conference Series **898** (2017) 052016 (cit. on p. 27).

[66] Martin Barisits et al., *Rucio: Scientific Data Management*, Computing and Software for Big Science **3** (2019) 11 (cit. on p. 27).

[67] Marian Babik et al., *HEPiX Network Functions Virtualisation Working Group Report*, version 2.0, Zenodo (2020) (cit. on p. 29).

[68] Oxana Smirnova et al., "Building and Operating a Distributed Regional Centre for LHC Computing and Data Storage", *Proceedings of the 7th IEEE International Conference on e-Science and Grid Computing*, 2011 181 (cit. on p. 29).

[69] F Barreiro Megino et al., *Integration of Titan supercomputer at OLCF with ATLAS Production System*, Journal of Physics: Conference Series **898** (2017) 092002 (cit. on p. 29).

[70] *WLCG Memorandum of Understanding*, URL: https://wlcg.web.cern.ch/mou (cit. on p. 29).

[71] Elastic.co, *Elastic Search*, URL: https://www.elastic.co/elasticsearch/ (cit. on p. 31).

[72] Andrea Valassi et al., *COOL, LCG conditions database for the LHC experiments: Development and deployment status*, IEEE Nuclear Science Symposium conference record. Nuclear Science Symposium (2008) 3021 (cit. on p. 31).

[73] Roland Sipos, Andrea Formica, Giovanni Franzoni, Giacomo Govi and Andreas Pfeiffer, *Functional tests of a prototype for the CMS-ATLAS common non-event data handling framework*, Journal of Physics: Conference Series **898** (2017) 042047 (cit. on p. 31).

[74] *LCG conditions database project overview*, 2007, URL: `http://frontier.cern.ch/` (cit. on p. 31).

[75] Apache Software Foundation, *Apache Hadoop*, URL: `https://hadoop.apache.org` (cit. on p. 31).

[76] Apache Software Foundation, *Apache HBase*, URL: `https://hbase.apache.org` (cit. on p. 32).

[77] S. Binet, P. Calafiura, S. Snyder, W. Wiedenmann and F. Winklmeier, *Harnessing multicores: Strategies and implementations in ATLAS*, Journal of Physics: Conference Series **219** (2010) 042002 (cit. on p. 44).

[78] Charles Leggett et al., *AthenaMT: upgrading the ATLAS software framework for the many-core world with multi-threading*, Journal of Physics: Conference Series **898** (2017) 042009 (cit. on p. 44).

[79] G. Barrand et al., *GAUDI – A software architecture and framework for building HEP data processing applications*, Computer Physics Communications **140** (2001) 45, CHEP2000 (cit. on p. 44).

[80] HEPiX, *Benchmarking Working Group*, 2017, URL: `https://w3.hepix.org/benchmarking.html` (cit. on p. 44).

Acronyms

ADC ATLAS Distributed Computing. 26
AMSG3 Analysis Model Study Group for Run 3. 18
AOD Analysis Object Data. 5, 24, 43

CDR Conceptual Design Report. 1
CP Combined Performance. 25
CUDA Nvidia Compute Unified Device Architecture. 4

DAOD Derived AOD. 5, 24
DCS Detector Control System. 25, 31
DESD Derived ESD. 24
DOMA Data Organisation and Management. 2
DPC++ Intel Data Parallel C++. 4
DQM Data Quality Monitoring. 25

EDM Event Data Model. 5, 19
ESD Event Summary Data. 19, 43

HLT High-Level Trigger. 2
HPC High-Performance Computing. 4, 27
HS06 HEP-SPEC06. 6
HSF HEP Software Foundation. 1

iDDS intelligent Data Delivery Service. 20

NLO Next to Leading Order. 1, 7
NNLO Next to Next to Leading Order. 1, 7

PEB Partial Event Building. 23

RDO detector digitization format. 9

TBB Intel Thread Building Blocks. 3
TDAQ Trigger and Data Acquisition. 22, 30
TDR Technical Design Report. 22
TLA Trigger-object Level Analysis. 23, 24

Glossary

AthenaMP The event-parallel version [77] of the athena application framework, based on the Gaudi project. 9

AthenaMT The multi-threaded version [78] of the athena application framework, based on the Gaudi project. 3

CMake A cross-platform free and open-source software tool used by ATLAS for managing the build process of software. 4

DRAW A filtered ("skimmed") raw data stream. 19, 24

Event Generator A Monte Carlo simulation program, describing a physics process in an LHC particle collision at the particle level. 1

Gaudi An open project for providing the necessary interfaces and services for building HEP experiment frameworks in the domain of event data processing applications [79]. 3, 44

HEP-SPEC06 The HEP-wide benchmark for measuring CPU performance [80]. HS06 measures CPU power, therefore HS06·s is a unit of computational work. 43

SYCL SYCL (pronounced "sickle") is a royalty-free, cross-platform abstraction layer that enables code for heterogeneous processors to be written in a "single-source" style using completely standard C++. 4

Transient-Persistent separation The Gaudi architecture calls for separate in-memory (transient) and disk-resident (persistent) event data model designs and implementations. This allows independent optimization of data structures for efficient storage, and for algorithm development[11] . 5

xAOD The ATLAS columnar Data Model. Collections of data are organized column-wise as Structures of Arrays, rather than row-wise as Arrays of Structures[9]. 5, 19