An ATLAS distributed computing architecture for HL-LHC

S Campana[1] and T Wenaus[2] for the ATLAS Collaboration

[1] CERN, Geneva, Switzerland
[2] Brookhaven National Laboratory, Upton, NY 11973-5000, USA

Abstract. The ATLAS collaboration started a process to understand the computing needs for the High Luminosity LHC era. Based on our best understanding of the computing model input parameters for the HL-LHC data taking conditions, results indicate the need for a larger amount of computational and storage resources with respect to the projection of constant yearly budget for computing in 2026. Filling the gap between the projection and the needs will be one of the challenges in preparation for HL-LHC. While the gains from improvements in offline software will play a crucial role in this process, a different model for data processing, management, access and bookkeeping should also be envisaged to optimise resource usage. In this contribution we will describe a straw man of this model, founded on basic principles such as single event level granularity for data processing and virtual data. We will explain how the current architecture will evolve adiabatically into the future distributed computing system, through the prototyping of building blocks that would be integrated in the production infrastructure as early as possible, so that specific use cases can be covered much earlier with respect to the HL-LHC time scale. We will also discuss how such system would adapt to and drive the evolution of the WLCG infrastructure in terms of facilities and services.

1. Introduction

The High Luminosity data-taking period of the LHC (HL-LHC) scheduled starting in 2026 will present unprecedented challenges in terms of computing. The ATLAS experiment [1], after a major hardware upgrade, will collect and retain 10,000 proton-proton collision events per second, compared to the 1,000 collisions in the current LHC run. In addition, the complexity of the events will increase considerably, as up to 200 particle interactions will overlap (pile-up). Reconstructing one event is expected therefore to consume more CPU resources as the growth in reconstruction time exponentially increases with the pile-up. The event size is expected to increase linearly with the pile-up as it strongly correlates with the number of particles in the event. The LHC experiment computing infrastructures, founded on the Worldwide LHC Computing Grid (WLCG), organise computing resources distributed over 200 centers. The budget for computing hardware in WLCG is not expected to increase in the next years and at best will remain constant. Considering the needs to replace old hardware and the technology improvements we can expect roughly a 20% growth of processing capacity (CPUs) and a 15% growth of storage (disk and tape) every year for the same constant funding. In 2016 ATLAS presented a study of the computing challenges in HL-LHC, comparing the resource needs with the expected available resources assuming constant funding evolution. In this contribution we present an update of the study, based on a more refined model, a better understanding of the input parameters, the new ATLAS detector layout and recent improvements of the reconstruction algorithms. We show how, despite the recent improvements, the required level of resources is still considerably larger than what we will realistically be able to obtain. We focus on the storage needs and present the architecture of a new computing model and infrastructure capable to at least partially address the likely shortage of resources.

2. The HL-LHC computing challenge

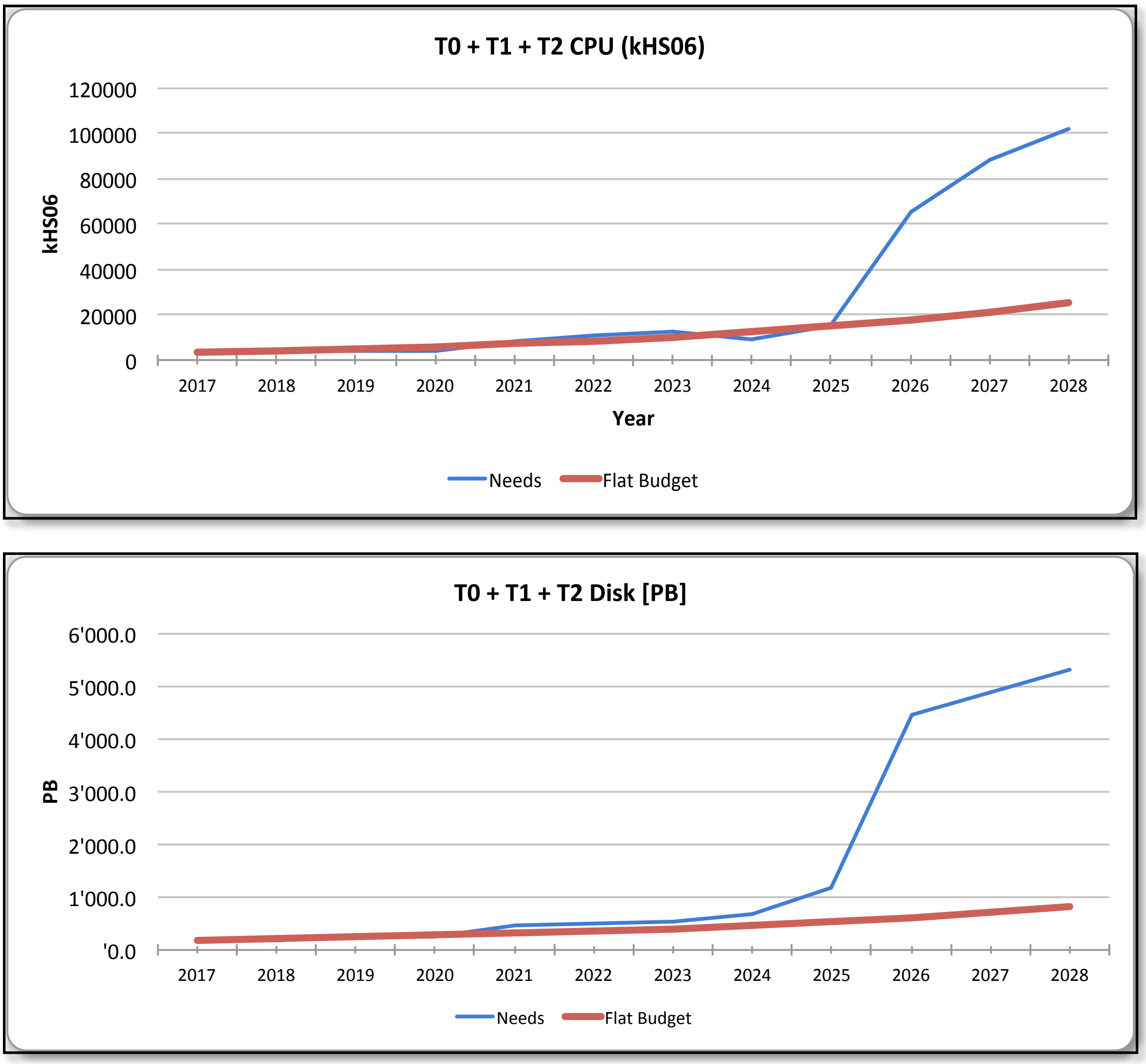

Figure 1. Estimates of the ATLAS CPU and storage (disk) needs in the next 10 years, in the top and bottom panel respectively. The needs (blue line) are compared with the projected available resources based on a constant funding model (red line)

The most recent study of the HL-LHC computing challenge for the four experiments was presented in Ref. [2]. In Ref. [3] ATLAS produced a more detailed analysis showing how different input parameters impact the expected resource needs. Such analysis estimated a gap of a factor 9 for CPUs and a factor 7 for storage between the needed resource and what likely will be available assuming constant funding. We recently updated the study and in particular:

1. The expected number of data-taking seconds is increased from 5.5 million to 7.3 million based on the current LHC running experience.
2. The contribution of the CERN computing facility is included in the model.
3. Data retained from previous data taking years is now properly accounted for.
4. The reconstruction time per event was lowered from 2880 HS06 seconds to 1300 HS06 seconds. The new number reflects the improvement after introducing the new layout of the inner detector and tuning the software accordingly.
5. The Monte Carlo needs are now estimated as 1.5 times more events than data, based on the current run experience. One third of such events can be produced with Fast Simulation while the rest should be simulated through Geant4 [4].

Figure 1 shows the evolution of the ATLAS computing needs in the next 10 years compared with the expected available resources obtained spending a constant budget every year. With the new parameters and model the gap in CPU needs decrease from a factor 9 to a factor 3.7. In fact, more improvements can be expected in the next years, as for the moment only the inner detector tracking was optimised, while more gains can be expected once the similar work is performed for calorimeters and the muon spectrometer.

There is no considerable improvement for storage however. The cost of disk space today exceeds by 4 times the cost of the same volume on tape, therefore we focus on disk. The vast majority of the disk space is needed to store analysis formats such as AODs and their derived ones (DAODs) for both data and Monte Carlo. The Computing Models evolved during Run-2 implementing data reduction based on popularity and implementing a lifetime model removing data from disk and tape once the datasets lifetime expires and the data is not recently accessed. Therefore, at the moment there is little contingency in further reducing the number of disk replicas.

3. The Run-2 analysis model

The DAODs were introduced in the ATLAS analysis model in 2014, during the LHC shutdown between Run-1 and Run-2 (2013-2015). The outputs of the reconstruction (AODs) are centrally processed through organised analysis campaigns called trains. Each train processes events in AOD formats, possibly updates some information, filters the ones suitable for a given analysis, removes the variables not relevant for that analysis and stores the derived format (DAODs) on disk. A train can run multiple selections (carriages) on the same events and produce multiple DAODs. The DAODs can be further processed through the ATLAS offline software (Athena) or the ROOT [5] analysis package, to produce NTUP files for final analyses. Today ATLAS runs approximately 20 trains for a total of 100 carriages, and therefore produces 100 DAOD types with the average size of 1% of the AODs each. The total volume of a full DAOD set is therefore similar to the one of the AOD it has been produced from. DAOD datasets are frequently accessed by analysers: to ensure data integrity and a rapid turnaround of the analyses, two copies of the most recent version of the DAODs are kept on disk. Eliminating the needs to store DAODs would reduce by more than a factor 2 the disk needs.

A possible data processing architecture for HL-LHC

A diagram of the proposed data processing architecture for HL-LHC is shown in Figure 2.

3.1 *The Event Whiteboard*

A considerable reduction of the storage needs can be achieved by reducing or eliminating the need to duplicate physical information between AODs and DAODs. Central to the new data processing architecture is a Data Knowledge Base (DKB) service, hosting the meta-information associated to all ATLAS events. Such an “Event Whiteboard” would contain all the relevant meta-data obtained from distributed computing services such as data management and workload management. It would in addition allow to annotate at the event level user defined meta information. The derivation framework could at this point avoid creating DAOD files but simply annotate the events with a tag indicating that the event passes some selection criteria. The Whiteboard would then be able to provide event lists for a given tag together with the necessary information for the Offline I/O libraries to extract single events or event ranges. The workload management system would be able to define data processing units (jobs) accessing a defined list of events.

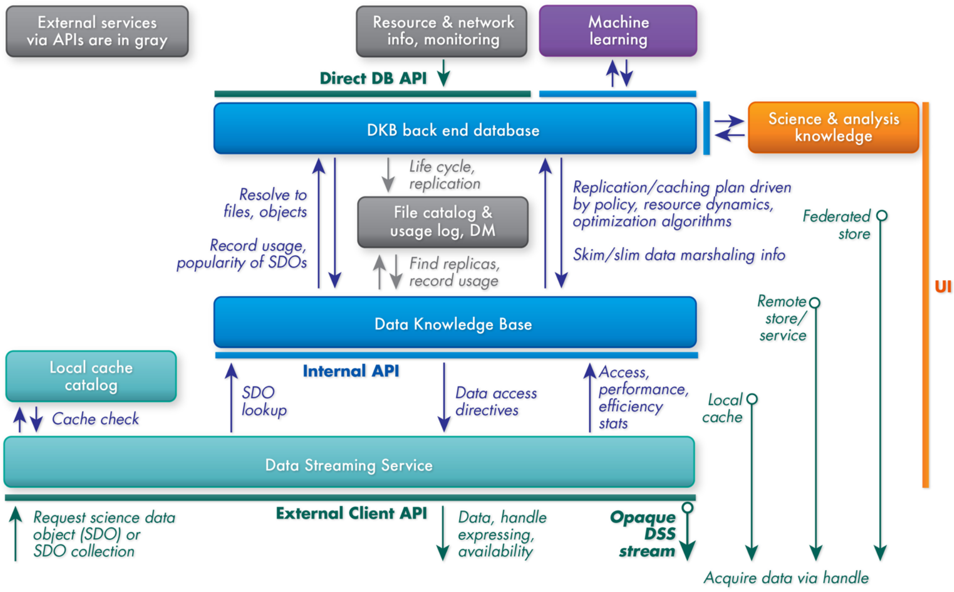

Figure 2. The Data Knowledge Base service and the interaction with external components, among which is the Event Streaming Service

3.2 *Asynchronous pre-fetching and Event Streaming Service*

Currently the ATLAS workload management system and particularly the jobs running at the worker nodes use two different data access techniques, depending on the workflow and storage I/O capabilities: the full copy of the input file containing the events to process into the worker node and the direct access using streaming protocols such as XRootD or HTTP from the storage endpoint. None of the two methods would be efficient in the scenario where the DAOD datasets are simply a logical collection of events. The full copy of the input file would end up downloading a large number of events not to be processed from the AOD, particularly in case the DAOD collection was the result of heavy skimming. Direct streaming would result in I/OWait while data are obtained from the storage locations, leaving idle CPU cycles. An Event Streaming Service would provide a solution. In its first implementation, the client on the worker node would pre-fetch event parcels (collections of events) asynchronously from the actual processing and store them on the local disk, so that CPUs would be able to open and process data locally. In a more advanced implementation, the pre-fetch would be directed at a service deployed in the network neighbourhood of the storage location. This service would prepare event parcels on disk caches under the direction of the workload management system, which would activate payloads accessing such parcels only when the parcels are available and possibly replicated in a convenient location. The system would be more robust and effective if the source data would be consolidated in a small number of storage locations interconnected with fast networks, while the event streaming service caches would need to be distributed as close as possible to the processing units and as broadly as possible. Today ATLAS processes 1 Exabyte of data per year. The event streaming service would need therefore to prepare that volume of DAODs, resulting in approximately 30 GB/s. There is however a considerable reuse of the DAODs (more than 10 times per sample).

Therefore a system delivering 3GB/s of cached data would suffice for Run-2 and Run-3 while one would need 10 times more for Run-4.
The potential gains are considerable: in principle we could eliminate all physical copies of DAODs for both data and Monte Carlo, consisting of 45% of the total disk needs. In reality many caveats apply and such estimates could be only the theoretical limit. The derivation framework does not only filter events but also retains only the relevant quantities based on the physics analysis that such derivation will support. Such reduction would be lost in the proposed model. Therefore we would likely continue to produce physical DAOD files but in a much smaller variety, say one per physics group (10 in total to be compared with 100 formats today). Additionally, today in producing DAOD files we order the event information to optimise data compression and we need to understand how that would happen in a model where no physical DAOD file is produced.

3.3 *Virtual Data*
A second application of the DKB consists in capturing all the necessary information to reproduce derived data. As of today, reproducing derived data is a tedious and error prone operation, requiring manual work. To ensure data integrity we store multiple replicas of the vast majority of the datasets and to avoid analysis reproducibility issues we tend to retain such copies for a long time. Only a very small fraction of such data is accessed after one year since it was produced. Capturing all necessary information to reproduce data in the Knowledge Base system and integrating the workload and data management system with it would allow treating part of the samples as *virtual data.* Knowing the cost to reproduce data, the cost to store replicas on different media and estimating the likelihood for that data to be accessed after a given period of time (or that data to be lost) the system could decide to retain physical copies or re-generate the data in case of need. The system could also decide to create more copies to facilitate the access or avoid the loss. The information sources could be multiple and extensible, including machine learning algorithms for anomaly detection and data popularity forecasts. The automation of the system is the key element in this model. The DKB would provide also the backbone for analysis and data preservation models being developed in ATLAS.

4. Conclusions

We presented an update of the computing challenges for HL-LHC in term of hardware resources. The computing and analysis model we use in Run-2 will not be able to accommodate the HL-LHC hardware needs in a constant funding profile. We propose a different analysis model based on event level processing granularity, virtual data and data reproducibility. While such a model presents a radical change with respect to the current system, single components can be developed and deployed early, complementing the data processing and storage tools we have available today.

References

[1] The ATLAS Experiment at the CERN Large Hadron Collider, **JINST 3** (2008) S08003

[2] S. Campana, “Software and Computing requirements and possible architecture”, ECFA High Luminosity LHC Experiments Workshop, Aix-Les-Bains, France, (2016)

[3] S. Campana, “The ATLAS Computing Challenge for HL-LHC”, Computing in High Energy and Nuclear Physics, San Francisco, United States, (2016)

[4] Agostinelli, S., Allison, J., Amako, K., et al., Geant4 – a simulation toolkit, Nuclear Instruments and Methods in Physics Research, A 506 (2003) 250-303

[5] Rene Brun and Fons Rademakers, ROOT - An Object Oriented Data Analysis Framework, Proceedings AIHENP'96 Workshop, Lausanne, Sep. 1996, Nuclear Instruments and Methods in Physics Research A 389 (1997) 81-86