PHYSLITE - A new reduced common data format for ATLAS

Jana Schaarschmidt[1,*], *James* Catmore[2], *Johannes* Elmsheuser[3], *Lukas* Heinrich[4], *Nils* Krumnack[5], *Serhan* Mete[6], and *Nurcan* Ozturk[7]

[1]University of Washington
[2]University of Oslo
[3]Brookhaven National Laboratory
[4]Technical University of Munich
[5]Iowa State University
[6]Argonne National Laboratory
[7]University of Texas at Arlington

Abstract. The High Luminosity LHC (HL-LHC) era brings unprecedented computing challenges that call for novel approaches to reduce the amount of real and Monte Carlo-simulated data that is stored, while continuing to support the rich physics program of the ATLAS experiment. With the beginning of LHC Run 3, ATLAS introduced a new common data format, PHYS, that replaces most of the analysis-specific formats that were used in Run 2, and therefore reduces the disk storage significantly. ATLAS also launched the prototype of another common format, PHYSLITE, that is about a third of the size of PHYS. PHYSLITE will be the main format for ATLAS at the HL-LHC and aims to serve 80% of all physics analyses. To simplify analysis workloads and further reduce disk usage it is designed to largely replace user-defined analysis n-tuples and consequently contains pre-calibrated objects. Various forms of validations are in place to ensure correct functionality for users. Developments continue towards HL-LHC to improve the PHYSLITE format further.

1 Introduction

ATLAS [1] is one of the main LHC experiments with a diverse and successful physics program, including for example Higgs boson searches and measurements, searches for a huge variety of exotic signatures, high precision Standard Model particle measurements, as well as B-physics, light states and heavy ion physics.
The High Luminosity LHC [2] (HL-LHC) era will start with Run 4 in 2029 and lead to an increase in event sizes and rates: The integrated luminosity expected in Run 4 and Run 5 is 270 fb^{-1} and 350 fb^{-1} per year, respectively. The average pile-up rises from 60 interactions now in Run 3 to about 140 and finally 200 in Runs 4 and 5. The center-of-mass energy will be 13.6-14 TeV. These parameters pose a challenging problem for computing, as budgets and resources won't scale accordingly. To remedy this situation, CPU and storage needs have to be reduced, but without compromising the multiplicity and quality of the physics output.
ATLAS has launched a strong R&D program for HL-LHC software and computing [3] and

*e-mail: jana.schaarschmidt@cern.ch

laid out a road map [4] to advance these efforts to production readiness. One of these efforts is the development of the new common reduced data format PHYSLITE.

2 Data Production and Reduction Chain

In ATLAS, the production of simulated Monte Carlo (MC) data starts with the event generation, followed by the detector simulation; the next step is the digitization, and finally the reconstruction. Detector data is also reconstructed. The standard output format of the reconstruction is called Analysis Object Data (AOD) with a typical size of 300-500 kB per event, containing physics objects and some lower-level information, such as particle flow objects and a limited amount of calorimeter cell information. In another reduction step, the AOD format is trimmed to a DAOD (derived AOD, also called derivation). The DAOD format is the input for physics analyses.

There are several ways in which information is reduced during the derivation step, based on pre-defined selection criteria: Entire events can be removed (*skimming*), objects can be removed (*thinning*), or variables within objects are removed uniformly across events (*slimming*).

Typically, information is reduced when going from AOD to DAOD, but some information is also added. Most prominently, jets are not stored at AOD level, but they are reconstructed from the inputs stored in the AODs and various jet collections are then written out only at DAOD level.

3 Evolution of the Analysis Model from Run 2 to Run 4

In the Run 2 analysis model [5], there were about 100 customized DAOD formats for various physics and performance studies. The DAODs were skimmed, slimmed and thinned, yet there was significant overlap between the various formats in events written, a result of often high skim fractions in particular for MC background samples, but also overlap of the stored branches, as many analyses needed similar inputs. Since generally MC is much bigger than data, this led to a huge disk footprint. The average DAOD size was about 30-50 kB per event, though some formats exceeded this size by a huge factor. Overall, this analysis model led to an inefficient use of storage resources: The entire storage of DAODs was about the same size as all AODs. Furthermore, users typically would create custom flat n-tuples from the DAODs as input to their plots and statistical frameworks, which resulted in a huge number of additional files that are stored on disk.

To improve the analysis model in Run 3 and beyond, a taskforce was created that concluded with specific recommendations. One key idea is the creation of a new common format for physics analysis, called DAOD_PHYS (or just PHYS), which is an unskimmed derivation suitable for 80% of all physics analyses, with a target size of 30 (50) kB per event for data (MC). PHYS contains information about objects and trigger, as well as a thinned track collection, generator-level (*truth*) information, and additional data needed for applying calibration tools and evaluate systematic uncertainties (these tools are also called *CP algorithms*, where CP stands for combined performance). PHYS is a *monolithic* format, meaning there is only one version (though the content of the format can change over time). PHYS typically cannot be used for performance studies (e.g. to derive calibrations), and due to the thinning of low p_{T} inner detector tracks is also not suitable for specialized analyses such as B-physics and heavy ions, and finally also does not contain non-standard objects, such as long-lived particles or emerging objects. For such cases, residual DAOD formats, similar to those used during Run 2, are still needed, but reducing the number of such special formats is the goal

and will directly impact the storage model projections.
A second key idea is the creation of a new reduced common data format, DAOD_PHYSLITE (or just PHYSLITE), that is being prototyped and tested during Run 3. The main idea behind PHYSLITE is that it contains already calibrated and pre-selected objects (imposing loose working points and low cuts on p_{T}) and high level information, such as discriminants or machine learning scores, rather than inputs to such algorithms. This reduces the need to store extra information that would be required to apply calibrations, and it reduces CPU needs since calibrations don't have to be applied again. Preliminary evaluations of the performance of PHYSLITE analyses compared to PHYS resulted in approximately 25% CPU reduction. The target file size of PHYSLITE is 10 (12) kB per event for data (MC). It is expected that in Run 3 the total size of all DAODs will be about 50-70% of the size of all AODs. PHYSLITE is also unskimmed, and also monolithic, meaning one version fits all use cases. Skims, however, can be requested for use cases in which the event selection efficiency is less than 1%. Finally, it is envisaged that PHYSLITE will be analysed directly, without writing out flat n-tuples, which will further decrease the storage needs.
In Run 4 finally, PHYSLITE will replace PHYS with the aim of serving 80% of all physics analyses, and consequently huge savings in storage can be achieved. However, it is anticipated that also during the HL era a number of residual derivation formats will be needed to accommodate special workflows. Those special formats ultimately drive the projected overall storage needs. Additional PHYS- or PHYSLITE-like formats (skims or augmented formats, see Section 6) are also envisaged.
To optimally take advantage of the strengths of tape and disk systems, the current and future analysis models also specify which format is to be stored on which system. AODs are stored exclusively on tape, which is cheaper but offers only slow access. Derivations are stored on disk for fast access but with a limited lifetime. AODs are staged to disk for the central bulk production of DAODs, which typically happens four times per year. Only a fraction of AODs are copied to disk at a time, and once they have been processed, the copies are deleted and the next portion is staged to disk (*data carousel* [6]). A special case, however, is the production of PHYSLITE. Nominally, PHYSLITE is produced by running a series of CP algorithms [7] on the objects in the AODs. An alternative workflow is in place that allows the creation of PHYSLITE from PHYS. This workflow has recently been enabled and requires careful remapping of the missing energy associations, since they are different between the two formats. Since PHYS is stored on disk, the production of PHYSLITE can occur at a higher frequency, up to eight times per year, whenever the format has been updated or when calibrations change. This workflow also reduces CPU needed for the PHYSLITE production by being approximately six times faster than production from AOD. PHYSLITE is not a fixed format, it will continuously evolve to optimally support physics analysers.

4 PHYSLITE Current Size and Composition

The currently available Run 3 version of PHYSLITE is still a prototype, but it has all basic functionalities, so analysers can already use the format for standard analyses, to gain experience but also to help optimize the format.
The file sizes per event for PHYS and PHYSLITE and for various campaigns are listed in Table 1. Real data events are smaller than simulated events since real data does not contain any truth information. The size increased from Run 2 to Run 3 due to larger trigger collections. Work is ongoing to further reduce the file size of PHYSLITE, see also Section 6. Assuming the target sizes for PHYS and PHYSLITE, the total size of the entire Run 3 dataset has been extrapolated in Table 2. This estimate includes multiplicative factors for a number of replicas and versions that are stored concurrently, but do not account for differences in pile-up. Given

these numbers, it would be possible to store the entire PHYSLITE production on a single grid site, which is expected to reduce network traffic and allows easier access for users at local sites.
The composition of PHYS and PHYSLITE for a Run 3 semi-leptonic $t\bar{t}$ MC sample is displayed in Fig. 1. The average pile-up in this sample is 45 (larger pile-up will increase the file size). PHYS is dominated by the jet collections, taking about a quarter of the whole size, followed by the EGamma containers, which include electron and photon candidates but also cluster information. PHYSLITE is currently dominated by trigger data (27%), but work is ongoing to reduce this. Tracking and truth information are also sizable fractions. The analysis objects (AnalysisJets, AnalysisPhotons, etc.) are the calibrated objects ready for use in physics analysis. Those individual fractions are small, the largest one being the AnalysisJets container with 12%. Additional information stored in PHYSLITE is currently needed to run CP algorithms, in particular those that perform systematic variations.

Table 1: The current file sizes (in kB per event) for PHYS and PHYSLITE for various data and MC campaigns. Work is ongoing to reduce the size of PHYSLITE further.

Format	Run 2 MC $t\bar{t}$	Run 3 MC $t\bar{t}$	Data 16	Data 22
PHYS (kB/event)	33.8	40.9	18.2	20.5
PHYSLITE (kB/event)	13.0	16.1	6.2	6.2

Table 2: Some parameters and the total size estimated for the entire Run 3 data and MC production, assuming the target sizes of PHYS and PHYSLITE are met.

Dataset	Events [10^9]	Size [kB/evt]	Replicas	Versions	Total [PB]
PHYS Data	19	30	4	2	4.6
PHYS MC	24	50	4	2	9.6
PHYSLITE Data	19	10	4	2	1.5
PHYSLITE MC	24	12	4	2	2.3

5 PHYSLITE Validation

The new PHYSLITE format is validated by comparing it to PHYS. The PHYS format itself has been available since Run 2 and its validation has been completed previously so it serves as the reference. Standard objects in PHYSLITE should agree exactly with PHYS objects that are passed through the same calibrations. Since objects stored in PHYSLITE are already calibrated during PHYSLITE production, two sorts of comparisons can be done: Firstly the object from PHYSLITE can be compared directly to those in PHYS that undergo calibration, and secondly the objects in PHYSLITE can be passed again through the respective series of CP algorithms and also be compared to PHYS. The second workflow is typically used when evaluating systematic uncertainties, and some of those CP algorithms currently also rerun calibrations, depending on the object type. Figure 2 shows both types of comparison, and perfect agreement is seen for both workflows and all physics objects. These manual comparisons are performed with analysis frameworks and closely resemble workflows used for physics analysis.
Since PHYSLITE is subject to frequent updates, and a huge number of analysers are able

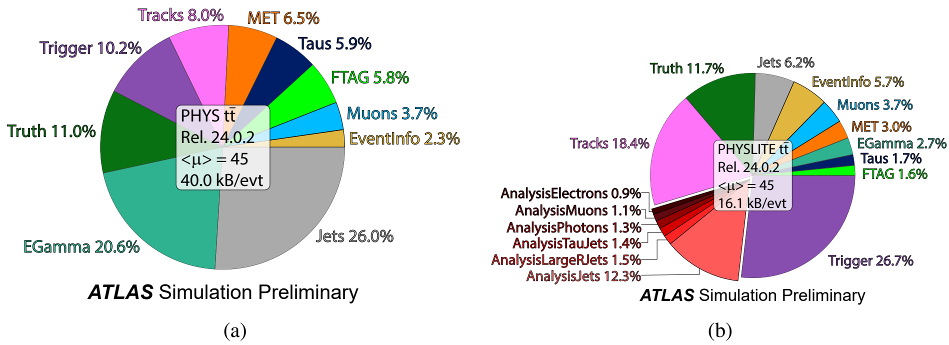

Figure 1: The composition of a Run 3 $t\bar{t}$ sample in (a) PHYS and (b) PHYSLITE format. The disk size from PHYS is dominated by the jet collections, while PHYSLITE is currently dominated by the trigger information. The containers that hold the calibrated analysis objects in PHYSLITE are shown in the red slices. Plots taken from Ref. [8].

(and encouraged) to modify the format, tools are in place that enable automatized checks and validations, complementing the detailed manual comparisons using analysis frameworks. Various checks are available:

- Command line scripts can print the content of AODs and DAODs, check container sizes and also list object counts and compression rates. Other scripts can perform integrity checks, such as printing differences between branches comparing PHYS and PHYSLITE files, or producing plots for all ROOT-readable [9] variables and their ratio.
- ATLAS Release Tests (ART) run daily and produce PHYS and PHYSLITE files from the current main branch of Athena [10]. They also create comparison plots for basic kinematics that are displayed on a web interface. Differences between the test and reference files are automatically detected and flagged.
- Another test runs CP algorithms on all the objects for existing PHYS and PHYSLITE files, the tests also output n-tuples that are compared. This test is designed to run as part of the continuous integration (CI) pipeline for merge requests in git, which will fail in case any of the standard tools do not work.
- SPOT (Software Performance Optimization Team) monitoring [11] is a tool that displays graphs of CPU utilization and memory allocations for the PHYS and PHYSLITE productions. It also checks the container sizes for various formats and visualizes their fractions relative to the total size and collects the results on a web interface.
- An ongoing development is the creation of web sites that dynamically show the content of PHYSLITE and display doc-strings and compression levels for variables.

6 Future Developments

The current Run 3 version of PHYSLITE is a good basis for analysers to start using the new format and provide feedback. Minor bugs are still being fixed, some variables (for example to support various working points of physics object definitions) are added, and the applied preselections are currently revisited and optimized based on early analyser feedback. But further developments are ongoing to elevate the current prototype to a new level, to be ready for HL-LHC:

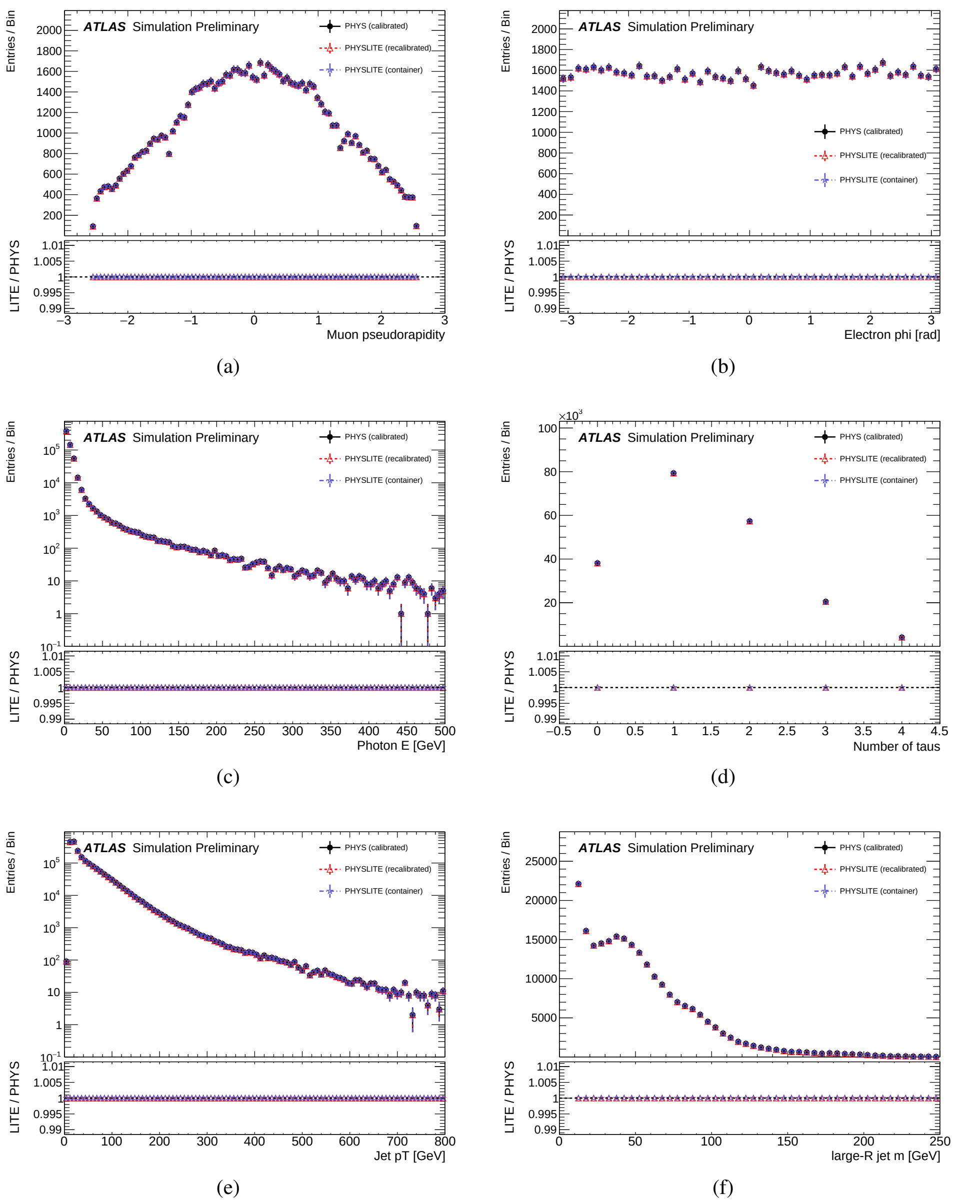

Figure 2: Validation of PHYSLITE by comparing object kinematics to those from objects in PHYS that have been calibrated using CP algorithms (black circles), for (a) muons, (b) electrons, (c) photons, (d) hadronic taus, (e) small-radius jets and (f) large-radius jets. Two comparisons are overlayed, either directly using the stored objects from PHYSLITE (red triangles) or passing them again through their respective CP algorithms (blue stars). Plots taken from Ref. [8].

- RNtuple-version:
  Currently, ROOT's 25+ years old TTree is at the core of any AOD-like format in ATLAS for the actual storage of information in branches. While both TTree and RNTuple are columnar formats, RNTuple is a modernized version, a type-safe format with exception-based I/O error handling, which is also more storage-efficient (up to 25% anticipated) and allows for larger read throughput, according to first evaluations [12].
- Lossless and lossy compression:
  Storing data can be more efficient when compression is applied. Several lossless compression algorithms are provided by ROOT. Performance studies for these algorithms have revealed that a compromise needs to be found between the level of compression and readout speed [13]. These studies will be considered in future iterations of data and MC production. Another approach is lossy compression, achieved by reducing the number of bits stored for the mantissa of floating point numbers for variables stored in PHYSLITE, which is planned to be applied in the future. The level of lossy compression should be guided by the detector precision and validations must ensure that physics results are not impacted.
- Event sample augmentation:
  Not all physics analysis will be able to use PHYS or PHYSLITE, due to the absence of non-standard objects, such as very low p_{T} objects or long-lived particles, or special requirements such as non-standard triggers or vertices. It is essential that ATLAS continues to support such workflows in the HL era, particularily motivated by the fact that beyond the Standard Model physics might very well be hiding in places that have not been explored yet. Such non-standard analyses can use the custom, skimmed DAODs as during the Run 2 analysis model. However, the concept of event sample augmentation [14] offers an alternative. It enables the extension of the common formats with supplementary data obtained from a secondary skimmed format, but only for the subset of events that meet the skim criteria. In the event sample augmentation configuration, the secondary format is incorporated as a ROOT friend tree into the base format and stored within the same file or in a separate file. This approach will assist in reducing the storage needs overall.
- Columnar analysis:
  So far, DAODs are processed event-by-event, but the same information can be processed object-by-object, i.e. in columnar style. There are several advantages to this approach: Since vectors of objects are stored contiguously in memory, they can be processed very rapidly, and columnar analysis can integrate well with modern python tools and industry standards. PHYSLITE is a testbed for developing a columnar analysis prototype in ATLAS. The main work required is to adapt the CP algorithms that are written in C++, by building an interface to use these tools within a python environment for handling the inputs from PHYSLITE and outputs for downstream analysis [15].

The goal for Run 4 is that PHYSLITE replaces PHYS as the default analysis format. It will be even more streamlined and very fast to process, and it will enable timely physics results within available resource budgets.

7 Summary

PHYSLITE is the new reduced common data format for ATLAS, that has been introduced into the analysis model since Run 3, and will become the default format for 80% of all physics analyses in Run 4, replacing a huge number of custom DAODs formerly used. PHYSLITE already contains calibrated objects, therefore objects can be used directly in physics analysis, saving CPU resources. It is also a streamlined format, containing only the minimal information necessary for standard analyses, making it a factor 3-5 smaller compared to PHYS.

Being such a small format, it can be stored on a single grid site, thus reducing network traffic. It is also a dynamic format, meant to evolve over time to optimally support physics analyses. Automatized validations perform checks and complement detailed manual comparisons to PHYS. Future versions of PHYSLITE will be based on RNTuple, include optimal lossless and lossy compression, and additional data can be stored in extra trees. Finally, PHYSLITE is to be used within columnar analysis frameworks.

References

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

[2] O. Aberle, et al., *High-Luminosity Large Hadron Collider (HL-LHC): Technical design report*, CERN Yellow Reports: Monographs (CERN, Geneva, 2020), `https://cds.cern.ch/record/2749422`

[3] P. Calafiura, J. Catmore, D. Costanzo, A. Di Girolamo, *ATLAS HL-LHC Computing Conceptual Design Report* (2020), `https://cds.cern.ch/record/2729668`

[4] ATLAS Collaboration, *ATLAS Software and Computing HL-LHC Roadmap* (2022), `https://cds.cern.ch/record/2802918`

[5] J. Catmore, J. Cranshaw, T. Gillam, E. Gramstad, P. Laycock, N. Ozturk, G.A. Stewart, on behalf of the ATLAS Collaboration, *A new petabyte-scale data derivation framework for ATLAS* (2015), `https://dx.doi.org/10.1088/1742-6596/664/7/072007`

[6] M.S. Barisits, M. Borodin, A. Di Girolamo, J. Elmsheuser, D. Golubkov, A. Klimentov, M. Lassnig, T. Maeno, R. Walker, X. Zhao (ATLAS), *ATLAS Data Carousel* (2020), `https://cds.cern.ch/record/2709950`

[7] N. Krumnack, *Common CP Algorithms at ATLAS*, these proceedings (2023)

[8] *Validation of PHYSLITE Derivations in Run 3*, `https://atlas-swc.web.cern.ch/PlotPages/2023-05-PHYSLITE/`

[9] F. Rademakers, et al., *root-project/root: v6.20/06* (2020), `https://doi.org/10.5281/zenodo.3895852`

[10] ATLAS Collaboration, *Athena* (2021), `https://doi.org/10.5281/zenodo.4772550`

[11] *ATLAS SPOT main page*, `https://atlaspmb.web.cern.ch/atlaspmb/`

[12] F. De Geus, *Integration of RNTuple in ATLAS Athena*, these proceedings (2023)

[13] C. Marcon, *Optimizing ATLAS data storage: the impact of compression algorithms on ATLAS physics analysis data formats*, these proceedings (2023)

[14] P. van Gemmeren, *Framework for custom event sample augmentations for ATLAS analysis data*, these proceedings (2023)

[15] N. Krumnack, *Columnar analysis and on-the-fly analysis corrections at ATLAS*, these proceedings (2023)