EPJ Web of Conferences **295**, 01018 (2024) https://doi.org/10.1051/epjconf/202429501018
CHEP 2023

Deployment and Operation of the ATLAS EventIndex for LHC Run 3

Elizabeth J. Gallas[1,*], *Evgeny* Alexandrov[2], *Igor* Alexandrov[2], *Dario* Barberis[3,**], *Luca* Canali[4], *Elizaveta* Cherepanova[5], *Alvaro* Fernandez Casani[6], *Carlos* Garcia Montoro[6], *Santiago* Gonzalez de la Hoz[6], *Alexander* Iakovlev[2], *Fedor* Prokoshin[2], *Jose* Salt Cairols[6], *Javier* Sanchez[6], *Grigori* Rybkine[7], and *Miguel* Villaplana Perez[6]

[1]University of Oxford, Oxford, UK
[2]Joint Institute for Nuclear Physics, Dubna, Russia
[3]University of Genoa and INFN, Genoa, Italy
[4]CERN, Geneva, Switzerland
[5]University of Amsterdam and NIKHEF, Amsterdam, Netherlands
[6]IFIC, University of Valencia and CSIC, Valencia, Spain
[7]IJCLab, Université Paris-Saclay, Orsay, France

Abstract. The ATLAS EventIndex is the global catalogue of all ATLAS real and simulated events. During the LHC long shutdown between Run 2 (2015-2018) and Run 3 (2022-2025) all its components were substantially revised and a new system was deployed for the start of Run 3 in Spring 2022. The new core storage system, based on HBase tables with a SQL interface provided by Phoenix, allows much faster data ingestion rates and scales much better than the old one to the data rates expected for the end of Run 3 and beyond. All user interfaces were also revised and a new command-line interface and web services were also deployed. The new system was initially populated with all existing data relative to Run 1 and Run 2 datasets, and then put online to receive Run 3 data in real time. After extensive testing, the old system, which ran in parallel to the new one for a few months, was finally switched off in October 2022. This paper describes the new system, the move of all existing data from the old to the new storage schemas and the operational experience gathered so far.

1 Introduction

The ATLAS experiment [1] collects several billion proton-proton, proton-ion and ion-ion interactions at the LHC accelerator at CERN every year. These "events" are then processed several times, resulting in many different formats and versions of the same original information. In addition, simulated events are generated to compare the results of the analysis of real data with different physics models. All these data need to be catalogued in a large, high performance and high reliability system that can provide information on single events out of several hundred billion records. The EventIndex [2] system catalogues all ATLAS events,

*Presenter: Elizabeth.Gallas@physics.ox.ac.uk
**Corresponding author: Dario.Barberis@cern.ch

both real and simulated, and provides a set of tools to search and retrieve information on single events or on event groups, following user selections. Its design started in 2013 [3] and the first implementation was fully functional for the start of LHC Run 2 in 2015 [4]. The original version of the EventIndex provided a stable and reliable service throughout LHC Run 2 (2015-2018), but like all software projects it had to be upgraded in order to stand the expected higher data rates for Run 3 (2022 onwards) and beyond. This paper describes the modifications implemented for this purpose in 2021-2022 and the operational experience with the new, now current, system.

2 Architecture

The *raison d'être* of the EventIndex is to enable ATLAS members to search for and retrieve one or more individual events from the tens of millions of data files, in order to perform detailed checks, more refined analyses or produce event displays – the so-called "event picking" operations. For this purpose a metadata catalogue containing the main event identification variables, such as run number and event number, and the event location in different formats and processing versions, the GUIDs [5] of the files containing it, is needed. Once such a catalogue exists, it can be integrated with trigger information and other event metadata, widening the range of searches and allowing the computation of event counts based on combinations of trigger selections, calculations of trigger overlaps within a dataset or of event overlaps in different derivation streams. Another important application of the EventIndex is the production consistency checks, as the EventIndex Producer is the first process that reads back all produced files and checks their integrity and completeness.

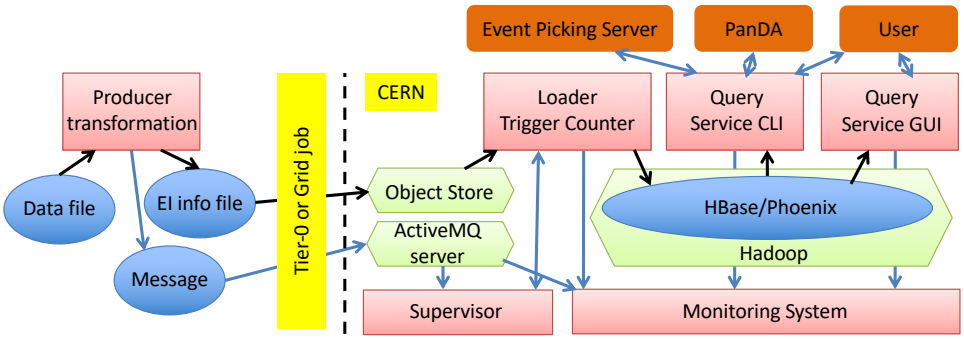

Figure 1. Architecture of the EventIndex system as implemented for LHC Run 3. The individual components are described in the text. The black arrows indicate the indexing data flow; the blue arrows indicate the information flow or exchange between the different tools.

The EventIndex system was designed from the beginning with a modular architecture, following the main data flow. Figure 1 shows a schema of the architecture and its components, as implemented for LHC Run 3. The system must be able to scale to eventually store trillions of event records, stand ingestion rates in excess of 10 kHz and react to queries in times that are independent of the volume of stored data. The implementation is based on the BigData tools that are available in the Apache Hadoop [6] ecosystem, including its native HDFS file system, the Apache HBase database [7] and Apache Phoenix [8], a SQL layer over HBase.

The main system components are:

- *Data Production*: The EventIndex Producer transformation processes datasets as soon as they are produced and extracts indexing information (event metadata) to files that are sent to an Object Store at CERN [9]. At the same time, information about these files is sent to the CERN ActiveMQ messaging server [10].
- *Data Collection*: The Supervisor process receives the messages from the ActiveMQ server and orchestrates the retrieval of the indexing files from the Object Store, their merging and validation, and the upload of the indexing data to the Hadoop cluster.
- *Data Storage*: The EventIndex records are stored in HBase [7] tables. The main tables are the *Dataset* table, with global information for each indexed dataset, and the *Events* table, which holds all individual records for all events. The Phoenix [8] layer allows operations on HBase tables through SQL commands.
- *Data Access*: A command-line interface provides data query and retrieval tools that can be used by authorised people (members of the ATLAS Collaboration); it can be used interactively or through the PanDA [11] system to submit Grid jobs. A graphical interface that connects the EventIndex to other ATLAS databases [12] and integrates the dataset-level information is in preparation.
- *Event Picking Server*: A new service with a graphical interface, the Event Picking Server [13] automates the operations needed for event picking and returns asynchronously the resulting files to the requestor in a common area.
- *Monitoring*: The health of all servers running EventIndex processes is constantly monitored along with their performance [14]. In addition, a suite of functional and performance tests [15] mimicking typical client usage are run periodically and notify experts when results are incomplete or otherwise different from expectations.

3 Revised and New Components

The EventIndex is a software project that uses almost exclusively free open-source software, in particular BigData tools in the Hadoop ecosystem. These tools evolve quite rapidly in time, as additional features become available and their performance improves; any system using them must also evolve in parallel. All EventIndex components were revised or re-implemented in advance of the start of LHC Run 3 in 2022 [16].

3.1 Producer

The Producer transformation extracts event metadata from files containing real or simulated events as they are produced, and packs them into files that are sent to the CERN Object Store. It is implemented as a Python script that runs within the ATLAS Athena software framework [17]; for this update it was re-implemented in Python3. At the same time, the code dealing with the previous data collection mechanism (that used the ActiveMQ messaging system also for the data files) was removed and the network connectivity was made more resilient.

3.2 Data Collection

As the data transport mechanism through the Object Store at CERN was demonstrably more performant than the older messaging system, code supporting the old mechanism was removed and a failover solution using the CERN EOS data store [18] was implemented instead. In this case, which happened so far only a few times each year, the Supervisor can recover files from EOS at a later stage and push them to the Object Store when it becomes available.

A new component, the Loader, was developed to store all indexing information related to a given dataset into the permanent data store in HBase. The Supervisor gathers from the ActiveMQ server the links to the location of all files with indexing information for each dataset and passes them to the Loader that will insert the records into the the HBase tables in the correct format and update the internal indices [19]. Apache Spark [20], with the Phoenix connector, is used in the data ingestion pipeline.

3.3 Data Store

The data store is the core component of the EventIndex system. The original version based on Hadoop MapFiles was developed in 2014 and soon complemented by an additional data store implemented in HBase in order to meet the requirements on query response time. The current version is based on HBase tables, with Phoenix as the SQL interface [16, 19]. As all events in ATLAS belong to datasets (groups of files containing statistically similar events), it was logical to organise the data into a *Dataset* table and an *Events* table; the *Events* table is very large, with over 500 billion event records in Spring 2023, but with data grouped into column families and an optimized primary key that ensures a balanced use of all HBase regions. In this way the query and retrieval performance is independent of the total data volume. Snappy compression is applied to keep the table size under control. Smaller auxiliary tables are used to store additional quantities, such as data types and trigger chains, that are referenced in the main tables.

3.4 Data Access

As the data store was replaced by a new implementation, all data access tools had to be re-implemented too. A new Query Service was developed to interact with HBase through the SQL interface provided by Phoenix [21]. Its Command-Line Interface comprising the `dataset-list`, `event-list` and `event-lookup` utilities gives the user the possibility to list and select datasets and events according to specifications supplied through their many options, and to retrieve detailed information about the selected events.

3.5 Event Picking Server

The Event Picking Server [13] is a new tool designed and developed since 2019 to automate the event picking workflow, especially when users need to extract several thousand events at once. Its entry point for the users is a graphical interface where the user supplies a text file with the list of run and event numbers of all events to be retrieved, the event format and data (trigger) stream. Behind the scenes, a process splits the event list by run number, uses the Query Service to get the locations (as GUIDs) of the files with those events, submits the event picking jobs to the PanDA workflow management system [11] that will run the jobs on the WLCG Grid, and finally notifies the user upon completion. The graphical interface can also be used to monitor the progress of the requests, which is a very useful feature given that the completion time can be of the order of several days when retrieving raw data from files on tape.

3.6 Testing and Monitoring

Two groups of jobs are run several times a day, related respectively to functional and performance tests [15]. The functional tests run event picking jobs in different configurations

(separate event lookup or integrated in the PanDA job), different data types (real and simulated data) on datasets of different ages (from just registered to several years old), in order to check that the established functionalities are always available. The performance tests always run the same event lookup processes, on different numbers of events and accessing different datasets, to make sure that there is no performance degradation with the constant increase of data volume. Both functional and performance tests were revised and adapted to the new client tools described above.

4 Operation and Performance

The operation of the EventIndex Producers started back in 2015 at the beginning of LHC Run 2 and continued without interruption, indexing all newly collected real data during data-taking periods, plus reprocessed and simulated data. Figure 2 (left) shows the number of datasets indexed daily between January 2022 and March 2023, on average a few hundred datasets each day.

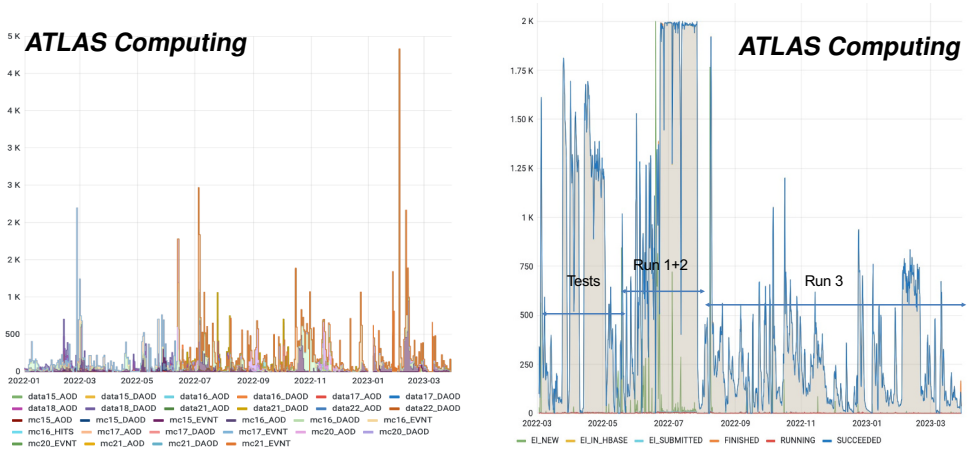

Figure 2. Left: datasets indexed daily by the EventIndex Producer between January 2022 and March 2023, divided by data type. Right: number of dataset loader processes active daily in HBase between March 2022 and March 2023.

The system went through extensive tests during the early months of 2022, and was put in operation at the beginning of the LHC Run 3 data-taking period in parallel to the old system. As the performance was satisfactory, all existing previous data from Run 1 and Run 2 were copied over during Summer 2022; at that point the old system was decommissioned. Figure 2 (right) shows the number of Loader processes running in parallel during the testing phase, the old data import and then the new Run 3 data inserts.

Most of the indexed datasets contain simulated data and are much smaller than datasets containing real data, so the global number of event records in the EventIndex store is dominated by real data, as shown in Figure 3 (left). At the end of March 2023 the data store contained 540 billion event records belonging to 280'000 datasets, occupying 47 TB in HBase (times the factor 3 replication). The EventIndex collects and stores the trigger information only once for each event, when indexing the datasets in AOD (Analysis Object Data) format, so there are two groups of event record size, about 50 bytes without trigger information, and about 150 bytes with trigger information.

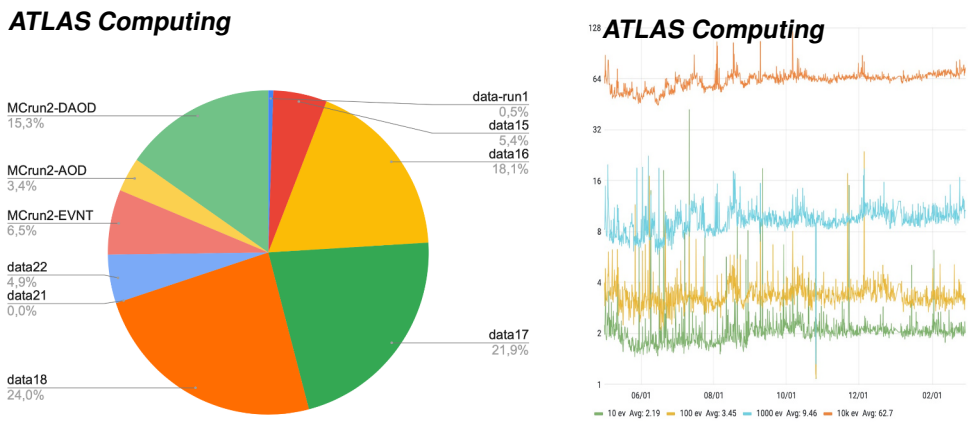

Figure 3. Left: fraction of event records stored by data type at the end of March 2023. Right: lookup times in seconds for queries for 10, 100, 1000 and 10'000 events from performance tests executed between May 2022 and February 2023.

The functional and performance tests showed a better stability and reduced response times compared to the previous implementation [15]. Queries for a few events return their results in seconds, and even queries for 10'000 events return in about a minute. This performance is definitely sufficient for interactive users through the command-line interface and asynchronous users through the Event Picking Server. Figure 3 (right) shows the response times to queries for 10, 100, 1000 and 10'000 events over a long time period; they represent typical user queries, which are normally for a few events for event displays and for several thousand events for massive picking in case of special analyses. The variations and spikes are due to other activities taking place on the same Hadoop cluster, which is not dedicated to the EventIndex but has other users and their applications running at different times.

5 Conclusions

The comprehensive revisions and reimplementation of the ATLAS EventIndex for LHC Run 3 have demonstrated remarkable efficacy in handling large amounts of data and improved accessibility for users. The migration from the HDFS MapFiles based storage system and the additional data store implemented in HBase to a single data store based on HBase tables with the Phoenix SQL interface has streamlined the data workflow and allowed greater data ingestion rates, which is especially crucial considering the increasing data rates projected for the completion of Run 3 and future runs.

The development of a new command-line interface and web services has not only enhanced the user experience but also facilitated real-time data management, including the seamless integration of the existing Run 1 and Run 2 datasets. The successful parallel operation of the new system alongside the old, followed by the decommissioning of the older system, further underscores the efficiency and superior performance of the revised system.

Furthermore, the incorporation of new tools such as the Event Picking Server has significantly automated the event picking workflow, thereby streamlining the analysis process for researchers. The stability and improved response times of the revised system, proven by the functional and performance tests, have underscored the robustness of the system in handling extensive queries.

In conclusion, the revamped ATLAS EventIndex system has shown superior performance and greater potential to address the increasing demands of the LHC runs, paving the way for more refined analyses and efficient data management. Future work will focus on further optimizing this system and adapting to the evolving needs of LHC runs.

References

[1] ATLAS Collaboration, *The ATLAS experiment at the CERN Large Hadron Collider*, *JINST* **3** (2008) S08003. https://doi.org/10.1088/1748-0221/3/08/S08003

[2] Barberis D. et al., *The ATLAS EventIndex: A BigData Catalogue for All ATLAS Experiment Events*, *Comput.Softw.Big Sci.* **7** (2023) 1,2. https://doi.org/10.1007/s41781-023-00096-8

[3] Barberis D. et al., *The ATLAS EventIndex: an event catalogue for experiments collecting large amounts of data*, *J.Phys.Conf.Ser.* **513** (2014) 042002. https://doi.org/10.1088/1742-6596/513/4/042002

[4] Barberis D. et al., *The ATLAS EventIndex: architecture, design choices, deployment and first operation experience*, *J.Phys.Conf.Ser.* **664** (2015) 4, 042003. https://doi.org/10.1088/1742-6596/664/4/042003

[5] GUID: https://www.rfc-editor.org/rfc/pdfrfc/rfc4122.txt.pdf

[6] Apache Hadoop and associated tools: https://hadoop.apache.org

[7] Apache HBase: https://hbase.apache.org

[8] Apache Phoenix: https://phoenix.apache.org

[9] Fernández Casaní A. et al., *A reliable large distributed object store based platform for collecting event metadata*, *J Grid Comp* (2021) 19:39. https://doi.org/10.1007/s10723-021-09580-0

[10] ActiveMQ: http://activemq.apache.org

[11] Barreiro Megino F.H. et al., *PanDA for ATLAS distributed computing in the next decade*, *J.Phys.Conf.Ser.* **898**:052002 (2017). https://doi.org/10.1088/1742-6596/898/5/052002

[12] Gallas E.J. et al., *Utility of collecting metadata to manage a large scale conditions database in ATLAS*, *J.Phys.Conf.Ser.* **513**:042020 (2014). https://doi.org/10.1088/1742-6596/513/4/042020

[13] Alexandrov E. et al., *Development of the ATLAS Event Picking Server.*, *Proc. 9th Int. Conf. "Distributed Computing and Grid Technologies in Science and Education" (GRID'2021)* (2021) Dubna (Russia). https://doi.org/10.54546/MLIT.2021.35.43.001

[14] Alexandrov E. et al., *BigData Tools for the Monitoring of the ATLAS EventIndex*, *Proc. VIII Int. Conf. "Distributed Computing and Grid technologies in Science and Education" (GRID'2018)* (2018) Dubna (Russia), http://ceur-ws.org/Vol-2267/91-94-paper-15.pdf

[15] Cherepanova E. et al., *Testing framework and monitoring system for the ATLAS EventIndex*, these proceedings (2023).

[16] Barberis D. et al., *The ATLAS EventIndex for LHC Run 3*, *EPJ Web of Conferences* **245**, 04017 (2020). https://doi.org/10.1051/epjconf/202024504017

[17] Stewart G.A. et al., *Multi-threaded software framework development for the ATLAS experiment*. *J.Phys.Conf.Ser.* **762**:012024 (2016). https://doi.org/10.1088/1742-6596/762/1/012024

[18] EOS: https://eos-docs.web.cern.ch

[19] Garcia Montoro C. et al., *HBase/Phoenix-based Data Collection and Storage for the ATLAS EventIndex*, these proceedings (2023).

[20] Apache Spark: https://spark.apache.org
[21] Rybkine G., *Query Service for the new ATLAS EventIndex system*, these proceedings (2023).