

PAPER • OPEN ACCESS

Towards a coherent Data Life Cycle in Astroparticle Physics

To cite this article: V Tokareva *et al* 2020 *J. Phys.: Conf. Ser.* **1525** 012070

View the [article online](#) for updates and enhancements.



IOP | ebooks™

Bringing together innovative digital publishing with leading authors from the global scientific community.

Start exploring the collection—download the first chapter of every title for free.

Towards a coherent Data Life Cycle in Astroparticle Physics

V Tokareva¹, A Haungs¹, D Kang¹, D Kostunin², F Polgart¹,
D Wochele¹ and J Wochele¹

¹ Karlsruhe Institute of Technology, Institute for Nuclear Physics, 76021 Karlsruhe, Germany

² Deutsches Elektronen-Synchrotron, 15738 Zeuthen, Germany

E-mail: victoria.tokareva@kit.edu

Abstract. The German-Russian Astroparticle Data Life Cycle Initiative (GRADLCI) aims to develop a data life cycle (DLC), namely a clearly defined and maximally automated data processing pipeline for a combined analysis of data from the experiment KASCADE-Grande (Karlsruhe, Germany) and experiments installed at the Tunka Valley in Russia (TAIGA). The important features of such an astroparticle DLC include scalability for handling large amounts of data, heterogeneous data integration, and exploiting parallel and distributed computing at every possible stage of the data processing. In this work we provide an overview of the technical challenges and solutions worked out so far by the GRADLCI group in the framework of a far-reaching analysis and data center. We will touch the peculiarities of data management in astroparticle physics and employing distributed computing for simulations and physics analyses in this field.

1. Introduction

Many promising concepts and techniques are currently of high interest for astroparticle physicists. Rapidly developing multi-messenger astronomy demands sophisticated analysis techniques, e.g. deep learning methods. By the validation of new theoretical models in the field through employing advanced statistical techniques to handle the increasing amount of available data big questions are approached to expand the human knowledge of nature.

Thus, two fundamental issues, data volume growth and increasing analysis complexity, encourage scientists in the field to look for effective data management solutions.

GRADLCI [1] is a project by the KASCADE (Karlsruhe, Germany) [2,3] and TAIGA (Tunka Valley, Russia) [4] astroparticle physics experiments, aimed to join the efforts in building a common data and analysis center for multi-messenger astroparticle physics. The initial objectives of the project include adding access to some of the TAIGA data to the KCDC [5] data portal of the KASCADE experiment, developing data mapping plugins and software for joint data analysis as well as providing capabilities for the analysis directly on the side of the KCDC data center.

2. Experiments

2.1. KASCADE

The Karlsruhe Shower Core and Array DEtector (KASCADE) is a cosmic-ray detector, which was operating on the site of the Karlsruhe Institute of Technology (KIT), Campus North,



Germany. From October 1996 till December 2013 (4383 days of effective exposure) about 1.7 billion events were measured, which resulted in about 450 million reconstructed events with 4 TB of data. The experiment was aimed to study the spectrum and composition of cosmic rays at primary energies from 10^{14} to 10^{17} eV via the observation of extensive air showers. To achieve this goal, 252 scintillation detectors were placed on an area of 200×200 m² at 110 m asl, corresponding to an average atmospheric depth of 10^{22} g/cm². Later the setup was extended to the KASCADE-Grande installation, extending the effective KASCADE area and the energy range by a factor of 10, as well as with the LOPES radio extension [6] which enabled the study of radio emission in extensive air-showers.

2.2. TAIGA observatory

The Tunka Advanced Instrument for cosmic ray physics and Gamma Astronomy (TAIGA) is a facility designed for ground-based gamma-ray astronomy in the energy range from a few TeV to several PeV as well as for investigations of cosmic rays with primary energy from 100 TeV to several EeV.

It is a complex hybrid detector system, which consists of several components. There are an array of imaging atmospheric Cherenkov telescopes (TAIGA-IACT [7]), wide-angle atmospheric Cherenkov timing arrays (TAIGA-HiSCORE [8]) for lower energies and Tunka-133 [9] for higher energies, a surface scintillator array (Tunka-Grande [10]) and a radio antenna extension (Tunka-Rex [11]).

Since the TAIGA and KASCADE experiments are at the same latitude and monitor the same region of the celestial sphere, measure the same part of the energy spectrum of cosmic rays and using the same models of hadronic interactions to interpret the data, joint analyses of the data were shown to be possible and to be of a particular interest [12].

3. Towards a global cosmic ray data center

3.1. Update of the KASCADE Cosmic Ray Data Center

The KASCADE Cosmic Ray Data Center (KCDC) [13] is intended to provide an open access to the full cosmic ray data collected by the KASCADE experiment.

It was established in 2013 following the idea of the Berlin Declaration on Open Data and Open Access [14]. With the last platform release NABOO [15], the events from the whole measuring time of KASCADE-Grande are delivered, including 433 million air showers. At the KCDC website one can further find published spectra from various experiments, and detailed educational examples on data analysis [5].

With respect to the open access conception, only non-commercial open source software is used in KCDC. Thus, the KCDC web site was developed using the *Django* web framework [16] and runs on an *nginx* [17] server. It communicates with the database server and worker nodes using the *RabbitMQ* [18] message-broker software. In order to facilitate adding new detector components without the restraint of a fixed database scheme, the experimental data are stored in a NoSQL database *MongoDB* [19]. The worker nodes perform data selections issued by users with custom Python tools. Each node is monitored and managed via the asynchronous task queue *Celery* [20] based on distributed message passing. The finished selections are then stored on a dedicated FTP server accessible to the registered users.

Within the GRADLCI project we are performing the modification and expansion of the KCDC portal so that this resource can also be used to access the data from the TAIGA experiment, and in particular, the Tunka-133 setup. The data life cycle scheme, including data from both experiments, is shown in fig. 1.

In fig. 1 the distributed data storages S_i are connected to an aggregation server (AS) over the internet using adapters A_i . To speed up data retrieval on the AS side, search over metadata

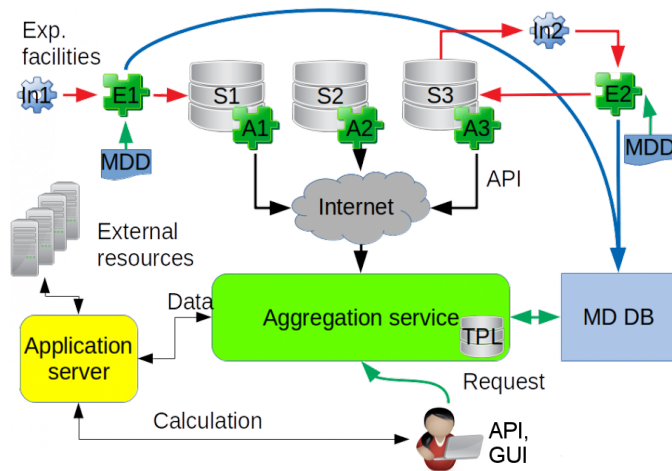


Figure 1. Scheme of the joint data life cycle architecture by GRADLCI. Here S_i stand for local data storages, In_i represent data sources of different types, MDD is a metadata description, E_i are metadata extractors, A_i are adapters that provide APIs for the data access, TPL is a template library, and $MD DB$ is the metadata database.

database is employed. Data caching and storage screening are included into the AS functionality as well. From the AS side data can be transferred to an application server for analysis.

In the next subsection we concentrate on the important features the application server and the system in whole should fulfill. In particular, special plugins for data mapping are required for the joint analysis of data from different detectors. Also we consider reduction of queue time for the analysis job, up to performing the analysis online (immediately after receiving a user request). Because of the fact that the data amount will increase, as well as a possible extension of the data center with data from new detectors, we require in general a scalability for the entire system.

3.2. Data mapping plugins

The development of common access interfaces for heterogeneous, distributed data storage naturally leads to the idea of joint data analyses. Approaching such a task, one should keep in mind, that data mapping for different setups is a non-trivial task, which should be proven to be possible and solved for each selected group of setups individually.

For the time being, as the possibility of data mapping between KASCADE and Tunka-133 has been shown [12], a corresponding data-mapping plugin is under development.

3.3. System scalability

A significant advantage of the concept presented in fig. 1 is the flexibility of setting up individual components of the system, making it easy to expand access to the distributed data storage by adding new storage resources. To add a new data store to the system, one needs to configure a storage adapter that unifies data for working with them in a general way. Caching, data selection and preprocessing functionality, employed at the AS server site reduces the load on the storage and should ensure the stable operation of the system with an increase in the amount of data.

3.4. Online data processing

A typical data-processing cycle in astroparticle physics includes the direct data acquisition, followed by preliminary data processing that includes calibration and combining coincident events measured by different detectors of the experiment. Next is the event reconstruction, in case of KASCADE, carried out at three levels (for further details see e.g. [21]). Reconstruction depends heavily on the simulation of air showers and detector responses, which itself is computationally expensive. Next, the researcher analyzes the data at the desired level of

reconstruction. The results of the analysis are published and can be used for further research, while the analysis can be later repeated with an extended data set. Thus, simulations and data analysis are the most resource-intensive data processing steps, where distributed computations are required to perform operations on data online.

3.4.1. Simulations Monte Carlo simulation is the main method to study properties of systems with many degrees of freedom when the analytical solution is either too complicated or computationally expensive, which is the case for estimating the behavior of advanced physical setups. In the case of KASCADE, simulations are mostly used to estimate the detector efficiency depending on various properties of the incoming primaries and to define the constants for the correction functions.

One of the main advantages of the method is that Monte Carlo jobs are loosely coupled and produce independent outcomes. Thus, parallel computations can be made straightforwardly with minimal efforts, decreasing the wall time almost proportionally to the number of processes employed.

3.4.2. Data analysis During the data analysis it is typical to perform both massive elementary operations (for example, applying cuts with linear conditions) and computationally complex ones (for example, constructing specific data projections, applying cuts with correlated conditions).

While elementary data cuts are performed in the framework of data retrieval pipeline on the KCDC site, heavy resource-consuming operations for the main analysis are still performed on the user side. Our plans include extending the application server functionality in the way such operations could be performed on the server side employing distributed computing.

4. Distributed computations

Currently, there are three main types of distributed computing resources that a scientist can employ in his work: opportunistic resources, computing clouds, and computing clusters. Here we consider each of them in terms of applicability in our project. The key parameters that we keep in mind are the ability to perform calculations online by user request and the possibility of easy access for outreach purposes.

4.1. Employing opportunistic resources

In our case, employing opportunistic resources means that users on the local network provide the computing resources of their PCs for third-party calculations while their own computers are idle. However, the owners keep the priority to use their computers, so if the main user needs the local computing resources, the task should be transferred to another free computer on the network if possible. A potential issue one might face opting for such an approach is that availability of free opportunistic resources can develop a shortage. On the other hand, when working with the internal network, it is relatively easy to provide access to resources for users registered on the outreach portal and not having special user certificates.

4.2. Scaling to clouds

In case of a lack of local computing resources on the network, calculations can be scaled to the cloud. Thus, the desired computational resource can always be found to keep performing computations online. However, the lease of cloud resources is relatively expensive. The problem of funding for calculations performed by third parties, including those made by students for educational purposes, requires a separate consideration.

4.3. Scientific clusters

Most of third-party scientific computing clusters provide limited computing resources to specific certified users (for example, GridKa [22] or BW HPC [23] cluster). Accordingly, users of the outreach sector may experience problems with obtaining a license. Also, the operating mode of scientific clusters implies maintaining a high load for all nodes of the system, which means that user requests are queued and executed according to a certain system of priorities. That makes online data processing at external scientific clusters inefficient for the GRADLCI project.

5. Outlook

In modern astroparticle physics joint analysis of data obtained from different sources is of high interest. Even further, open access to scientific data makes knowledge transfer much faster and helps to diminish the gap between fundamental science and the broad audience. To fulfill the concepts above, the GRADLCI project was established to develop a data life cycle in astroparticle physics and the supporting infrastructure. The challenges to solve in a framework of the project are: arranging reliable, fast access to the data retrieval and analysis, data mapping plugins development and keeping the system architecture scalable for growing data volumes as well as some others beyond the scope of this article.

Acknowledgments

The work is supported by KRAD, the Karlsruhe-Russian Astroparticle Data Life Cycle Initiative (Helmholtz HRSF-0027).

The authors acknowledge the help of and cooperation with the colleagues of the APPDS initiative (especially A. Kruykov, A. Mikhailov, M.D. Nguyen, A. Shigarov) and the SCC GridKa infrastructure at KIT (especially A. Heiss, A. Streit, G. Quast, M. Schnepf, C. Heidecker).

References

- [1] Bychkov I *et al.* 2018 *Data J.* **3** 56
- [2] Antoni T *et al.* 2003 *Nucl. Instrum. Methods Phys. Res. Sect. A* **513** 490–510
- [3] Apel W D *et al.* 2010 *Nucl. Instrum. Methods Phys. Res. Sect. A* **620** 202–216
- [4] Budnev N *et al.* 2017 *Nucl. Instrum. Methods Phys. Res. Sect. A* **845** 330–333
- [5] Haungs A *et al.* 2018 *Eur. Phys. J. C* **78** 741
- [6] Haungs A *et al.* 2009 *Nucl. Instrum. Methods Phys. Res. Sect. A* **604** S1–S8
- [7] Zhurov D *et al.* 2019 *J. Phys.: Conf. Ser.* **1181** 012045
- [8] Gress O *et al.* 2013 *Nucl. Instrum. Methods Phys. Res. Sect. A* **732** 290–294
- [9] Berezhnev S *et al.* 2012 *Nucl. Instrum. Methods Phys. Res. Sect. A* **692** 98–105
- [10] Budnev N *et al.* 2017 *J. of Instr.* **12(06)** C06019–C06019
- [11] Bezyazeev P A *et al.* 2015 *Nucl. Instrum. Methods Phys. Res. Sect. A* **802** 89–96
- [12] Apel W D *et al.* (Tunka-Rex and LOPES Collaborations) 2016 *Phys. Lett. B* **763** 179–185
- [13] KASCADE Cosmic-ray Data Center (KCDC) <https://kcdc.i kp.kit.edu>
- [14] Berlin declaration on open access to knowledge in the sciences and humanities <https://openaccess.mpg.de/Berlin-Declaration> published: January 2015
- [15] Kang D *et al.* 2017 *35th International Cosmic Ray Conference (ICRC2017)* Proceedings of Science p 452
- [16] Django web framework documentation <https://docs.djangoproject.com/en/2.2/>
- [17] NGINX documentation <https://docs.nginx.com/>
- [18] RabbitMQ message-broker <https://www.rabbitmq.com/>
- [19] MongoDB cross-platform document-oriented database <https://www.mongodb.com/>
- [20] Celery distributed task queue <http://www.celeryproject.org/>
- [21] Wochele J, Kang D, Wochele D, Haungs A and Schoo S 2018 KCDC user manual Tech. rep. Karlsruhe Institute of Technology (KIT) URL http://kcdc.i kp.kit.edu/static/pdf/kcdc_mainpage/kcdc-Manual.pdf
- [22] Grid Computing Centre Karlsruhe (GridKa) <http://www.gridka.de/cgi-bin/frame.pl?seite=/welcome.html>
- [23] Project bwHPC-C5 <https://www.bwhpc.de/bwhpc-c5.html>