RUSSIAN DATA LAKE PROTOTYPE AS AN APPROACH TOWARDS NATIONAL FEDERATED STORAGE FOR MEGASCIENCE

A. Alekseev[5,6,7], A. Kiryanov[1,2,7,a], A. Klimentov[4], T. Korchuganova[5,6,7], D. Oleynik[7,8], A. Zarochentsev[3,7]

[1] *NRC "Kurchatov Institute", 1 Akademika Kurchatova sq., Moscow, 123182, Russia*

[2] *Petersburg Nuclear Physics Institute of NRC "Kurchatov Institute", 1 Orlova Rocha, Gatchina, 188300, Russia*

[3] *Saint Petersburg State University, 7-9 Universitetskaya emb., Saint Petersburg, 199034, Russia*

[4] *Brookhaven National Laboratory, Upton, NY, USA*

[5] *Ivannikov Institute for System Programming RAS, 25 Alexander Solzhenitsyn st., Moscow, 109004, Russia*

[6] *University Andres Bello, Santiago, Chili*

[7] *Plekhanov Russian University of Economics, 36 Stremyanny lane, Moscow, 117997, Russia*

[8] *Joint Institute for Nuclear Research, 6 Joliot-Curie st., Dubna, 141980, Russia*

E-mail: [a] globus@pnpi.nw.ru

A substantial data volume growth will appear with the start of the HL-LHC era. It is not well covered by the current LHC computing model, even taking into account the hardware evolution. The WLCG DOMA project was established to provide data management and storage researches. National data lake r&d's, as a part of the DOMA project, should address the study of possible technology solutions for the organization of intelligent distributed federated storage. This talk will present the current status of the Russian Scientific Data lake prototype and the methodology, which is used for the validation and functional testing of deployed infrastructure.

Keywords: HL-LHC, WLCG, Data Lake, Distributed Storage, Megascience

Aleksandr Alekseev, Andrey Kiryanov, Aleksey Klimentov,
Tatyana Korchuganova, Danila Oleynik, Anderey Zarochentsev

1. Introduction

High Luminosity LHC (HL-LHC) will be a multi-Exabyte challenge where the envisaged Storage and Compute needs are a factor 10 above the expected technology evolution. WLCG community needs to evolve current computing and data organization models in order to introduce changes in the way it uses and manages the infrastructure, focused on optimizations to bring performance and efficiency, not forgetting simplification of operations.

Technologies that will address the HL-LHC computing challenges may also be applicable to other communities, such as SKA, DUNE, CTA, LSST, BELLE-II, JUNO, etc. and allow them to manage large-scale data volumes. One of such technologies that we will discuss in this paper is Data Lake [1].

Generally speaking, Data Lake is a set of sites, associated by proximity, providing together storage services, possibly accompanied by compute ones, to an identified set of user communities, capable to carry out independently well-defined tasks. Proximity could be defined by geography, connectivity, funding or a shared user community. This requires that their combined storage capacity and network bandwidth can meet the demands of the designated task and that the usage of different sites is transparent to the users. This implies some form of trust relationship between the sites and a way to orchestrate their shared resources.

In this work we will focus on a specific scenario of a national Data Lake with a common namespace, providing storage for a set of relatively small CPU-oriented sites alongside a large national research center with a reliable storage system [fig.1]. Our goal is to optimize the computational effectiveness by using techniques like read caching and write buffering, minimizing the CPU idle time during heavy data I/O operations.

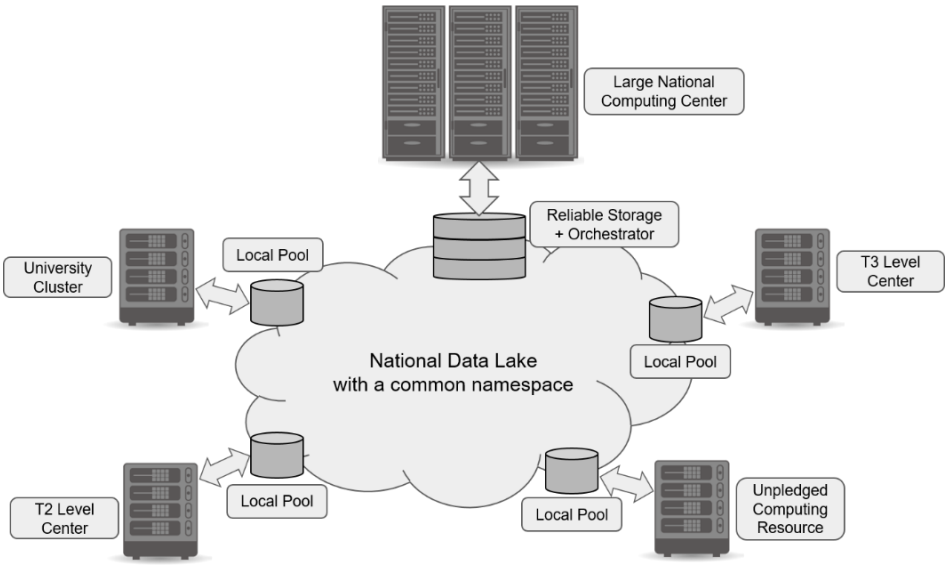

Figure 1. Data Lake with CPU-oriented smaller sites

2. Prototype Implementation

A national Data Lake prototype was deployed on a number of Russian sites connected by a modern network infrastructure also used for processing of LHC data (part of the LHC ONE network) [fig. 2]. In our prototype JINR acts as a Data Lake entry point providing data storage and orchestration. Three other sites (PNPI, MEPhI and PRUE) act as CPU-oriented resources with relatively small volatile local storages used as caches.

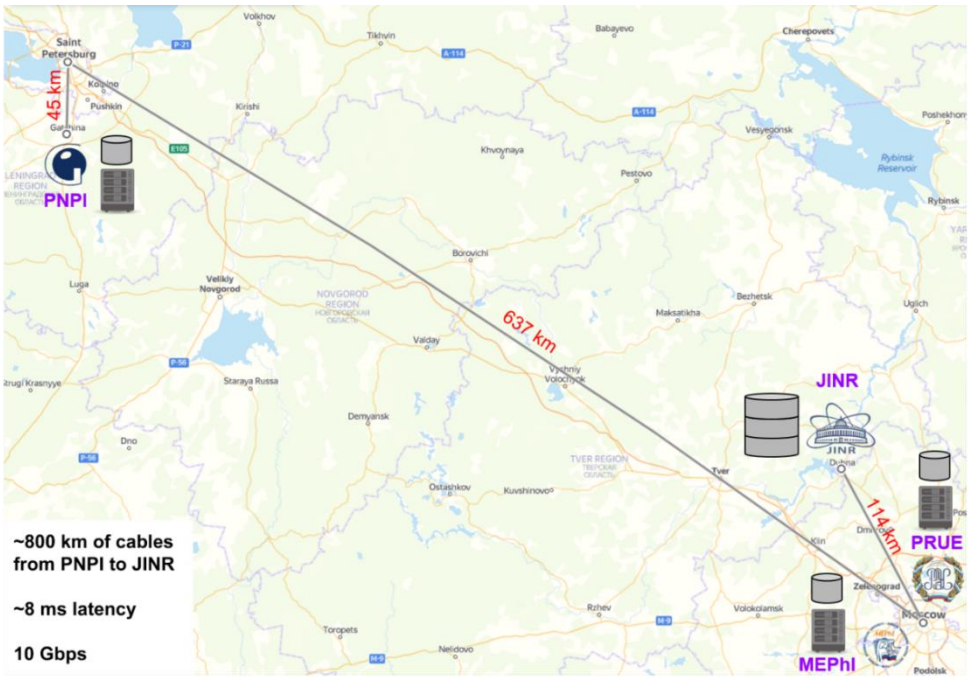

Figure 2. Russian DataLake prototype map

The prototype is being built using the following building blocks:

- Resources
  - Bare-metal at JINR and MEPhI
  - Virtualized at PNPI and PRUE
- Storage systems
  - EOS [2]
  - dCache [3]
  - XCache [4]
- Payloads configuration, submission and testing
  - Custom synthetic tests
  - PanDA and ProdSys2 (ATLAS) [5, 6]
  - HammerCloud [7]
  - CRIC (former AGIS) [8]
- Monitoring infrastructure
  - perfSONAR
  - Logstash
  - ElasticSearch
  - Kibana
  - Custom web app extending Kibana interface

3. Tests

In order to conduct efficiency tests authors have used both synthetic and real-life tests. Synthetic tests were basically generating files of a given size with random contents and measuring read/write performance of these files in various conditions. Real life tests involved application from ATLAS software stack reading and processing event files. Different set of tests was used for read- and write-intensive scenarios.

3.1. Caching

The first synthetic tests were mostly designed as functional tests with an additional goal of measuring maximum "theoretical" efficiency in ideal conditions. The tests were reading a predefined set of files of a certain size with a certain number of repetitions. Three scenarios were examined:

1) No cache (reference value)
2) Dedicated cache on a single server

3) Distributed cache on the worker nodes

Caching scenarios were primarily targeted at workloads that reprocess data, i.e. read the same input data file more than once. Such payloads are not uncommon as can be seen in the ATLAS data popularity studies [9].

The next step was to test the real-life experiment payloads. This was achieved using the HammerCloud stress-testing system submitting ATLAS analysis tasks.

3.2. Buffering

Write buffering is targeted at workloads that mostly write data. In order to optimize the CPU efficiency on the worker nodes, generated data is written into the local buffer and then migrated into the Data Lake in the background automatically by the storage system. Currently only synthetic tests are developed for this scenario. HammerCloud requires some modifications in order to be used for this kind of workloads. A fundamental agreement has been reached with the developers of this system, but the implementation is still a work in progress.

Unlike in caching tests, in this case it was decided not to measure the I/O speed directly, but to compare the CPU usage efficiency on the worker nodes. In order to do this one needs a real-life CPU-intensive workload that will generate data and on top of that the CPU time it consumes must be significantly higher than the time that is necessary for saving the generated data on the local storage.

As with the caching tests, buffering must be examined using multiple scenarios:

1) No buffer (reference value)
2) Dedicated buffer on a single server
3) Distributed buffer on the worker nodes

Due to technical difficulties the third scenario has not been implemented yet.

4. Monitoring

For the monitoring of the Russian Data Lake prototype, a unified monitoring system was created using the ELK stack [fig. 3]. This stack includes ElasticSearch (non-relational data storage), Logstash (collector and data processor) and Kibana (data visualization system). In total, four dashboards were implemented in the Kibana system:

1. Xrootd dashboard based on the processed data from the event logs for monitoring the Xrootd component
2. Billing monitoring dashboard to get information about dCache
3. Jobs monitoring provides information about all test jobs which run on the Russian Data Lake prototype computing resources
4. Accounting dashboard containing the information about the consumed CPU resources

Additionally, a web application was developed for monitoring and analytics of the Russian Data Lake prototype testing. It uses data from the same job monitoring index in the unified ElasticSearch storage and mitigates a lack of advanced charts customizing features in Kibana visualization platform providing ready-for-publication quality comparison plots.

The perfSONAR system is used separately to collect and analyze information about data transmission on the global network. The prototype structure assumes that each site has a dedicated perfSONAR node for network measurements.

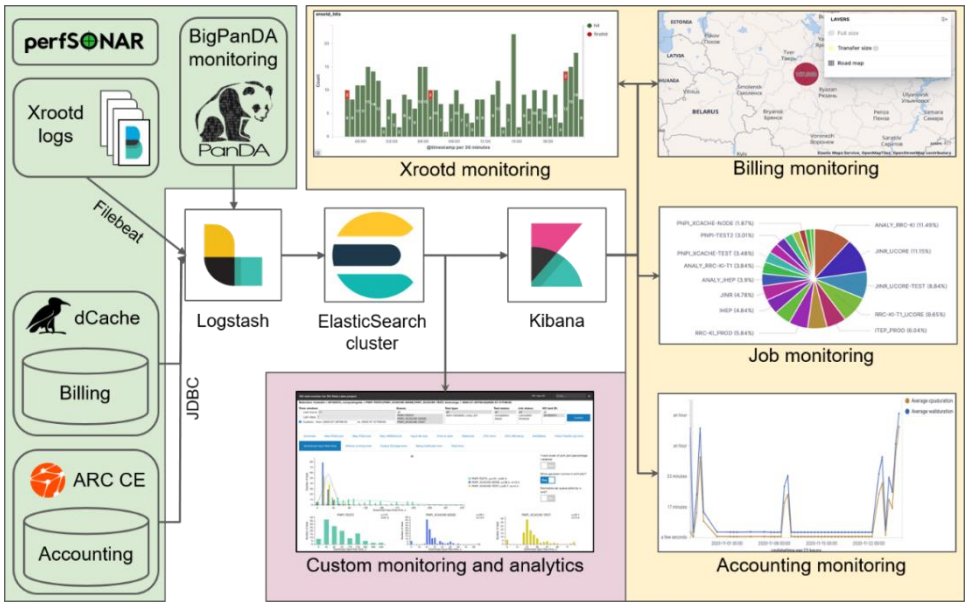

Figure 3. Russian DataLake prototype monitoring

5. Acknowledgements

This work was funded in part by the Russian Science Foundation under contract No.19-71-30008 (research is conducted in Plekhanov Russian University of Economics)

References

[1] Kiryanov, A. Klimentov, A. Zarochentsev. Russian scientific data lake // Open Systems Journal, issue 4, 2018. Available at: https://www.osp.ru/os/2018/04/13054563/

[2] Luca Mascetti, Massimo Lamanna, Daniel Van Der Ster, at al., CERN Disk Storage Services: Report from last data taking, evolution and future outlook towards Exabyte-scale storage // EPJ Web Conf., 245 (2020) 04038 - DOI: https://doi.org/10.1051/epjconf/202024504038

[3] Tigran Mkrtchyan, Patrick Fuhrmann, at al., dCache - storage for advanced scientific use cases and beyond // EPJ Web Conf., 214 (2019) 04042 - DOI: https://doi.org/10.1051/epjconf/201921404042

[4] Andrew Hanushevsky, Mario Lassnig, Vincent Garonne, J Ilija Vukotic, at al. Xcache in the ATLAS Distributed Computing Environment // EPJ Web Conf., 214 (2019) 04008 - DOI: https://doi.org/10.1051/epjconf/201921404008

[5] The Production and Distributed Analysis (PanDA) system. Available at: https://panda-wms.readthedocs.io/en/latest/index.html

[6] Borodin M. et al., Scaling up ATLAS production system for the LHC Run 2 and beyond: project ProdSys2 // Journal of Physics: Conference Series. - 2015. - Vol. 664. - DOI: 10.1088/1742-6596/664/6/062005. - URL:https://iopscience.iop.org/article/10.1088/1742-6596/664/6/062005

[7] HammerCloud Distributed Analysis testing system. Available at: http://hammercloud.cern.ch/hc/

[8] Alexey Anisenkov, Julia Andreeva, Alessandro Di Girolamo, at al., CRIC: Computing Resource Information Catalogue as a unified topology system for a large scale, heterogeneous and dynamic computing infrastructure // EPJ Web Conf. 245 03032 (2020) - DOI: 10.1051/epjconf/202024503032

[9] Thomas Beermann, Alessandro Di Girolamo, Alexei Klimentov, Mario Lassnig, Markus Schulz, Andrea Sciaba, at al. Methods of Data Popularity Evaluation in the ATLAS Experiment at the LHC // EPJ Web Conf., 251 (2021) 02013 -DOI: https://doi.org/10.1051/epjconf/202125102013