PREDICTIVE ANALYTICS AS AN ESSENTIAL MECHANISM FOR SITUATIONAL AWARENESS AT THE ATLAS PRODUCTION SYSTEM

M.A. Titov [1,a], M.Y. Gubin [2], A.A. Klimentov [1,3], F.H. Barreiro Megino [4], M.S. Borodin [1,5], D.V. Golubkov [1,6] on behalf of the ATLAS Collaboration

[1] *National Research Centre "Kurchatov Institute", 1 Akademika Kurchatova pl., Moscow, 123182, Russia*

[2] *National Research Tomsk Polytechnic University, 30 Lenin Avenue, Tomsk, 634050, Russia*

[3] *Brookhaven National Laboratory, P.O. Box 5000, Upton, NY, 11973, USA*

[4] *University of Texas at Arlington, 701 South Nedderman Drive, Arlington, TX, 76019, USA*

[5] *University of Iowa, 108 Calvin Hall, Iowa City, IA, 52242, USA*

[6] *Institute for High Energy Physics, 1 Ulitsa Pobedy, Protvino, Moscow region, 142280, Russia*

E-mail: [a] mikhail.titov@cern.ch

The workflow management process should be under control of a specific service that is able to forecast the processing time dynamically according to the status of the processing environment and workflow itself, and to react immediately on any abnormal behavior of the execution process. Such situational awareness analytic service would provide the possibility to monitor the execution process, to detect the source of any malfunction, and to optimize the management process. The stated service for the second generation of the ATLAS Production System (ProdSys2, an automated scheduling system) is based on predictive analytics approach. Its primary goal is to estimate the duration of the data processings (in terms of ProdSys2, it is task and chain of tasks) with possibility for later usage in decision making processes. Machine learning ensemble methods are chosen to estimate completion time (i.e., "Time-To-Complete", TTC) for every (production) task and chain of tasks, and "abnormal" task processing times would warn about possible failure state of the system. This is the primary phase of the service development that also includes the strategy for its precision enhancement. The first implementation of such analytic service is designed around Task TTC Estimator tool and it provides a comprehensive set of options to adjust the analysis process and possibility to extend its functionality.

Keywords: situation awareness, predictive analytics, production system, Apache Spark

1. Introduction

The key point of any control system is its ability to process a large amount of data and to extract the most valuable set of parameters and corresponding connections between them that would describe the behavior of the controlled system and forecast its future state with a certain level of confidence. The complexity of such control systems encapsulates a comprehensive model it is based on. A theoretical model of situation awareness contains workflow description and corresponding procedures regarding all above requirements. This model is viewed as "a state of knowledge" [1, 2].

The formal definition of situation awareness is presented as "the perception of the elements in the environment within a volume of time and space, the comprehension of their meaning and the projection of their status in the near future" [2]. It describes three essential levels / phases in achieving the situation awareness. The first phase (Level 1) is about perceiving the crucial factors of the environment (selection process of the most significant parameters to describe the controlled system's behavior). The second phase (Level 2) is aimed at understanding the meaning of the collected data and their possible impact on the whole controlled system. And the third phase (Level 3) gives an estimation of the (near) future state of the controlled system. Each of the phases is based on the outcome of the previous one, and it is a cyclic process.

The original scope of this model of situation awareness was the dynamic human decision making (in a variety of domains), and its structure is presented at the Figure 1 [2]. But also, this model represents a basic strategy to control the behavior of the system and might be used as a strategy (primary actions) for anomaly detection.

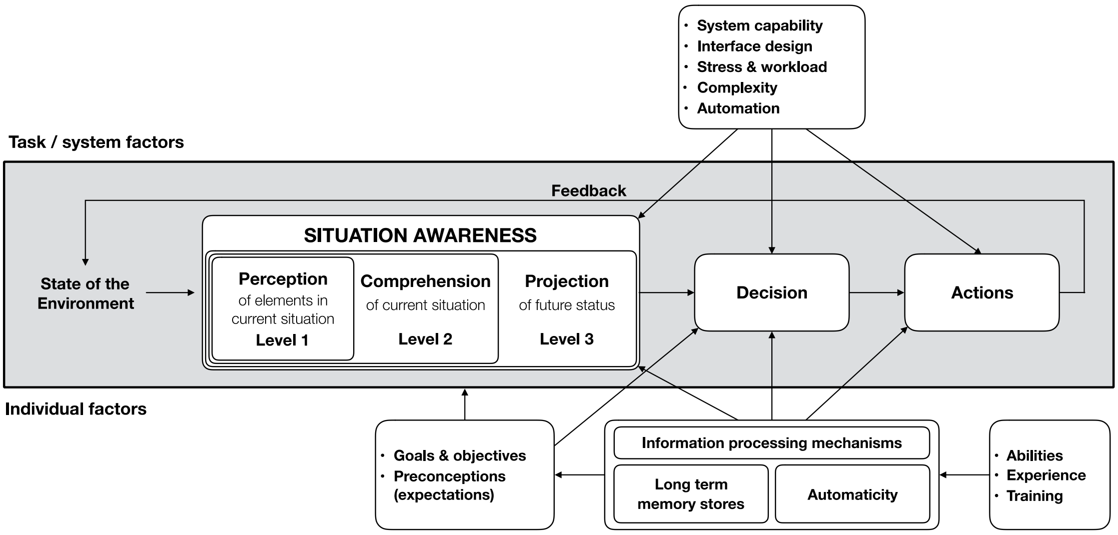

Figure 1. Model of situation awareness in dynamic decision making

Specifically, this schematic model is considered as a workflow for the control service of the ATLAS Production System (2nd generation, i.e., ProdSys or ProdSys2) [3]. ProdSys is a part of the distributed production/analysis and data management workflow in the ATLAS Experiment [4] (see Figure 2). ProdSys is a distributed task management and automated scheduling system with the current processing rate of about 2M tasks per year. Its major responsibilities include the following workflows: the central production of Monte-Carlo data, highly specialized production for physics groups, as well as data pre-processing and analysis using such facilities as grid infrastructures, clouds and supercomputers.

The system contains such core components / layers as Ref. [5]: i) web UI to manage tasks (that consists of jobs that all run the same program) and production requests (i.e., group of tasks); ii) DEfT (Database Engine for Tasks) to formulate tasks, chains and groups of tasks; iii) JEDI (Job

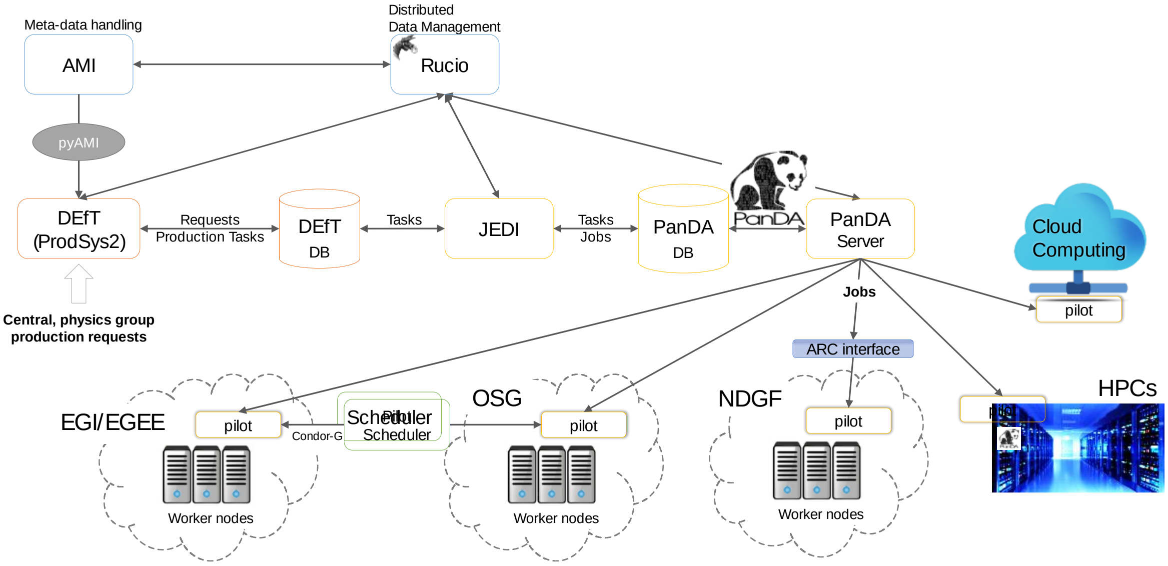

Figure 2. The ATLAS ProdSys workflow management

Execution and Definition Interface, which is an intelligent component in the PanDA system [6]) to manage task-level workload with such key feature as dynamic job definition and execution for resources usage optimization (see Figure 3). The 4th layer is represented by PanDA system (workload management system) that is responsible for job execution process.

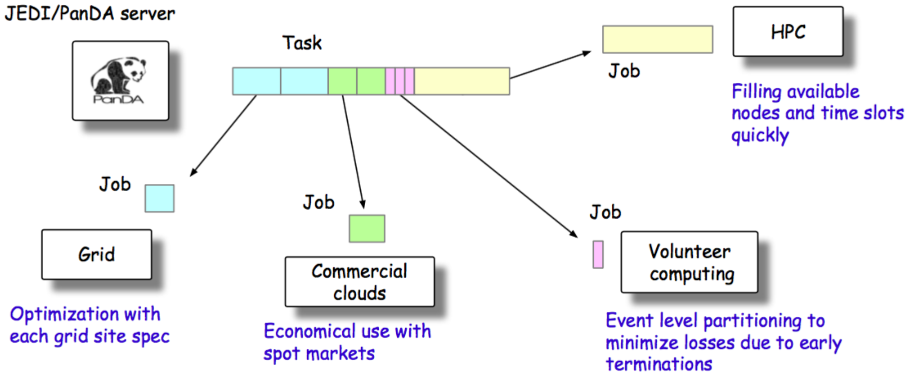
JEDI/PanDA server
PanDA
Task
Job
HPC
Filling available nodes and time slots quickly
Job
Grid
Optimization with each grid site spec
Job
Commercial clouds
Economical use with spot markets
Job
Volunteer computing
Event level partitioning to minimize losses due to early terminations

Figure 3. Workload partitioning for traditional and opportunistic resources

2. Problem statement

The possibility of controlling / tracking the state of a particular system requires being aware of its crucial parameters and processes, and their regular behavior. Such parameters and processes should be categorized following the potential failures of the controlled system. List of potential failures: i) overload in the system (that can cause for stuck handling processes); ii) malfunctioning of computing resources (that can cause failure of data processing); iii) components misconfiguration (that can cause improper operational processes); iv) general malicious activities (that can cause incorrect data results).

The following parameters, which are actual ProdSys parameters or derived ones, reflect the specific state of the system and indicates a strong correlation with its failures of a particular category presented above. Controlled parameters are:

- Duration of task (chain of tasks) execution (forecast and control Time-To-Complete);
- Rate of task submission over processing during a limited period of time;

- Cumulative number of failures or/and errors occurred;
- Metrics of resources utilization (which are compared with optimal values).

These parameters are considered as the initial set that has to be maintained, and should be extended later to improve the operational process.

The goal of the designed control system (service) is being able to track the defined set of parameters of the controlled system (ProdSys), to distinguish their non-valid values according to the running state of the system, to forecast their next values considering the state of the computing environment and system itself, and base on corresponding metrics and controlled parameters to detect the source of malfunctioning. The current project is focused on description of the task TTC estimation as a part of the control service infrastructure (i.e., as a part of a proof-of-concept).

3. Technology and methods

Predictive analytics

Predictive analytics is a form of conjunction of advanced analytics and decision optimization that uses both new and historical data to forecast future activity, behaviour and trends. Advanced analytics includes in itself: statistical analysis, machine learning, modeling, data visualization, and reporting; while decision optimization is represented by: scoring engine, rules engine, and recommendation engine. Thus, multiple variables are combined into a predictive model that is capable to assess future probabilities with a certain level of confidence. These predictive models (modeling) are used to look for correlations between different data elements (features). Predictive models are presented by two types: i) classification (predict the class membership for the considered object); ii) regression (predict a number for the specific feature of the object). Three of the most widely used predictive modeling techniques are decision trees, regression and neural networks.

These techniques (or a combination of them) is a foundation for the Level 3 of the model of situation awareness. Predictive analytics should be applied to the parameters which are under the control, thus, the forecasting of their next value (state) will trigger the control system to follow the predefined actions.

Processing framework and machine learning methods

Apache Spark [7] parallel processing framework and Spark.MLlib [8] machine learning library are used as a prediction tools for the control system / service. Apache Spark is a framework providing speedy and parallel processing of distributed data in real time (sophisticated analytics, real-time streaming and machine learning on large datasets). It provides powerful cache and persistence capabilities. Because of these features Apache Spark is applied to defined data analytics problems. There are two regression ensemble methods (ensemble of decision trees) that were considered as the initial implementation for the predictive analytics: i) Gradient-Boosted Trees (GBT) regression method; ii) Random Forests (RF) regression method. Both methods were evaluated with the current service implementation, and RF outreached GBT in key characteristics such as less prone to overfitting, marginally better performance of parallel implementations, robust when the training set contains outliers.

4. Conducted research

The initial focus in controlling crucial parameters was set on estimation of the duration of task (chain of tasks) execution. The ability of being aware of the task execution process assists in forecasting the near-future state of the controlled system (as a basic representation of the system's state). Three core stages for estimation of the task execution duration were highlighted, where each of the stage is applied at a certain time of the task lifecycle with available set of parameters (and applied corresponding techniques) and with a certain level of estimation accuracy: i) thresholds definition (based on statistical analysis); ii) initial / static predictions (based on descriptive data of task); iii) dynamic predictions (based on static and dynamic data of the running task).

The general workflow of the designed control system with its main components is presented at the Figure 4. It shows that the management process is concentrated in the "Control node", while data processing is executed in the "Analytics cluster". (Blocks "Collector" and "Model Handler" are responsible for the task TTC estimation.)

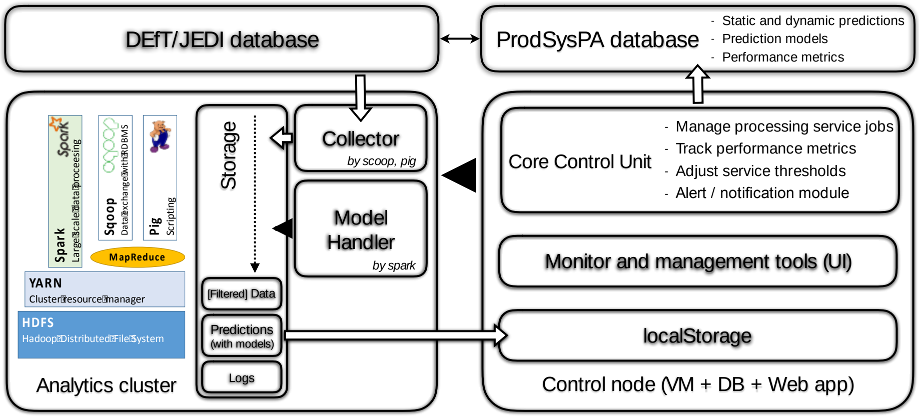
DEfT/JEDI database
ProdSysPA database
- Static and dynamic predictions
- Prediction models
- Performance metrics
Spark
Large scale data processing
Sqoop
Data exchange with RDBMS
Pig
Scripting
MapReduce
YARN
Cluster resource manager
HDFS
Hadoop Distributed File System
Analytics cluster
Storage
[Filtered] Data
Predictions (with models)
Logs
Collector
by scoop, pig
Model Handler
by spark
Core Control Unit
- Manage processing service jobs
- Track performance metrics
- Adjust service thresholds
- Alert / notification module
Monitor and management tools (UI)
localStorage
Control node (VM + DB + Web app)

Figure 4. Controlled system structure overview (analytics cluster with highlighted services as provided by CERN-IT)

Task Time-To-Complete by threshold definition

The set of thresholds of the task execution duration is calculated based on the statistical analysis of the historical data for the last several months (6 months was used for the conducted research, see Figure 5). This stage is considered as a rough estimation of the duration of task of a particular type for the possible and valid values. The type of the task is a combination of its features such as <projectType> (e.g., "mc" – Monte-Carlo simulated data, "data" – real data collected from

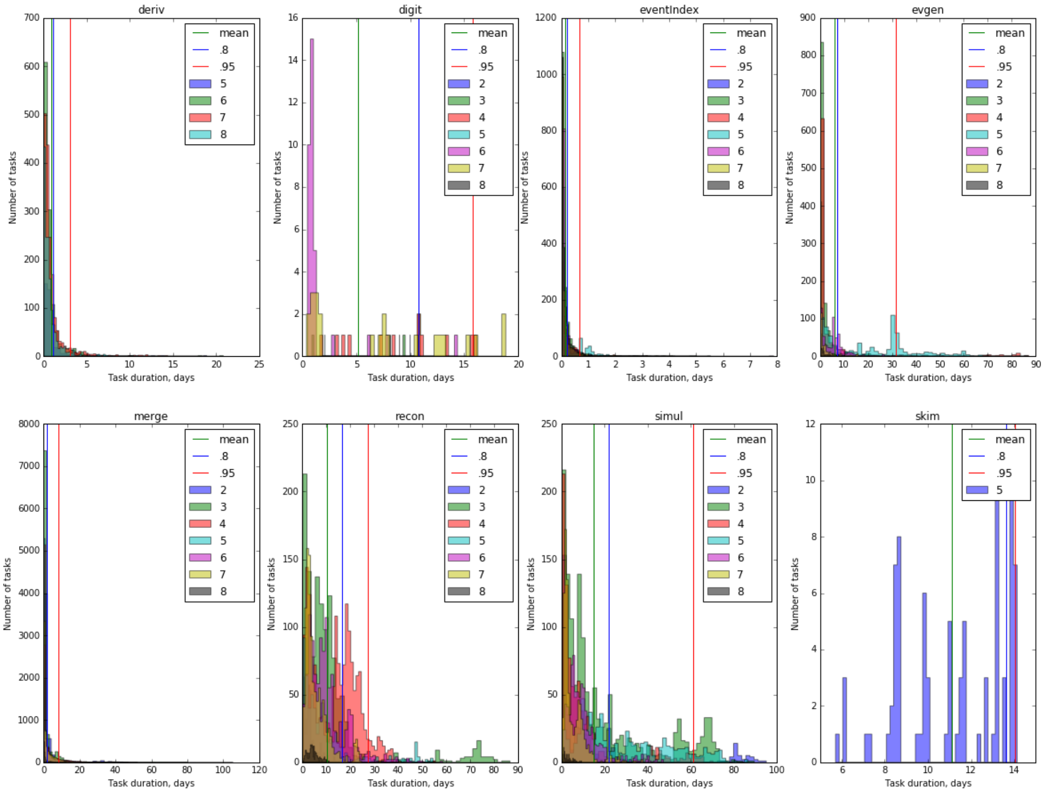

Figure 5. MC tasks duration distribution per month (for the last 180 days, collected on 2017-08-16)

the physical experiment) and <productionStep> (e.g., evgen, simul, recon, merge, etc.).

At Figure 5, there are plots with corresponding thresholds and columns per productionStep that are represented by: i) red vertical lines, which correspond to the values of maximum task execution duration for 95% of tasks that are finished during the defined period of time; ii) blue vertical lines are the same as red ones, but for 80% of finished tasks; iii) green vertical lines represent mean values of the task execution duration among all tasks of the defined type; iv) columns of a particular color represent the number of tasks that were started in a particular month (i.e., number in the legend corresponds to the number of the month) finished within a certain period of time (i.e., task duration). Thereby, the threshold for execution duration of 95% of tasks of a particular type was chosen as a rough estimation (other 5% of tasks were pruned as exceptions, which evaluation should be done separately), and which defines the range of valid values (i.e., "no warning" values).

Task Time-To-Complete by predictive modeling

The process of predictive modeling assumes to solve the problem (forecasting the value of a particular parameter) by creating a model from the sample data (training data based on historical data), thus take data with known relationships, and using learned relationships to make predictions on new data (to discover a certain outcome, e.g., numerical value of task execution duration).

There are several categories of prediction approaches (where sub-categories are named according to temperature states to represent the time point of the task lifecycle when it is applied):

- Initial / static predictions
  - "Cold"-prediction is based on task definition parameters that categorize the average execution process for the defined task type (with particular conditions). It gives the estimation of the task duration during its definition.
- Dynamic predictions
  - "Warm"-prediction is based on description and state of scout jobs that are used to check the processing environment. It gives the prediction for the task duration immediately after task is launched.
  - "Hot"-prediction is based on the current state of the task processing (states of environment and corresponding jobs). It gives the prediction adjustment during the task execution (this prediction is calculated multiple times during the task execution process).

The example of the static prediction based on Random Forest regression method is shown at the Figure 6. The following data was used for the prediction process: i) predictive models are based on "training" data ("mc-all" + "mc-recon" types) for the defined 3 months; ii) "test" data ("mc-recon" type) for the next month (after training data is collected) was used to generate predicted task

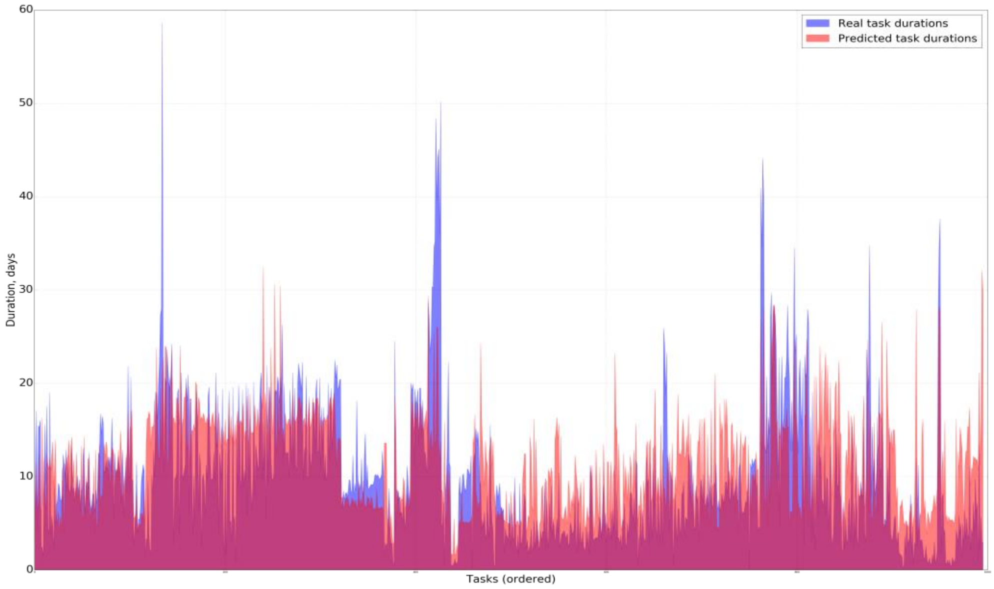

Figure 6. Real and predicted task durations (projectType=mc; productionStep=recon)

durations and compare it with real values. This result shows that predictions for tasks (execution duration of 95% of tasks of the defined type is under 27d) are actually following the real values of task durations (the accuracy for the defined example is presented with the following absolute error values: mean=1.57d, std=6.67d, RMSE=6.85d). Since the prediction approach of this category does not consider dynamic parameters and the state of the environment, the accuracy of predictions is low and can be considered as a rough estimation, same as for thresholds (further research will explore generated predictions based on different set of parameters for the training data). Predictions of "dynamic" category will increase the final accuracy (this is under the development).

5. Acknowledgements

We thank ProdSys/PanDA team for providing the data for the conducted analysis and for their continued support. This work has been carried out using computing resources of the federal collective usage center Complex for Simulation and Data Processing for Mega-science Facilities at NRC "Kurchatov Institute", http://ckp.nrcki.ru/. Also, this work was funded in part by the Russian Ministry of Science and Education under contract No. 14.Z50.31.0024.

6. Conclusion

Techniques and methods of predictive analytics would benefit the control and monitoring processes for the controlled system. The situational awareness analytic service (based on predictive analytics techniques) would provide the possibility to detect the source of any malfunction, and to optimize the whole management process.

The obtained results facilitate filtering the regular processes of tasks execution (with certain level of confidence) and form the basic layer for further sophisticated processes of highlighting abnormal operations and executions. The further improvement of the prediction process will increase the accuracy of the selection process of potentially failure task (chain of tasks), and the extension of this approach to other system's computing blocks and data will keep the system aware of potential malfunction.

References

[1] Endsley M.R. Design and evaluation for situation awareness enhancement. // Proceedings of the Human Factors and Ergonomics Society Annual Meeting, 1988, vol. 32, no. 2, pp. 97-101. DOI:10.1177/154193128803200221

[2] Endsley M.R. Toward a Theory of Situation Awareness in Dynamic Systems. // Human Factors Journal, 1995, vol. 37 no. 1, pp. 32-64. DOI:10.1518/001872095779049543

[3] 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, no. 6, 062005. DOI:10.1088/1742-6596/664/6/062005

[4] ATLAS Collaboration. The ATLAS Experiment at the CERN Large Hadron Collider // Journal of Instrumentation, 2008, vol. 3, no. 8, S08003. DOI:10.1088/1748-0221/3/08/S08003

[5] De K. et al. Task Management in the New ATLAS Production System. // Journal of Physics: Conference Series, 2014, vol. 513, no. 3, 032078. DOI:10.1088/1742-6596/513/3/032078

[6] Barreiro Megino F.H. et al. PanDA: Exascale Federation of Resources for the ATLAS Experiment at the LHC. // EPJ Web of Conferences, 2016, vol. 108, 01001. DOI:10.1051/epjconf/201610801001

[7] Apache Spark: https://spark.apache.org

[8] Meng X. et al. MLlib: Machine Learning in Apache Spark. // Journal of Machine Learning Research, 2016, vol. 17, no. 1, pp. 1235-1241.