Письма в ЭЧАЯ. 2016. Т. 13, № 5(203). С. 1074–1083

SIMULATION CONCEPT OF NICA–MPD–SPD TIER0–TIER1 COMPUTING FACILITIES

V. V. Korenkov, A. V. Nechaevskiy, G. A. Ososkov[1]*, D. I. Pryahina, V. V. Trofimov, A. V. Uzhinskiy*

Joint Institute for Nuclear Research, Dubna

The simulation concept for grid–cloud services of contemporary HENP experiments of the Big Data scale was formulated in practicing the simulation system developed in LIT JINR, Dubna. This system is intended to improve the efficiency of the design and development of a wide class of grid–cloud structures by using the work quality indicators of some real system to design and predict its evolution. For these purposes the simulation program is combined with a real monitoring system of the grid–cloud service through a special database (DB). The DB accomplishes acquisition and analysis of monitoring data to carry out dynamical corrections of the simulation. Such an approach allows us to construct a general model pattern which should not depend on a specific simulated object, while the parameters describing this object can be used as input to run the pattern. The simulation of some processes of the NICA–MPD–SPD Tier0–Tier1 distributed computing is considered as an example of our approach applications.

PACS: 29.20.D-; 07.05.Tp

INTRODUCTION

Distributed complex computing systems for data storage and processing are in common use in the majority of scientific centers. However, some of these distributed computing systems are distinguished from conventional ones by their focus on a large-scale resource sharing, innovative applications, a multi-institutional collaboration, high level of dynamic computing resources heterogeneity, and high performance orientation. This computing technology developed for very complex system is named the Grid [1].

As one of quite impressive examples illustrating the new challenging era of scientific data management in the coming decade named "Big Data" is the Large Hadron Collider (LHC) [2] data handling. Hundreds of petabytes of the LHC data were aggregated in the CERN Data Centre [3] during the first LHC run (2010–2012) to be distributed between several large data centers around the world. Subsequently, hundreds of thousands of computers from around the world come into action: harnessed in a distributed computing service, they form the Worldwide LHC Computing Grid (WLCG) — a hierarchical distributed computing infrastructure arranged in tiers (see Fig. 1), which provides the resources to store, distribute, and process the LHC data [4].

[1]E-mail: ososkov@jinr.ru

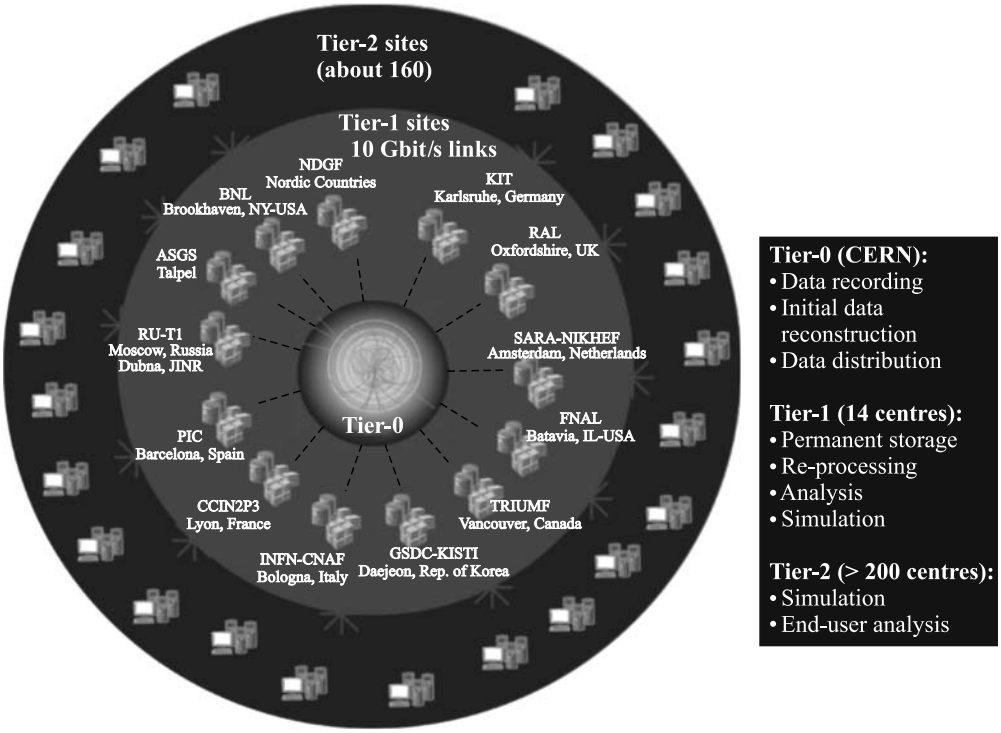

Fig. 1. Hierarchical tier structure of WLCG

It gives a community of over 8000 physicists near real-time access to LHC data, including the Joint Institute for Nuclear Research (JINR) in Dubna, Russia, where one of those WLCG Tier1 centers is serving the CMS — one of four LHC experiments [5].

During the LHC second run in 2015–2018, a considerable increase in the data volume in LHC experiments and corresponding transitions to the grid–cloud complexes are expected. It is necessary for new physics, but faces great challenges in distributed computing [6]: large increase of CPU and network resources; combined grid and cloud access; intelligent dynamic data placement; distributed parallel computing; renewal of most simulation and analysis software codes.

These problems are also inherent to such JINR projects as running Tier1 for CMS [7] and planning Tier0/1 for the new JINR NICA collider megaproject where two contemporary experiments MPD and SPD are now under design and development [8] (see Fig. 2).

To clarify the issue of combined grid and cloud access, one should note that the rigid grid structure was carried out to integrate already existing hardware and software resources fixed at the system, while the cloud structures of distributed computing are more flexible employing virtual clusters of virtual computers. Including cloud structures into the grid allows one to reduce the solution time of the wide range of experimental high-energy physics problems and to improve considerably the efficiency of resource usage.

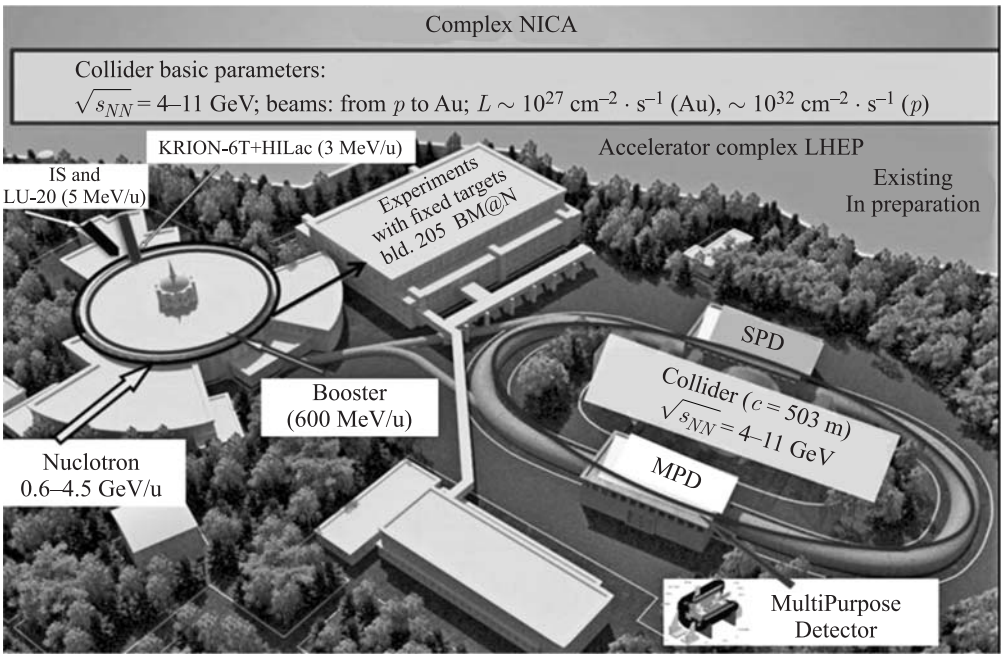

Fig. 2. Complex NICA of JINR, Dubna, Russia

The Production and Distributed Analysis (PanDA) [9] system is a good example of applying such "grid of clouds" technology. The ATLAS experiment uses PanDA for managing the workflow for all data processing jobs on the WLCG capable of operating at LHC data processing scale which can optimally make the distributed resources accessible to all users. PanDa workload management system (WMS) is based on pilot job execution system, automatic error handling and recovery, extensive monitoring, elaborated Database, flexible dataset sizing/containers for scalability, queues in clouds as additional resources.

After integration PanDA with dynamic network provisioning ATLAS is well advanced in integrating and bringing cloud computing in real analysis and production workflows. In particular, ATLAS has now quite a successful experience to work on Titan supercomputer [10].

Due to the notable success of PanDa WMS performance, US DoE recently selected "Next Generation Workload Management and Analysis System for Big Data" for ASCR (Advanced Scientific Computing Research) funding, what signifies beginning the BigPanda project [11].

PanDa is also considered as a possible base of WMS for the JINR NICA–MPD–SPD project.

Carrying out such unique JINR projects as Tier1 for CMS and Tier0/1 for the JINR NICA supposes great efforts for sophisticate grid–cloud systems intended to store, distribute, and process superbig volumes of experimental data.

Substantial optimality study is needed to avoid possible and quite expensive mistakes at design and development stages of any grid–cloud system. The study of such a system

optimality is based on the optimality criterion which minimizes the cost of its equipment set and maximizes its reliability under unconditional fulfillment of SLA (Service Level Agreement).

As to reliability of the contemporary giant grid–cloud services as WLCG, it is necessary to note that they use extended workload management systems like BigPanDa [11], which are able to cure emerging failures and other unplanned service downtimes automatically. It became possible due to embedded well-elaborated features based on automatic error handling and recovery, extensive monitoring and intelligent network services via cloud virtual networks on demand. An example of how it was carrying out for the ATLAS grid–cloud infrastructure is given in [10].

The optimality study is needed to keep an admissible balance between the total cost of any grid–cloud system and interrelated investments to its automatic fault tolerance, especially at its design stage. Such a study can be efficient when it is based on scrupulous simulations of computing resources (number of compute nodes, the architecture of a computer system, installed software, CPU consumption), job stream with knowledge of job types (simulation, analysis, reconstruction), and statistical information about distribution of their arrival and execution times.

SIMULATION OF GRID–CLOUD SYSTEMS

The efficient simulation of a grid–cloud system should be based on its functioning quality in order to evaluate its performance and to forecast its future, taking into account dynamics of its evolution. Besides, the simulations should clarify the main issues of the grid–cloud system especially important at its design stage:

1. Evaluate grid–cloud system performance and reserves under various changes: different workloads; system configuration; different scheduling heuristics; hardware malfunctions.

2. Balance the equipment needed for data transfers and storage by minimizing cost, malfunction risk and execution time.

3. Optimize resource distribution between user groups.

4. Predict and prevent a number of unexpected situations.

5. Test the system functioning to find bottlenecks.

There are known several approaches to the analytical simulation of grid and cloud systems which can be grouped into two types:

1. System is considered as a multichannel queuing system with state controlled by Markov process under restrictions on input stream distributions and priority discipline.

2. System is considered as a dynamic stochastic network, described by the system of equations that allows one to consider both routing and resource allocation in the network. Equilibrium and nonequilibrium network states are studied (see [12], for example).

Both approaches give a simulation result in the form of asymptotic distributions and in view of limited theoretical assumptions cannot be applied to simulate complex multitier architecture of computer networks with real distributions of the input streams of tasks, intricate multipriority service disciplines, and dynamic resource allocations.

Therefore, we choose the imitative simulation method oriented on the knowledge of dynamics of the system functioning. The group of specialists from the Laboratory of Information Technologies (LIT) of JINR, Dubna, experienced already with simulation grid

structures, developed the new simulation program called SyMSim (Synthesis of Monitoring and Simulation) [13]. The SyMSim development was inspired by the known simulation library GridSim [14], job scheduler ALEA [15] and based on the following *basic simulation concepts*:

1. The best way to evaluate dynamically the system functioning quality is to use its monitoring tools.

2. The simulation program is to be combined with a real monitoring system of the grid/cloud service through a special database (DB).

3. To ensure a developer from writing the simulation program from zero at each development stage, it is more feasible to accept a twofold model structure which consists of

a) a core — its stable main part independent on simulated object and

b) declarative module for input of model parameters defining a concrete distributed computing center — its setup and parameters obtained from monitoring information, as dataflow, job stream, etc.

4. DB intention is just to realize this declarative module work and provide means for output of simulation results.

5. Web portal is needed to communicate with DB assigning concrete simulation parameters and storing results in DB.

SYMSIM APPLICATION FOR THE CMS TIER1 AT JINR SIMULATION

SyMSim was successfully tested on simulation of the JINR CMS Tier1 center with a robotized tape library. One of the main problems was to simulate the running data storage system of JINR Tier1 with robotized tape library, where RAW data are to be regularly and without losses transferred from CMS experiment disks of Tier0 at CERN.

The goal of our study was to make sure that our combined approach with obtaining initial simulation parameters for Tier1 from CMS monitoring via data base is working correctly and SyMSim program can be applied to simulate a more sophisticate and planning yet the Tier0–Tier1 system of the NICA project. The following improvements were made in SyMSin to accomplish that: (1) new classes were invented to declare the data store specific for the tape robot library including a needed disk buffer; (2) input job and data flows were formed via data base on the basis of Tier1 monitoring information; (3) data exchange process was modified from packet flow simulation into file transfer simulation; (4) software means for handling simulation results were provided.

Besides, there are some simplifications in the model: the number of the active sites is limited; the same jobs flow are used for different strategies; each task demands only one file; few tasks can use the same file; at start files are statically distributed between the sites, disks, and tapes; the files which are written on the sites disk pools remain there until the experiment is over; each file has only one copy.

As one example of comparing the real and simulated characteristics of Tier1 workflow, in Fig. 3 two graphics of completed job numbers during one month in 2015, on the right — real number obtained from Tier1 monitoring with average 24000 and rms 6100, on the left — simulated one with average 19700 and rms 6700, that is quite close within the error corridor, are shown.

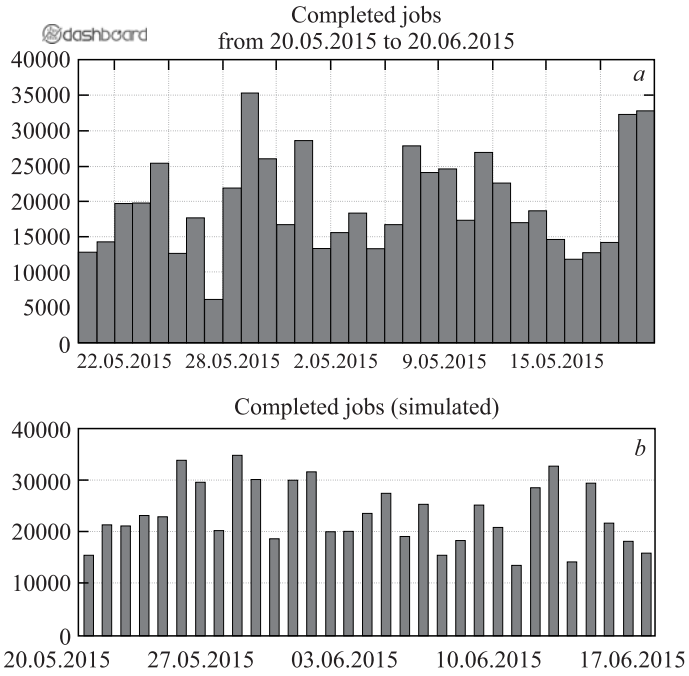

Fig. 3. Real (*a*) and simulated (*b*) Tier1 completed job numbers during one month in 2015

This example among some others was used for a positive validation of the running CMS T1 model and encouraged us to simulate the Tier0 –Tier1 system of the NICA project.

SYMSIM APPLICATION FOR THE NICA TIER 0/1 AT JINR SIMULATION

One of the planned tasks is to recommend the volume of the disk store and a temp of data transfer from Tier0 center to the robotized library which is part of the Tier1 center. The data storage and processing scheme of the NICA Tier0–Tier1 is depicted in Fig. 4, where Tier0 module denotes the center of data gathering from an experiment (either MPD or SPD).

The obtained raw data are to be stored on disks. This two-level structure is interconnected by a local area network. DQ in Fig. 4. denotes not only DAQ of the corresponding experiment, but also includes means of communications and buffer cleaning.

Initial information to start simulation includes setup parameters of designed hardware as well as data flow, job stream, which characteristics are taken from real data of CMS Tier1 monitoring and the DAQ MPD Technical Design Report.

Let us recall two issues of the basic simulation concepts, namely, database and web portal, which needed to be expounded more in details for a better understanding of the simulation process.

Database (DB) contains the description of the grid structure, each of its nodes and links between them, the monitoring results of the various subsystems of the grid. Then DB provides a job flow generation, as the job number in the given time interval, their starting and execution times. So, DB contains all information about running jobs and, eventually, the simulation results.

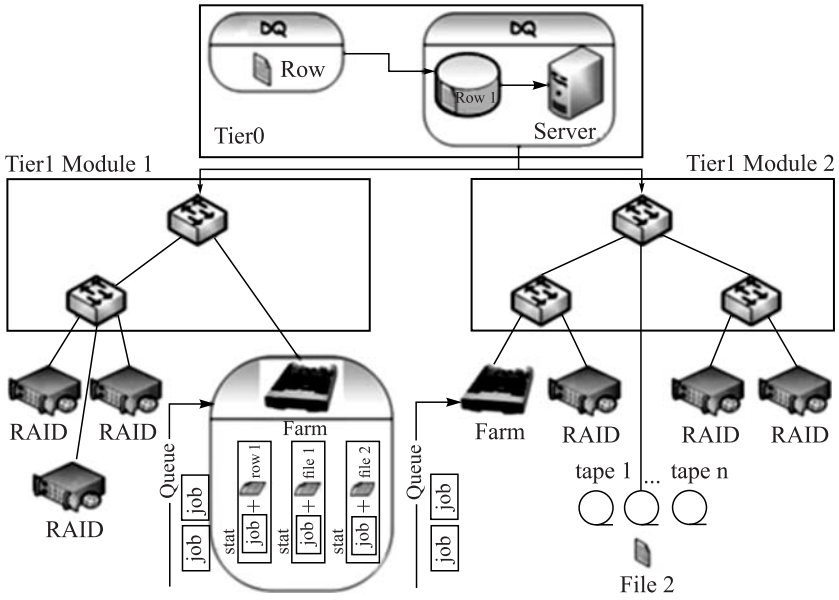

Fig. 4. Data storage and processing scheme of the NICA Tier0–Tier1

To ensure its compatibility with the monitoring system, the format of the database has been chosen to coincide with the database of the workflow management system (WMS) PanDA [9] which is considered as a perspective for a number of new and upgraded experiments.

Therefore, the main table (jobswaiting4) is the same as a similar table of the PanDA's database. This table contains a description of the job flow, which is used as the simulation input data. This compatibility makes it possible to use the monitoring data without changing parameters. The simulation results are written to the database in the PanDA required format. However, we have to add the tables to describe grid or cloud sites to communicate with the web portal and also the managing system of the computational experiments.

Four types of jobs are chosen to be generated according to the CMS and ATLAS experience:

1. Data acquisition (DQ) — simulated "raw" data to be stored.
2. Monte Carlo (MC) — do not need input data.
3. Express analysis (EA) — jobs use recently obtained files.
4. Reconstruction processing (PR) — jobs consume the most of resources.

The database was designed to make further use of it via a web portal. It means that multiple users can work simultaneously, store in DB input data and simulation results for different experiments.

Web Portal Development. Web portal for the description of a designed experiment computing structure and workflow parameters is developed.

The web-portal functions are as follows: interact with the database; describe the current model structure; generate new workflow with given parameters (number of events, numbers of DQ, MC, EA, PR jobs, the needed CPU time and memory); represent simulation results.

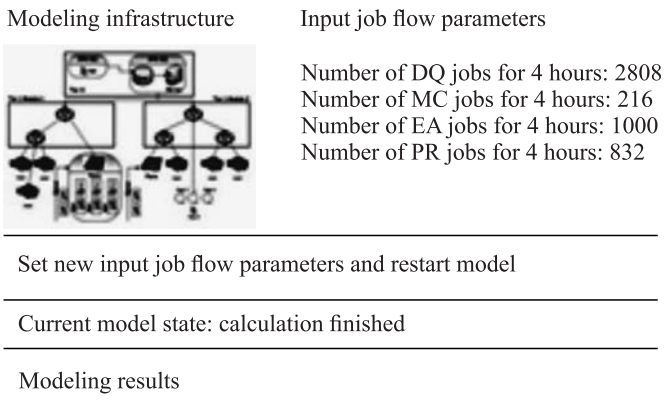

Fig. 5. Snapshot of the SyMSim web portal

The description of assigned IDs (of user, experiment, hardware, configuration, etc.) is specified in the startup parameters of a model. The latter reads the information from the database and builds a description of the computational structure. After that a set of tools is launched. These tools allow one to use the monitoring data, generate a workflow with different parameters. Since the graphical presentation of simulation and statistics results was found as the most suitable, the corresponding software means were provided to plot graphics and diagrams.

An example of the SyMSim web portal snapshot is shown in Fig. 5.

There is a point to keep in mind when one interprets simulation results, it is the initial transition process. The simulation algorithms are inevitably designed in such a way that at the initial time all buffers are empty, the processors are not loaded, and the data are not transferred. Therefore, the initial transition process must be excluded from the analysis. It also happens when the current job flow stops.

The result of the simulation program is a sequence of records in the database, which reflects all the events occurring at the system during the simulation run.

After obtaining simulation results one can analyze the following: the intensity of the data flow on communication equipment and lines; the utilization of the computing elements; the usage level of the data storage elements.

Based on those results and the optimization criteria, one can select the hardware configuration, data transfer and storage algorithms, and priorities for queues for the future data processing center.

In the first study the authors propose the following list of charts built depending of the system time used during the simulation:

— load on the communication line between the detector and the data center;

— load on each communication node;

— number of jobs executed on each farm;

— load on the interchange device of the tape robot library;

— job waiting time in the queue;

— value of the free space on the disk buffer provided by a cleaning feature to prevent its overfull.

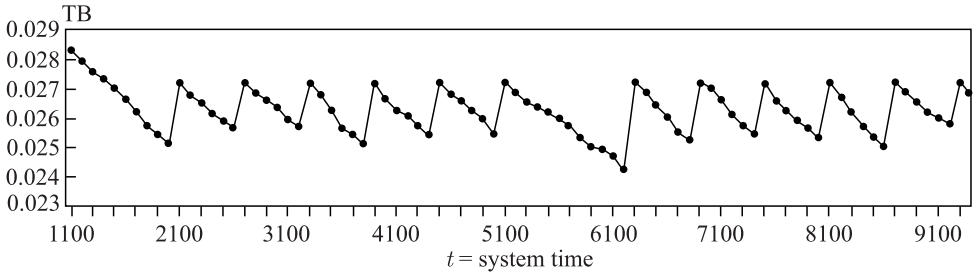

Fig. 6. Disk available space (in terabytes)

This list is not exhaustive, of course, and just gives an example of possibilities, which the SyMSim simulation can provide at the stage of a grid–cloud system design. One example illustrating the last chart from the above list is depicted in Fig. 6, where the disk available space vs. system time is shown in order to find out what buffer size is needed to store input files on tapes without losses.

Zigzag shape of the curve in Fig. 6 is due to regular buffer cleaning. The sharp slump in the middle is caused by end-of-tape delay. This result proves that due to clever buffer cleaning the buffer should not be too big, so we can place it in the RAM operational memory.

The test results of the NICA project simulation proves that the SyMSim program, in particular, all database queries, are executed correctly.

Several computational experiments were also accomplished. The specification of simulated experiments and results are described in [13, 16]. However, at its present simplified state the SyMSim program needs further development to face new challenges of the future distributed computing in experimental physics.

CONCLUSIONS

The originality of the proposed simulation approach consists in combining a simulation program with a real monitoring system of the grid–cloud service through a special database in the framework of the same program. The program SyMSim for simulation of grid–cloud structures has been developed and tested on a simplified model of the JINR Tier1 site. The next simulation was accomplished for Tier0–Tier1 computing facilities of JINR NICA–MPD–SPD project. It confirms a good potential of our simulation approach, but also shows some of its incompletenesses needed to be retreated.

SyMSim structure is sufficiently general and flexible to replace our present simplifications into more real conditions in future developments. It can also be used to solve some design problems and the subsequent development of data repositories, not limited by the physical experiments area.

Acknowledgements. This work was supported by the RFBR grants Nos. 14-07-00215 and 15-29-07027.

REFERENCES

1. *Foster I. et al.* The Physiology of the Grid: An Open Grid Services Architecture for Distributed Systems Integration. http://www.globus.org/alliance/publications/papers.php.
2. LHC. http://home.web.cern.ch/topics/large-hadron-collider.

3. CERN Computing. http://home.web.cern.ch/about/computing.
4. WLCG. http://home.web.cern.ch/about/computing/worldwide-lhc-computing-grid.
5. JINR CMS Tier1. http://lit.jinr.ru/Reports/SC_report_12-13/p16.pdf.
6. *Bloom K., Gerber R.* https://www.nersc.gov/assets/pubs_presos/1311.2208v1.pdf.
7. *Korenkov V.* GRID in JINR and Participation in the WLCG Project // Proc. of the 5th lntern. Conf. GRID-2012. Dubna, 2012. P. 254–265 (in Russian); http://grid2012.jinr.ru/programme.php.
8. NICA — Nuclotron-Based Ion Collider Facility. http://nica.jinr.ru.
9. PANDA. https://poland.jinr.ru/docs/de_panda-grid2012.pdf.
10. *Golubkov D. et al. (ATLAS Collab.).* ATLAS Grid Data Processing: System Evolution and Scalability // J. Phys.: Conf. Ser. 2012. V. 396. P. 032049.
11. BigPanda. https://indico.cern.ch/event/258092/session/0/contribution/136/attachments/454145/629534/4.Klimentov.pdf.
12. *Popkov Yu.* Macro Systems and Grid-Technology: Simulation of Dynamic Stochastic Networks // Control Sci. 2003. No. 3. P. 10–20 (in Russian).
13. *Korenkov V. et al.* Simulation of Grid–Cloud Services as Important Stage of Their Development // Systems and Means of Informatics, 2015. V. 25, No. 1. P. 3–19 (in Russian).
14. GridSim: A Grid Simulation Toolkit for Resource Modelling and Application Scheduling for Parallel and Distributed Computing. http://www.gridbus.org/gridsim/. The Univ. of Melbourne, 2014.
15. *Klusacek D., Matyska L., Rudova H.* Alea — Grid Scheduling Simulation Environment // 7th Intern. Conf. on Parallel Proc. and Appl. Math. (PPAM 2007). 2008. V. 4967 of LNCS. P. 1029–1038.
16. *Korenkov V. et al.* Grid–Cloud Services Simulation for NICA Project, as a Mean of the Efficiency Increasing of Their Development // Comp. Res. Modeling. V. 6, No. 5. P. 635–642 (in Russian).