USING DISTRIBUTED CLOUDS FOR SCIENTIFIC COMPUTING

N.A. Kutovskiy[1] [0000-0003-1001-4687], **I.S. Pelevanyuk**[1,3,a] [0000-0002-4353-493X], **D.N. Zaborov**[2] [0000-0002-9335-1410]

[1] *Joint Institute for Nuclear Research, 6 Joliot-Curie St., Dubna, Moscow Region, Russia, 141980*

[2] *Institute for Nuclear Research, Russian Academy of Sciences, 7a 60-letiya Oktyabrya Prospekt, Moscow, 117312*

[3] *Plekhanov Russian University of Economics, 36 Stremyanny Lane, Moscow, 117997*

E-mail: [a] pelevanyuk@jinr.ru

Nowadays, cloud resources are the most flexible tool to provide access to infrastructures for establishing services and applications. However, it is also a valuable resource for scientific computing. At the Joint Institute for Nuclear Research, the computing cloud was integrated with the DIRAC system. It allowed for the submission of scientific computing jobs directly to the cloud. Thanks to the experience, the cloud resources of several organizations from the JINR Member States were integrated in the same way. It increased the total amount of cloud resources accessible in a uniform way through DIRAC, in the scope of the so-called Distributed Information and Computing Environment (DICE). Folding@Home tasks related to the SARS-CoV-2 virus were submitted to all available cloud resources. In addition to useful scientific results, such experience was also helpful in obtaining information about the performance, limitations, strengths, and weaknesses of the combined system. Based on the gained experience, the DICE infrastructure was tuned to successfully perform real user jobs related to Monte-Carlo simulation for the Baikal-GVD experiment.

Keywords: data processing, cloud computing, distributed computing, GRID applications

Nikolay Kutovskiy, Igor Pelevanyuk, Dmitry Zaborov

1. Introduction

The Joint Institute for Nuclear Research (JINR) is an international intergovernmental organization. It is developing as a large multidisciplinary international scientific center incorporating basic research in the field of modern nuclear physics, the development and application of high technologies, as well as university education in the relevant fields of knowledge. Currently, JINR has 18 Member States and 6 countries participating in JINR's activities on the basis of bilateral agreements signed at the governmental level.

The cloud infrastructure [1] deployed at JINR was created in 2013 to manage the IT services and servers of the Meshcheryakov Laboratory of Information Technologies (MLIT) more efficiently using modern technologies, to combine resources for solving common tasks, to increase the efficiency of hardware utilization and service reliability, to simplify access to application software and optimize the use of proprietary software, as well as to provide a modern computing facility for JINR users.

Cloud infrastructures are a flexible tool for many different tasks. One of these tasks is massive scientific computing. Clouds can provide a resource capable of handling a High-Throughput Computing workload. JINR takes part in several experiments where extensive Monte-Carlo generation is required. The use of cloud infrastructures of the JINR Member States can also be used for such workloads.

The integration of the cloud resources of JINR and its Member States was performed with the help of the DIRAC Interware [2]. DIRAC is a platform that provides basic tools and methods for combining heterogeneous infrastructures and their operation for large sets of computing tasks. A special module was developed at JINR in order to integrate clouds to the system [3]. At the moment, 9 clouds from JINR and its Member States are included in the DIRAC system. All of these distributed heterogeneous computing resources represent a valuable and powerful resource. The main features of these clouds are shown in Table 1. The major questions related to this unique resource are how to use it efficiently in scientific computing, what are the limitations of the combined system, how to support it in the operational state, and what to do with possible failures.

Organization	Location	CPU cores	RAM GB
Joint Institute for Nuclear Research	Russia	80	320
Plekhanov Russian University of Economics	Russia	132	608
Astana branch of the Institute of Nuclear Physics	Kazakhstan	84	840
Institute of Physics of the National Academy of Sciences of Azerbaijan	Azerbaijan	16	96
North Ossetian State University	Russia	84	672
Academy of Scientific Research & Technology - Egyptian National STI Network	Egypt	98	704
Institute for Nuclear Research and Nuclear Energy	Bulgaria	20	64
St. Sophia University "St. Kliment Ohridski"	Bulgaria	48	250
Scientific Research Institute for Nuclear Problems of Belarusian State University	Belarus	132	290
Total		614	3524

Table 1. Cloud resources accessible through DIRAC at JINR.

2. Features of the distributed cloud infrastructure

The DIRAC Interware platform is used to integrate different clouds into a united infrastructure. For this purpose, a dedicated user account should be created on each cloud. All resources provided to this user within their quota will be available for creating virtual machines.

The DIRAC provides a job queue where users can send their jobs. The job is described in a special way, either using the DIRAC Python API or with a special Job Description Language. The API is the best way for massive job submissions. The parameters of the job include the following information: name of the executable on the virtual machine, arguments passed to the executable, list of files to be uploaded with the job ("Input Sandbox"), list of files to be downloaded back ("Output Sandbox"), and optionally, list of appropriate clouds that can complete this particular job. It is not possible to include large files (more than 5 MB) in input and output sandboxes. The general recommendation is to put only logs and configuration in sandboxes.

The standard approach is to create a Shell script that describes the whole workflow of a particular job. Generally, each job described in the Shell script consists of several major steps: initial configuration, input data download, processing, output data upload, and finalization. Any step can be omitted if not required. The Shell script and the necessary configuration files are included in the input sandbox. The standard output of the job is automatically redirected to a file that is included in the output sandbox.

All data files should be placed on storage elements integrated with DIRAC. Currently, only systems with the support of the root protocol for access and VOMS (Virtual Organization Membership Service) for authentication can be used as storage elements in DIRAC. The use of VOMS ensures that a user with the correct membership in a VOMS group will be able to access storage elements from anywhere.

User jobs are not directly submitted to the cloud. At first, when jobs suitable for the clouds appear, DIRAC creates virtual machines on one of the appropriate clouds with special instructions to execute after the boot process. These instructions contain information about the installation of DIRAC Pilot. DIRAC Pilot is a special process that performs basic checks of the resource it is running on and submits the results to the DIRAC Matcher service. The DIRAC Matcher service chooses a job that can be completed on a resource with the received parameters. This job and the corresponding input sandbox are downloaded by DIRAC Pilot. After that, the user job starts as a child process of DIRAC Pilot. This scheme eliminates resources that cannot complete the job, for example, due to the lack of RAM or an inappropriate OS. After completing the first job, DIRAC Pilot can request another.

3. Limitations on job execution in the cloud

The main concern during job submission is related to the fact that the network connection between the cloud and the storage element is a limited resource. It means that even if a single job is successfully completed during the testing stage, it does not guarantee that hundreds of similar jobs will work in the same way. All these jobs will need to retrieve user software, download some input files and upload results to a storage element. This can be a bottleneck for the whole pack of jobs and can increase the execution time. What in turn decreases the efficiency of CPU utilization, since jobs will be waiting for data transfers for a substantial amount of time.

The CernVM File System (CVMFS) is used for the software distribution across the computational resources of all participating organizations. It caches the queried files in a dedicated directory on the local CVMFS caching node, and the rest of the queries to the same files are served from this local cache. The client-side CVMFS cache also helps to lower a load on the network when multiple similar jobs reusing the same CVMFS files are executed subsequently on the same virtual machine; in the clouds, where virtual machines are spawned and deleted, often the benefits of such file caching cannot be pronounced. DIRAC provides the necessary options for configuring the IP addresses of caching CVMFS servers for different clouds.

The cloud infrastructure does not provide a standardized solution for caching input data files. Hence, custom solutions may need to be considered. For example, some input data can be downloaded once and cached in a temporary directory on the local file system of the virtual machine. Then jobs requested by DIRAC Pilot after the first job can use the cached data. It can also be done by placing the input data on the CVMFS file system, but due to the limited frequency of CVMFS file system synchronization, this will only work for very static data. Mainly due to the network bandwidth limitations, clouds are not currently proposed as resources for massive data processing. On the other hand, clouds can be highly efficient at executing a wide range of CPU-intensive jobs characterized by modest storage requirements. This is typical of jobs performing Monte Carlo simulation of particle physics experiments, which normally require very little input data, but are highly CPU-intensive. During Monte-Carlo data generation, the only data that will definitely require a real network transfer is the output. The output data size is usually predictable, and it is possible to estimate the number of jobs that are reasonable for simultaneous execution on a single cloud, given the network bandwidth restrictions. The described problem does not relate to every cloud. Some of them were designed as tools for massive data processing and possess a high-bandwidth external network connection, while others were designed to host services that are not so demanding in terms of network and have a very limited external network bandwidth.

Sometimes it is possible to negotiate with cloud owners to upgrade the network connection. Demonstration that their cloud can participate in computing for large scientific collaborations can be a good reason for the network upgrade. Thus, we prepared two successful use cases of scientific workload execution in the clouds.

4. Folding@Home jobs in the clouds

Folding@Home is a community of volunteers, researchers and organizations that help with their intelligent and computing resources to understand the dynamics of proteins, their functions and dysfunctions in order to find new proteins and drugs. Every person in the world can install the Folding@Home client on their home computer and calculate scientific jobs. When SARS-CoV-2 appeared, Folding@Home created a queue devoted to the study of this virus. Many people and organizations worldwide donated their resources to this cause.

We decided that Folding@Home jobs were perfect for the demonstration of distributed cloud performance, not with artificial tests, but with real jobs. The main feature of Folding@Home jobs is that the input and output are small, but the amount of computing work is large. This is a perfect mode for cloud resources. The Folding@Home software was installed in advance in all images used by DIRAC on all clouds. A special DIRAC job was designed for this task. It starts the Folding@Home client, but only for one work unit from Folding@Home. After its execution, the job is considered completed. After that, a new DIRAC job can take its place.

The total number of successfully completed Folding@Home jobs exceeds 13 thousand. The distribution of the jobs across the clouds is shown in Figure 1. The completion of each job took on average 15 hours. Only single-core jobs were executed. The total amount of normalized consumed computer power is around 135 kHS06 days.

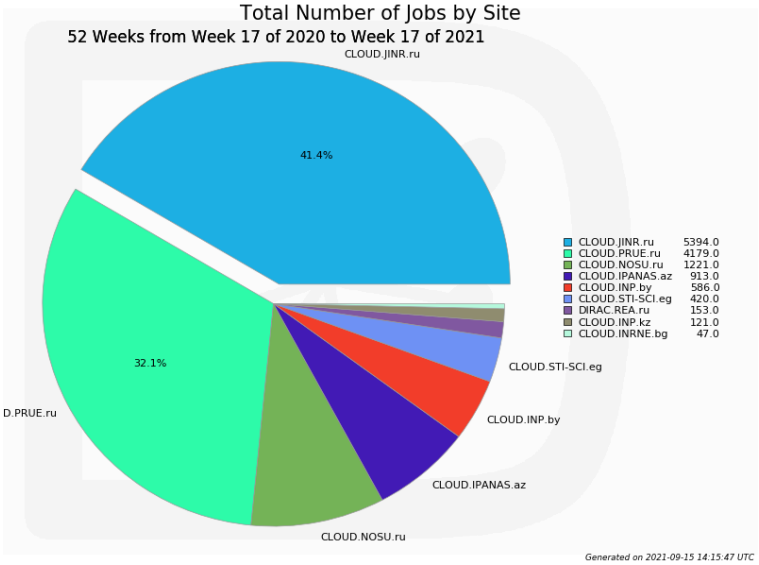

Figure 1. Total number of Folding@Home jobs completed by different clouds.

4. Baikal-GVD jobs in the clouds

Baikal-GVD is a cubic-kilometer scale underwater neutrino detector currently under construction on Lake Baikal (Russia) [4]. At JINR, Baikal-GVD is the first large particle physics experiment that massively used cloud resources combined by DIRAC for Monte-Carlo simulation purposes. The majority of jobs were part of large-scale Monte-Carlo production that involved simulating the propagation of high-energy muons in water and the detector response. The standard workflow of a user job with input data download and output data upload was used. Each job required to download an input file of about 2 GB. The result of Monte-Carlo generation had an average size of 370 MB. Only two cloud resources were ready for this type of workload. The PRUE cloud was busy at the time of production, so most of the jobs were completed by the JINR cloud. Only single-core jobs were executed. The total number of successfully executed jobs is 67.5 thousand. The utilization of the JINR cloud was around 80% during this simulation campaign, which is demonstrated in Figure 2. The completion of each job took on average 6 hours. The total amount of normalized consumed computer power is around 280 kHS06 days.

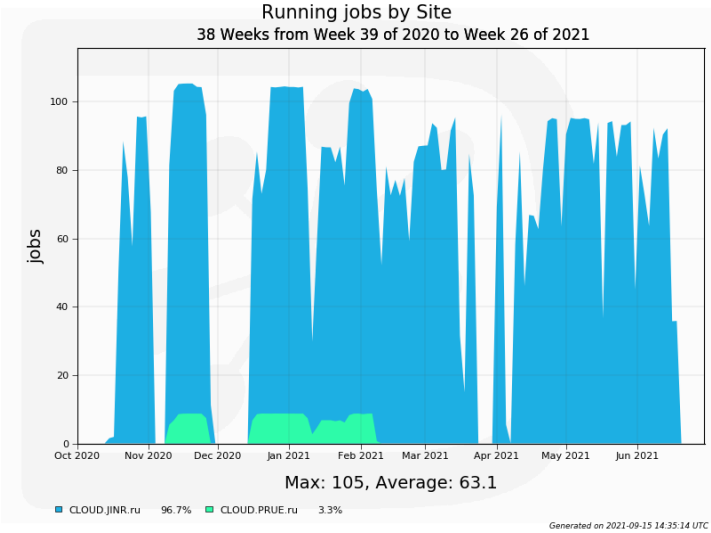

Figure 2. Number of running Baikal-GVD jobs at each moment during production.

4. Conclusion

Clouds are a valuable computing resource for scientific computing. Their flexibility provides a unique opportunity for tuning working environments for the requirements of different jobs: OS, amount of local storage, amount of RAM per core; all of which can be configured on the cloud. The low network bandwidth of any participating organization is the main critical parameter in defining the number of jobs that can be simultaneously executed.

We showed an example of cloud integration via the DIRAC Interware. User jobs should be configured in order to be executed by DIRAC. It requires some significant effort from users. However, once it is done, it is possible to greatly increase the amount of accessible computing resources. This is good not only for users, but also for clouds. DIRAC jobs are a good way to improve the utilization of cloud resources, and a possibility to participate in computing for scientific collaborations.

Folding@Home jobs were successfully executed in the clouds. That gave us experience in running Monte-Carlo jobs of Baikal-GVD.

References

[1] Balashov N.A. et al., Present Status and Main Directions of the JINR Cloud Development // Proceedings of the 27th International Symposium Nuclear Electronics and Computing (NEC'2019), CEUR Workshop Proceedings, ISSN:1613-0073, vol. 2507 (2019), pp. 185-189

[2] Korenkov V., Pelevanyuk I., Tsaregorodtsev A. Integration of the JINR Hybrid Computing Resources with the DIRAC Interware for Data Intensive Applications // Data Analytics and Management in Data Intensive Domains. 2020. P. 31-46. DOI: 10.1007/978-3-030-51913-1_3

[3] N. Balashov, R. Kuchumov, N. Kutovskiy, I. Pelevanyuk, V. Petrunin, and A. Tsaregorodtsev, CEUR Workshop Proceedings 2507, 256 (2019), URL http://ceur-ws.org/Vol-2507/256-260-paper-45.pdf.

[4] Baikal-GVD Collaboration: V.A. Allakhverdyan et al., Measuring muon tracks in Baikal-GVD using a fast reconstruction algorithm, submitted to EPJ C, arXiv:2106.06288.