

JINR cloud infrastructure development

**A.V. Baranov¹, N.A. Balashov¹, K.V. Fedorov¹, R.R. Gainanov²,
N.A. Kutovskiy^{1,3,a}, R.N. Semenov^{1,3}**

¹ Laboratory of Information Technologies, Joint Institute for Nuclear Research, 6 Joliot-Curie, Dubna,
Moscow region, 141980, Russia

² Department of Information Technologies and Computer-Aided Systems, Perm National Research Polytechnic
University, 29 Komsomolsky prospekt, Perm, 614990, Russia

³ Plekhanov Russian University of Economics, 36 Stremyanny per., Moscow, 117997, Russia

E-mail: ^a kut@jinr.ru

A number of the JINR cloud users as well as a demand in cloud resources from both individual users and experiments are growing permanently. To increase a reliability and availability of the service, the JINR cloud has been migrated from a single server-based front-end node setup to a high availability one. To cover the resources deficit, work is in progress in three directions: 1) the resources usage optimization with the help of a smart scheduler which consolidates underloaded virtual machines and containers within the hosts; 2) an external clouds integration with the JINR one following the so-called “cloud bursting” approach what a custom driver was developed for; 3) buying more servers. An aggregation and visualization of the JINR cloud resources utilization was re-implemented and moved to external services based on Grafana and InfluxDB. All these aspects are covered in more details in that article.

Keywords: cloud computing, OpenNebula, clouds integration, cloud bursting, smart cloud scheduler, Grafana, InfluxDB

© 2016 Alexandr V. Baranov, Nikita A. Balashov, Konstantin V. Fedorov, Ruslan R. Gainanov,
Nikolay A. Kutovskiy, Roman N. Semenov

High availability setup

To increase a reliability and availability of the JINR cloud resources delivery to its end-users, a front-end node (FN) was migrated from a single server-based server setup to a high availability (HA) one with two servers and shared partition based on distributed replicated block device (DRDB) where OpenVZ container (CT) with OpenNebula core components deployed is running on. Apart from increasing the reliability and availability by eliminating a single point of failure a new setup let to overcome such limitations of previous configuration as an impossibility to extend a size of FN cloud datastores on the fly and to maintain without any downtime a server where FN is running. A schema of JINR HA-cloud setup is shown on the Figure 1. Two hardware nodes (HNs) marked on that figure as HN1-FN and HN2-FN are physical servers of HA cluster with one dedicated DRBD partition in each node where CT with all cloud components deployed is running.

To be able to dynamically extend the cloud datastores capacities and to increase the reliability of data preservation, a distributed network file system (DNFS) based on LizardFS was deployed and configured as a storage backend for cloud. It consists of three storage nodes (SNs) with the following roles each: cloudfs1 is master, chunk-server and web-based admin GUI, cloudfs2 is metalogger and chunk-server, cloudfs3 is shadow and chunk-server.

The total capacity of that storage is 16 TB, but due to a redundancy set to 2 the effective size is 8 TB.

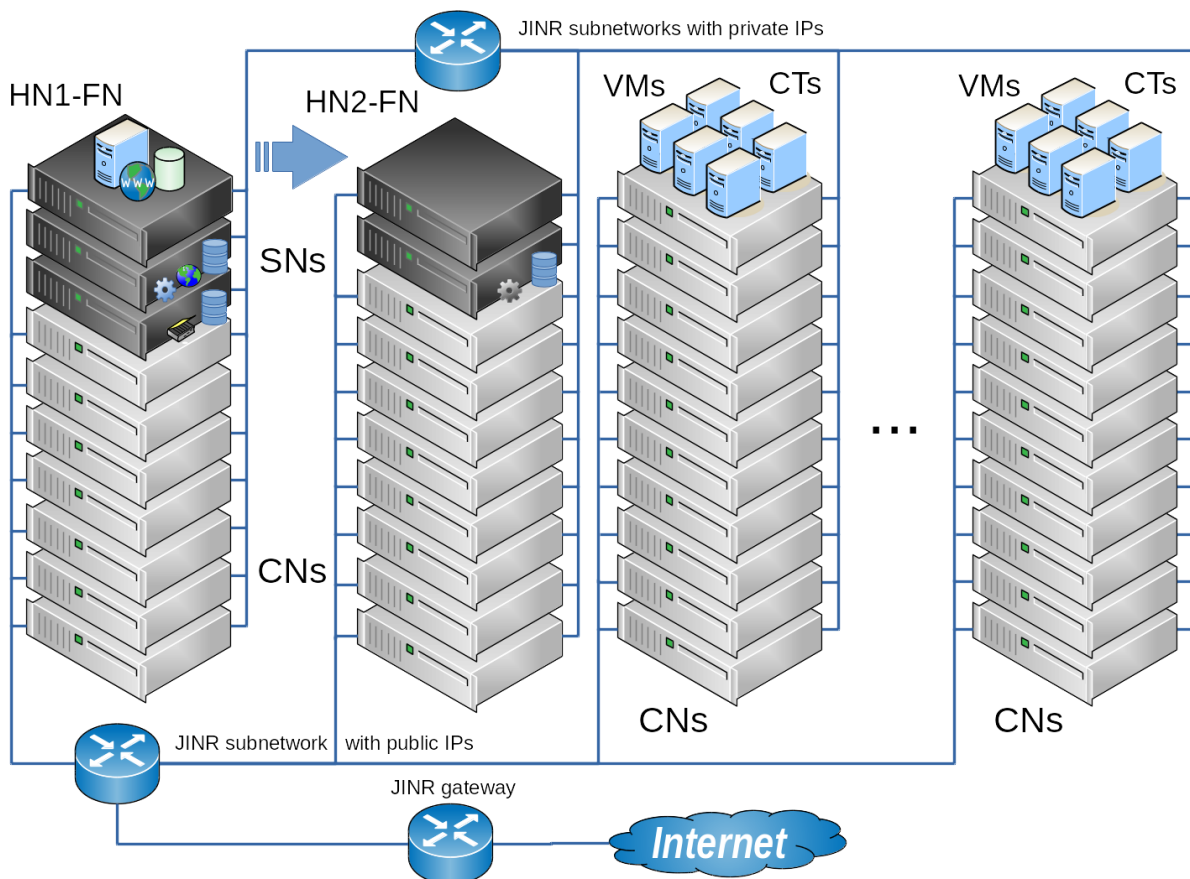


Figure 1. A schema of high availability JINR cloud setup

More details on tested HA-configurations can be found in [JINR cloud..., 2016]. Smart cloud scheduler

An experience of running the JINR cloud in production for almost three years shows that a lot of allocated resources are underutilized. That decreases an overall efficiency of the JINR cloud resources usage sufficiently. To change that it was decided to develop a smart cloud scheduler (SCS) with the following logic behind. All cloud servers are ranked: new and most powerful servers have maximal rank but elder and weaker hosts have a minimal one. Initially new virtual instance (VI, it can be either a virtual machine or container) is deployed on the cloud server with highest rank. During VI's life the smart cloud scheduler tracks a real consumption of CPU and RAM resources by each VI. If such consumption is less than a defined threshold then the SCS will initiate a live migration of such VI to the less ranked server. That procedure is constantly repeated until underloaded VI reaches the worst ranked server. If such VI starts consuming more resources the SCS initiates its live migration into backward direction, i.e. to the host with higher rank than a previous one.

More details on smart cloud scheduler are expounded in [Smart cloud scheduler..., 2016].

Clouds integration

The amount of resources available for the JINR cloud users has been increased lately due to 1) maintaining additional servers as cloud nodes and 2) integration of part of computing resources of the partner organizations' clouds.

To join the cloud resources of partner organizations from JINR Member State for solving common tasks as well as to distribute a peak load across them, a cloud bursting driver [Approaches to cloud..., 2016] was developed by the JINR cloud team. It allows one to integrate the JINR cloud with partner clouds either OpenNebula-based one (and in that case it is possible to enable real time external cloud resources monitoring) or any other cloud platform which supports Open Cloud Computing Interface (OCCI) but then statically defined values of external cloud resources are possible only. The driver is based on ruby implementation of OCCI (rOCCI) and OpenNebula eXtensible Markup Language Remote Procedure Call (XML-RPC) interface. The reason why both interfaces are used at the same time is that OCCI allows a better re-use of the code since it is provided by other popular cloud platforms like OpenStack, while OpenNebula XML-RPC is OpenNebula-specific and cannot be used for integration with other platforms. At the same time, OCCI does not provide monitoring operations, while OpenNebula XML-RPC does. It was decided to use rOCCI for management operations and XML-RPC for monitoring.

The clouds of the following partner organizations from the JINR Member States are integrated with the JINR one:

- 1) Institute of Physics of Azerbaijan National Academy of Sciences (Baku, Azerbaijan);
- 2) Bogolyubov Institute for Theoretical Physics of the National Academy of Sciences of Ukraine (Kiev, Ukraine);
- 3) Plekhanov Russian University of Economics – PRUE (Moscow, Russia).

Apart from that the JINR cloud is integrated into EGI Federated cloud thus enabling a possibility to use a part of JINR computing resources by EGI FedCloud Virtual Organizations.

At the moment of writing that article a total number of CPU cores available for users in the JINR cloud is 330 (it was 200 in 2015) and total amount of RAM is 840 GB (was 400 GB).

Re-implementation of statistics visualization

Initially a visualization of statistics on the JINR cloud resources utilization was done by JINR cloud team as an additional item in a menu of OpenNebula graphical web-interface (which is called "Sunstone"). But a drawback of such implementation was a necessity to check a compatibility of that visualization add-on against each new release of OpenNebula because its web-interface might be

changed. It was needed to adopt the code of visualization module for the new Sunstone in case of its incompatibility. So in order to avoid such extra step before software update on the JINR cloud FN as well as to store a collecting metrics in a database for further analysis against their changes over time and to obtain the dynamics for the selected period it was decided to implement an aggregation and visualization statistics on JINR cloud resources utilization using such tools as InfluxDB and Grafana software. One of such pie-chart diagram is shown on the Figure 2.

Future plans

The re-implemented statistics visualization module decreases an amount of components to be checked for compatibility against new version of OpenNebula software. So as soon as the validation of the rest custom components such as OpenVZ driver and Kerberos authentication plug-in are finished, it is planned to update software on OpenNebula front-end node to the latest stable version.

The OpenVZ driver for OpenNebula needs to be updated to work with 7th version of OpenVZ software which is intended to be deployed on RHEL7-compatible Linux distributions. That would allow one to upgrade the operating system to the latest stable 7.x branch of CentOS on all cloud worker nodes.

Upon smart cloud scheduler readiness it will be integrated into the JINR cloud.

There is a plan to buy and mount 10 Gbps network equipment and disk servers to build CEPH-based storage. That would let host a disk images of running VIs on shared CEPH directly instead of

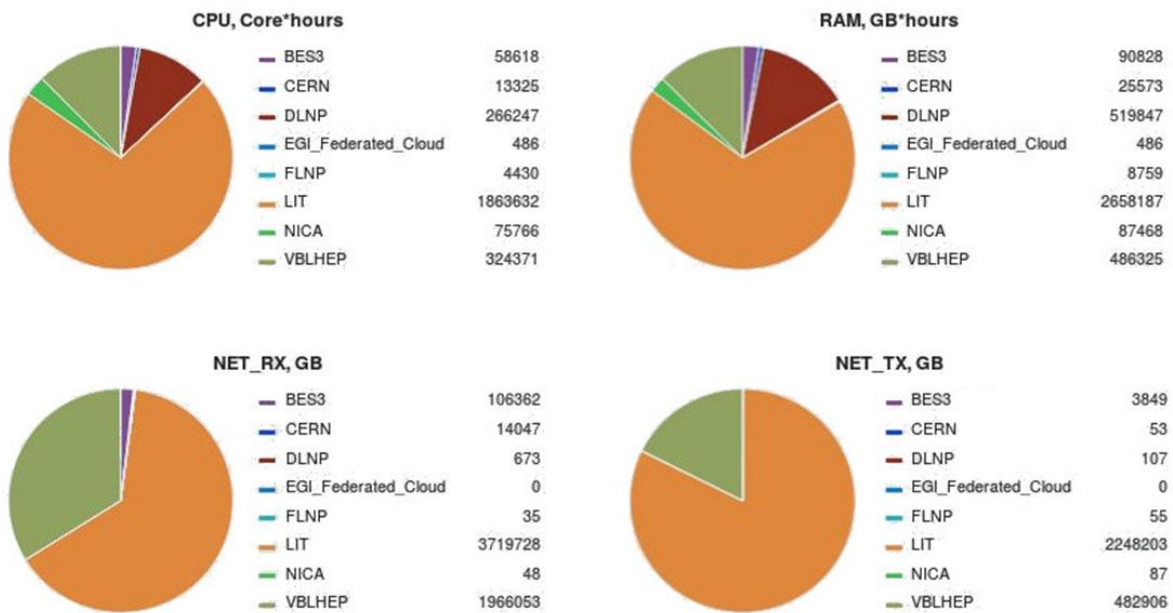


Figure 2. A screenshot of Grafana dashboard with information on consumed resources distribution by departments and projects from 5-Oct-2015 till 5-Oct-2016

running them on local CNs disks causing a necessity to copy them over the network during deployment or stop operations.

In order to use the cloud resources as a computing backend for batch-systems and thus grid computing elements, it is planned to validate a set of helping components (e.g. VCycle, Vac, VCondor/VPManager) implementing this functionality.

Acknowledgement

The work on smart scheduler part was supported by the Russian Foundation for Basic Research (RFBR), grant #15-29-07027 "Intelligent scheduling and adaptive self-organization software development for the cloud computing resources of LIT JINR".

Conclusion

The JINR cloud is rapidly developing. A demand in its resources as well as a spectrum of tasks it is used for is growing permanently.

The JINR cloud front-end node configuration was migrated to HA one to increase the reliability and availability of the service for the end-users and to overcome other limitations of previous setup.

The amount of resources available for the JINR cloud users had been increased in 2016 sufficiently due to maintaining additional servers as cloud worker nodes and an integration a part of computing resources of partner's organizations clouds.

Re-implementation of the statistics visualization module as an external component allows one to reduce the amount of time needed for custom components validation against new version of OpenNebula software.

References

- Baranov A.V. et al.* JINR cloud infrastructure evolution // Physics of Particles and Nuclei Letters, ISSN 1547-4771, eISSN: 1531-8567, 2016, vol. 13, No. 5, pp. 672–675.
DOI: 10.1134/S1547477116050071.
- Baranov A.V. et al.* Smart cloud scheduler for OpenNebula, to appear in proceedings of GRID'2016 conference, 04 – 09 July 2016, Dubna, Russia.
- Baranov A.V. et al.* Approaches to cloud infrastructures integration // Computer Research and Modeling, 2016, vol. 8, No. 3, pp. 583–590 (in Russian).