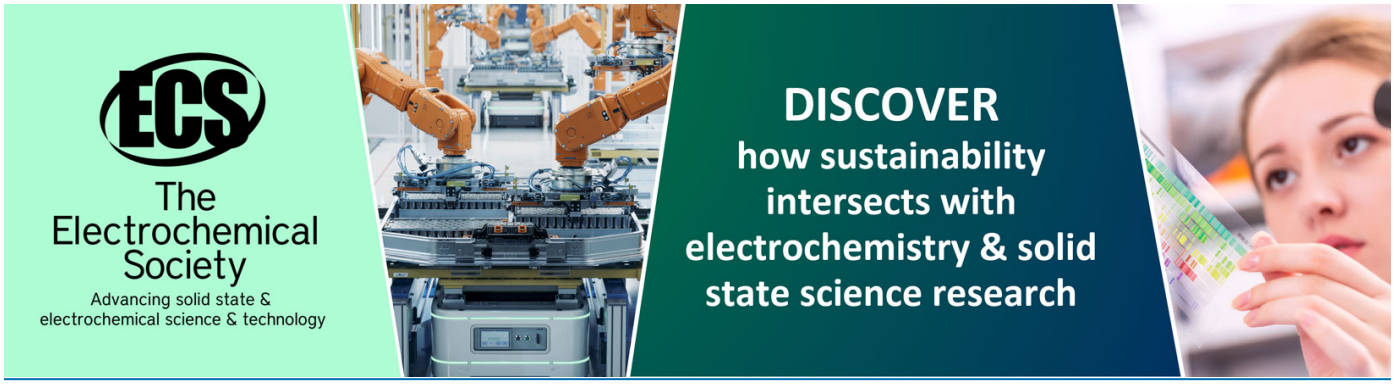
The Cloud Area Padovana: from pilot to production



IOP Conf. Series: Journal of Physics: Conf. Series **898** (2017) 052007 doi:10.1088/1742-6596/898/5/052007

The Cloud Area Padovana: from pilot to production

P Andreetto[1], F Costa[1], A Crescente[1], A Dorigo[1], S Fantinel[2], F Fanzago[1], M Sgaravatto[1], S Traldi[1], M Verlato[1] and L Zangrando[1]

[1] INFN - Sezione di Padova, Via Marzolo 8, 35131 Padova, Italy
[2] INFN Laboratori Nazionali di Legnaro, Viale dellUniversità 2, 35020 Legnaro (Padova) Italy

E-mail: cloud@lists.pd.infn.it

Abstract. The Cloud Area Padovana has been running for almost two years. This is an OpenStack-based scientific cloud, spread across two different sites: the INFN Padova Unit and the INFN Legnaro National Labs. The hardware resources have been scaled horizontally and vertically, by upgrading some hypervisors and by adding new ones: currently it provides about 1100 cores. Some in-house developments were also integrated in the OpenStack dashboard, such as a tool for user and project registrations with direct support for the INFN-AAI Identity Provider as a new option for the user authentication. In collaboration with the EU-funded Indigo DataCloud project, the integration with Docker-based containers has been experimented with and will be available in production soon. This computing facility now satisfies the computational and storage demands of more than 70 users affiliated with about 20 research projects.

We present here the architecture of this Cloud infrastructure, the tools and procedures used to operate it. We also focus on the lessons learnt in these two years, describing the problems that were found and the corrective actions that had to be applied. We also discuss about the chosen strategy for upgrades, which combines the need to promptly integrate the OpenStack new developments, the demand to reduce the downtimes of the infrastructure, and the need to limit the effort requested for such updates. We also discuss how this Cloud infrastructure is being used. In particular we focus on two big physics experiments which are intensively exploiting this computing facility: CMS and SPES. CMS deployed on the cloud a complex computational infrastructure, composed of several user interfaces for job submission in the Grid environment/local batch queues or for interactive processes; this is fully integrated with the local Tier-2 facility. To avoid a static allocation of the resources, an elastic cluster, based on cernVM, has been configured: it allows to automatically create and delete virtual machines according to the user needs. SPES, using a client-server system called TraceWin, exploits INFN's virtual resources performing a very large number of simulations on about a thousand nodes elastically managed.

1. Introduction

At the end of 2013, INFN Padova division and Legnaro National Laboratories (LNL) decided to launch a project for the provision of a computational and storage cloud service.

In summary the identified goals were:

- to provide a facility aimed to satisfy the computing needs that can not be easily addressed by the existing worldwide Grid infrastructures;
- to provide a pool of resources that can be easily and efficiently shared among all the relevant stakeholders;

- to limit the deployment of private clusters, which have a very high cost in terms of administration, and whose resources are often poorly used (often under-utilized, while insufficient to meet the usage peak requests concentrated in short periods).

This led to the implementation of the Cloud Area Padovana [1]: a single OpenStack [2] based cloud whose resources are spread across the two INFN sites.

After a prototyping phase where the infrastructure was set up and validated by a few pilot applications, the cloud facility was declared production ready and made available to all users at the end of 2014.

Since then the Cloud Area Padovana infrastructure has been kept evolving. New hardware resources were added (now the Cloud Area Padovana provides about 1100 cores) and new capabilities and functionalities were introduced, also with some in-house developments.

Also the use of such infrastructure has been increasing, in terms of number of users and experiments using the Cloud, and of the actual resource usage.

The increased workloads on the Cloud infrastructure also raised a few problems, not spotted before, that had to be handled also revising some design and implementation choices.

This paper is organized as follows. Sect. 2 provides an overview on the characteristics and architecture of the Cloud Area Padovana. Sect. 3 discusses about some applications run on such Cloud infrastructure. In sect. 4 we discuss about the operations of the Cloud, focusing on some lessons learnt. Sect. 5 concludes the article.

2. Overview of the Cloud Area Padovana

The Cloud Area Padovana is a Cloud service which seamlessly integrates resources spread in two different sites 10 km far away: INFN Padova and INFN National Laboratories.

The high level view of the Cloud Area Padovana is shown in Figure 1, where also the network layout is described.

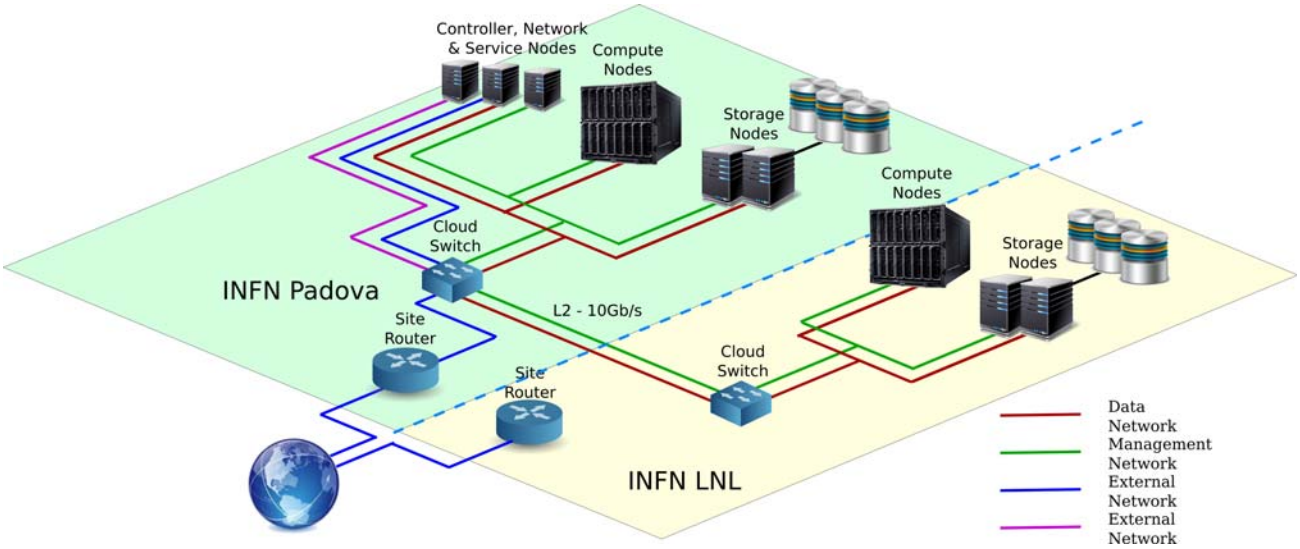


Figure 1. Layout of the Cloud Area Padovana

Compute nodes have been installed in both sites: 15 compute nodes in Padova and 13 compute nodes at INFN-LNL. They globally provide about 1100 cores, available for virtual machines or containers instantiated by the end users. The OpenStack services (and all the relevant ancillary services) were instead deployed only in Padova.

The two sites are connected by a dedicated 10Gbps network link.

Cloud storage is provided by an iSCSI storage server installed in Padova: it provides about 45 TB of disk space.

Besides the "core" OpenStack services (Keystone for identity, Glance for managing images, Nova for compute, Neutron for networking, Horizon for the dashboard, Cinder for the permanent block storage) the following optional services were deployed:

- heat, to provide an orchestration engine, supporting the deployment of complex applications on the Cloud;
- ceilometer: this is used to collect accounting information about resource usage; as explained later the retrieval of such information is instead managed through an in-house developed tool;
- EC2-service: this was deployed to provide an Amazon EC2 [3] compatible interface to manage instances, as requested by some applications;
- nova-docker: this allows the instantiation of docker containers with the very same tools and procedures used to create virtual machines. The assessment and testing of such service (which is not part of the official OpenStack distribution) was done in the context of the INDIGO-DataCloud project [4].

The OpenStack dashboard was also enhanced, including a software module to manage the registration of users and projects, as shown in Figure 2. Extensions to the OpenStack identity and dashboard services were also implemented, to integrate with SAML based identity providers. In particular the integration with the *INFN Authentication and Authorization Infrastructure* (INFN-AAI) was implemented, so that INFN users can authenticate to the Cloud Area Padovana using the same procedure used to access other INFN services.

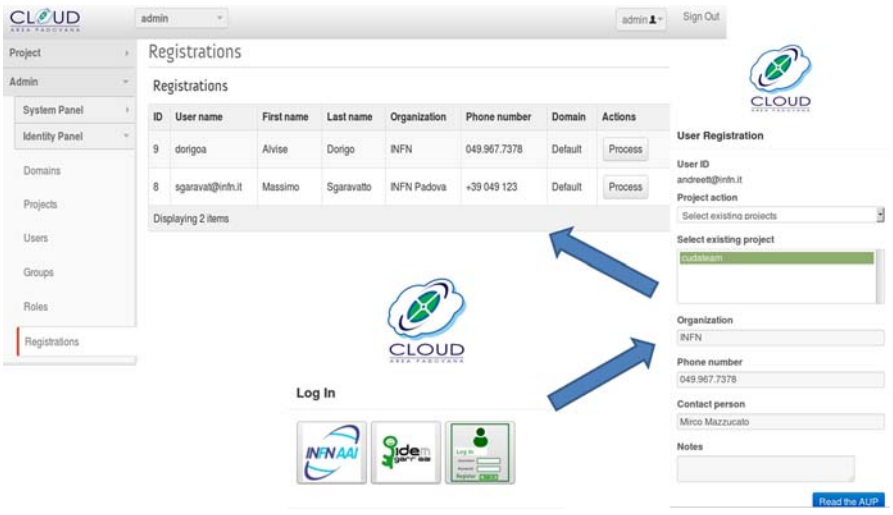


Figure 2. The customizations to manage user authentication and registration

For what concerns networking, OpenStack Neutron with Open vSwitch and GRE (Generic Router Encapsulation) [5] is used. By using two provider routers (one with the external gateway on the public network and one with the external gateway towards the internal LAN) instances can alternatively be given a floating (public) IP and be accessed from outside, or they can be reachable from the Padova and Legnaro LANs using their private IP.

Figure 3 shows how the relevant services were deployed in the Cloud Area Padovana.

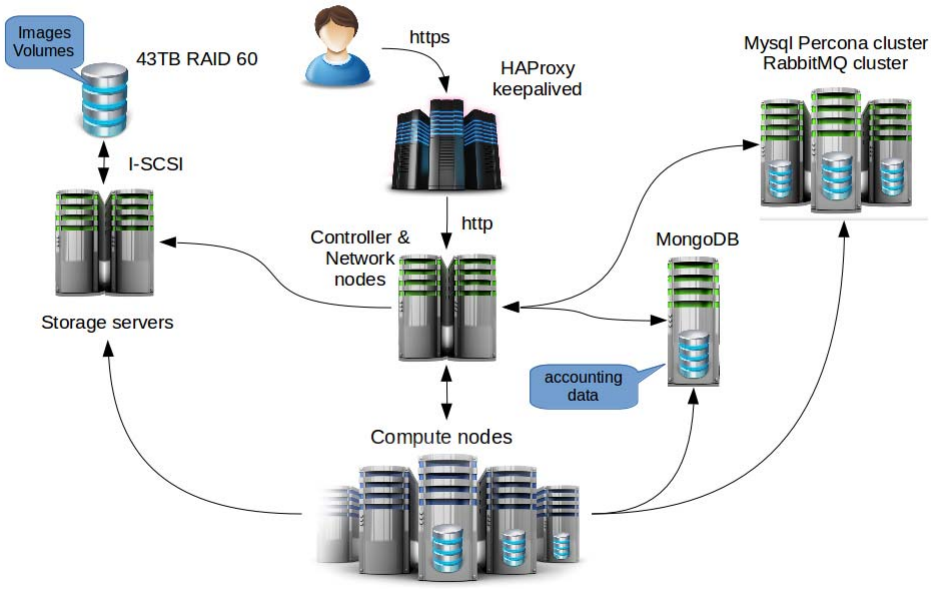


Figure 3. Architecture of the Cloud Area Padovana

The OpenStack services were deployed in high availability mode on two nodes, which act as both controller and network nodes. The high availability, in active-active mode, is implemented through a HAProxy [6] - Keepalived [7] cluster composed of three instances (these are VMs running on a Proxmox cluster).

HAProxy is also used to expose SSL interfaces for the OpenStack endpoints. Requests to the OpenStack services, generated from other OpenStack services or coming from generic clients, are encrypted and targeted to HTTPS URIs. HAProxy is responsible for extracting the plain payload and forwarding it to the OpenStack APIs on the controller nodes which actually use a plain HTTP protocol.

The relational databases needed by the OpenStack services are hosted on a Percona XtraDB cluster [8], composed of three instances. The same nodes are also used for the AMQP (messaging) service: the RabbitMQ implementation was chosen.

Accounting information, collected by the Ceilometer service, are stored on a single MongoDB instance. Because of scalability and performance problems with the ceilometer APIs, the retrieval of such accounting data is implemented using an in-house developed tool called CAOS, which extracts this information interacting directly with the database. CAOS also manages the presentation of such accounting information (e.g. to show the wall clok time and CPU time consumed by each project in a given time period).

The Cloud service storage (provided by an iSCSI box) has been configured using GlusterFS [9]. It is exposed through two storage servers. This is used for the OpenStack image service (Glance), for the block storage service available to the Cloud users (Cinder service) and, just for a couple of compute nodes, for the ephemeral storage of the virtual machines (the other compute nodes use instead the local disk devices).

3. Usage of the Cloud Area Padovana

At the time of writing this article, about 100 users are registered in the Cloud Area Padovana. They belong to about 30 projects: each project maps to an experiment or another research group.

They use the Cloud in different ways: interactive access (e.g. for analysis jobs, code development and testing, etc.), batch mode, etc.

CMS and SPES are at the moment the main customers of the Cloud Area Padovana.

The users of the local CMS group use on the Cloud some instances for interactive activities, such as running analysis and production jobs or simply submitting jobs to the Grid.

Experiment software is taken from CVMFS, while user data are read/written from/to a Lustre filesystem (which is also used by other CMS local resources). The CMS project on the Cloud is integrated with the local Tier-2 network, to allow an efficient access to the dCache storage system.

As shown in Figure 4, CMS also uses the Cloud resources in batch mode: using the elastiq [10] system, a HTCondor [11] batch system is created on the Cloud and it is then used to run user jobs. The size of this cluster is automatically scaled up or down as needed: worker nodes are created when jobs are waiting in the queue and are eventually destroyed when they don't have jobs to run.

One example showing how the Cloud helped the activities of the local CMS group is the generation of about 50k pseudo-experiment (toy Monte Carlo) events, followed by unbinned extended maximum likelihood fits, needed for the definition of confidence intervals for the angular parameters of the rare decay $B_0 \to K^\star \mu\mu$ analysis. This was done executing in total about 50k batch jobs in the HTCondor based elastic cluster, running simultaneously up to 750 jobs (using virtual machines with 6 Virtual CPU).

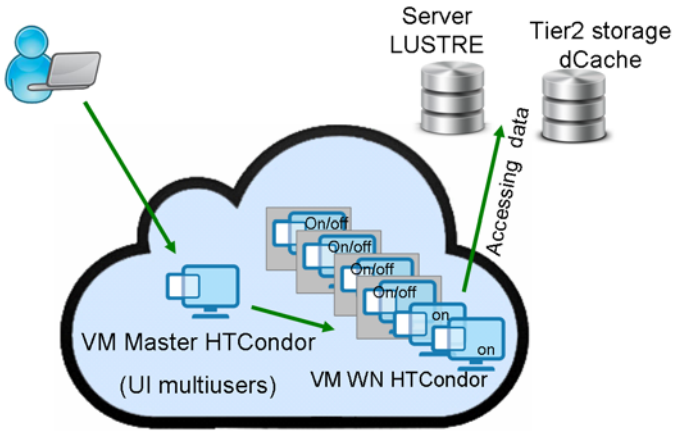


Figure 4. CMS usage of the Cloud

The SPES experiment is instead using the Cloud to address some modeling computational requirements. In particular the Cloud Area Padovana supported the activity of Beam Dynamics characterization of the European Spallation Source - Drift Tube Linac (ESS-DTL), that is mandatory to the SPES characterization. For this task, 100k different, 39m long, DTL configurations, based on Monte Carlo simulations (each one with 100k macroparticles), were performed. The configurations were split into 10k groups, each one resolved by 2k parallel job running on the cloud in batch mode. A framework, called TraceWin, was used for this use case: a TraceWin master service is instantiated on a Virtual Machine and then a number of

TraceWin clients are elastically instantiated on the Cloud: each client receives tasks coming from the master, and execute them. Up to 500 cores were used simultaneously.

4. Operations and lessons learnt

In this section we elaborate on some aspects of the operations of the Cloud Area Padovana. We also discuss about some lessons learnt, highlighting some design/implementation choices that had to be revised.

4.1. Evolution of the Cloud Area Padovana architecture

The architecture of the Cloud Area Padovana, previously described, is not the one originally envisioned when the infrastructure was planned.

We started with a configuration where cloud controllers and network nodes were deployed on different hosts, in high availability (two nodes for the controllers, other two hosts for the network nodes). The same hosts used as network nodes were used also as storage servers.

We found that mixing storage with other services was not a good idea (e.g. when a network node had to be rebooted, this had an impact also on the storage exposed by that node). Moreover, according to our experience, deploying the controller and network nodes on different hosts is not strictly needed.

This is why we decided to change the architecture: now two servers are used just for the storage services, and other two nodes are used as controller-network nodes, in active-active high availability mode.

We also had to revise the configuration of the Percona database cluster. Originally this was hosted on three virtual machines, but, for performance reasons, it was then migrated to three physical machines. The access to this database by the OpenStack services had to be properly tuned. In particular different primary instances are used by the different services.

4.2. Ephemeral storage for virtual machines

In the first setup of the Cloud Area Padovana, all the compute nodes used a shared file system for the nova service (i.e. for the ephemeral storage of the hosted virtual machines). This was provided, as a GlusterFS file system, by the iSCSI storage server. This was done because the possibility of live-migrating virtual machines was considered a very important functionality.

Unfortunately we experienced some scalability and performance problems with such setup: some particular workloads stressed too much the storage system, impacting the whole infrastructure.

Moreover we found out that, only for a very small subset of the applications deployed on our Cloud, the possibility of live migrating instances without any service interruption was considered a must-have functionality.

For these reasons we decided to consider a different setup: now most of the compute nodes use their local storage disks for the nova service. Only a few compute nodes use a shared file system, and therefore for their hosted instances live migration is supported. These compute nodes are targeted to host critical services (and are exposed as an ad-hoc availability zone).

4.3. Installation and configuration

According to our experiences (also related to the operations of other computing infrastructures) any manual configuration should be avoided. This is important in particular to limit the chance of operator mistakes, which can lead to problems that can be hard to debug.

We chose Puppet [12] and Foreman [13] as configuration and provisioning tools for the Cloud Area Padovana. Puppet in particular is used not only to configure OpenStack, but also all the other deployed additional services.

These tools also proved to help in reducing the cost of running the infrastructure. E.g. a new compute node of our infrastructure can be very quickly (re)installed and (re)configured in a completed automated way.

4.4. Security auditing

Security auditing is particularly challenging in a Cloud environment, since the relevant instances are dynamically created and destroyed. This is even more complex in the Cloud Area Padovana, because of its peculiar network setup.

By using some specific tools (in particular the ulogd [14] software) and by archiving all the relevant log files, we are now able to trace any internet connections initiated by instances on the Cloud and e.g. to find the author of a security incident, even if in the meantime the relevant Cloud instance was deleted.

4.5. Monitoring

In order to prevent or at least early detect problems, most of the systems and services of the Cloud infrastructure are monitored.

Several tools are used.

Ganglia [15] is used to monitor the system parameters of all servers.

Nagios [16] is used in particular to check the functionality and the performance of the Cloud services. E.g. Nagios is used to test the registration of new images, the instantiation of new virtual machines, their network connectivity, etc. While the most generic Nagios sensors are available on internet, we had to implement some specific plugins to perform some specific checks tailored to our infrastructure.

Cacti [17] is also part of our monitoring suite: it is used in particular for network related information.

4.6. Managing updates

Every change done on the production cloud is first tested and validated on a testbed. This is a small infrastructure which however resembles the production one (i.e. there are two controller-network nodes where services are deployed in high-availability, there is a percona cluster, etc.). The monitoring sensors are also active on such testbed, to be able to test the applied changes.

For what concerns the updates of the OpenStack middleware, one OpenStack update per year (i.e. skipping one OpenStack release) is done. We think this is a right balance between having latest features and fixes, and the need of limiting the manpower needed for such updates.

We are currently running the Mitaka version of OpenStack.

5. Conclusions and future work

We discussed in this paper about the Cloud Area Padovana which, after a period of prototyping, is now a fully production service.

A hundred users registered to such cloud service. They are using the infrastructure in different ways to meet their computing needs, also using some high level tools to efficiently leverage the available resources.

The increased usage of the infrastructure revealed some problems that had to be properly addressed, also revising some design and implementation choices. The infrastructure keeps evolving also in terms of provided resources and offered services.

Activities foreseen for the next future include:

- Deployment of a ceph [18] based storage service. We aim to use it for all Cloud related storage needs.

- Deployment of the Synergy service [19] (developed in the context of the Indigo-DataCloud project). This will allow to efficiently share the resources among all the relevant user groups, without the need of a static partitioning.
- Integration of the Cloud Area Padovana with the Cloud infrastructure owned by the University of Padova [20].

References

[1] Aiftimiei C et al., *Implementation and use of a highly available and innovative IaaS solution: the Cloud Area Padovana*, J.Phys.Conf.Ser. 664 (2015) 022016
[2] Home page for the OpenStack project, http://www.openstack.org
[3] Amazon EC2, https://aws.amazon.com/ec2/
[4] Salomoni D et al., *INDIGO-Datacloud: foundations and architectural description of a Platform as a Service oriented to scientific computing*, arXiv:1603.09536v3 [cs.SE]
[5] Hanks S et al., *Generic routing encapsulation (GRE)* (2000). http://tools.ietf.org/html/rfc2784.html
[6] Tarreau W, *HAProxy-The Reliable, High-Performance TCP/HTTP Load Balancer*, http://haproxy.1wt.eu
[7] Cassen A, *Keepalived: Health checking for LVS and high availability*, (2002), http://www.linuxvirtualserver.org
[8] Percona XtraDB Cluster, https://www.percona.com/software/mysql-database/percona-xtradb-cluster
[9] The Gluster web site, http://www.gluster.org/
[10] https://github.com/dberzano/elastiq
[11] Thain D, Tannenbaum T, and Livny M, *Distributed Computing in Practice: The Condor Experience*, Concurrency and Computation: Practice and Experience, Vol. 17, No. 2-4, pages 323-356, February-April, 2005.
[12] Puppet Labs home page, https://puppetlabs.com/
[13] Foreman home page, http://theforeman.org/
[14] The netfilter.org ulogd project home page, https://www.netfilter.org/projects/ulogd/
[15] Massie M, Chun B, Culler D, *The Ganglia Distributed Monitoring System: Design, Implementation, and Experience*, Journal of Parallel Computing, vol. 30, no. 7, July 2004.
[16] Barth W, *Nagios. System and Network Monitoring*, No Starch Press, u.s (2006)
[17] Cacti home page, http://www.cacti.net/
[18] http://ceph.com/
[19] Zangrando L et al., *Synergy: a service for optimising the resource allocation in the cloud based environments*, in Proc. of International Symposium on Grids and Clouds (ISGC) 2016, PoS(ISGC 2016)032
[20] Mazzon P E et al., *Progetto CloudVeneto.it - Status Report*, University of Padova Technical Report, May 2016.