Consortium GARR
THE ITALIAN EDUCATION & RESEARCH NETWORK

White Paper

GARR CLOUD: a Federated and Replicable Model for the Provisioning of Community Cloud Services

June 2022

White Paper

GARR Cloud: a Federated and Replicable Model for the Provisioning of Community Cloud Services

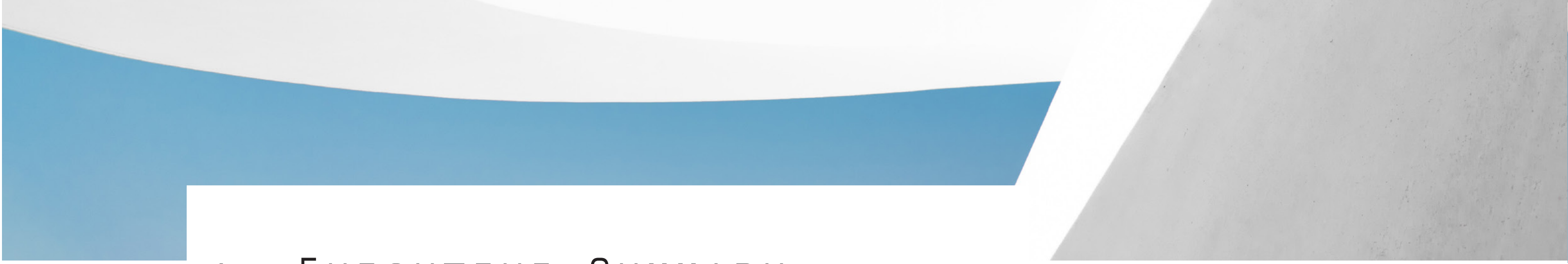
DOI 10.26315/GARR_Federated_Cloud_WhitePaper

Authors:
V. Ardizzone[1], G. Attardi[2], A. Barchiesi[3], A. Colla[4], M. Di Fazio[5],
R. di Lallo[6], F. Galeazzi[7], M. Lorini[8], D. Passalacqua[9], C. Pisa[10], M. Reale[11],
F. Ruggieri[12], D. Vaghetti[13]

Abstract:
This paper presents the current model of the GARR federated Cloud starting from the motivations and going through the strategic and technical choices that were made.
A brief description of the provided services and applications is also included.

1 EGI Foundation - valeria.ardizzone@egi.eu - orcid 0000-0001-7513-8678
2 University of Pisa - orcid.org/0000-0003-3875-6404
3 GARR - orcid 0000-0002-1170-9985
4 GARR - orcid 0000-0002-8248-4103
5 GARR - orcid 0000-0002-2229-3815
6 GARR - orcid 0000-0002-6098-7455
7 GARR - orcid 0000-0002-6830-9982
8 GARR - orcid 0000-0002-2877-7749
9 GARR - orcid 0000-0003-3452-0736
10 GARR - orcid 0000-0003-3452-0736
11 GEANT - mario.reale@geant.org - orcid 0000-0002-3502-258X
12 GARR - orcid 0000-0002-1013-3929
13 GARR - orcid 0000-0003-2337-549X

Index



1. Executive Summary

Distributed computing platforms have always been used to fulfil the requirements of geographically distributed research communities willing to collaborate effectively on common data sets, tools and applications. The GARR Cloud computing infrastructure is motivated by both internal needs of flexible computing resources and the opportunity to serve at the national level a large number of researchers in different disciplines and with various requirements.

A further motivation for the GARR Cloud is to leverage the platform to build a community of Cloud experts, able to create, develop and maintain in-house part of the infrastructure, to share expertise, a common vision and roadmap, and to provide users with a complete alternative to commercial solutions.

This motivation led to the choice of Open Source software as the foundation of the platform.

The Federated Cloud developed by GARR uses Open Source solutions and customised configurations to provide a number of cloud based services spanning from simple IaaS (Infrastructure as a Service) to different flavours of PaaS (Platform as a Service) and SaaS (Software as a Service). An additional service layer, combining IaaS and PaaS features, named Deployment as a Service (DaaS), was included to deploy specific applications.

GARR does not aim to become a single large-scale provider of cloud resources for the whole research community in Italy. We believe a Federated Cloud infrastructure, built around the model described in this document, can take this role, realizing a complex and rich infrastructure, operated and maintained by multiple entities, where computing and data services are accessible to a wide range of research activities.

The vision at the basis of the GARR Cloud is twofold:

- on one hand, we aim to provide Cloud services for the research community, with special focus to the small research groups that do not have enough expertise or manpower to develop those services on their own (long tail of science);
- on the other hand, we aim to become a resource aggregator, i.e. to create a reference architecture for the research Institutes that aim to build Cloud platforms in their data centres and eventually join together in a federation.

In relation to this twofold vision, currently the cloud end users are both researchers and organizations.

Therefore on the users side the GARR federated cloud provides CPU and storage resources and higher level services to a spectrum of users ranging from IT specialists to Biomedical researchers. On the organization side the federated approach offers a complete reference architecture and deployment model to virtualize and add a region of resources into the federated Cloud. This approach has already been adopted by a few Universities and Research Organisations.

In the following chapters we briefly expose the path followed and the current state of the art of the GARR Federated Cloud infrastructure, broadly designed as a "Community Cloud". We believe it better serves the needs of the Universities and Research communities. We thus present the general architecture with the developed services and an overview of the use cases behind them.

Last but not least, in this document the organizational aspects related to the activity of developing and maintaining a cloud infrastructure are underlined. A valuable side-effect of the realization of the GARR Federated Cloud is that it helps the Community to acquire and maintain knowledge of this technology from hands-on experience, and supports the development of specialised personnel with specific skills, thus preventing brain drain to occur.

Finally some considerations on the position of the GARR Federated Cloud in the European context and in connection to the European Open Science Cloud (EOSC) initiative are exposed.

2. Introduction and motivation

Since the beginning of the 2010's, GARR decided to optimise the computing infrastructure for its own services and to provide the Community of its users with custom services developed and maintained on their own infrastructure.
There are several motivations and benefits to proceed towards a "Community Cloud" solution:

- **Keeping competences inside the community**: investing on competences and empowering the staff with the knowledge to introduce new technologies that fulfil the users' requirements and allow to evaluate and compare the contents of commercial offers;
- **Tailoring the Cloud to community needs**: commercial offers provide only a price list for off-the-shelf solutions, while customisation comes as a separate and expensive support service;
- **Keeping the services on the R&E network**: not using third-party clouds helps avoiding the need for high-capacity peering with commercial providers and other bottlenecks;
- **Giving more space to community requirements**: Software tools dedicated to data sharing and collaboration can be evolved free from commercial constraints and under the control of the research community, minimising vendor lock-in risks;
- **Investing on the community expertise**: collaborate in the evolution of the cloud environment through the sharing of best practices and support to the newcomers;
- **Obtaining more value for money**: if the use of resources are optimised to reach continuos 24x7 usage, the cost of comparable commercial solutions is higher or comparable, but it implies much less flexibility;
- **Planning investments**: costs are related to investments (e.g. hardware) and operations (e.g. electrical power, personnel) that can be managed and planned, while a "pay-per-use" model of commercial services has no easy ways to be planned for the researchers and can possibly go out of control;
- **Keeping Data and Software sovereignty**: data and applications remain on storage and computing infrastructure located in the country, and in premises that are under the community's control.
- **Ensuring compliance to privacy and data protection regulations**: commercial providers based in non-EU countries (e.g. U.S.A.) can be subject to local legislation regarding privacy on data hosted on their servers, even if those servers are in other countries (i.e. the so-called CLOUD Act, Privacy-Shield framework). In community clouds, the data remain on the community's servers and it is under its direct control.

The roadmap to realise this vision was implemented through a new project proposed to and approved in July 2013 by the Italian Ministry of Education, University and Research, "GARR-X Progress", aiming at deploying a new network infrastructure based on optical fibres in the South of Italy. One of its planned actions was to deploy an ICT infrastructure distributed over 5 sites (Bari, Catania, Cosenza, Napoli, Palermo). The full infrastructure amounts to some 40k Cores and about 10 PB of raw disk storage organized in 11 racks distributed among the sites (3 each in Bari, Catania and Palermo; 1 each in Cosenza and Napoli) like shown in the following scheme.

During following years the GARR Cloud platform has been implemented and evolved on part of this ICT infrastructure.

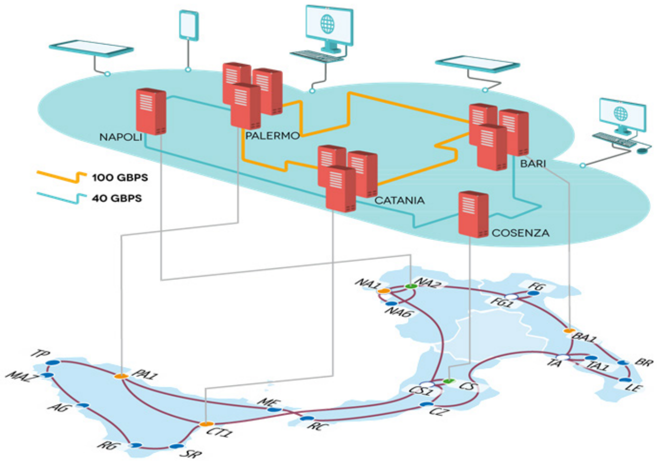


Figure 1 - GARR-X Progress distributed computing and data infrastructure

3. GARR Cloud Platform

The GARR Cloud spans across three separate geographical regions (Catania, Palermo, Napoli) interconnected through high capacity fiber optic links.

Users can login to the OpenStack dashboard using their federated identities (IDEM/eduGAIN), thus employing credentials from their home institutions.

Along with the OpenStack deployment, GARR operates a (PaaS) Kubernetes service featuring GPUs, which can be accessed with the same credentials as OpenStack.

3.1 Reference architecture

The architecture is based on the open-source OpenStack cloud technology, which is currently the de facto standard in this field.

OpenStack manages data center resources through a set of modular services which interoperate in a service-oriented architecture. Its components provide abstractions for computational, storage and networking resources and allow to provide them dynamically and securely to different tenants.

As it is typical with open-source solutions, OpenStack offers reduced investment and operational costs and large scalability without licensing limitations. However, building a cloud remains a complex task, which requires the integration of different components.

Using tools for automating the steps needed to create, implement and support a production-ready OpenStack cloud is key to simplify processes, reduce the required manpower, reduce risks, and keep services always up

to date. To this end, we implemented an automation solution based on two opensource tools developed by Canonical: Juju and MAAS .

3.1.1 Service Architecture

The logical service architecture is organized into four layers, shown in Figure 2:

- **Physical**: consists of the physical hardware and network resources
- **Operating System**: corresponds to the services provided by the operating system
- **Infrastructure Virtualization**: provides an abstraction of Operating System services through LXC containers. This choice allows abstracting from lower layers.
- **Application Services**: includes services provided by the automation stack. In the case of IaaS the service will be OpenStack itself; in the case of PaaS the service will be Kubernetes.

3.1.2 Orchestration and Automation

In this section we describe the automation techniques used for the installation, deployment and maintenance of the cloud platform and the deployed services, starting from the hardware resources.

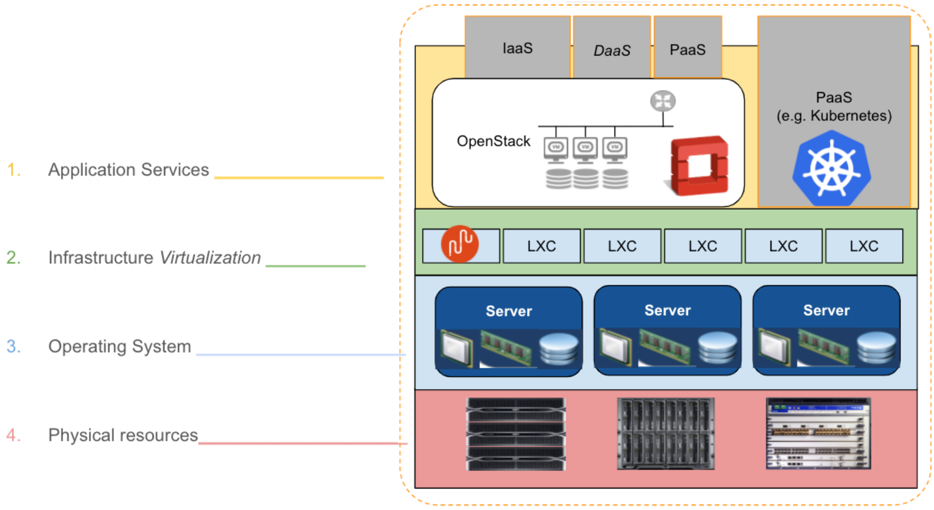


Figure 2 - GARR Cloud Service Architecture

MAAS

MAAS (Metal As A Service) is a tool that helps managing the physical infrastructure with the same ease and flexibility as virtual machines in the cloud. Specifically, MAAS allows to:

- Discover, commission and deploy physical servers;
- Dynamically allocate physical resources to match workload requirements;

- Release servers when they are no longer needed and make them available for new workloads as required.

MAAS is particularly suited for a distributed, multi-site environment such as the GARR infrastructure, thanks to its modular architecture: a "region controller" hosts the catalogue and the web interface, and more "rack controllers" installed on each datacentre host the tools to manage the local infrastructure.

MAAS is responsible for hardware discovery and for installing the OS, making basic hardware configurations and allowing servers to be recognized by network and system management software. When a new node boots up, MAAS steps in, supplies it with all the required information and provides an OS image to install. This is done via PXE and DHCP. MAAS provides both a web interface and an API to manage bare metal systems under its control. Moreover, it provides a Highly Available DNS for the application layer.

Juju

MAAS works in conjunction with Juju, a service orchestration tool allowing the administrator to configure, manage, maintain, deploy and scale cloud services (workloads) quickly and efficiently on public clouds as well as on bare metal, leveraging MAAS to control the hardware.

Juju uses interoperable operators called Charms to deploy and scale a variety of architectures, from simple workloads like a web server or a database, up to complex infrastructures such as OpenStack. Juju can be controlled via a web GUI, the command line, or APIs. Juju's core (the Juju controller) manages the lifecycle of the deployment: upgrades, integrations, management, and operations on the workloads.

Charms implement the intelligence necessary for connecting different applications together. These inter-application connections are called relations, and they are formed by connecting the applications' endpoints. Endpoints can only be connected if they support the same interface and are of a compatible role ("requires" to "provides", "peers" to "peers").

Bundles are collections of charms that link applications together. They enable administrators to deploy large, complex systems from one file, where all of the configuration and relations are pre-defined. Where a charm represents a single application or service, a bundle represents an entire Juju model. A bundle is fundamentally a YAML file that contains details of the charms, relationships and configuration to deploy.

Finally, Juju provides dynamic configuration, which allows re-configuring services on the fly, adding, removing, or changing relationships between services, and scaling in or out.

Upgrades and administrative operations

The automation layers provide also a powerful tool to reduce the administrative effort. In particular the possibility to upgrade and roll back proved important in the choice of the automation architecture, also in view of reducing the operation time. This optimisation works for the upgrades of both the Operating System layers (when all the control plane can be upgraded with few api call of clicks of the dashboards) and of the OpenStack layer itself.

3.1.3 Ceph

For the storage part, we rely on Ceph , the software-defined storage solution which has long been the de-facto standard for providing block and object storage to Openstack clusters. The main advantages of Ceph are its versatility (different possibilities for storage provisioning), scalability, lack of single points of failure, support by the community.

In our architecture, storage can be managed using the same tools (Juju/MAAS) used for the rest of the OpenStack cluster or other automation tool. The automation framework was refined after the CEPH cluster, so we still use Juju/MAAS only for later deployed Ceph clusters (and for test/development clusters), while for the big production Ceph clusters (in particular on the datacenters of Catania and Palermo) we relied on the automation provided by ceph-ansible . Once setup, the configuration of the Ceph cluster is rather stable and does not particularly benefit from the dynamics made possible by the mechanism of charm relations so we had no need of leveraging the automation layer for the first born CEPH clusters, but integrate each one of the distinct Ceph cluster per region, through the Ceph-proxy charm into the juju MAAS ecosystem.

3.2 GARR Cloud regions

Leveraging the federative approach described in Chapter 5, the GARR Cloud has been deployed on 3 datacentres (Palermo, Catania, Napoli), each one corresponding to an OpenStack **region** (Fig. 3)

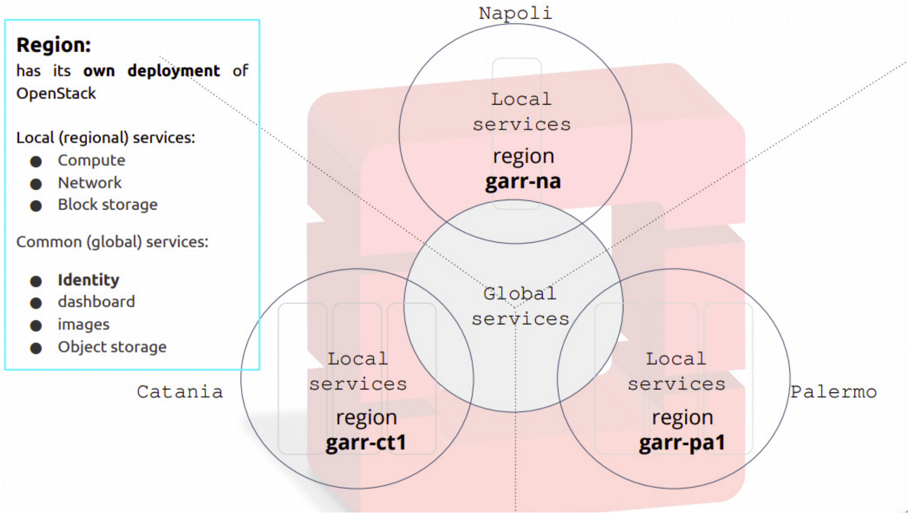
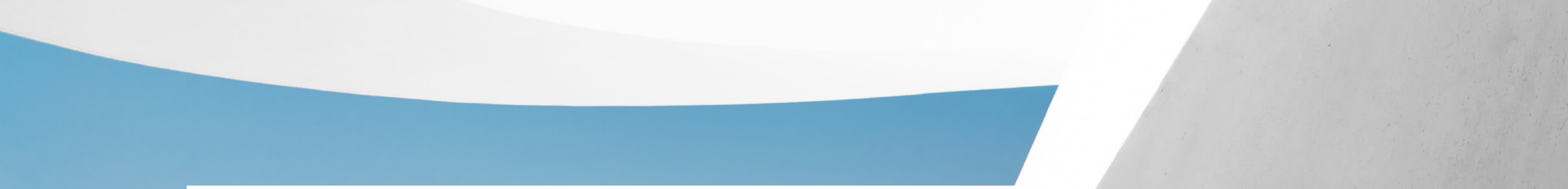


Figure 3 - GARR Cloud Regions

Each region hosts the OpenStack services managing compute, network and block storage resources. All regions share the GARR Cloud general services, i.e.:

- Identity management (User accounts, projects, roles)
- Images repository (the repository of Operating System images used to deploy Virtual Machines)
- Object storage
- The GARR Cloud dashboard



4. GARR Cloud Service Offering

4.1 Virtual Machines (IaaS)

The GARR cloud allows creating and managing Virtual Machines (IaaS) running in the data centres of the GARR Federated Cloud, interconnected through the GARR high speed fibre network.

Virtual machines belong to **projects**, i.e. organizational units in the cloud, to which the Cloud administrators can assign users and set **quotas** (i.e. predetermined maximum shares of computing, storage and networking resources). Users can be members of one or more projects. **Roles** define which actions users can perform. Cloud administrators assign roles to user/project pairs.

The OpenStack networking service handles the creation and management of a virtual networking infrastructure, including networks, switches, subnets, and routers. Advanced services such as firewalls or virtual private networks (VPNs) can also be used.

Each virtual machine has a private, fixed IP address and can also have a public, or floating IP address. Private IP addresses are used for communication between instances, and public addresses are used for communication with networks outside the cloud, including the Internet.

Security groups are sets of IP filter rules that are applied to all project instances, which define networking access to the instance. Group rules are project-specific; project members can edit the default rules for their group and add new rule sets. All projects have a default security group which is applied to any instance that has no other defined security group. Unless users change the **default**, this security group denies all incoming traffic and allows only outgoing traffic to their instances.

4.1.1 Virtual Datacentre

A Virtual Datacentre (vDC) consists of a set of virtual resources (vCPUs, memory, storage, networking) assigned to an administrator, who can manage them by creating VMs, virtual network architectures, and deploying services to them. The vDC administrator can also:

- create a tree of (sub)-projects within the vDC;
- enable registered users to use the resources in one or more sub project;
- distribute the total amount of resources assigned to the vDC (by the Cloud administrator) among the sub-projects.

4.1.2 Storage

The GARR Cloud provides an object storage service, which allows to design scalable distributed applications and to simplify backup strategies. Users can upload and download objects, which are replicated on the GARR Cloud storage, leveraging their connection to the GARR high speed symmetric network. To ensure reliabiliy, object storage relies on erasure-coded Ceph data pools.

Storage for volumes attached to virtual machines (block storage) is also provided via Ceph. To ensure reliability, block storage relies on replica-3 pools.

4.2 Deployment as a Service (DaaS)

Based on Juju, the same orchestration tool used for the deployment of the GARR Cloud itself, the DaaS platform allows Cloud users to declaratively deploy and scale-out pre-built applications, consisting of a single service or a combination of interacting ones. Through DaaS, users set up and launch applications and services directly on their virtual data centers, leveraging a huge public catalogue of community built applications, that GARR and its community contribute to expand. Juju takes care of the deployment and configuration of the virtual machines (OS installation, networking, access security, etc), while the user maintains full administrative control on the VMs dedicated to the services. Among the applications available on the DaaS (Juju) store we mention:

- Kubernetes;
- CMS: WordPress, Joomla, Drupal, Moodle;
- Big Data: Hadoop, Spark;
- Collaboration Tools: Mattermost (Slack alternative);
- Stacks: LAMP Stack;
- Storage: MongoDB, MySQL, Cassandra, OwnCloud.

4.3 GARR Container Platform (PaaS)

The GARR Cloud Container Platform is an environment for automating the deployment, scaling, and management of containerized applications, based on Kubernetes.

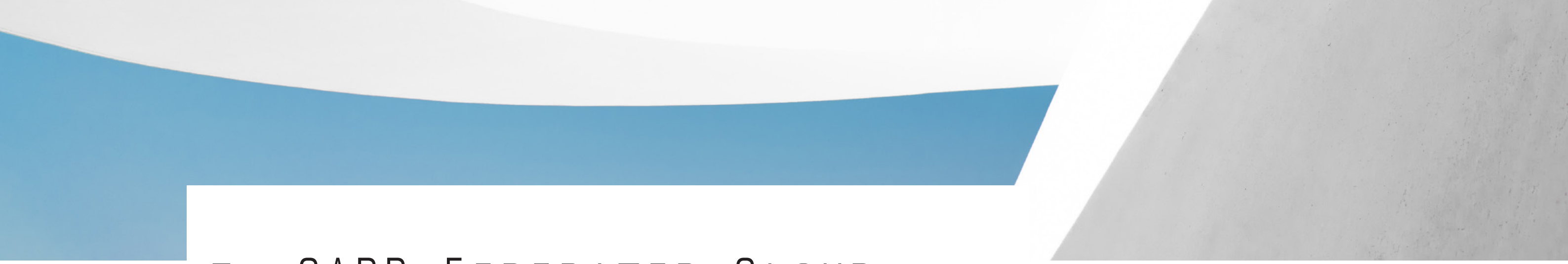
It enables rapid application development and iteration by making it easy to deploy, update, and manage users' applications and services. Users simply need to describe the compute, memory, and storage resources their application containers require, and Kubernetes provisions and manages the underlying cloud resources automatically. An additional provided functionality is the possibility to attach persistent storage to instances. Support for hardware accelerators enables running Machine Learning, General Purpose GPU, High-Performance Computing, and other workloads that benefit from specialized hardware accelerators.

The GARR Container Platform is based on CDK (the Canonical Distribution of Kubernetes). The current release has been deployed with MAAS and Juju on bare metal servers (including a server with GPU).

To allow a better user experience and avoid account duplication, the GARR Container Platform authenticates to the GARR Cloud Platform's Keystone authentication server. To this regard, the GARR Cloud department has developed, in collaboration with SWITCH (the Swiss NREN, https://www.switch.ch/), an authentication mechanism that leverages Kubernetes webhooks and OpenStack application credentials. The result is that GARR Cloud users, in a few clicks through the OpenStack dashboard, can download a Kubernetes configuration file to access their own environment (user namespace).

Leveraging the DaaS layer, users can then easily deploy (or ask GARR Support to do so) fully functional Kubernetes platforms directly on their Cloud projects/vDCs, on which they will have full administrative privileges.

Our roadmap foresees the migration of the bare metal Container Platform to a "virtual" one (built on OpenStack), maintaining the current multi-tenant, privilege-limited access.



5. GARR Federated Cloud

In addition to managing the GARR Cloud Platform, offering cloud services to the Italian academic and research community, GARR also coordinates a federation of clouds, located in national data-centres owned by members of the GARR community, which participate to the federation by sharing resources and services.

Moreover, the GARR Cloud Platform aims at setting an open, live and working model which can be replicated by other members of the research community.

GARR promotes the aggregation of national cloud infrastructures through a federated approach. Indeed, deploying and maintaining a cloud infrastructure is an endeavour which requires an initial substantial knowledge acquisition, followed by significant and continuous operational activities. The GARR Federated Cloud aims at relieving the burden of these tasks for the organizations who plan, for strategic or regulatory reasons, to retain the ownership of their infrastructure and of their know-how.

The GARR Federated Cloud approach reflects GARR's mission as infrastructure aggregator and harmonizer, taking inspiration from GARR's role for research and education networking infrastructures. In this perspective, GARR federated cloud team regularly organizes knowledge sharing sessions and tutorials that foster the community building. The objective is to foster the creation of an ecosystem that could retain all the knowledge at stake in the process of building such a complex infrastructures .

5.1 Region - Domain model

The proposed solution is based on a multi-region OpenStack model, which is practically feasible assuming to federate clouds designed and implemented in a coordinated manner, based on compatible OpenStack versions. The solution leverages the OpenStack mechanisms to scale to thousands of nodes and to expand onto different data centers and geographical areas, exploiting in particular the concept of Region.

Each Region has its own deployment of OpenStack, including endpoint API, network and computing resources, which is linked to other regions using shared centralized services such as OpenStack Identity and dashboard. The concept of regions is flexible; it may contain OpenStack service endpoints located within a distinct geographic area or areas, or it may be smaller in scope, where a region is a single rack within a data centre, with multiple regions existing in adjacent racks in the same data centre.

5.1.1 Central authentication

The authentication and authorization services are provided by a single central service that exposes a single API and a single web interface to the whole federated cloud. Since authentication itself is performed in a federated form, users are globally recognized and therefore can be enabled to use resources on the whole federation. The authentication system allows the use of credentials provided by Identity providers in the IDEM /eduGAIN federations as well as OpenID Connect providers.

5.1.2 Administrative delegation

The logic of the federation allows each organization to maintain control over the use of its own resources, dynamically setting which part to keep for private use and which part to give for sharing within the federation.

Technically this is done by associating to each region its own Keystone Domain[1], through which the local manager will administer resources as shown in the following scheme.

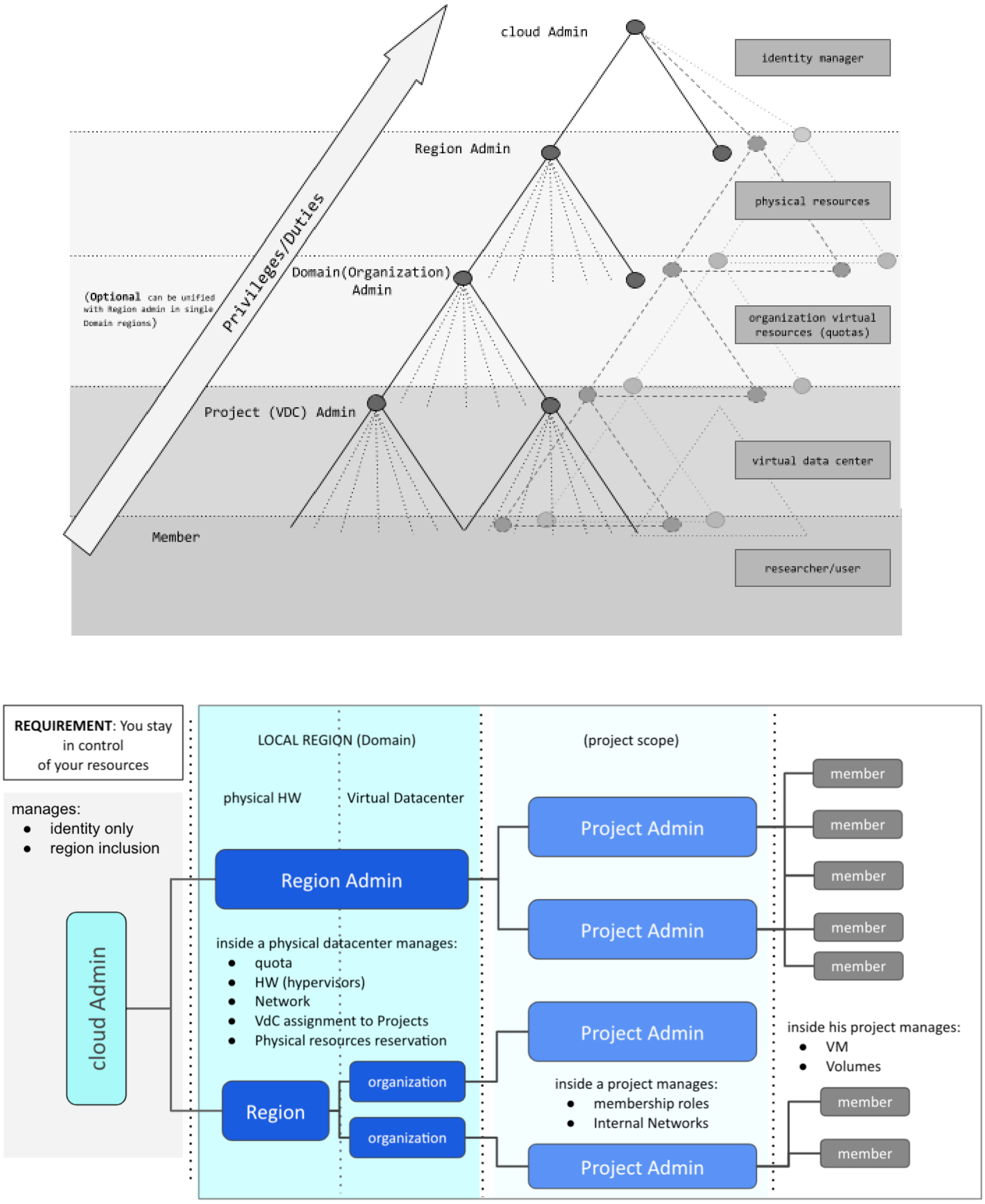


Figure 4 – Administrative delegation scheme

[1] Keystone is the OpenStack service that provides API client authentication, service discovery, and distributed multi-tenant authorization. Domains are high-level containers for projects, users and groups. As such, they can be used to centrally manage all keystone-based identity components

Administration authority is delegated as follows (for more details see table 1).

The Cloud Administrator can:

- Create/Update/Delete/List Regions and Region Administrators
- Create/Update/Delete/List Domains and Users
- Grant/Revoke/List Roles on Domains

Inside each Region the Cloud administrator delegates the following tasks to the relevant Region Administrators:

- Add/Remove/Activate/Deactivate Hypervisors (compute nodes) within their region
- Add/Remove External networks within their region
- Grant/Revoke/List Roles to Project administrators (inside their Region)
- Set quotas to top-projects (vDC)
- Create/Update/Delete/List (Virtual Datacenter) Projects within their Domains
- Grant/Revoke/List Roles to Users within their Domain
- Grant/Revoke/List Roles on Projects within their Domain

🟩	Operation allowed on any resource
🟧	Operation allowed only on resources assigned to the role (e.g. a domain_admin can create a project only in his domain)
🟥	Operation forbidden

Identity		cloud admin	region admin	vDC admin	member
Domain	create	🟩	🟥	🟥	🟥
	update	🟩	🟥	🟥	🟥
	delete	🟩	🟥	🟥	🟥
	list	🟩	🟥	🟥	🟥
Project	create	🟥	🟧	🟧	🟥
	update	🟥	🟧	🟧	🟥
	delete	🟥	🟧	🟧	🟥
	list	🟥	🟧	🟧	🟧 SELF
	list user prj	🟥	🟧	🟧	🟧 SELF
	List users within prj	🟥	🟧	🟧	🟥
	get	🟥	🟧	🟧	🟧

User	create	🟩	🟥	🟥	🟥
	update	🟩	🟥	🟥	🟧 SELF
	delete	🟩	🟥	🟥	🟥
	list	🟩	🟧	🟧	🟥
	get	🟩	🟧	🟧	🟧 SELF
	change pass	🟩	🟥	🟥	🟧 SELF
Role	grant	🟩	🟧	🟧	🟥
	get	🟩	🟧	🟧	🟥
	list	🟩	🟧	🟧	🟥
	create	🟩	🟥	🟥	🟥
	delete	🟩	🟥	🟥	🟥
Group	Create, update, or delete				
	get				
	list				
	List groups for user				
	List users				
	add/remove users				

Compute		cloud admin	region admin	project admin	member
`flavour`	`create`	🟥	🟧	🟥	🟥
	`update`	🟥	🟧	🟥	🟥
	`delete`	🟥	🟧	🟥	🟥
	Set metadata	🟥	🟧	🟥	🟥
Quota	assign	🟥	🟧	🟥 X	🟥
	update	🟥	🟧	🟥 X	🟥
	list	🟥	🟧	🟧	🟧
Instance	create	🟥	🟧	🟧	🟧
	delete	🟥	🟧	🟧	🟧
	list	🟥	🟧	🟧	🟧
	start/stop	🟥	🟧	🟧	🟧
	Attach net	🟥	🟧	🟧	🟧
	Attach Vol	🟥	🟧	🟧	🟧
Hypervisor	enable	🟥	🟧	🟥	🟥
	disable	🟥	🟧	🟥	🟥
Host Aggregate	`create`	🟥	🟧	🟥	🟥
	`delete`	🟥	🟧	🟥	🟥
	`add/remove host`	🟥	🟧	🟥	🟥
	`Set metadata`	🟥	🟧	🟥	🟥
Availability Zone	`create`	🟥	🟧	🟥	🟥
	`delete`	🟥	🟧	🟥	🟥

Volume		cloud admin	region admin	project admin	member
Volume	create	🟥	🟧	🟧	🟧
	delete	🟥	🟧	🟧	🟧
	list	🟥	🟧	🟧	🟧
Snapshot	create	🟥	🟧	🟧	🟧
	delete	🟥	🟧	🟧	🟧
	list	🟥	🟧	🟧	🟧
Transfers	create	🟥	🟧	🟧	🟧
	delete	🟥	🟧	🟧	🟧
	list	🟥	🟧	🟧	🟧
Backups	create	🟥	🟧	🟧	🟧
	delete	🟥	🟧	🟧	🟧
	list	🟥	🟧	🟧	🟧

Image		cloud admin	region admin	project admin	member
Image	create	🟥	🟧	🟧	🟧
	update	🟥	🟧	🟧	🟧
	create	🟥	🟧	🟧	🟧
	delete	🟥	🟧	🟧	🟧
	list	🟥	🟧	🟧	🟧
	download	🟥	🟧	🟧	🟧
	upload	🟥	🟧	🟧	🟧
	publicize	🟥	🟧	🟥	🟥

Network		cloud admin	region admin	project admin	member
External Network	create	🟥	🟧	🟥	🟥
	delete	🟥	🟧	🟥	🟥
	list	🟥	🟧	🟥	🟥
Internal Network	create	🟥	🟧	🟧	🟧
	delete	🟥	🟧	🟧	🟧
	list	🟥	🟧	🟧	🟧
Router	create	🟥	🟧	🟧	🟧
	delete	🟥	🟧	🟧	🟧
	list	🟥	🟧	🟧	🟧

Table 1 - Schematic view of delegation of Administration authority

5.2 How to build a Federated Region

Any institutional member of the GARR Community can make part of their hardware resources available to the Cloud Federation. To this aim, the GARR team prepared a simple operational procedure. The only prerequisite is that the underlying network infrastructure is set up, e.g. physical resources are interconnected and have access to the Internet to perform network installation of the software.

The federation recipe consists of four steps:

- on the federating region side:

    - Installation and configuration of the automation tools (MAAS and Juju);

    - Registration of the physical resources in the MAAS catalogue servers, networks, VLANS, storage, etc.

    - Juju-driven deploy and configuration of the OpenStack cluster;

- on the federation identity region:

    - Injection of the region credentials and API endpoints through an encrypted mechanism that preserves the credentials security so to enable the region in the central Identity service.

6. GARR Cloud Status

The GARR Cloud platform is in production since 2017. It started with a single region (Catania). In 2018 the Palermo region was added, and in 2019 the third region, Napoli, joined. In October 2020 the first "external" institute, the Politecnico di Torino, joined the Federated Cloud integrating its datacentre as the fourth region. Finally, in December 2021 the fifth region joined, built on GARR hardware hosted at the datacenter of the University of Turin.

The GARR Cloud department is committed to maintaining the Cloud platform fully operational and keeping the software framework updated.

Since the beginning more than 1200 users registered to the platform (see fig.5), and at the time of writing we count more than 620 active users maintaining more than 1200 virtual machines (GARR regions only)

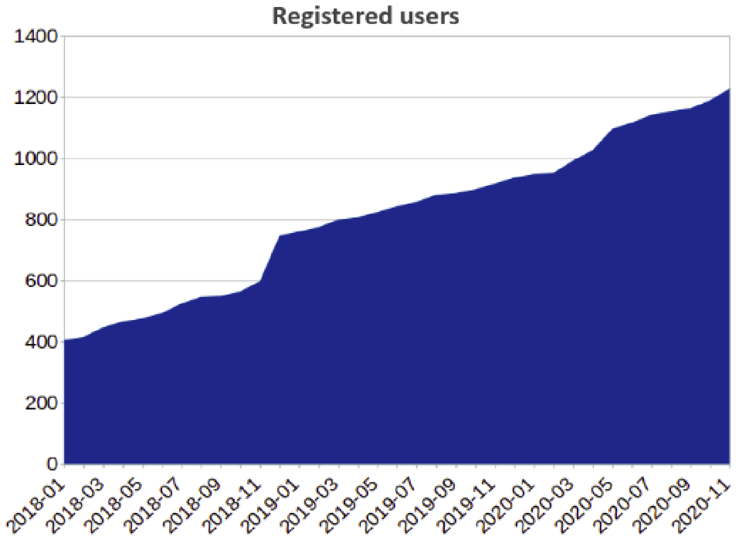


Figure 5 – Number of registered users as a function of time

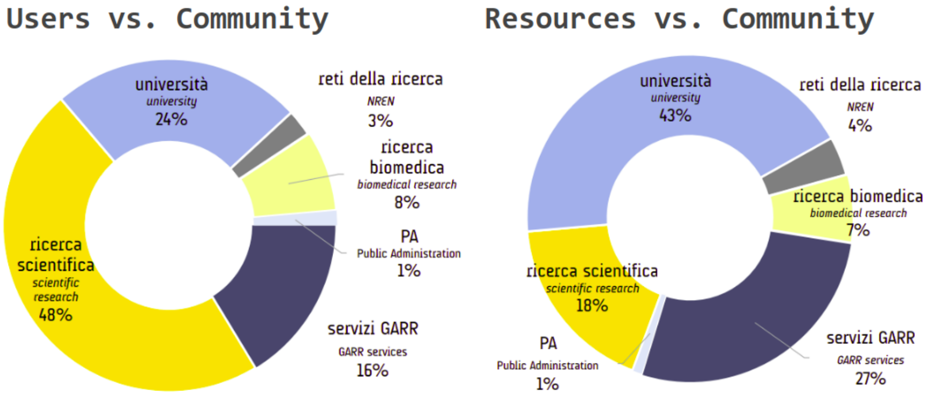


Figure 6 – Distribution of users and resources VS communities

6.1 User management

We typically deal with two kind of requests to access the GARR Cloud:

1. Users or research groups with an already defined project ask for a virtual Datacenter with resource quotas

tailored to their needs. There are currently more than 120 active vDCs.

2. Users who want to test the Cloud platform are given a "demo" project on the Cloud and a user namespace on the Container platform with limited resources. Unless otherwise agreed, demo projects are granted resources for a period of 6 months.

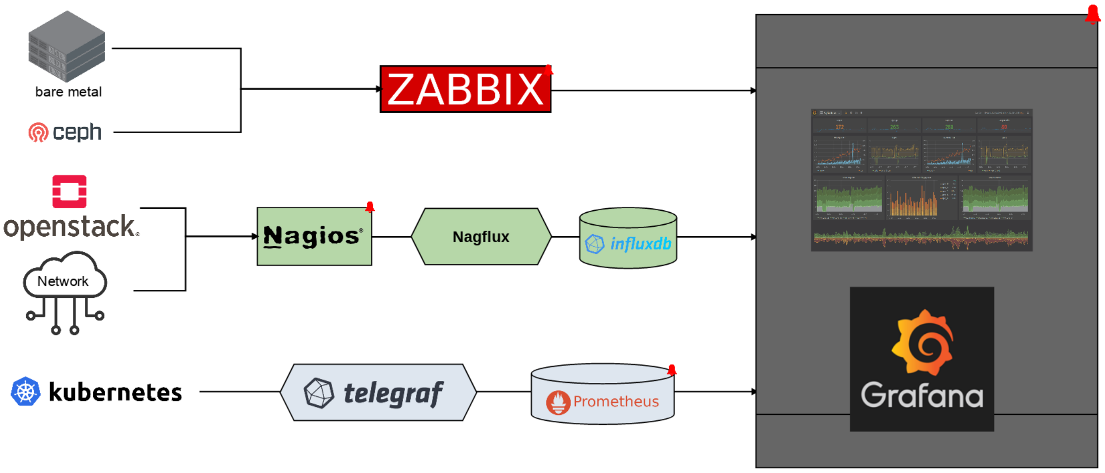
Users make a registration request to the GARR Cloud through the "signup" portal developed. The registration portal is available at https://cloud.garr.it/forms/register.

Cloud administrators take care of approving the request, and assign the user to the requested vDC or a demo project. For the latter case, user creation and assignment to a demo project and container platform namespace is done automatically with a single click on the administration page of the signup portal.

6.2 Monitoring

The monitoring of the GARR Cloud is performed on a region basis and relies on the combination of:

- Nagios and Zabbix: monitoring and alerting systems
- InfluxDB, a time series database
- Prometheus, a monitoring and alerting system
- Grafana, an interactive visualization web application



The Nagios monitoring and alerting system is well supported by OpenStack Juju Charms and can be configured to collect performance metrics. These metrics are processed by Nagflux which inserts the collected information into the InfluxDB time series database. Grafana can then use InfluxDB as a data source for visualization and analysis.Prometheus is instead well supported by Kubernetes. It provides its own internal time series database, which can also be queried by Grafana. Moreover, Grafana allows the definition of dashboards which use heterogeneous data sources.

This open monitoring model is designed to be replicated in each federated region.

7. GARR Cloud Use Cases

The GARR Cloud serves different use-cases. At the time of writing, users from the GARR Community are using the GARR Cloud for several applications: gravitational waves alert systems, artificial intelligence, computing platforms (e.g. Virtual Research Environments), open science (OpenAire, EOSC services), bioinformatics, seismic monitoring. This non-exhaustive list includes:

1. users of the actual GARR Cloud infrastructure, either through the OpenStack interface, the DaaS interface or the Kubernetes interface;
2. users from organizations which have joined the GARR Cloud federation with their own infrastructure;
3. users from organizations which are replicating the GARR Cloud model at their premises.

GARR itself is using the GARR Cloud for several applications, summarized below at time of writing.

i. The GARR.tv platform is the streaming service of audio / video content produced by the GARR community. It allows the uploading and management of video content (in on-demand mode), and the distribution of events in live streaming.

ii. GARR Workplace is a collaborative platform allowing users with strict regulatory requirements or concerns on data privacy and sovereignty, or on the geographical location of their data, to enjoy the advantages of a web office real-time collaborative platform, with a rich set of features and a high usability. GARR Workplaces is based on OnlyOffice: available as open source software, provides a multi-user online document editor which uses natively, and with 100% claimed compatibility, the Microsoft Office document formats, as well as tools for document management, communication (E-mail, CRM, calendar, chat, forums, blogs) and project management. It is designed to be integrated with pre-existing systems, as it features storage back-ends for OwnCloud, NextCloud, Google Drive, OneDrive, etc. and support for external authentication, through e.g. LDAP. As for the deployment, the preferred OnlyOffice installation method relies on a set of interconnected Docker-based microservices.

iii. Jira is a software suite that provides a ticketing system and tools to model and manage internal processes and projects. An instance of the Jira platform was installed on the GARR cloud in the second half of 2019, and made available to all GARR and offices departments.

iv. IdP in the Cloud is an Authentication and Authorization Infrastructure (AAI) service designed for organizations that lack both the expertise and the resources to run a full AAI on their own. The user base of the service is mostly composed of research hospitals and veterinary research centres, which have typically a relatively small number of end users (hundreds).

A glance at the "big picture" at European level and the EOSC

The development of the GARR Federated Cloud was inspired by GARR's mission and values: to name a few, to facilitate access to computing infrastructures to its users, co-create the best solutions to fit users' needs, engage the community in an active exchange of experiences and information.

The GARR Federated Cloud can also be regarded as an attempt to prevent knowledge drain. We believe this is extremely relevant for the research community because, while it is true that the IT world is changing at an

unprecedented pace and neither GARR nor most of its constituents have IT as their primary core-business, the risk of being a mere spectator of such changes is that expertise quickly fades and organisations soon become unable to govern the evolution of their own IT infrastructure and even become unable to properly compare different solutions.

A vision similar to GARR's one is shared by other European NRENs members of Géant, and the future project Géant5 will continue the multi-year effort in "Community Clouds development" which is ongoing since at least 10 years.

The GARR Federated Cloud is also relevant, we believe, in connection to the EOSC - European Open Science Cloud, although, at the moment, the EOSC is focusing on the "higher layers" of Open Science, such as interoperable services and FAIR data management, and is not directly involved with the lower e-Infrastructure layer. Nevertheless, we believe that a federated cloud model like the one presented in this paper is a perfect fit as the building block of the EOSC vision, acting as an enabler of the cross-border and cross-community services which national and European researchers are in need of.

There are challenges which still need to be tackled, about the specific Business Model behind the provisioning and consuming of Community Clouds, and related to the existing and forthcoming regulations on privacy and security. In order to achieve long-term sustainability, funding and business model issues related to cross-domain services also need to be addressed.

8. Conclusions

The architecture of the GARR Cloud presented here is a reference model of an infrastructure-level federated cloud entirely built with the use of open-source software.

We described how it is possible to create and maintain a federation of private cloud regions that fits the needs of a large number of researchers and scientific domains minimising the effort needed for maintenance and operations. The Federated approach allows a large number of organisations (Universities and Research institutions) to create similar infrastructures and federate them together without giving away neither data ownership, nor authority on the resources. At the same time, this approach fully supports cross-domain and cross-institutional research activity. Because of the focus at infrastructure level it fosters the creation and cross-fertilization of a knowledge building ecosystem that is a benefit for the research community.

From the funders' perspective, this approach allows shifting from a pay-per-use model, in which final costs are hardly predictable and researchers need to adapt to the commercial offering, to one in which investments in hardware and personnel training can be managed and programmed and services can be tailored, evolved, updated according to the users' needs. In turn, this also implies that a larger fraction of the overall investment stays within the national borders, supporting the construction and retention of knowledge assets empowering the research community by providing the background and skills needed to evaluate and compare the content of commercial offers.

It's in GARR mandate to foster community independence and know how for its stakeholders. With this investment in cloud federations GARR wanted to give a tool to navigate the continuously evolving landscape of cloud infrastructures and to preserve research independence both in the near future and in the long term.

9. References

1. OpenStack: https://www.openstack.org
2. MaaS: https://maas.io
3. Juju: https://jaas.ai
4. Ceph: https://ceph.io/
5. Kubernetes: https://kubernetes.io/
6. Swift: https://wiki.openstack.org/wiki/Swift
7. Cinder: https://wiki.openstack.org/wiki/Cinder
8. Keystone: https://wiki.openstack.org/wiki/Keystone
9. GARR-X Progress: https://www.garrxprogress.it/
10. Ceph-ansible: https://github.com/ceph/ceph-ansible
11. OpenStack region: https://docs.openstack.org/python-openstackclient/rocky/cli/command-objects/region.html
12. VPN: https://docs.openstack.org/mitaka/networking-guide/common/glossary.html#term-virtual-private-network-vpn
13. Idem GARR Federation: https://www.idem.garr.it/
14. Edugain: https://edugain.org/
15. OpenIDConnect: https://openid.net/connect/
16. GARR TV: https://www.garr.tv/
17. OnlyOffice: https://www.onlyoffice.com/
18. Jira: https://www.atlassian.com/software/jira
19. IdP in the Cloud GARR: https://www.servizi.garr.it/en/idp-in-the-cloud-en
20. GÉANT: https://www.geant.org/
21. EOSC: https://www.eosc.eu/