Cloud Infrastructures

Alessandro Costantini, INFN – CNAF
alessandro.costantini@cnaf.infn.it

Corso Big Data, INFN-CCR
9-12 Dicembre 2019

The trainer

- alessandro.costantini@cnaf.infn.it
- National Institute for Nuclear Physics (INFN)
- Involved in different cloud-oriented projects
- Based in Bologna, Italy
- Member of Distributed Systems Unit @ INFN-CNAF

https://www.cnaf.infn.it

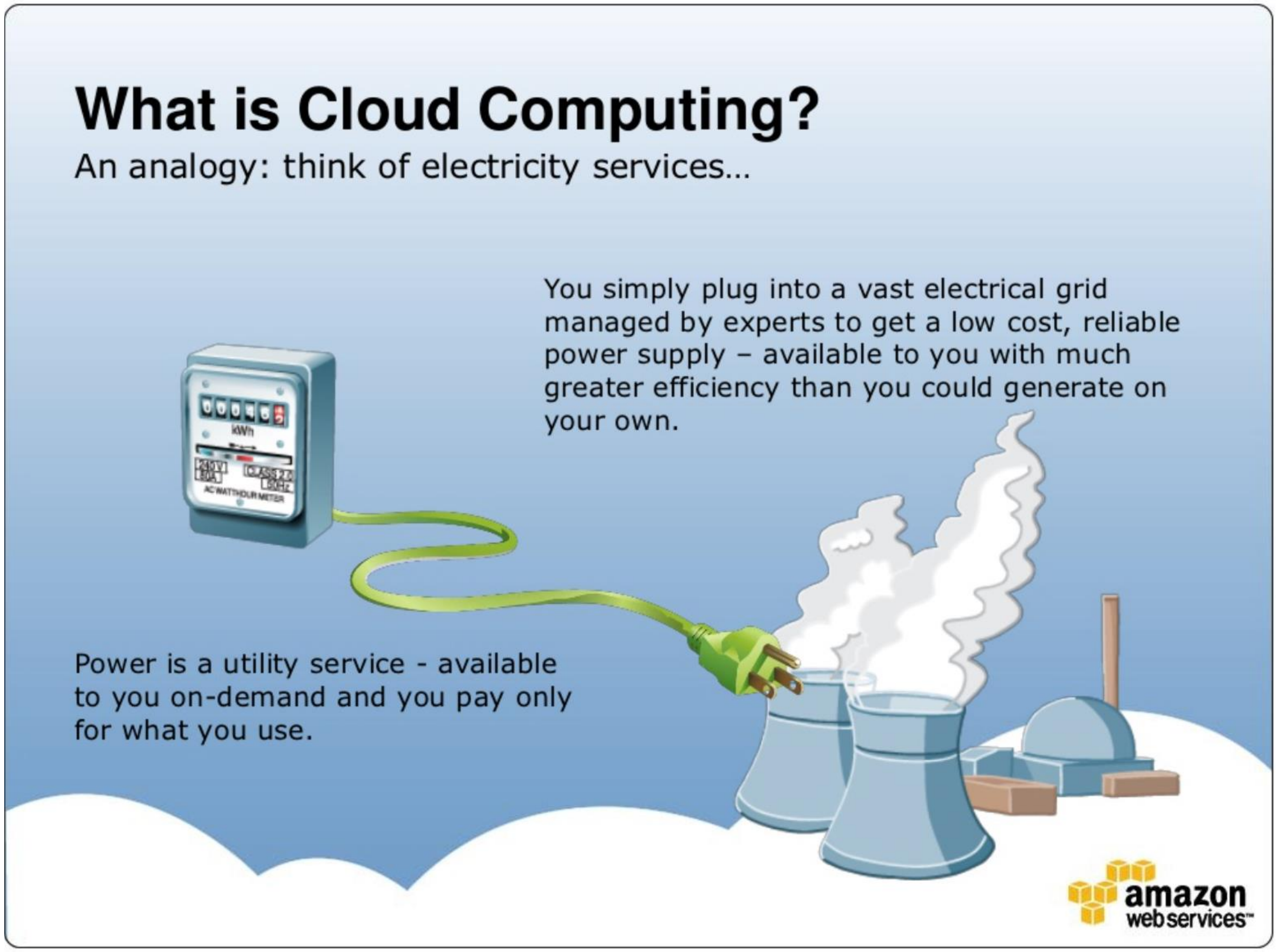
Training goals

1. Learn the concept of Cloud computing
2. IaaS approach: OpenStack
3. Hands-on with Cloud services
   - IaaS approach with OpenStack
   - Connection to the VMs

Introduction to cloud computing

Cloud Computing

Cloud computing: Analogy



What is Cloud Computing?

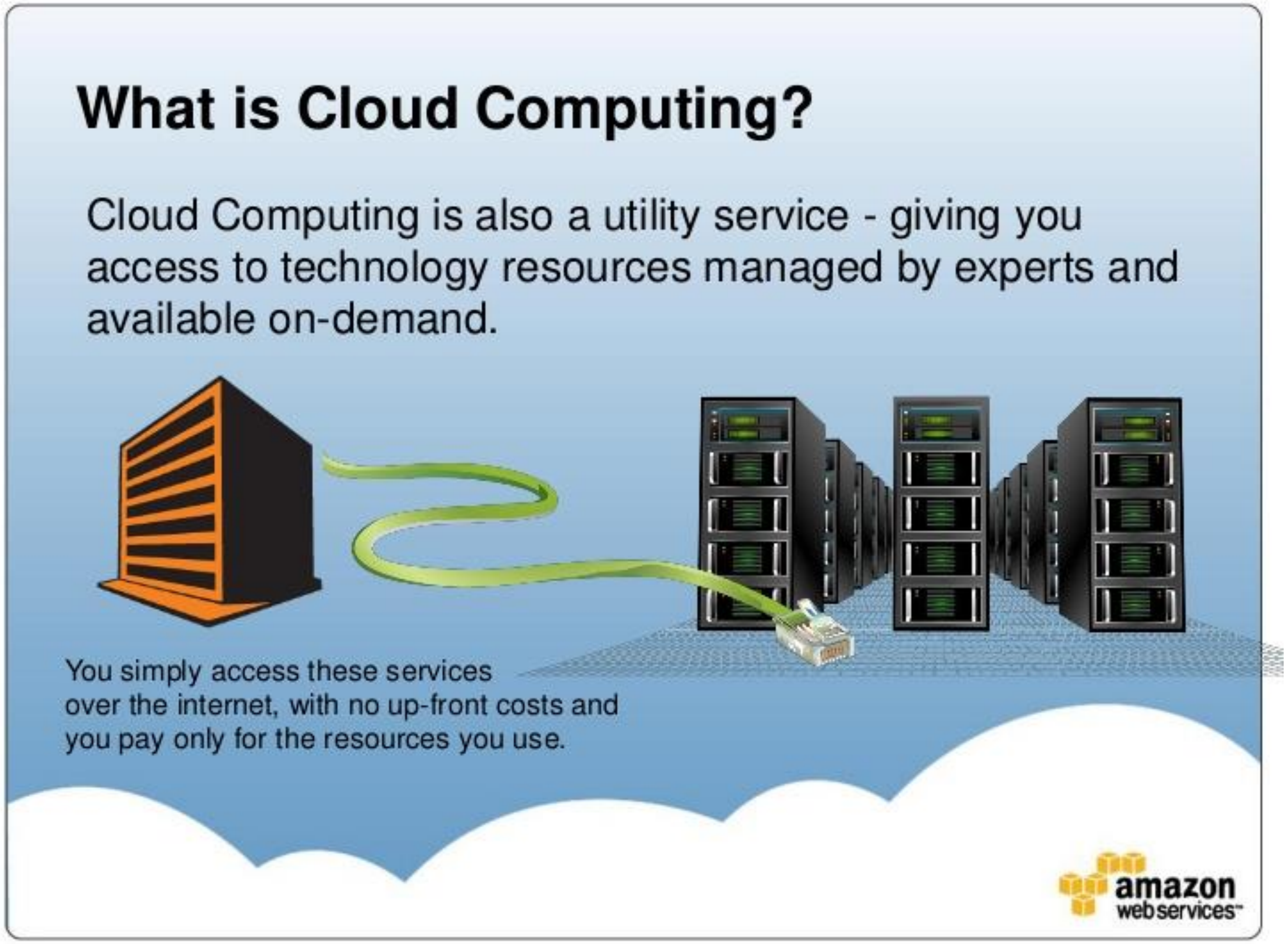
An analogy: think of electricity services...

You simply plug into a vast electrical grid managed by experts to get a low cost, reliable power supply – available to you with much greater efficiency than you could generate on your own.

Power is a utility service - available to you on-demand and you pay only for what you use.

Source: bit.ly/2Z4kHNG

Cloud computing: Concept



What is Cloud Computing?

Cloud Computing is also a utility service - giving you access to technology resources managed by experts and available on-demand.

You simply access these services over the internet, with no up-front costs and you pay only for the resources you use.

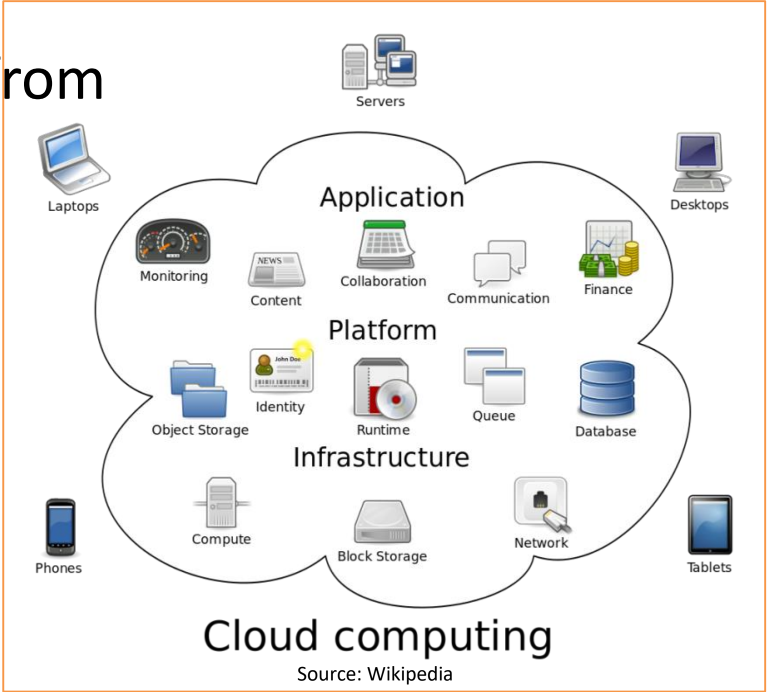
amazon
webservices

Source: bit.ly/2Z4kHNG

Cloud Computing: Definition

- The canonical definition comes from the **US National Institute of Standards and Technology (NIST)** bit.ly/2YOop2X

- In a nutshell, Cloud Computing deals with:



Source: Wikipedia

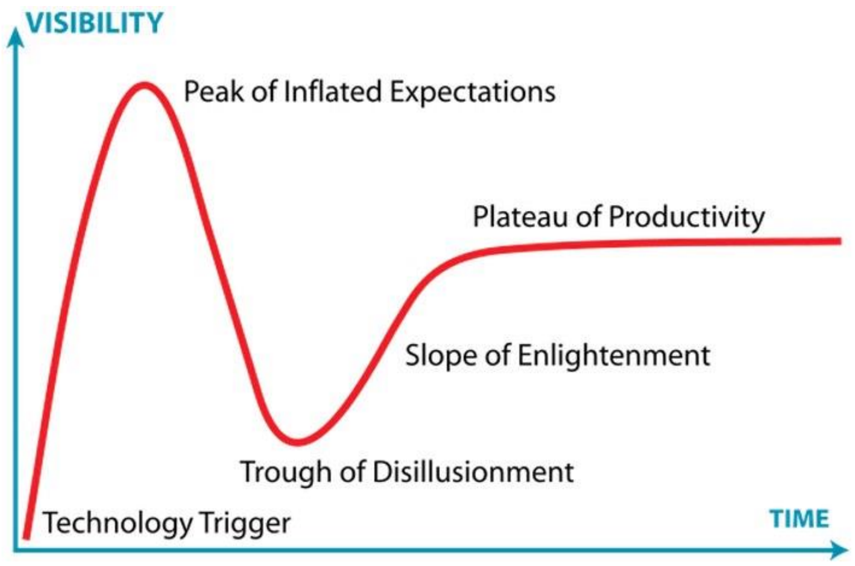
1 Supplying

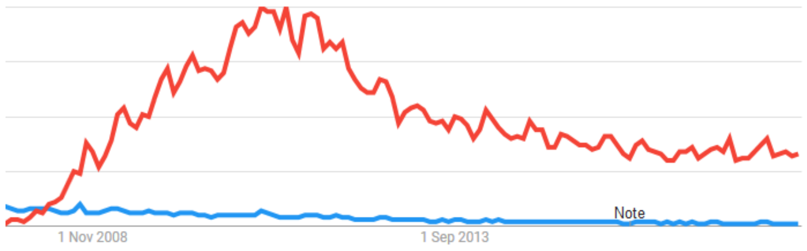
2 information and communication technologies

3 as a service

Cloud hype cycle

- The hype cycle is used to represent the maturity, adoption and social application of specific technologies, through five phases ([bit.ly/2H4iX0N](bit.ly/2H4iX0N)):
  - Technology trigger
  - Peak of inflated expectations
  - Trough of disillusion
  - Slope of enlightenment
  - Plateau of productivity
- See also the "Gartner Hype Cycle for Cloud Computing" for more details ([bit.ly/33p1UjA](bit.ly/33p1UjA))





Characteristics

- **Self-service, on-demand**
  - A consumer can unilaterally provision computing capabilities as needed automatically without requiring human interaction with each service provider.
- **Network-based access**
  - Capabilities are available over the network and accessed through standard mechanisms that promote use by heterogeneous thin or thick client platforms
- **Resource pooling**
  - The customer has no control or knowledge over the details of the provided resources, that are managed by the Cloud provider
- **Elasticity**
  - Capabilities can be elastically provisioned and released to scale rapidly commensurate with demand. To the consumer, the capabilities available for provisioning often appear to be unlimited
- **Pay-per-use**
  - The customer pay only for what he/she used.

An analogy: car rental

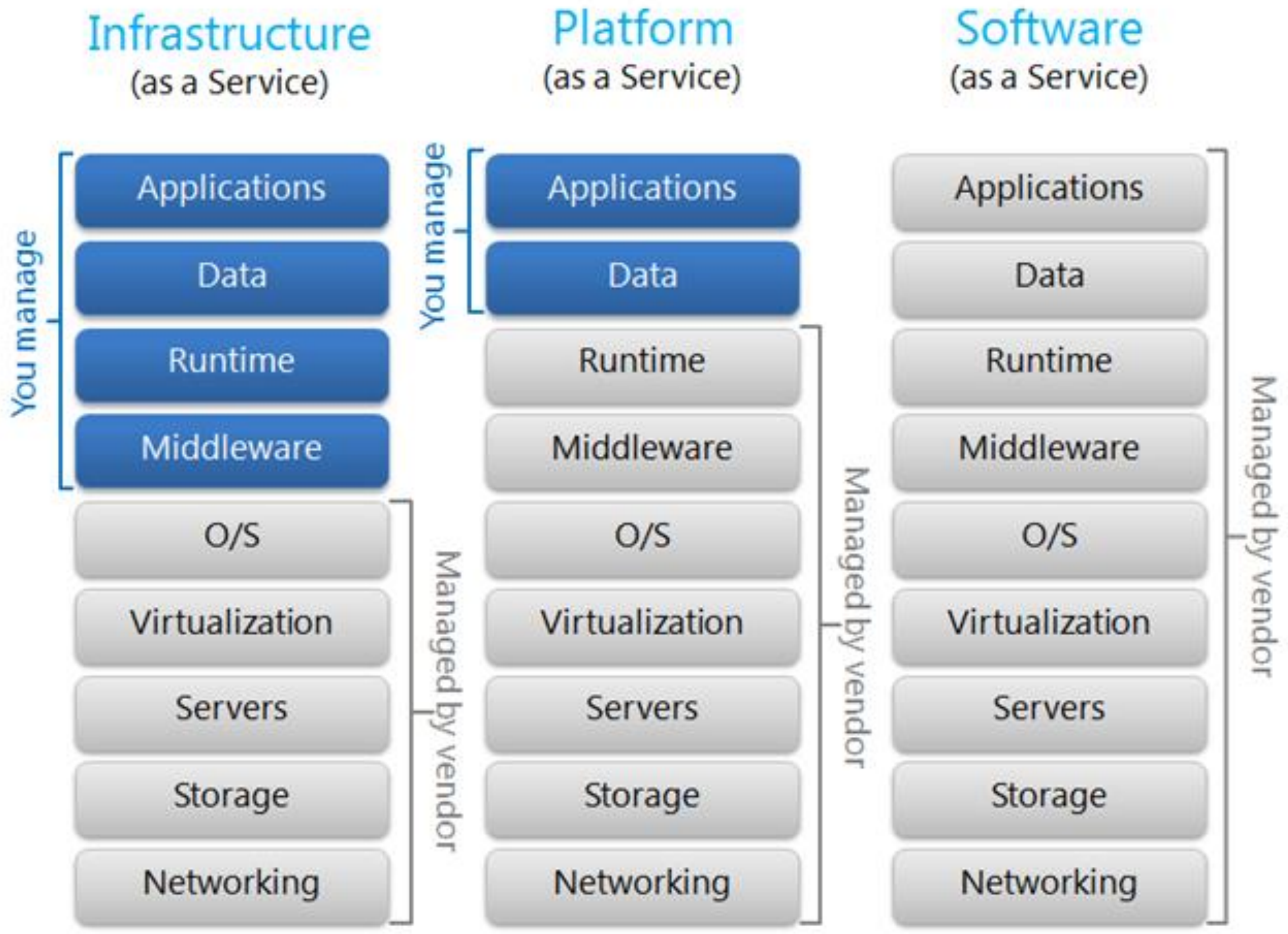
- **Self-service, on-demand**
  - Online or by telephone booking
- **Network**
  - Network of car rental all over the world
- **Resource pooling**
  - The car rental manages the availability of cars for customers
- **Elasticity**
  - The number of cars can vary depending on users demand
- **Pay-per-use**
  - The customer pays for the time he/she used the service (no matter about tires, insurance, etc.)

The emphasis on “Service”

- In the standard Cloud definition (“Supplying information and communication technologies as a service”), the ***service toward the Cloud users is the essential part*** – e.g. for usability, flexibility, reliability, etc.

- Cloud computing is indeed typically modeled around ***service models*** primarily linked to:
  - Infrastructure (**IaaS** → Infrastructure as a **Service**)
  - Platform (**PaaS** → Platform as a **Service**)
  - Software (**SaaS** → Software as a **Service**)

Service models



Source: bit.ly/2KuxiFW

IaaS - Infrastructure as a Service

- **IaaS**, the basic building blocks of a data center:
  - **Storage** → I want to store data, lots of data, at low cost
  - **Compute** → give me a machine where I can host my services or run my applications
  - **Network** → create a "Software-Defined Network" infrastructure for me
- In many cases, in a "virtual" form
- No need to know details, no need to contacts administrators to install something

PaaS - Platform as a Service

- **PaaS**, a computing platform providing you with several building blocks or components that you can request programmatically or statically. For example:
  – A cluster of systems with operating system and an entire execution environment installed and configured.
  – A web server (or a cluster of web servers) with database(s), virtual storage, load balancers, other dependencies.

SaaS - Software as a Service

- With **SaaS**, you are directly given access to some application software. You don't have to worry about the installation, setup and running of the application. You typically access SaaS applications via a web browser.
- For example: Gmail, social media such as Facebook, Twitter, Instagram, etc.

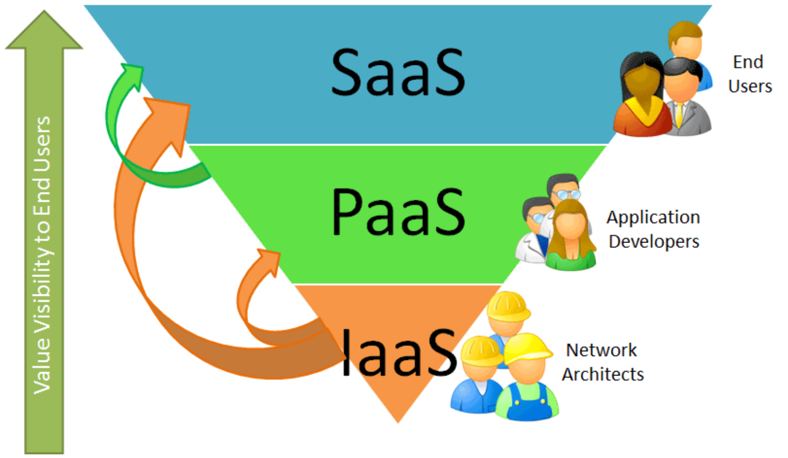
IaaS vs. PaaS vs. SaaS

	IaaS	PaaS	SaaS
What you get	You get the infrastructure. Freedom to use or install any OS or software	You get what you demand: software, hardware, OS, environment.	You don't have to worry about anything. A pre-installed, pre-configured package as per your requirement is given.
Deals with	Virtual Machines, Storage (Hard Disks), Servers, Network, Load Balancers etc	Runtimes (like java runtimes), Databases (like MySQL, Oracle), Web Servers	Applications like email (Gmail, Yahoo mail etc), Social Networking sites (Facebook etc)
Popularity	Highly skilled developers, researchers who require custom configuration as per their requirement or field of research.	Most popular among developers as they can directly focus on the development of their possibly complex apps or scripts.	Most popular among normal consumers or companies which rely on software such as email, file sharing, social networking as they don't have to worry about the technicalities.

Remember what matters…

What matters, at the end, *are the applications*.

TRUE!

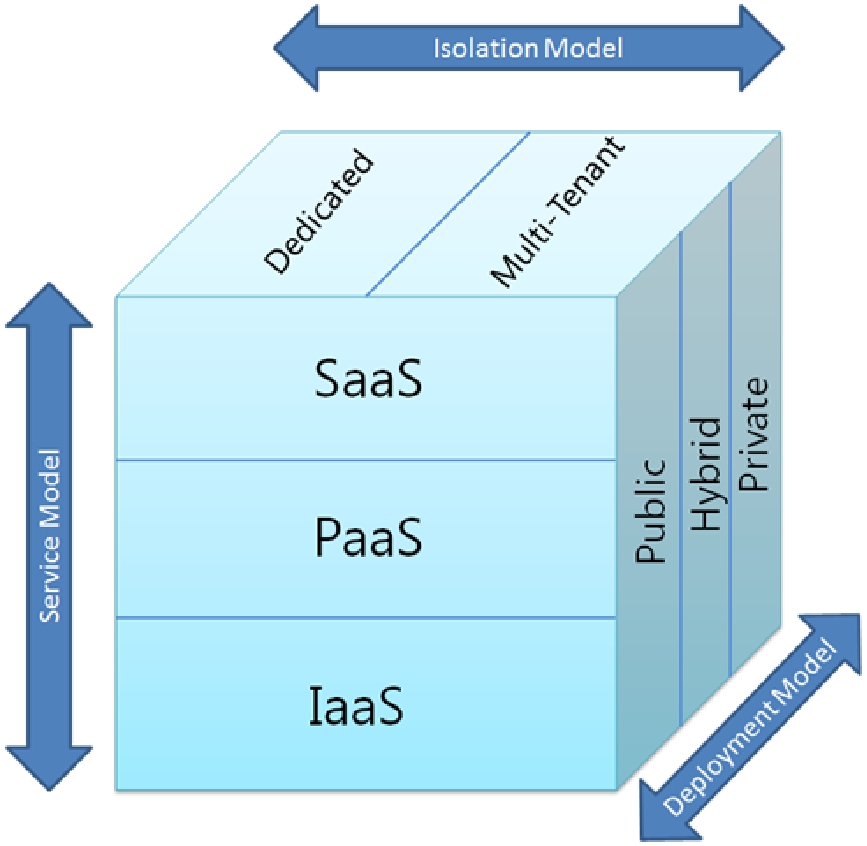


… however, **without Cloud providers (public or private)**, and without **efficient and effective** ways of managing distributed resources, applications cannot be deployed!
(rather obvious isn't it)

How do you find, provision and use resources in the Cloud, then?

Let’s add dimensions

- Beyond the *service models* (IaaS, PaaS, SaaS), important parts to define and understand Cloud computing are the models linked to:
  - ***deployment***
    - where distribute services
  - ***isolation***
    - how isolate services



Source: bit.ly/2KuxiFW

Isolation

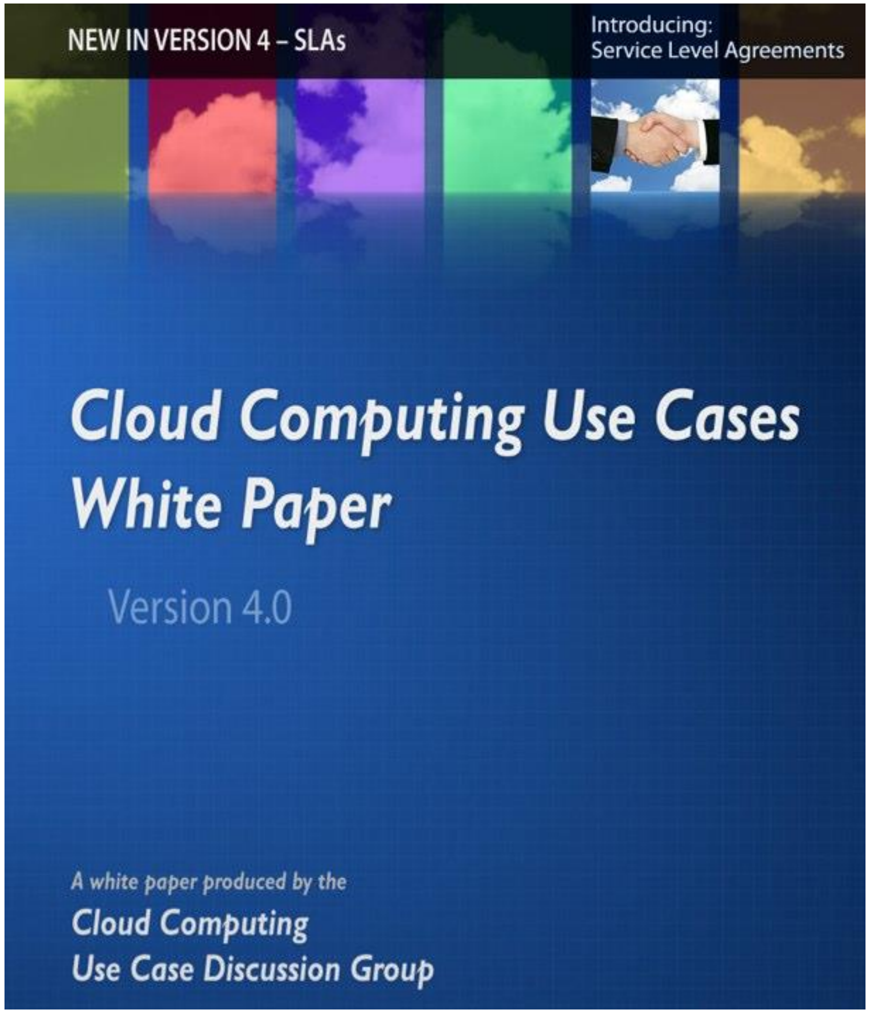
- Cloud **isolation models** are important and often ignored. We could have :
  - **Dedicated** infrastructures
  - **Multi-tenant** infrastructures (i.e., with several [types of] customers)
- The isolation type is essential in many regards, such as:
  - Resource segmentation
  - Data protection
  - Application security
  - Auditing
  - Disaster recovery

Deployment: “Cloud types”

- **Private Cloud:**
  - The infrastructure is **procured for exclusive *use*** by a single organization. Management, operation, ownership, location of the private cloud, however, can be independent by the organization using it.
- **Community Cloud:**
  - The infrastructure is **available to a community** of organizations sharing a common goal (for instance: mission, security requirements, adherence to common regulatory rules, etc.)
- **Public Cloud:**
  - The infrastructure is **available to the public** at large. Management can be either public or private. The location is at some service supplier premises.
- **Hybrid Cloud:**
  - The infrastructure is a **combination of two or more Cloud infrastructures** (private, public, community Cloud), connected so that there is some form of portability of e.g. data or applications.

Cloud use cases

- https://goo.gl/qxRtrw
- 7 principal cases:
  - End user → Cloud
  - Enterprise → Cloud → end user
  - Enterprise → Cloud
  - Enterprise → Cloud → enterprise
  - Private Cloud
  - Changing Cloud vendors
  - Hybrid Cloud


NEW IN VERSION 4 – SLAs

Introducing:
Service Level Agreements

Cloud Computing Use Cases White Paper

Version 4.0

A white paper produced by the Cloud Computing Use Case Discussion Group

Source: bit.ly/305Jfre

Cloud use cases: some examples

- **Testing and development.**
  - As with traffic bursting, you may not have the capacity to host lots of servers and storage in your data center for testing and development purposes. Using the public Cloud allows you to spin up servers as you need them, and then shut them down when you're finished.
- **Web hosting**
  - Many organizations choose to host their web services in the Cloud because it can balance the load across multiple servers and scale up and down quickly and automatically with traffic.
- **Big Data and data manipulation.**
  - Maintaining and implementing compute resources to handle huge datasets can be expensive and complicated. Using Cloud computing resources, you can use only the resources you need to analyze data when you need them. Some public cloud vendors offer specialized managed Big Data services.

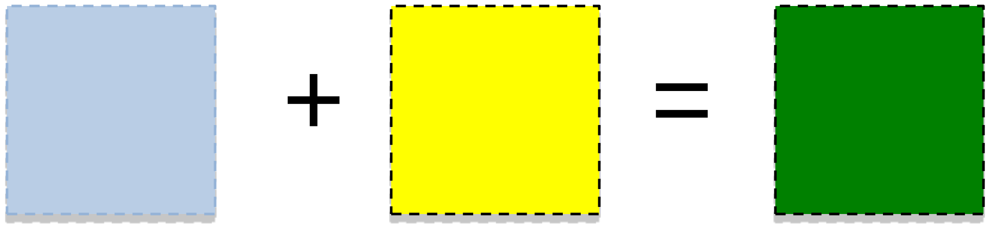
Source: https://goo.gl/29PQFH

Static vs Virtual

Virtualization and Cloud

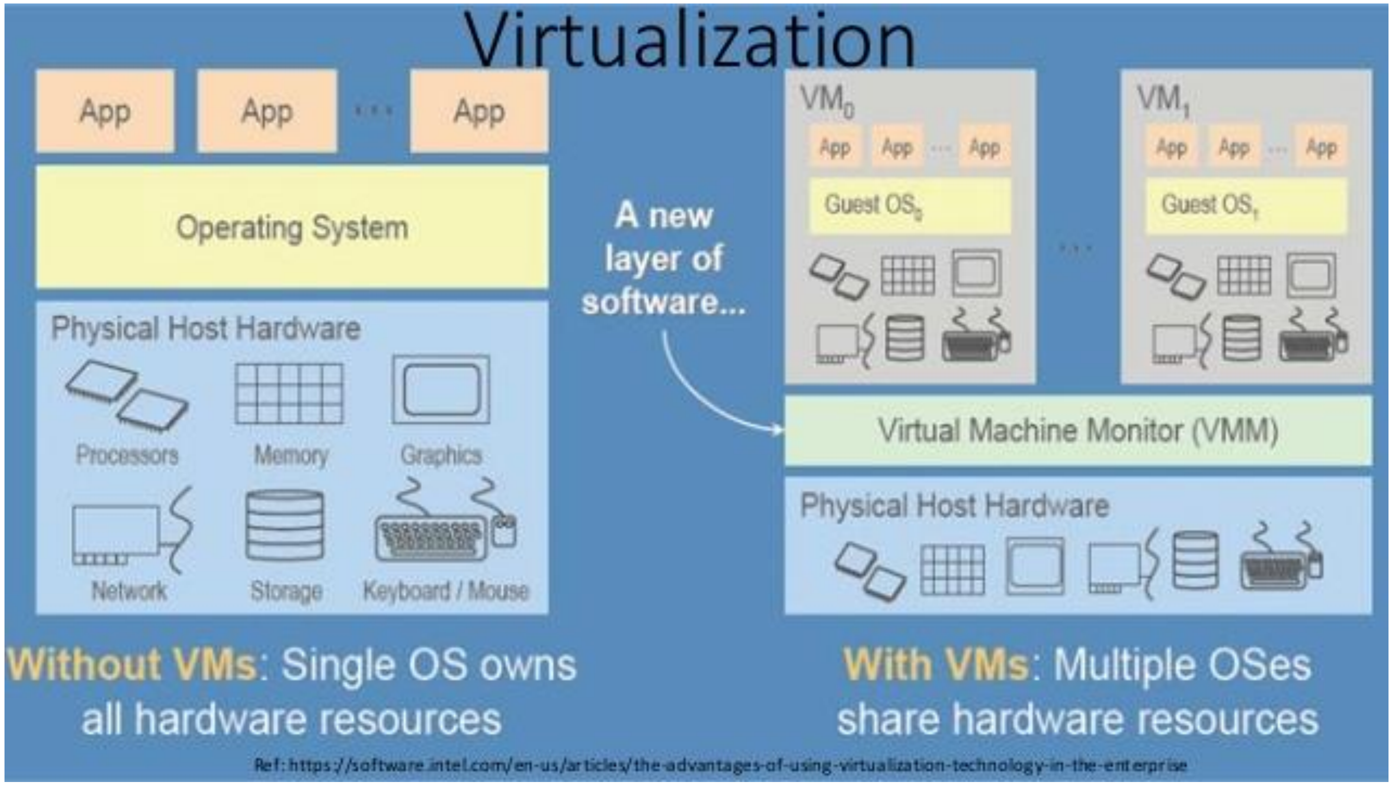
- Virtualization and Cloud computing

- An analogy

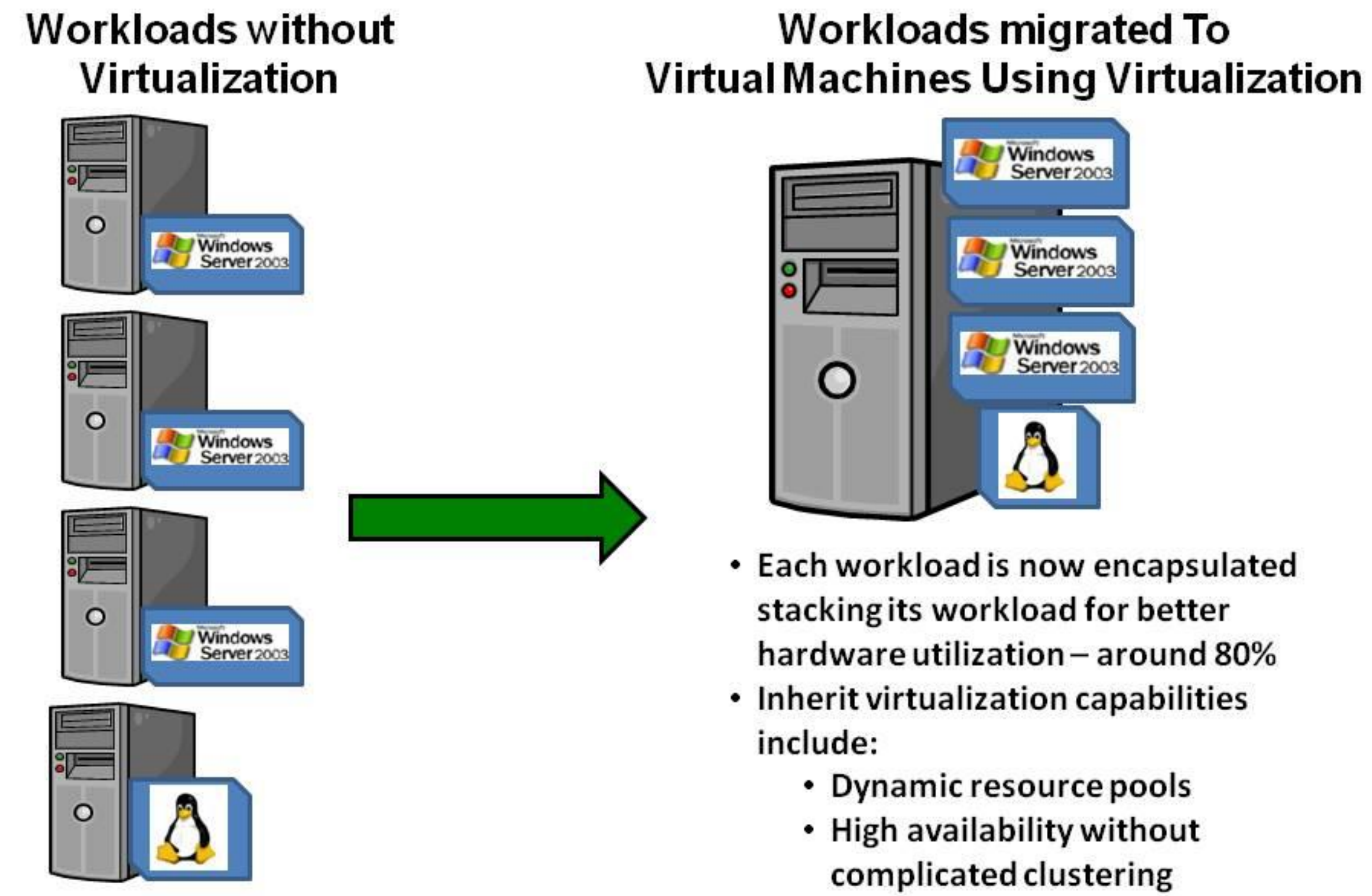


Virtualization

- Informally, by *virtualization* we mean the creation of a virtual version of *something*.
  - For instance, an hardware platform, an operating system, a storage device, a network resource.
  - Through an **abstraction**: an intermediate level between hardware/software and applications, simplifying and hiding underlying details.


Virtualization
App
App
App
Operating System
Physical Host Hardware
Processors
Memory
Graphics
Network
Storage
Keyboard / Mouse
A new layer of software...
VM0
Guest OS0
VM1
Guest OS1
Virtual Machine Monitor (VMM)
Physical Host Hardware
Without VMs: Single OS owns all hardware resources
With VMs: Multiple OSes share hardware resources
Ref: https://software.intel.com/en-us/articles/the-advantages-of-using-virtualization-technology-in-the-enterprise

Let’s go virtual



- Servers poorly utilized at average of 4% to 7% capacity
- Limited in failover capability
- Prone to hardware failure

- Each workload is now encapsulated stacking its workload for better hardware utilization – around 80%
- Inherit virtualization capabilities include:
  - Dynamic resource pools
  - High availability without complicated clustering
  - Provision new servers in minutes
- Virtual Machines are hardware independent

Source: https://technofirmsoftware.wordpress.com/tag/benefits-of-virtualization/

Virtualization advantages (1/2)

- **Server consolidation**
  - Multiple VMs on the same host.
  - Cost reduction for hardware provisioning that can simplify administrative and monitoring operations
- **Isolation (*sandboxing*)**
  - Application isolation.
  - Code development, testing and debugging.
  - Creating dedicated enviroment for legacy application.
- **On-demand VM provisioning**

Virtualization advantages (2/2)

- **Decoupling of hardware and software**
  - Suspend/Resume VMs.
  - Migration of VMs between physical hosts
- **Testing of new versions of Operationg System, applications**
  - Or of old versions: data preservation
- **Emulation of hardware**
  - different from that of the physical host
- **Execution of applications**
  - that can not run on the OS of the physical host

Virtualization disadvantages

- Security.
  - On the same hardware different OS cohexist, managed by a software – higher probability of bugs or *attack vectors*:
    - **VM-to-VM → network attacks**
    - **VM-to-HV** (KVM o XEN)
      KVM is a Linux kernel module
      Xen is a hosted hypervisor, directly connected to the hardware → **everything can be compromized**
    - **VM-to-QEMU**
      QEMU is a complex software. In case of attack, the OS can be compromized.

- Performance
  - Overhead for the physical host
  - Worse performance for the VM, especially I/O

Virtualization and/or Cloud Computing?

- Provisioning of VMs is not *Cloud computing*.
- Check the 5 Cloud characteristics:
  - **Self-service, on-demand → NO**
    - tipically an IT department provides VMs
  - **Network-based access → NO**
    - deployment limited to “internal customers”
  - **Resource pool → YES**
  - **Elasticity → NO**
    - tipically an IT department installs OS + software and maybe not in a scalable mode
  - **Pay per use → NO**
    - traditional billing

Docker

- “Open-platform for building, shipping and running distributed applications”

- Docker commoditizes containers
  - Hides and automates container management process
  - One-command-line deployment of applications
  - Easy to move from development to production
  - Provides ecosystem to create and share images

Migration to Cloud

Application migration to Cloud

- Migration of an application from an existing data center to a Cloud infrastructure

- Which technical and business factors move to migration?
  - **Cost reduction** → resource pooling, pay-per-use
  - **Business agility** → deployment simplification
  - **Management saving** → performance (e.g. auto-scaling), delegation of operations

- Public or private Cloud?
  - WAN traffic? (tipically expensive)
  - Security?
  - Integration with other *legacy* applications?

Cloud advantages

- A large cloud inrfrastructure permits to **reduce costs for a single server**.
  - More customers, **less management costs for each customer**.
- The resource aggregation allows their **more efficient utilization**.
- **Flexibility and Scalability,** Self-service provisioning and possibility to increase the resources through Cloud providers, instead of buying new resources
- **Collaboration and Business opportunities,** in terms of ubiquitous access to resources from any device (SaaS) and simple sale of software developed by someone else

Last but not least, the big misunderstanding

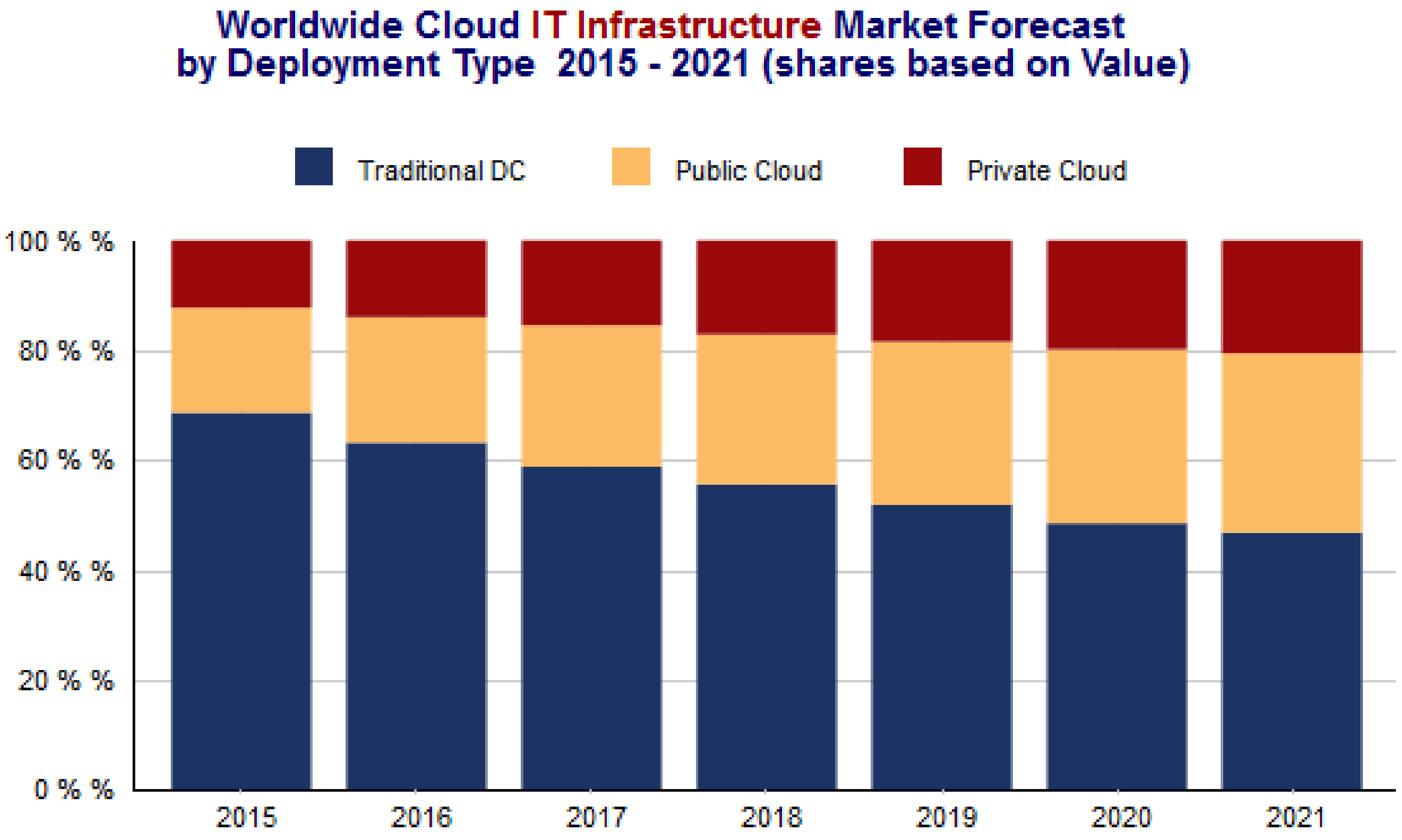
- **Capacity is not infinite** (although this is one of the postulates of Cloud computing). *Nor are credit card limits*.
  - Hence, resources might not be available when we need them; or, if available, they might not have the characteristics we need.
  - Unless maybe we are willing to pay some hefty over-provisioning costs.

Cost evaluation

- Understand **if by an economical point of view the best solution is a public or private Cloud** is not easy. It requires ad-hoc investigation. E.g.:
  - How important is a possible data lost? And information leakage towards my competitors? And what about the know-how lost?
  - The terms of agreement with the Cloud provider are completely clarified?

Cloud market Forecast

Source: goo.gl/cQk28g



EOSC-hub Science Cloud in Europe: EOSC

EUROPEAN
COMMISSION

Brussels, 19.4.2016
COM(2016) 178 final

COMMUNICATION FROM THE COMMISSION TO THE EUROPEAN PARLIAMENT, THE COUNCIL, THE EUROPEAN ECONOMIC AND SOCIAL COMMITTEE AND THE COMMITTEE OF THE REGIONS

European Cloud Initiative - Building a competitive data and knowledge economy in Europe

{SWD(2016) 106 final}
{SWD(2016) 107 final}

European Cloud Initiative by the European Commission (April 2016)

1. How to maximise the incentives for **sharing data** and to increase the capacity to **exploit them**?
2. How to ensure that **data can be used as widely as possible**, across scientific disciplines and between the public and the private sector?
3. How better to **interconnect** the existing and the new **data infrastructures** across Europe?
4. How best to **coordinate the support available** to European data infrastructures as they move towards exascale computing?

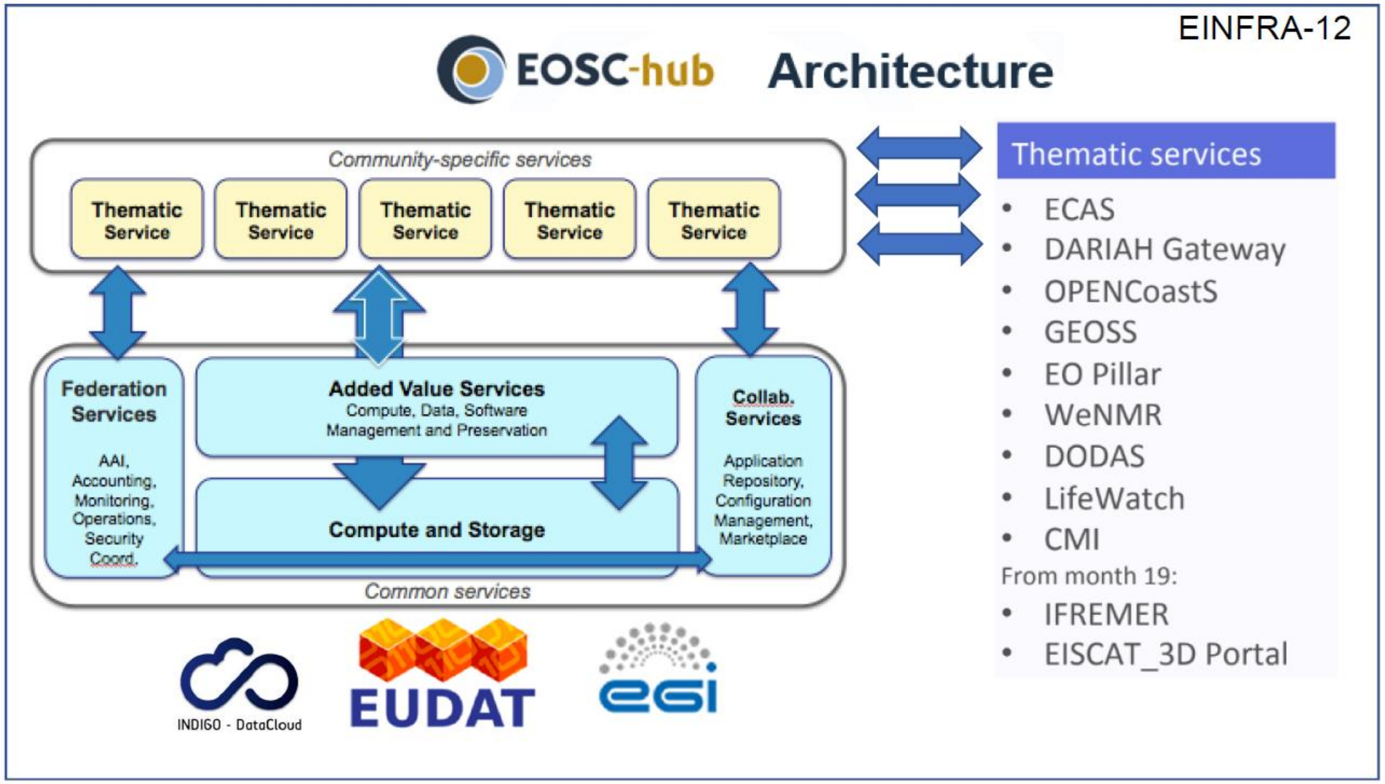
"…a trusted, open environment for the scientific community for storing, sharing and re- using scientific data and results…"

"…by 2020…"

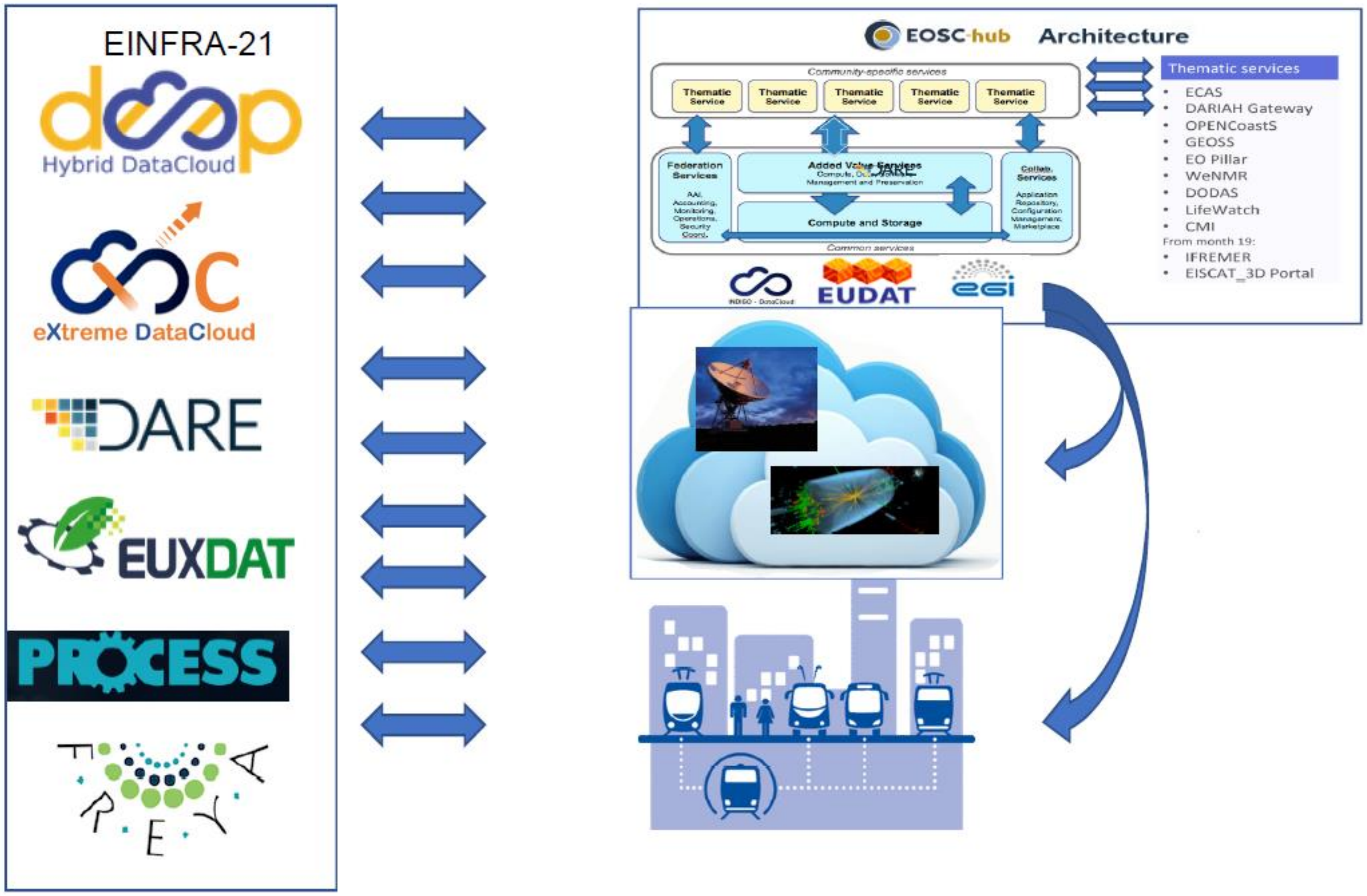
EOSC Ecosystem...the oversimplified story: EOSC-hub project

EOSC-hub mobilizes providers from European major digital infrastructures, **EGI**, **EUDAT CDI** and **INDIGO-DataCloud** jointly offering **services**, **software** and **data** for **advanced data-driven research and innovation**

- 100 Partners
- 76 beneficiaries
- €33M total budget
- 36 months
- Jan 2018 – Dec 2020



EOSC Ecosystem...the oversimplified story



Thematic services

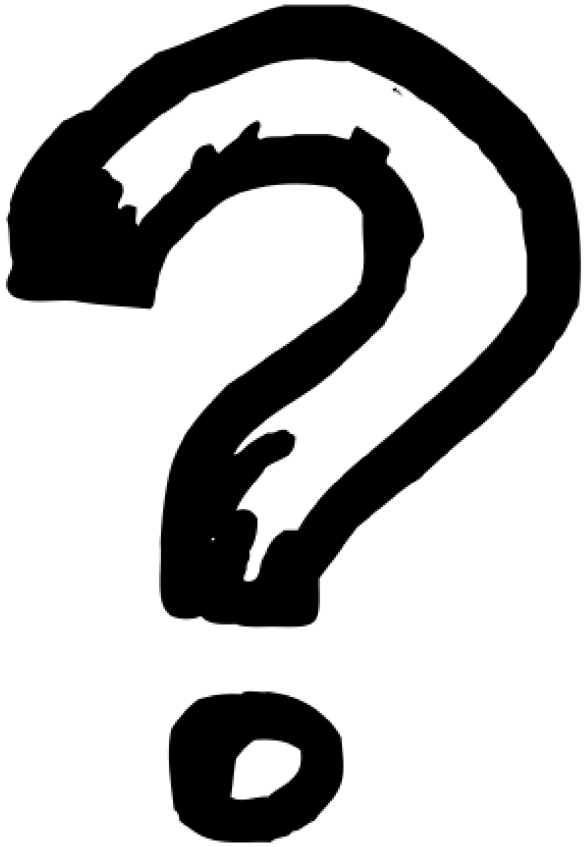
- ECAS
- DARIAH Gateway
- OPENCoastS
- GEOSS
- EO Pillar
- WeNMR
- DODAS
- LifeWatch
- CMI

From month 19:

- IFREMER
- EISCAT_3D Portal

Conclusions

- Cloud computing is a **distributed** technology more flexible and usable than Grid computing
- **Mature** technology, adopted not only in the scientific field
- It **extends** the virtualization concept
- As many complex technologies, it can have **pros** and **cons**
  - You should be careful in order to understand whether Cloud can help you
- The Cloud market is strongly **growing**



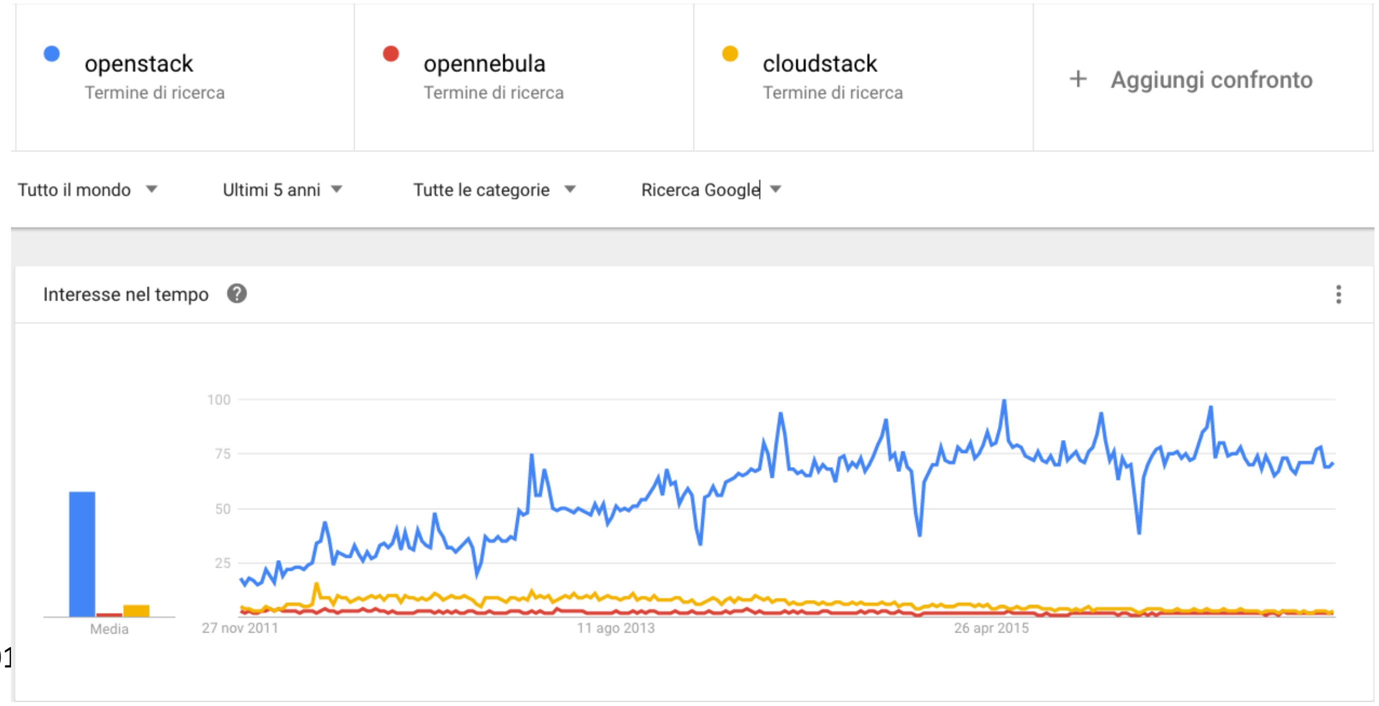
Dealing with IaaS:
The OpenStack implementation

OpenStack

What is OpenStack

OpenStack is a **free** and **open-source** software platform for cloud computing, mostly deployed as a **Infrastructure-as-a-Service** (IaaS)

– interrelated **components** that control diverse, multi-vendor hardware pools of **processing**, **storage**, and **networking resources** throughout a data center.



OpenStack principles

OpenStack is

- ***Open source***
  - Fully Functional Open Source
    - Pluggable functionalities
  - Acceptable Licensing - Apache License, 2.0
- ***Openly designed***
  - Common development cycle – (most) release every 6-months
    - Common cycle with development milestones
    - Common cycle with intermediary releases
- ***Openly developed***
  - Engage larger communities, boarder group of members
  - Project Team Leader (PTL), Code Reviewers
  - Specifications - http://specs.openstack.org/

OpenStack Community

- 2010 - started as joint project of Rackspace Hosting and NASA
- 2012 - **OpenStack Foundation** – *“Protect, Empower, and Promote OpenStack software and the community around it, including users, developers and the entire ecosystem.”*
  - Individual membership – free
  - Sponsors

INFN

Sponsor

AT&T | Ericsson | Huawei | Intel | Rackspace | Red Hat, Inc.

SUSE | Tencent Cloud

8 platinum ($500K/y)

99Cloud Inc. | Canonical | China Mobile | China Telecom | China Unicom | Cisco

City Network | Dell EMC | Deutsche Telekom | EasyStack | FiberHome | Fujitsu

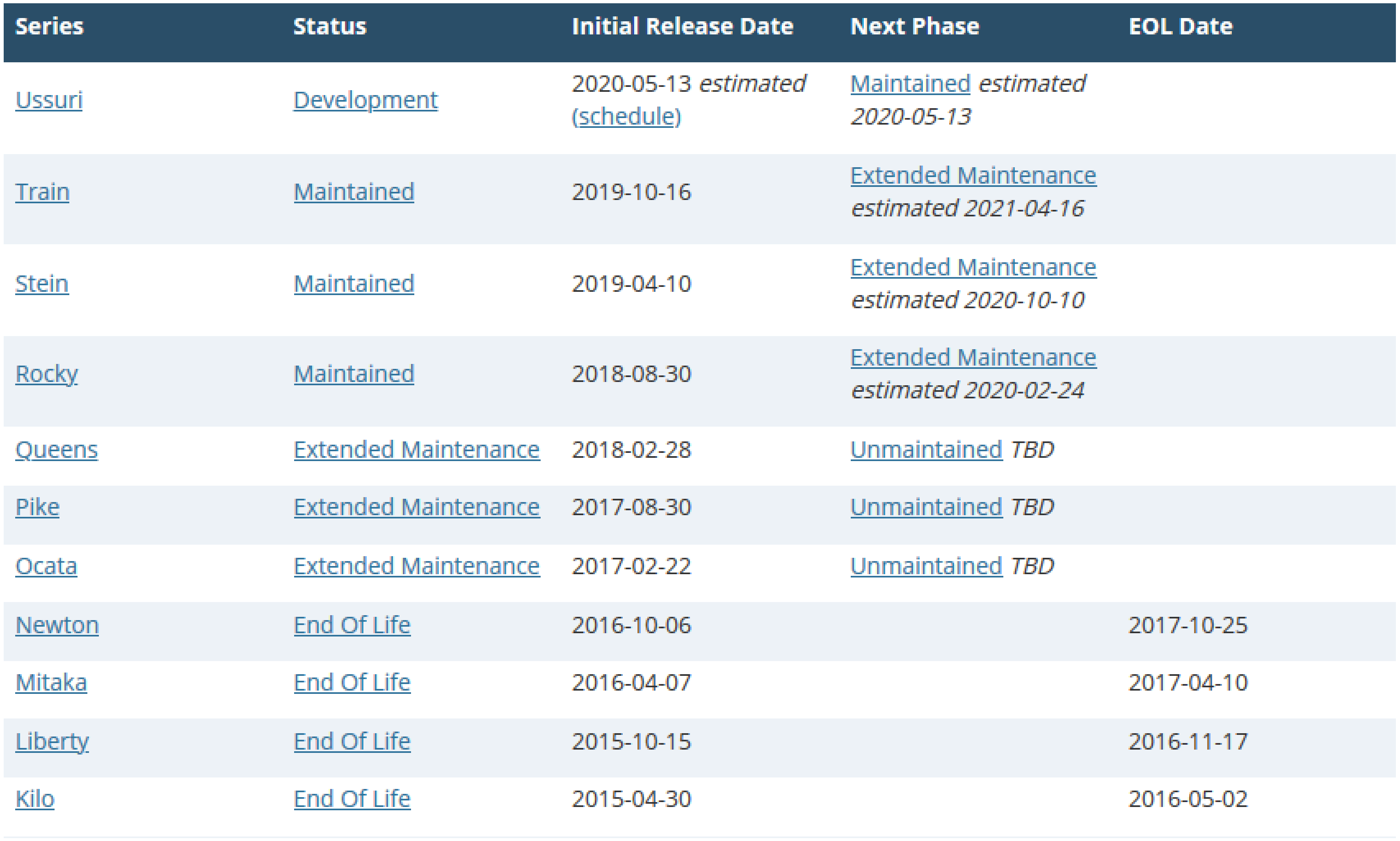
Inspur | inwinSTACK | Mirantis | NEC | NetApp | New H3C Technologies Co., Limited

UnitedStack Inc. | ZTE Corporation

20 gold ($50K/y – 200K/y)

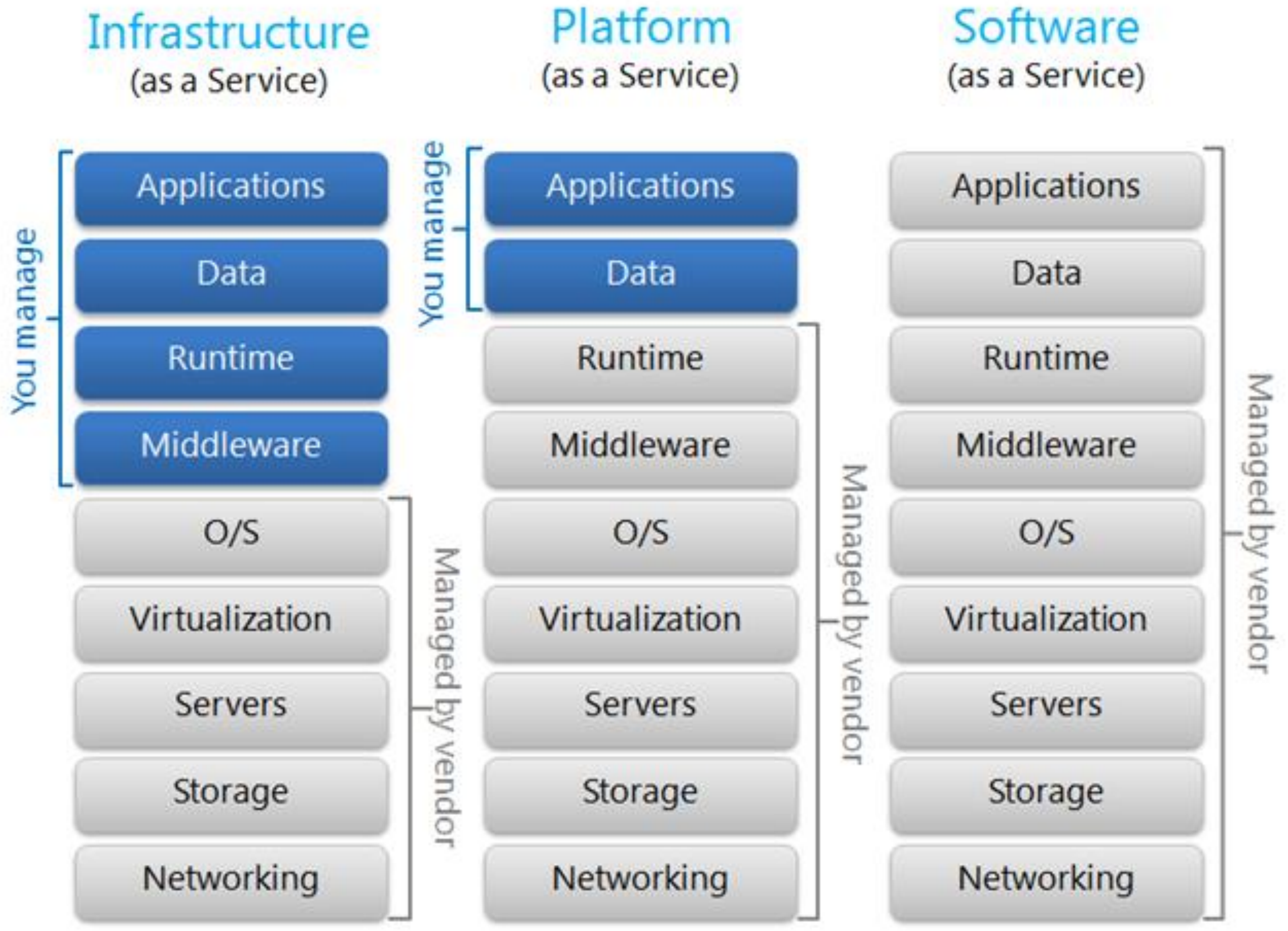
Release

- OpenStack is developed and released around 6-month

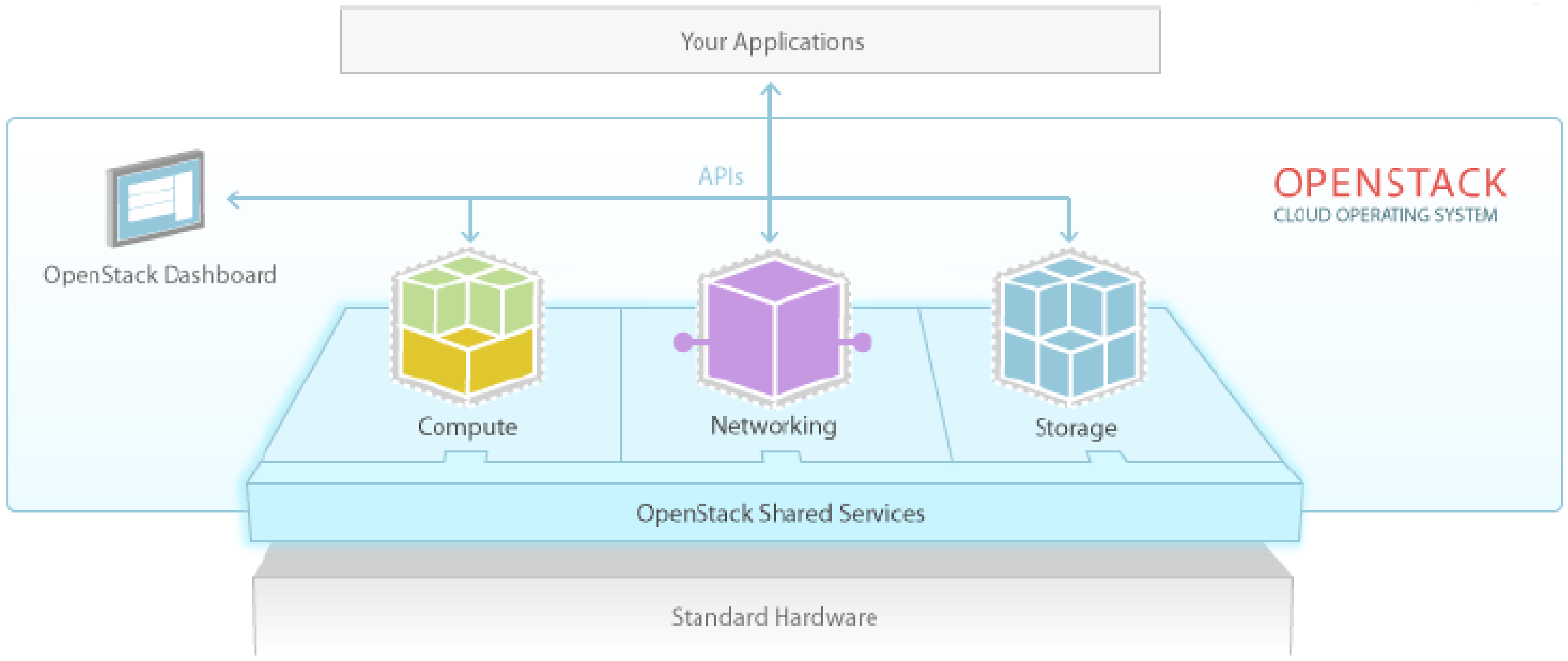


Series	Status	Initial Release Date	Next Phase	EOL Date
Ussuri	Development	2020-05-13 *estimated* (schedule)	Maintained *estimated 2020-05-13*	
Train	Maintained	2019-10-16	Extended Maintenance *estimated 2021-04-16*	
Stein	Maintained	2019-04-10	Extended Maintenance *estimated 2020-10-10*	
Rocky	Maintained	2018-08-30	Extended Maintenance *estimated 2020-02-24*	
Queens	Extended Maintenance	2018-02-28	Unmaintained *TBD*	
Pike	Extended Maintenance	2017-08-30	Unmaintained *TBD*	
Ocata	Extended Maintenance	2017-02-22	Unmaintained *TBD*	
Newton	End Of Life	2016-10-06		2017-10-25
Mitaka	End Of Life	2016-04-07		2017-04-10
Liberty	End Of Life	2015-10-15		2016-11-17
Kilo	End Of Life	2015-04-30		2016-05-02

OpenStack IaaS Platform

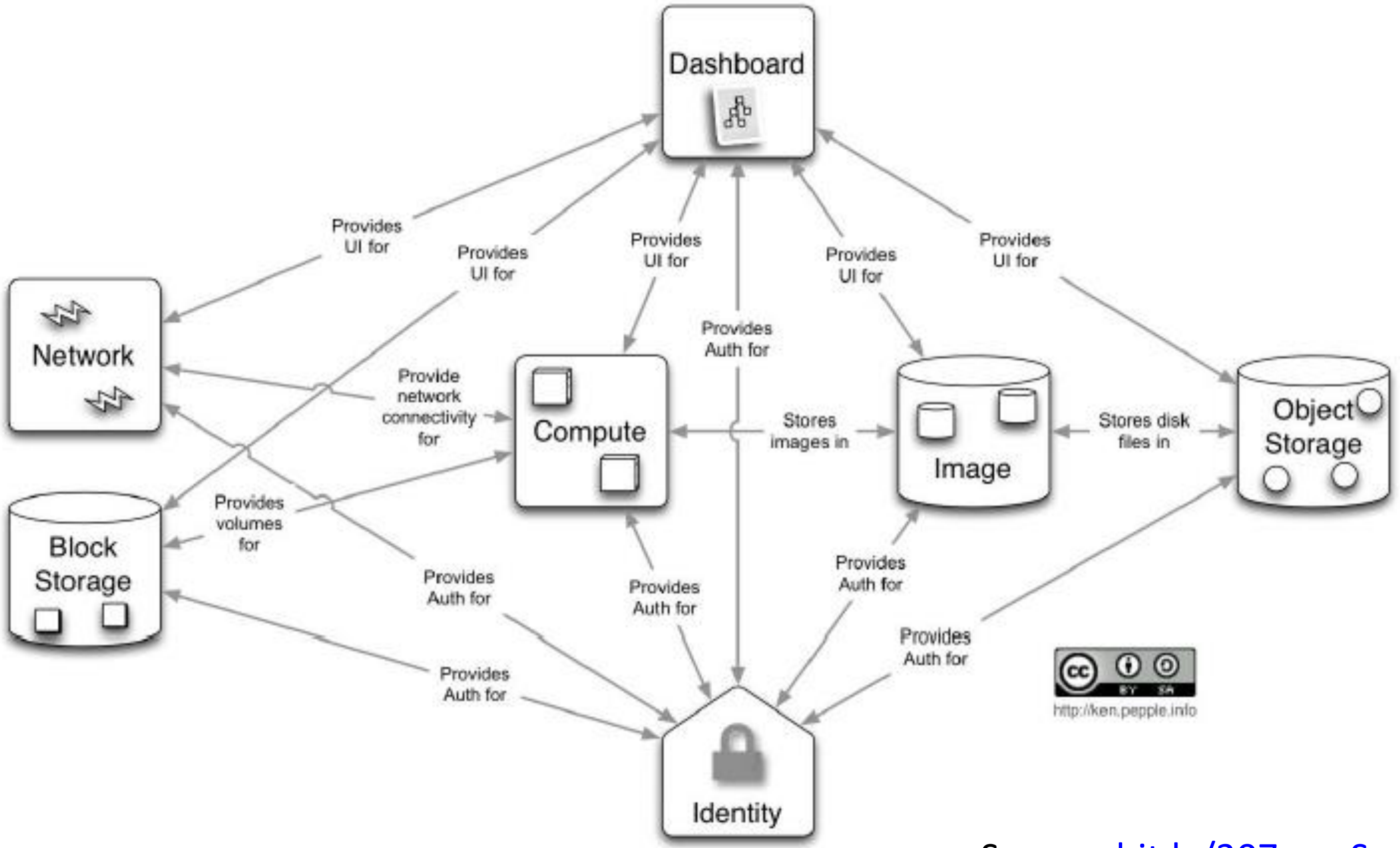


OpenStack: HL overview



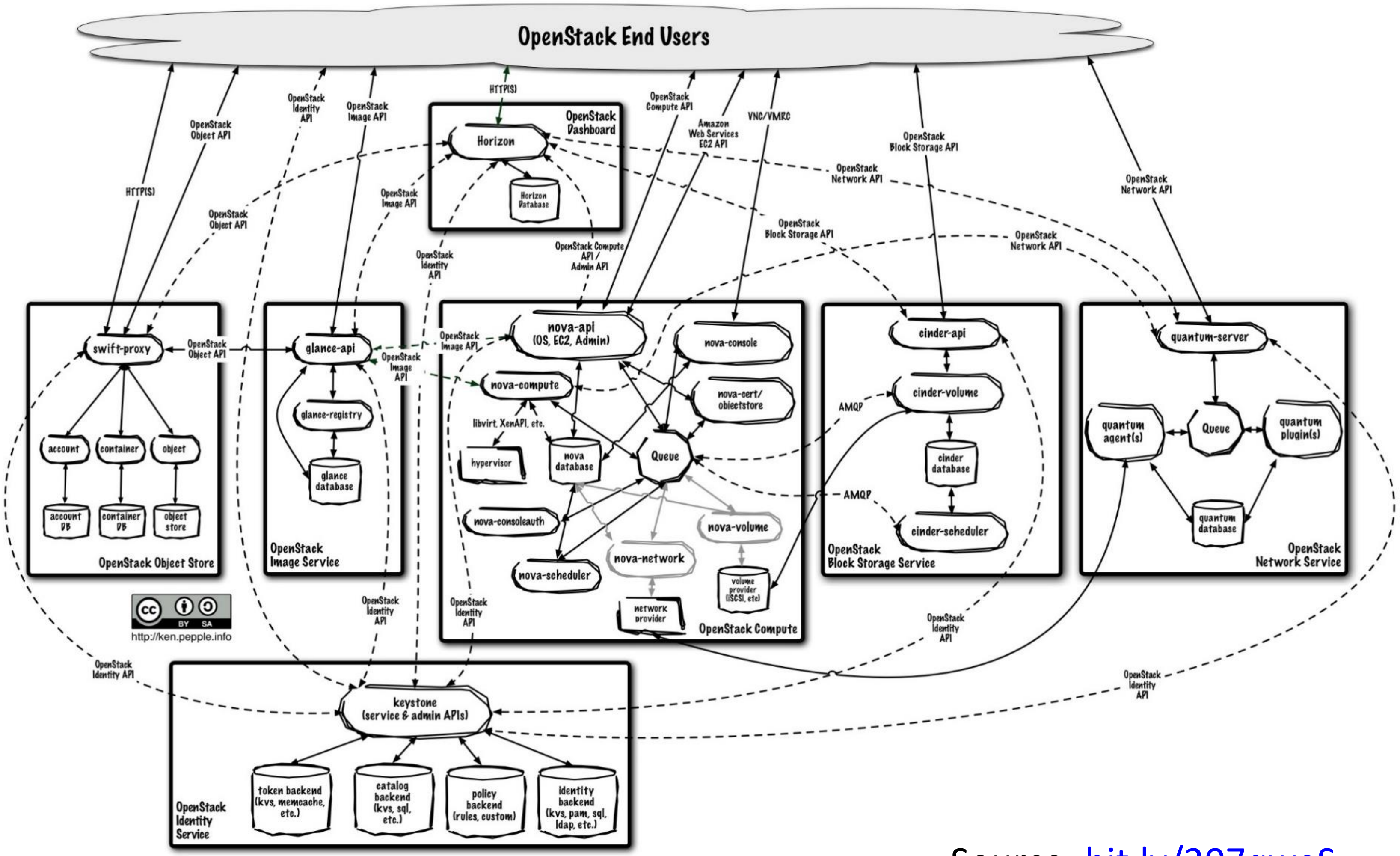
Source: OpenStack

OpenStack: Service Architecture



Source: bit.ly/307qweS

Architecture – deep view



Source: bit.ly/307qweS

OpenStack services

Service	Project Name	Description
Dashboard	Horizon	Provides a web-based self-service portal to interact with underlying OpenStack services, such as launching an instance, assigning IP addresses and configuring access controls.
Compute	Nova	Manages the lifecycle of compute instances in an OpenStack environment. Responsibilities include spawning, scheduling and decommissioning of virtual machines on demand.
Networking	Neutron	Enables Network-Connectivity-as-a-Service for other OpenStack services, such as OpenStack Compute. Provides an API for users to define networks and the attachments into them. Has a pluggable architecture that supports many popular networking vendors and technologies.

OpenStack services

Storage	Project Name	Description
Object Storage	Swift	Stores and retrieves arbitrary unstructured data objects via a *RESTful*, HTTP based API. It is highly fault tolerant with its data replication and scale-out architecture. Its implementation is not like a file server with mountable directories. In this case, it writes objects and files to multiple drives, ensuring the data is replicated across a server cluster.
Block Storage	Cinder	Provides persistent block storage to running instances. Its pluggable driver architecture facilitates the creation and management of block storage devices.

OpenStack services

Shared services	Project Name	Description
Identity service	Keystone	Provides an authentication and authorization service for other OpenStack services. Provides a catalog of endpoints for all OpenStack services.
Image service	Glance	Stores and retrieves virtual machine disk images. OpenStack Compute makes use of this during instance provisioning.
Telemetry	Ceilometer	Monitors and meters the OpenStack cloud for billing, benchmarking, scalability, and statistical purposes.
Higher-level services		
Orchestration	Heat	Orchestrates multiple composite cloud applications by using either the native *HOT* template format or the AWS CloudFormation template format, through both an OpenStack-native REST API and a CloudFormation-compatible Query API

And many more… For complete view see https://www.openstack.org/software/

Keystone (Identity Management)

- **Keystone** is the identity service used by OpenStack for
  - Authentication
  - Authorization
- Supported protocols
  - Lightweight Directory Active Protocol (LDAP)
  - Federation AuthN/AuthZ via OIDC/OAuth
- Keystone primary functions
  - Service catalogue
  - User management

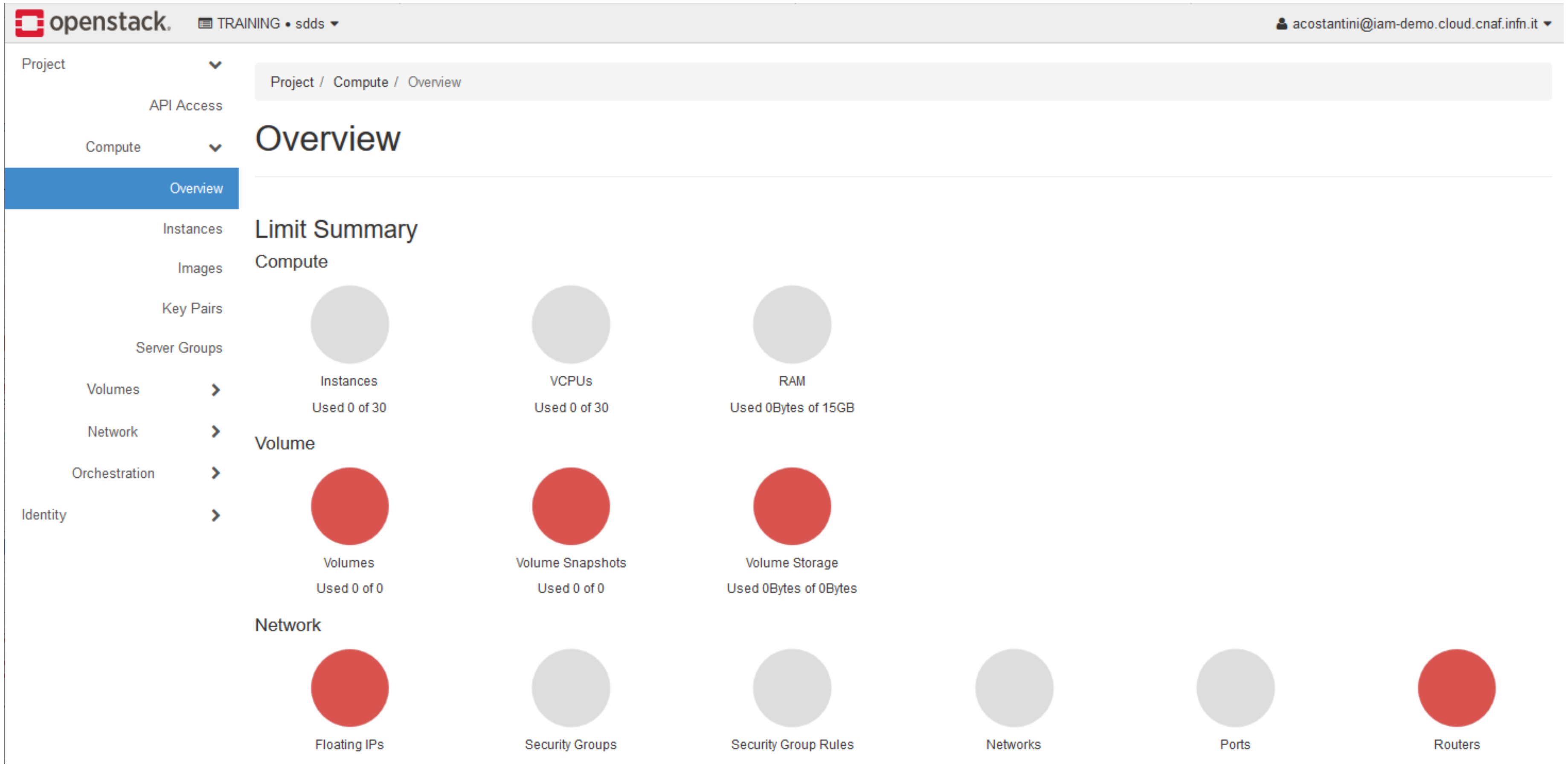
Domain, Project, Domain, Users, Roles

- **Project (Tenant)**
  - Base unit of “**ownership**” in OpenStack
  - All resources in OpenStack should be owned by a specific project
  - A project must be owned by a specific domain
- **Domain**
  - Collection of projects, groups and users that defines administrative boundaries for managing OpenStack Identity entities
- **Users**
  - OpenStack Identity - entities represent individual API consumers and are owned by a specific domain.
  - OpenStack Compute, a user can be associated with roles, projects, or both.
- **Roles**
  - A personality that a user assumes to perform a specific set of operations.
  - A role includes a set of rights and privileges.
  - A user assuming that role inherits those rights and privileges.

Horizon (Dashboard)

- **Horizon** provides the web interface for admin and final users
  - Written in Django (bit.ly/2YIce7E), a framework for the development of webapps in Python.
- CLI also available
- Use your own dashboard
  - OpenStack services based on APIs

Dashboard



openstack. TRAINING • sdds

acostantini@iam-demo.cloud.cnaf.infn.it

Project

API Access

Compute

Overview

Instances

Images

Key Pairs

Server Groups

Volumes

Network

Orchestration

Identity

Project / Compute / Overview

Overview

Limit Summary

Compute

Instances
Used 0 of 30

VCPUs
Used 0 of 30

RAM
Used 0Bytes of 15GB

Volume

Volumes
Used 0 of 0

Volume Snapshots
Used 0 of 0

Volume Storage
Used 0Bytes of 0Bytes

Network

Floating IPs

Security Groups

Security Group Rules

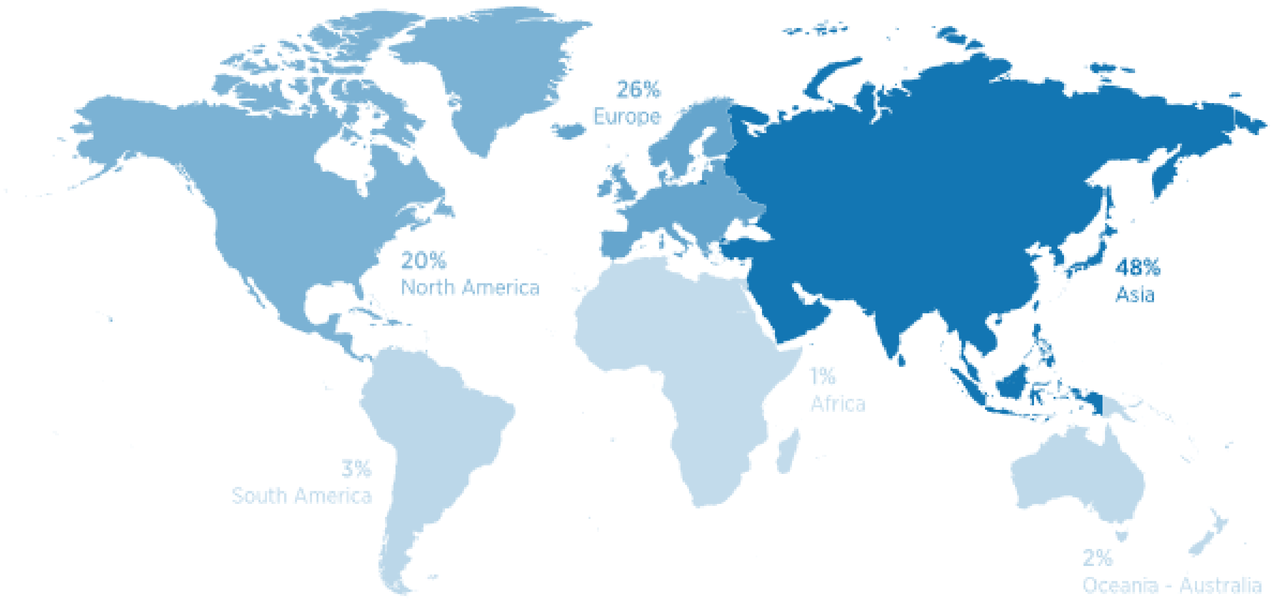
Networks

Ports

Routers

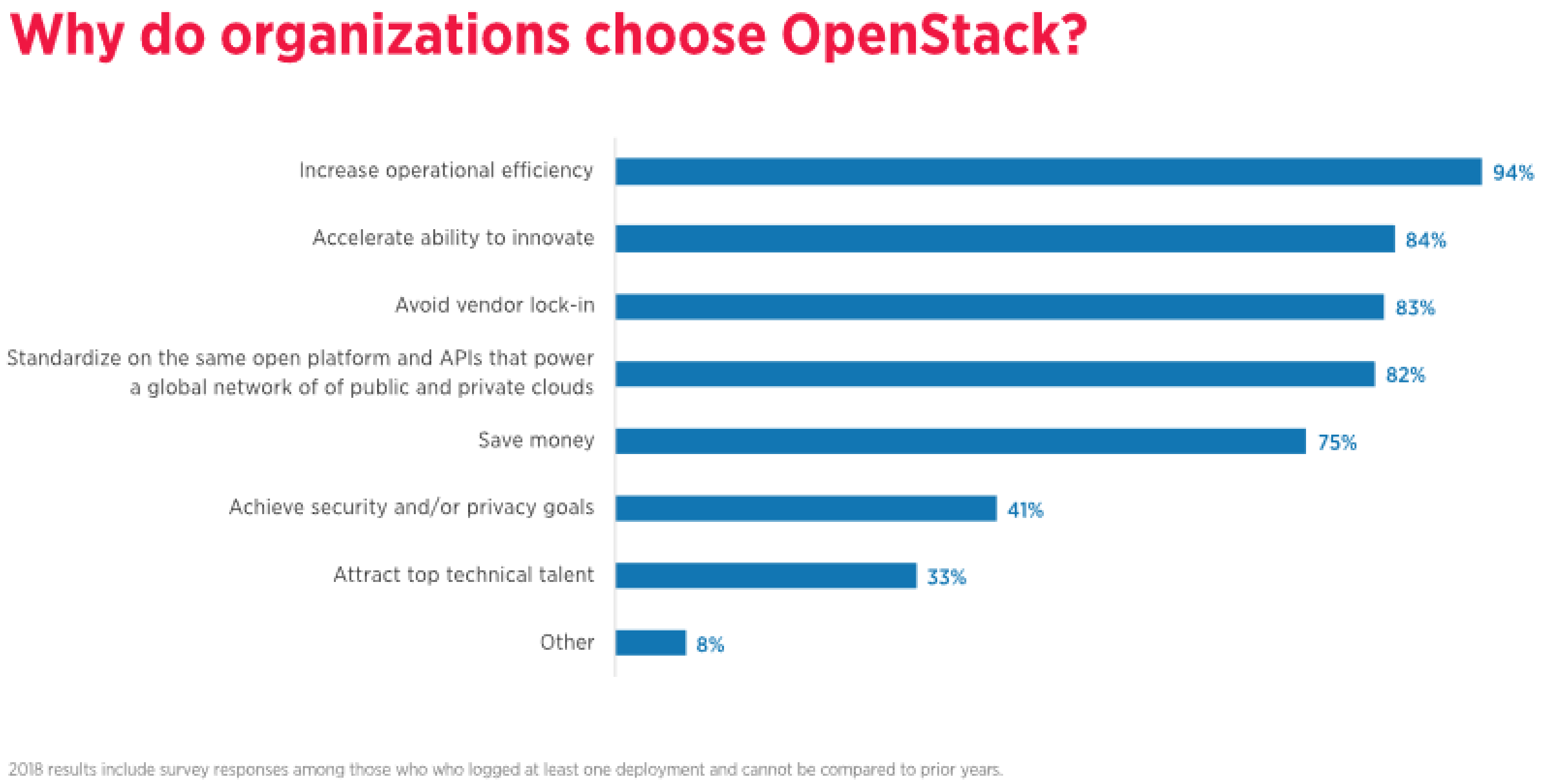
OpenStack survey

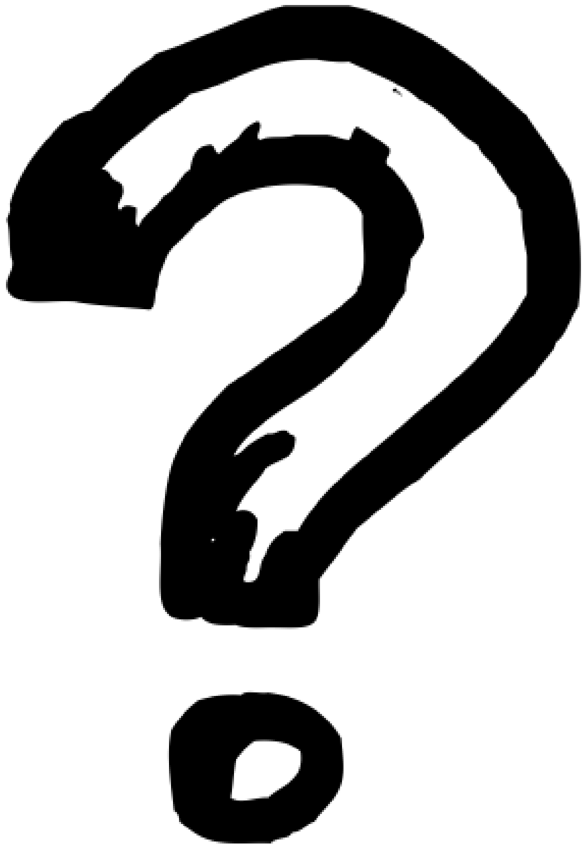
Where in the world are OpenStack users?



2018 shows significant increases in respondents in Asia and significant decreases in N. America, compared to both 2016 and 2017.
Sample size of 687.

OpenStack survey





Hands-on

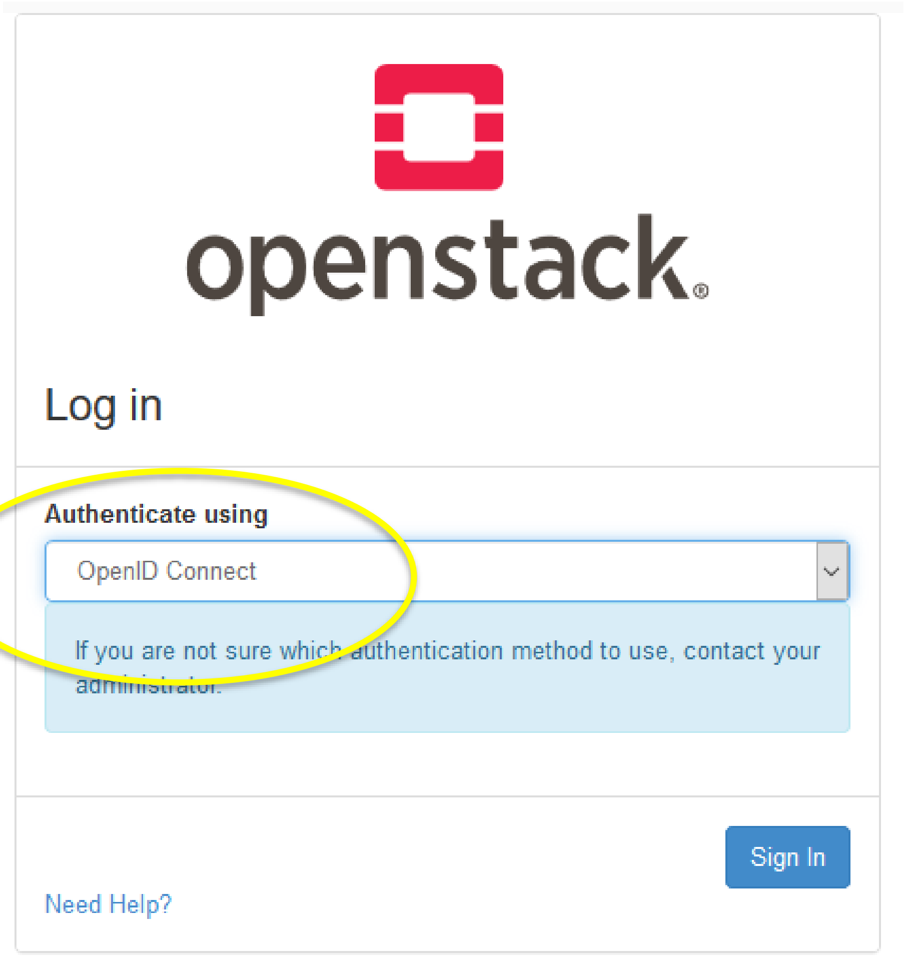
Deploy a VM in the OpenStack cloud infrastructure

- The access to the resources will be valid for the duration of the course

Deploy your first VM

Access the Openstack infrastructure

- https://cloud-dashboard.cnaf.infn.it/dashboard


Log in

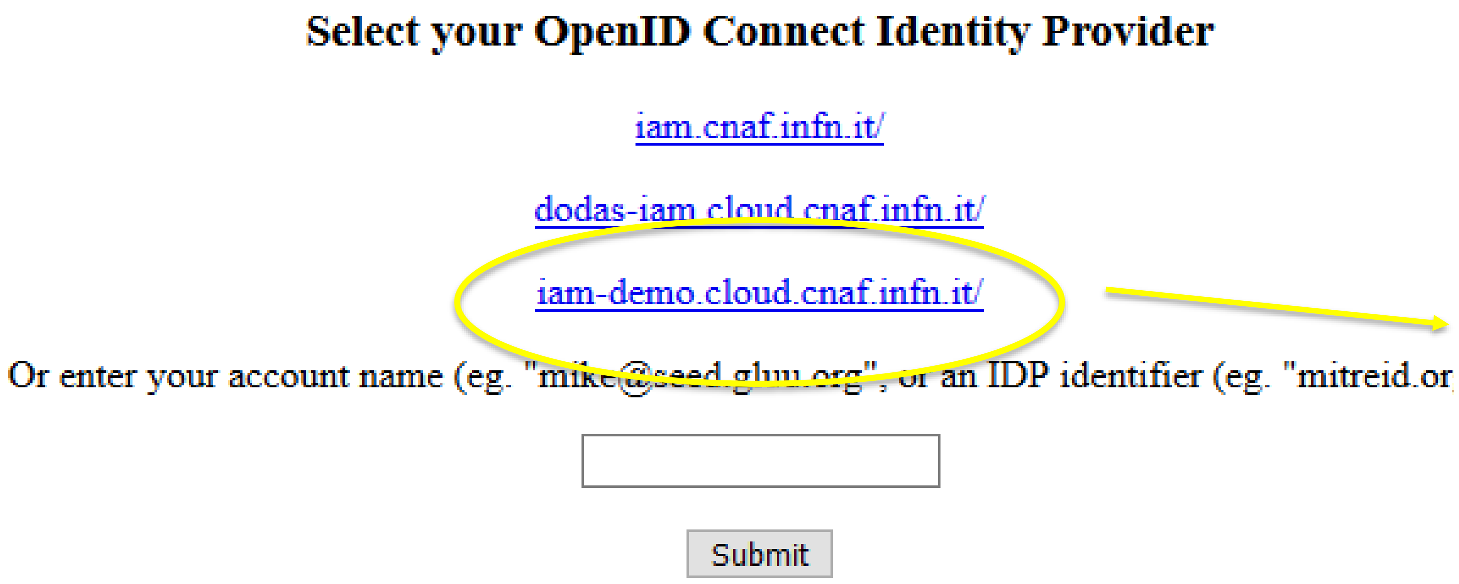
Authenticate using

OpenID Connect

If you are not sure which authentication method to use, contact your administrator.

Deploy your first VM

- Select iam-demo IDP



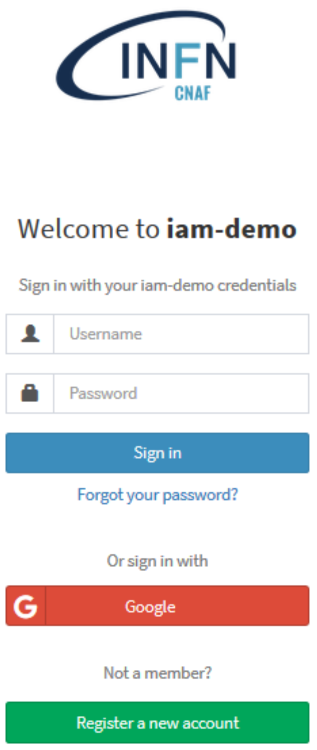
Select your OpenID Connect Identity Provider

iam.cnaf.infn.it/

dodas-iam.cloud.cnaf.infn.it/

iam-demo.cloud.cnaf.infn.it/

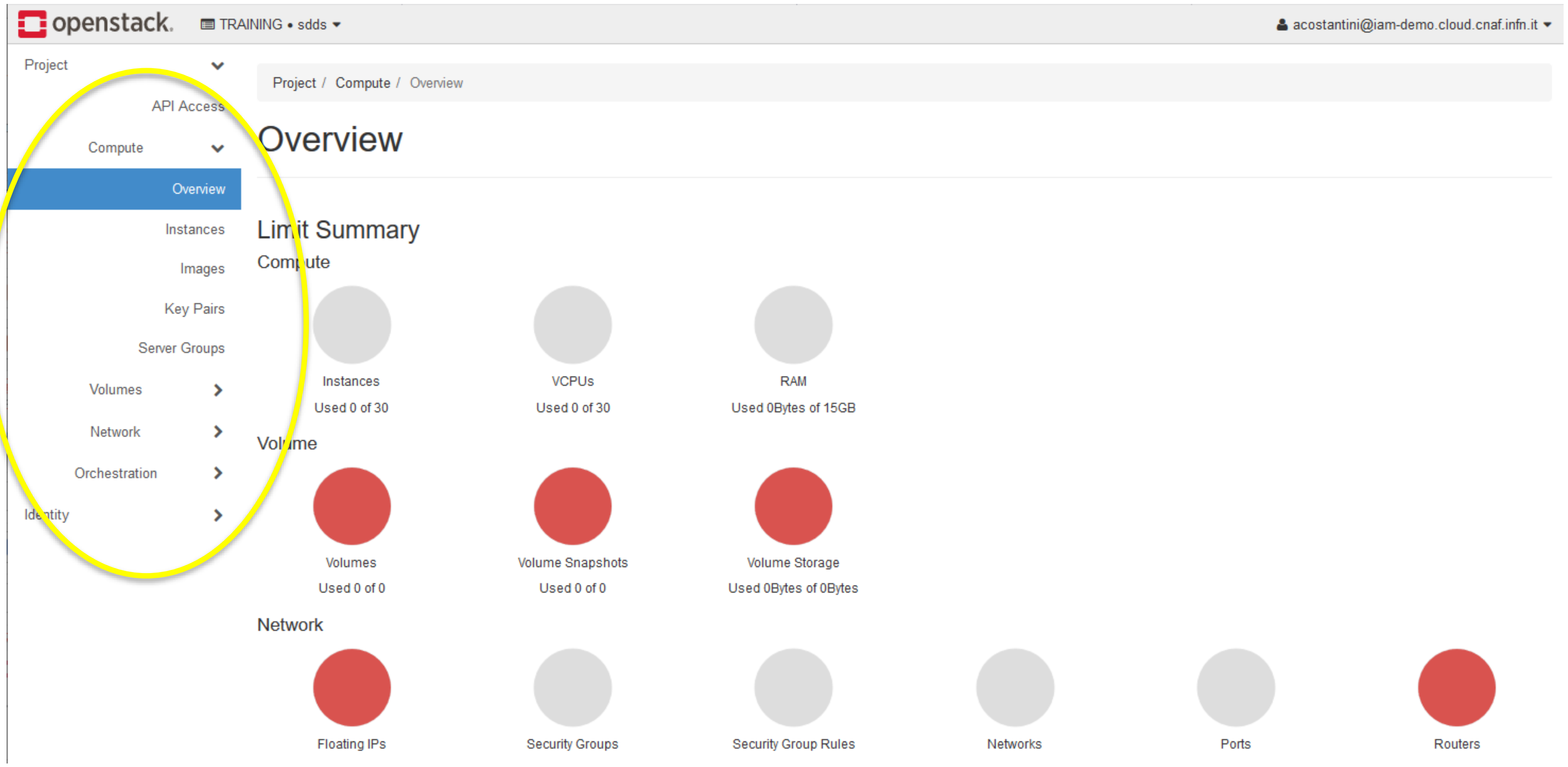
Or enter your account name (eg. "mike@seed.gluu.org", or an IDP identifier (eg. "mitreid.or



Welcome to **iam-demo**

Sign in with your iam-demo credentials

Resources overview



Overview

Limit Summary

Compute

Instances
Used 0 of 30

VCPUs
Used 0 of 30

RAM
Used 0Bytes of 15GB

Volume

Volumes
Used 0 of 0

Volume Snapshots
Used 0 of 0

Volume Storage
Used 0Bytes of 0Bytes

Network

Floating IPs

Security Groups

Security Group Rules

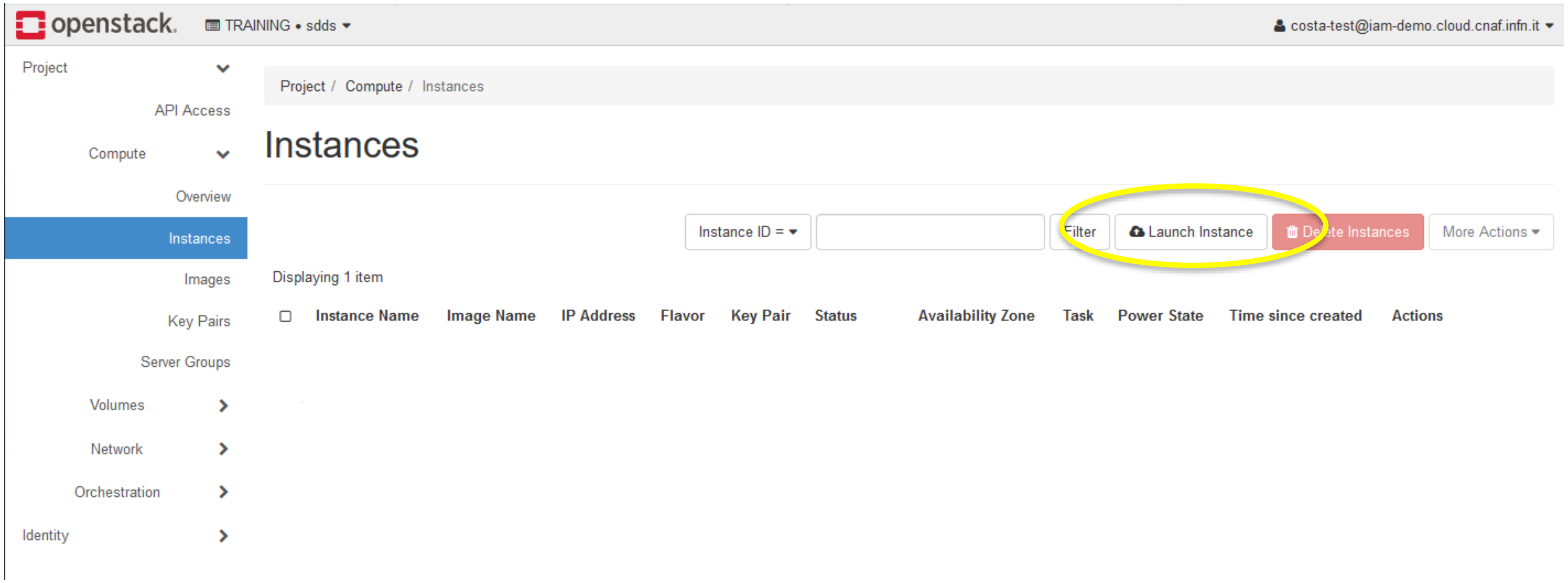
Networks

Ports

Routers

Instance

- Deploy new instance

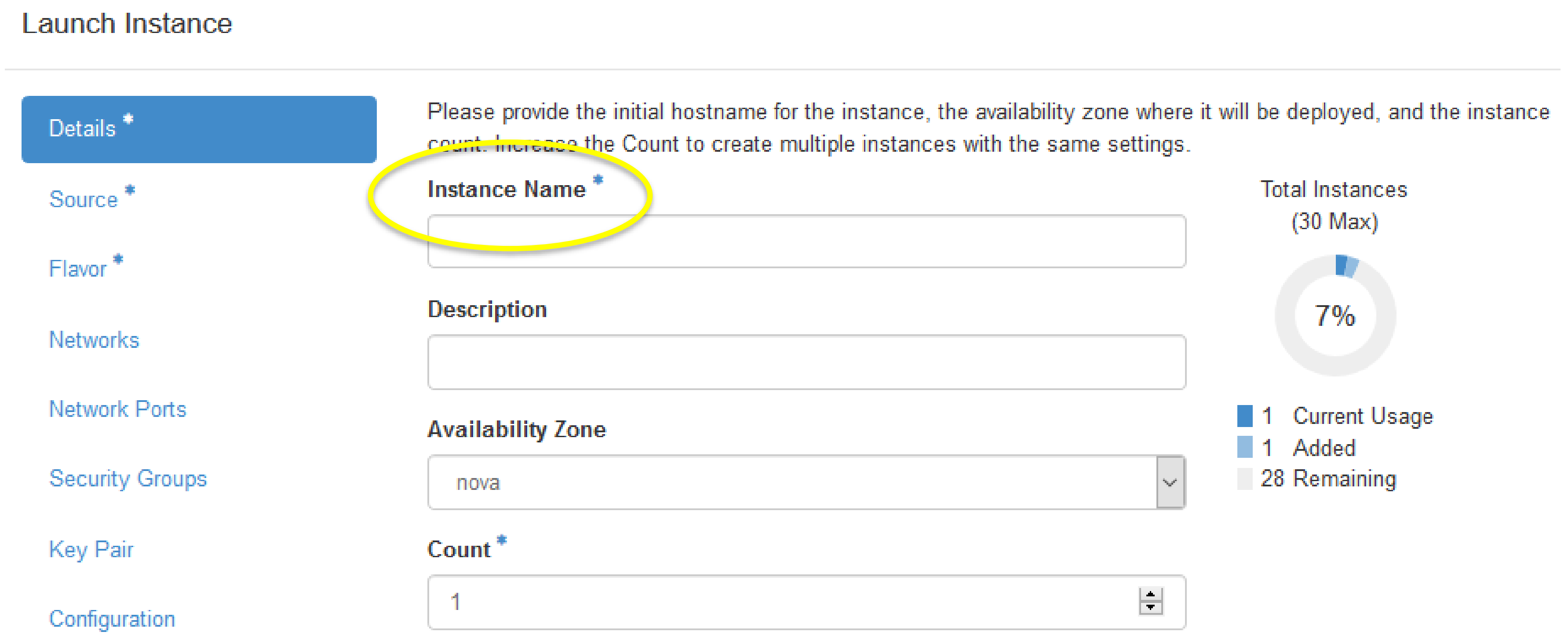


Instances

Displaying 1 item

Instance Name	Image Name	IP Address	Flavor	Key Pair	Status	Availability Zone	Task	Power State	Time since created	Actions

Define Instance name



Launch Instance

Please provide the initial hostname for the instance, the availability zone where it will be deployed, and the instance count. Increase the Count to create multiple instances with the same settings.

Instance Name *

Description

Availability Zone: nova

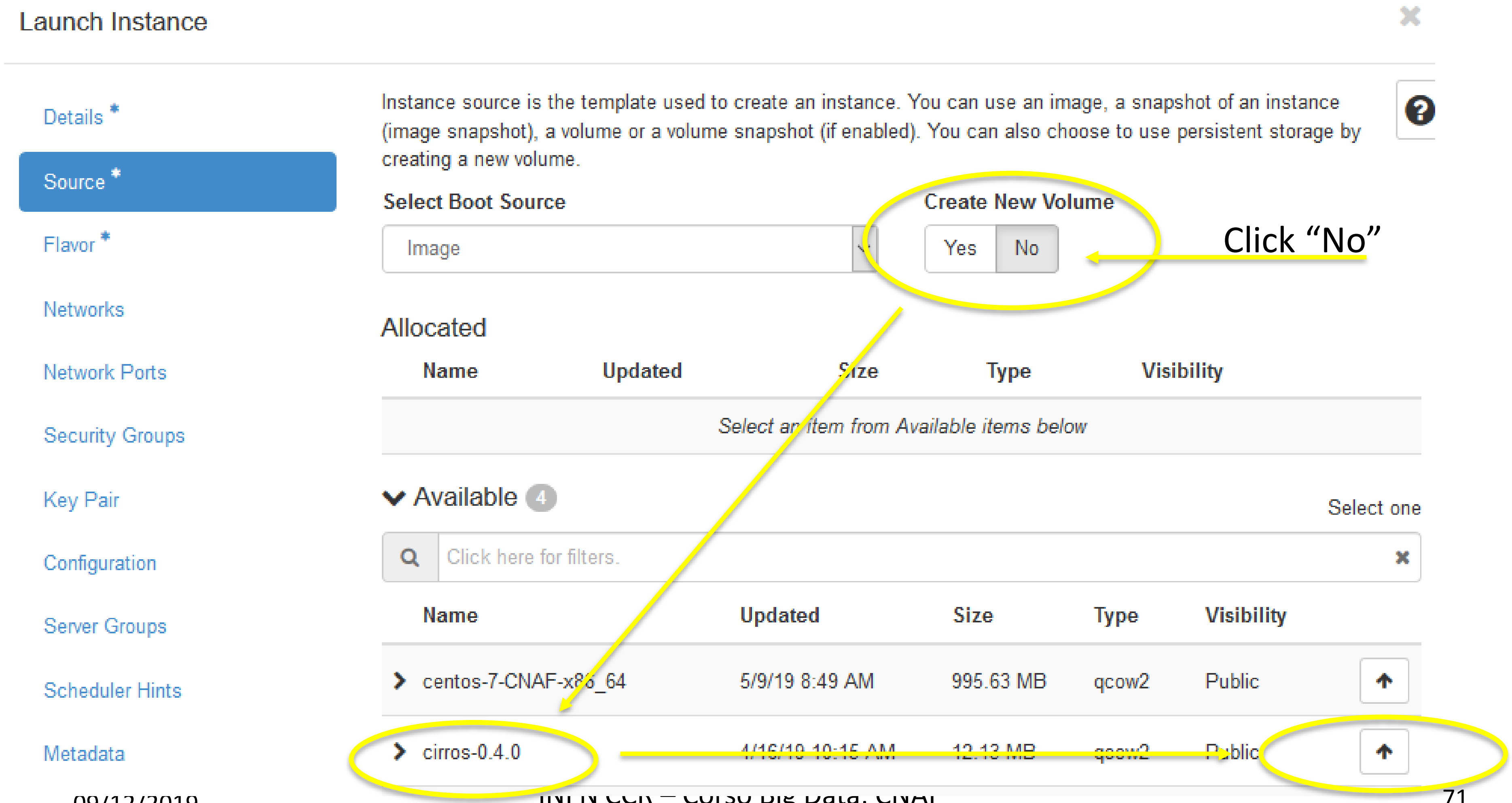
Count *: 1

Total Instances (30 Max)

7%

🟦	1	Current Usage
🟦	1	Added
⬜	28	Remaining

Select Image (cirros)


Launch Instance

Instance source is the template used to create an instance. You can use an image, a snapshot of an instance (image snapshot), a volume or a volume snapshot (if enabled). You can also choose to use persistent storage by creating a new volume.

Select Boot Source: Image

Create New Volume: Yes [✓ No]

Click "No"

Allocated

Name	Updated	Size	Type	Visibility
Select an item from Available items below				

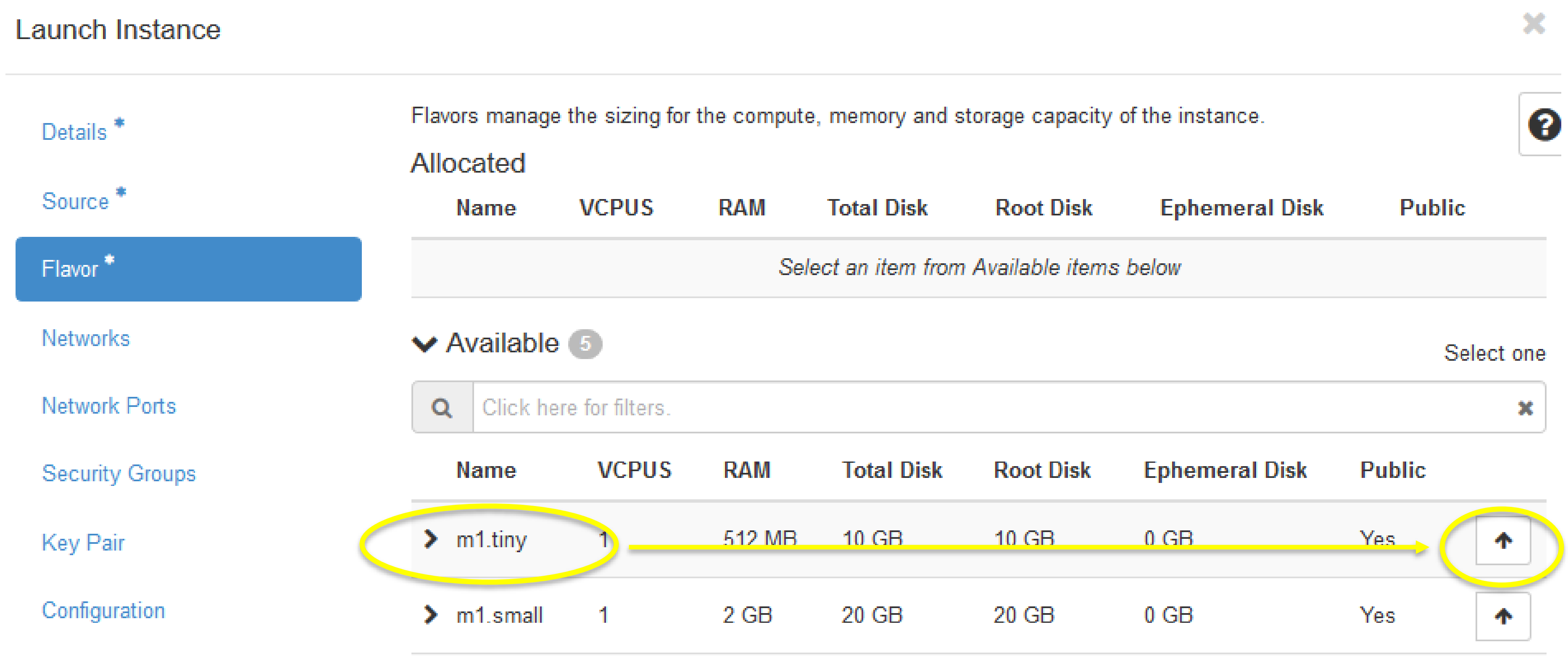
Available 4

Select one

Click here for filters.

Name	Updated	Size	Type	Visibility
centos-7-CNAF-x86_64	5/9/19 8:49 AM	995.63 MB	qcow2	Public
cirros-0.4.0	4/16/19 10:15 AM	12.13 MB	qcow2	Public

Select Flavour (m1.tiny)



Launch Instance

Flavors manage the sizing for the compute, memory and storage capacity of the instance.

Allocated

Name	VCPUS	RAM	Total Disk	Root Disk	Ephemeral Disk	Public
Select an item from Available items below						

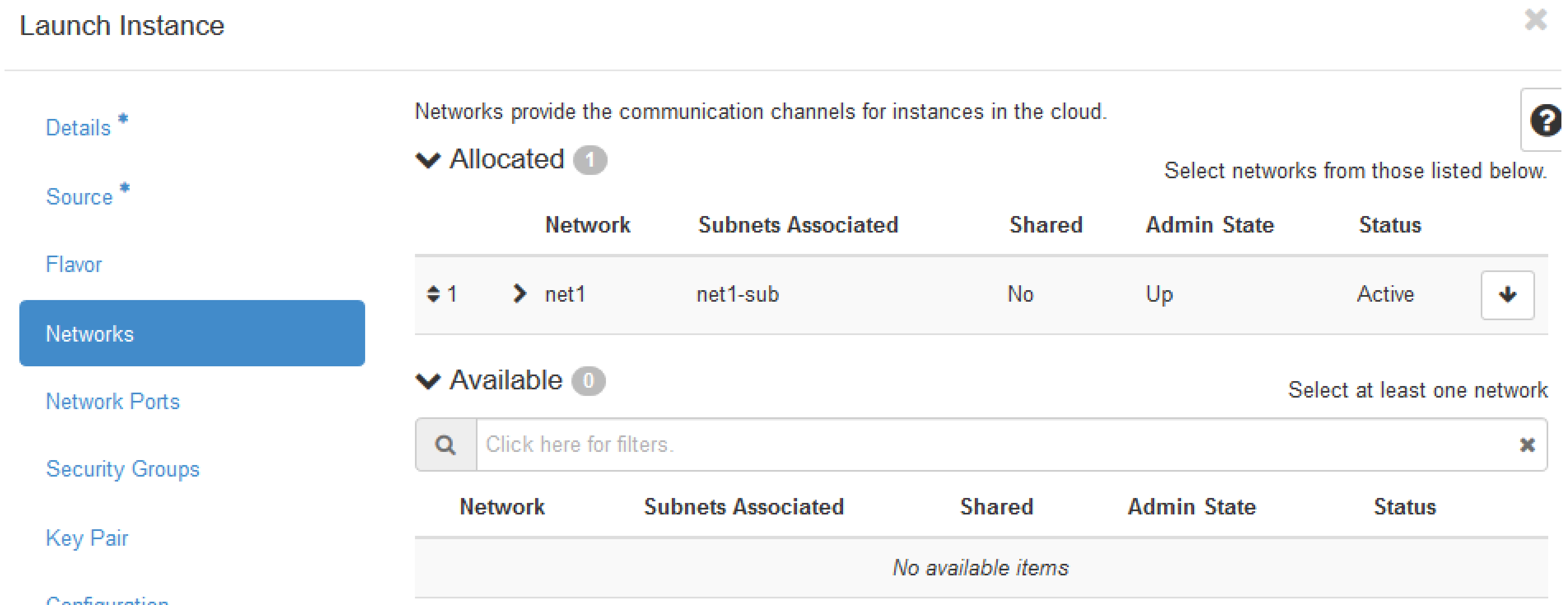
Available 5

Select one

Click here for filters.

Name	VCPUS	RAM	Total Disk	Root Disk	Ephemeral Disk	Public
m1.tiny	1	512 MB	10 GB	10 GB	0 GB	Yes
m1.small	1	2 GB	20 GB	20 GB	0 GB	Yes

Select Network (net1)


Launch Instance

Networks provide the communication channels for instances in the cloud.

Allocated 1

Select networks from those listed below.

	Network	Subnets Associated	Shared	Admin State	Status
1	net1	net1-sub	No	Up	Active

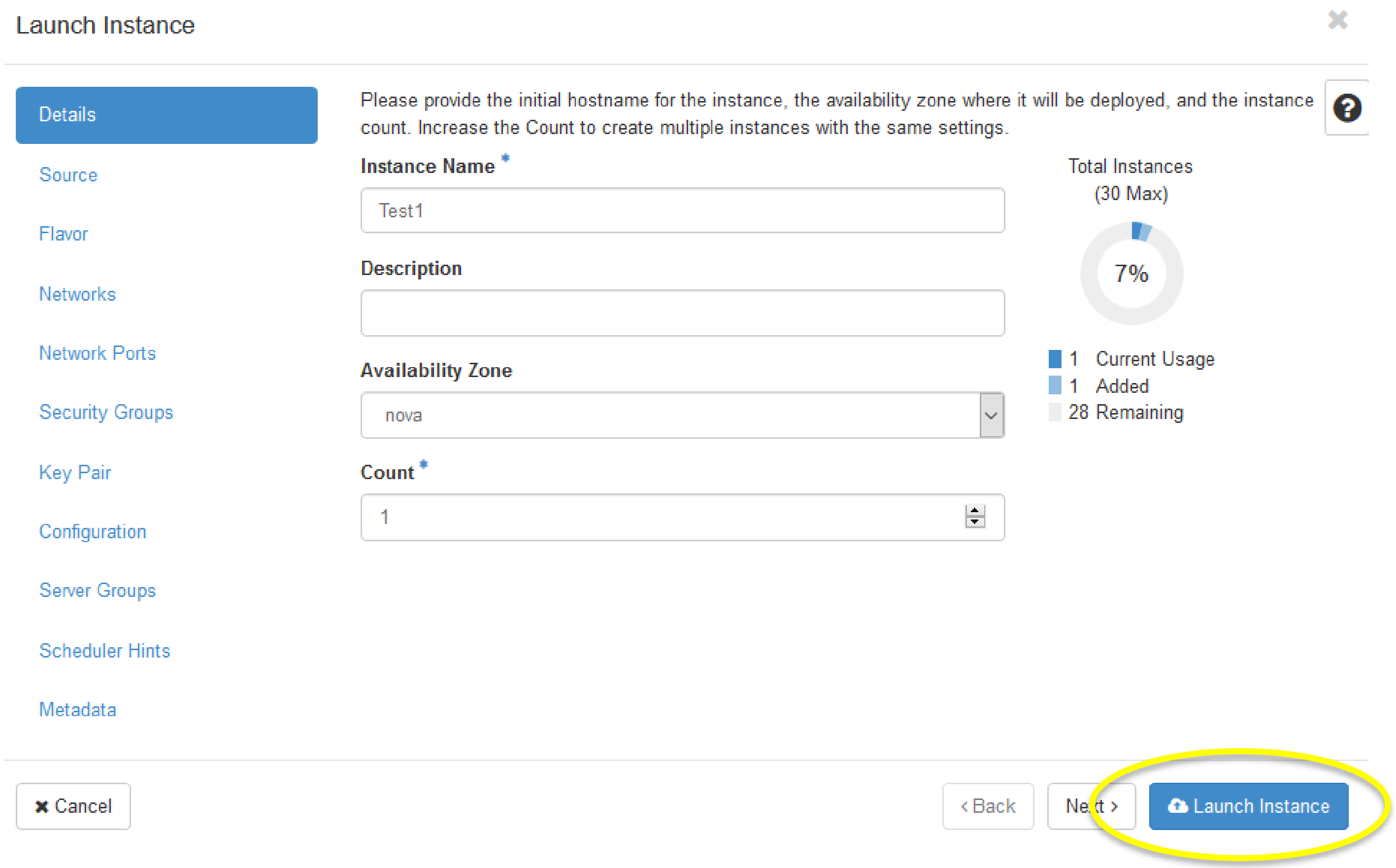
Available 0

Select at least one network

Click here for filters.

Network	Subnets Associated	Shared	Admin State	Status
No available items				

Launch Instance


Launch Instance

Please provide the initial hostname for the instance, the availability zone where it will be deployed, and the instance count. Increase the Count to create multiple instances with the same settings.

Instance Name *: Test1

Description:

Availability Zone: nova

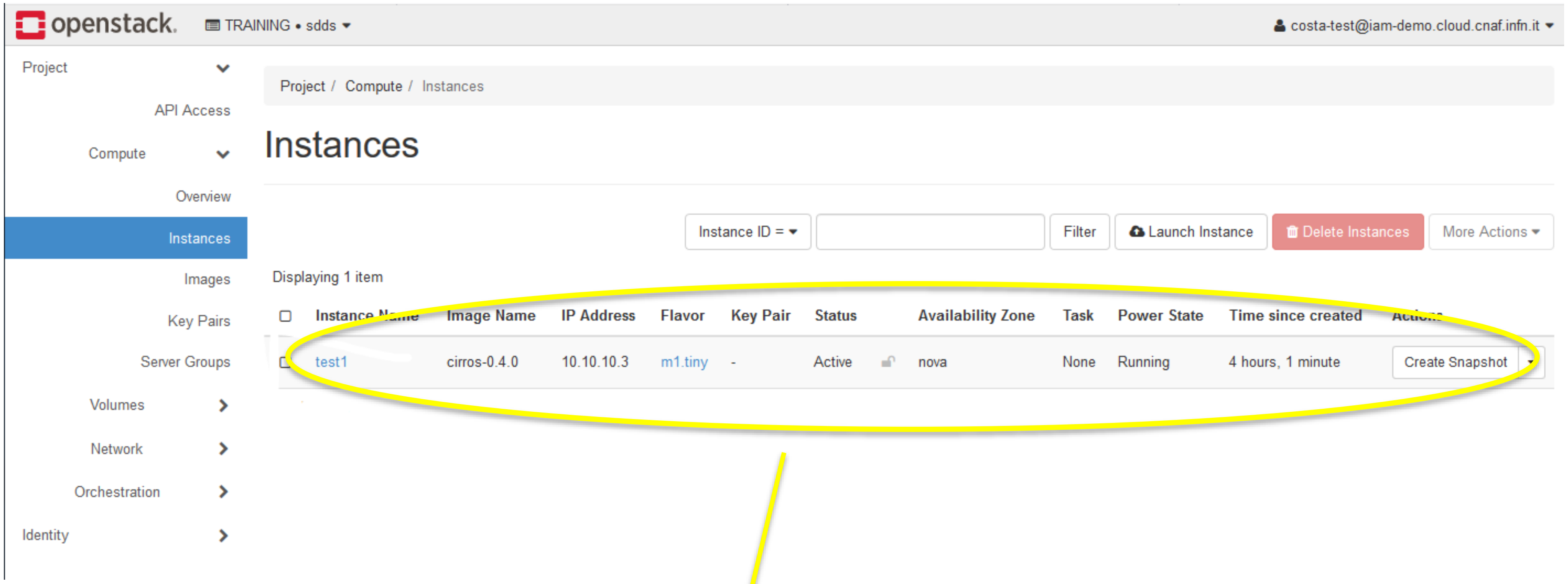
Count *: 1

Total Instances (30 Max)

7%

1	Current Usage
1	Added
28	Remaining

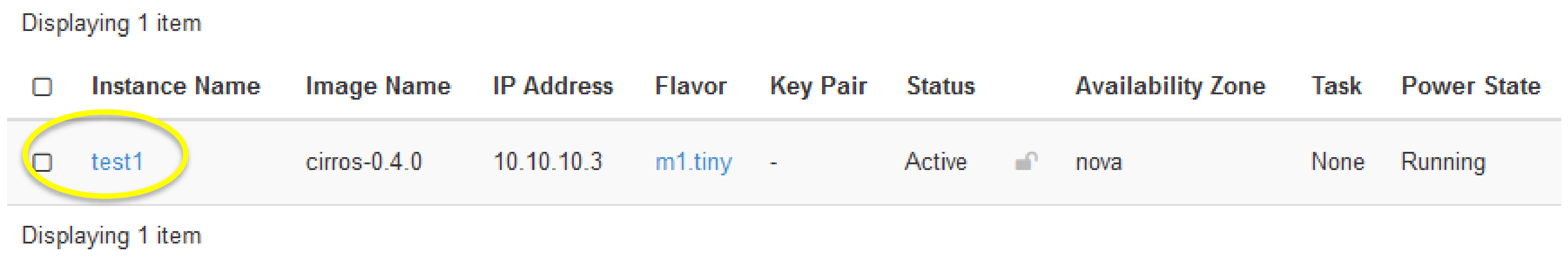
Explore your Instance



Instances

Displaying 1 item

	Instance Name	Image Name	IP Address	Flavor	Key Pair	Status	Availability Zone	Task	Power State	Time since created	Actions
☐	test1	cirros-0.4.0	10.10.10.3	m1.tiny	-	Active	nova	None	Running	4 hours, 1 minute	Create Snapshot

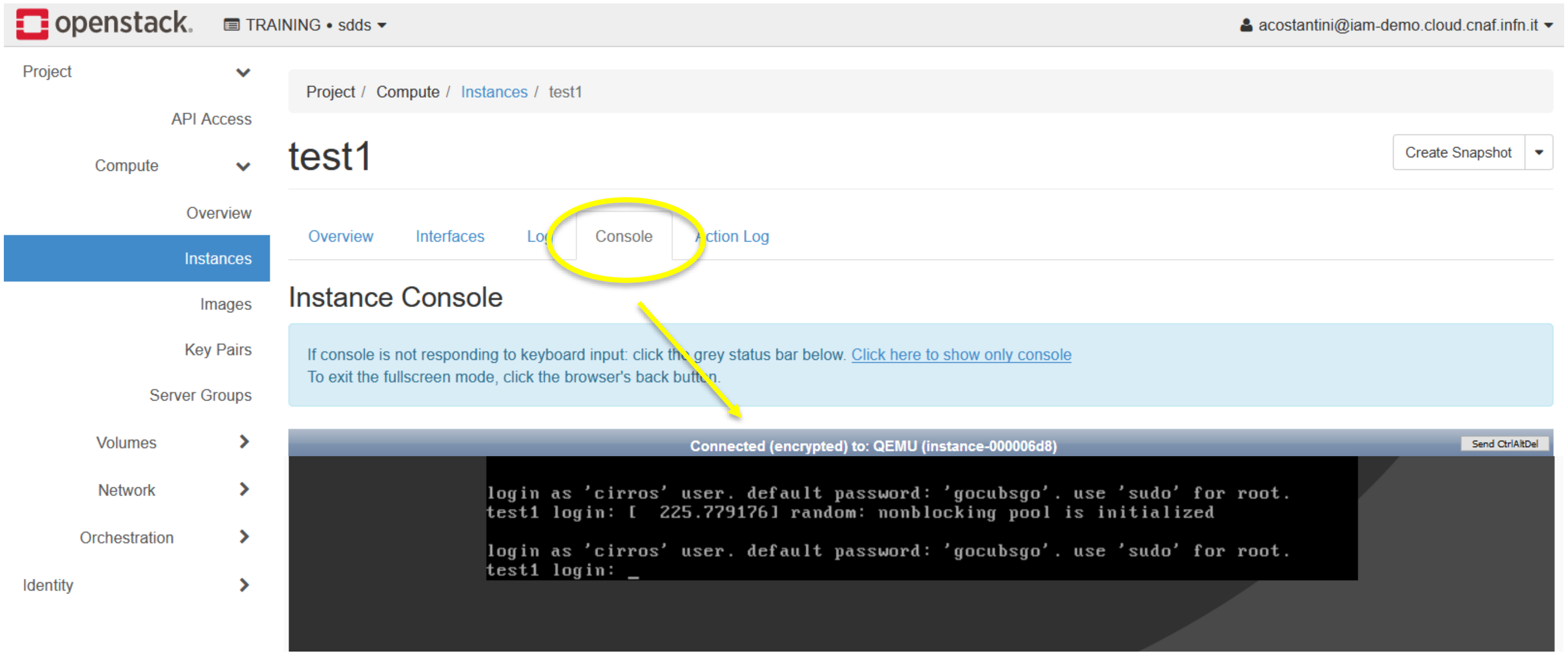


Displaying 1 item

☐	Instance Name	Image Name	IP Address	Flavor	Key Pair	Status	Availability Zone	Task	Power State
☐	test1	cirros-0.4.0	10.10.10.3	m1.tiny	-	Active	nova	None	Running

Displaying 1 item

Explore your Instance



If console is not responding to keyboard input: click the grey status bar below. Click here to show only console
To exit the fullscreen mode, click the browser's back button.

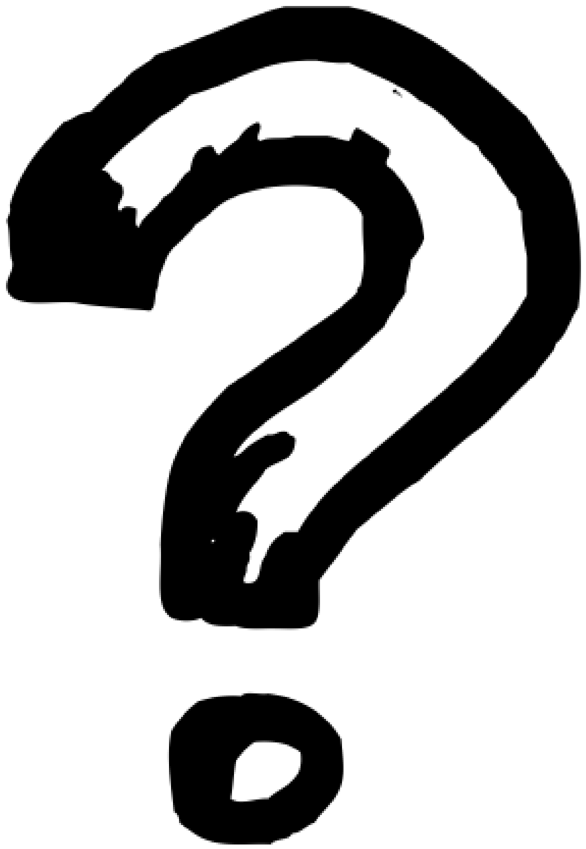
Connected (encrypted) to: QEMU (instance-000006d8)

Send CtrlAltDel

```
login as 'cirros' user. default password: 'gocubsgo'. use 'sudo' for root.
test1 login: [  225.779176] random: nonblocking pool is initialized

login as 'cirros' user. default password: 'gocubsgo'. use 'sudo' for root.
test1 login: _
```

Login
User: cirros
Password: gocubsgo



Thank you for your attention!

- **Contact**:
- alessandro.costantini@cnaf.infn.it

Extra slides

Application migration to Cloud

- Migration of an application from an existing data center to a Cloud infrastructure

- Which technical and business factors move to migration?
  - Cost reduction → resource pooling, pay-per-use
  - “Business agility” → deployment simplification
  - Management saving → performance (e.g. auto-scaling), delegation of operations

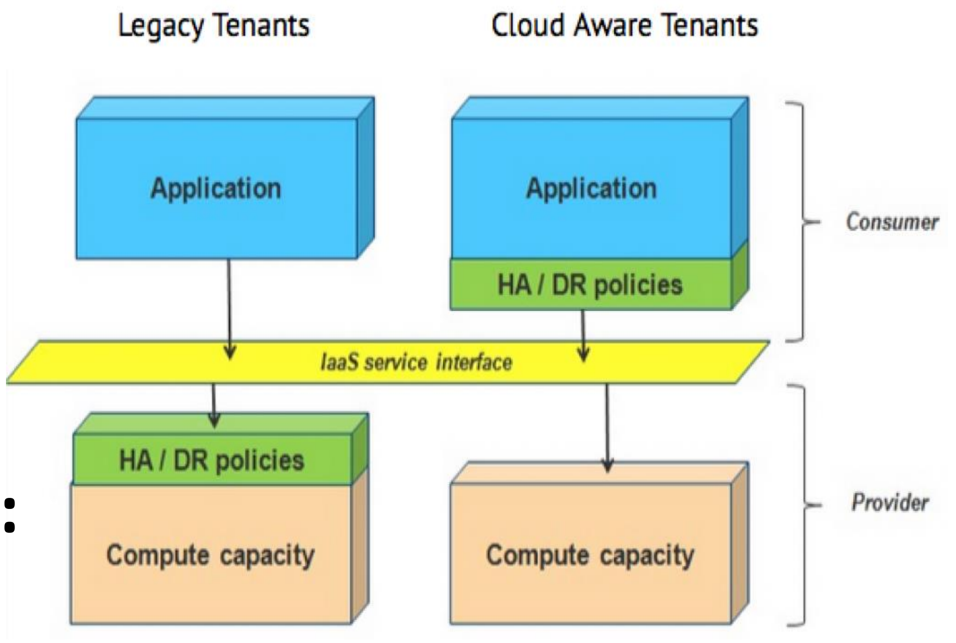
- Public or private Cloud?
  - WAN traffic? (tipically expensive)
  - Security?
  - Integration with other *legacy* applications?

Stateless vs Stateful service

- A **stateless** service
  - provides a response without storing any state.
  - E.g.: simple Web server
- A **stateful** service
  - provides response on the basis of the state of the previous requests
  - E.g.: Web server with a shopping cart

Cloud-friendly applications

- **“Cloud-aware” applications:**
  - Distributed
  - Stateless
  - Fail-over in the app
  - Scaling in the app

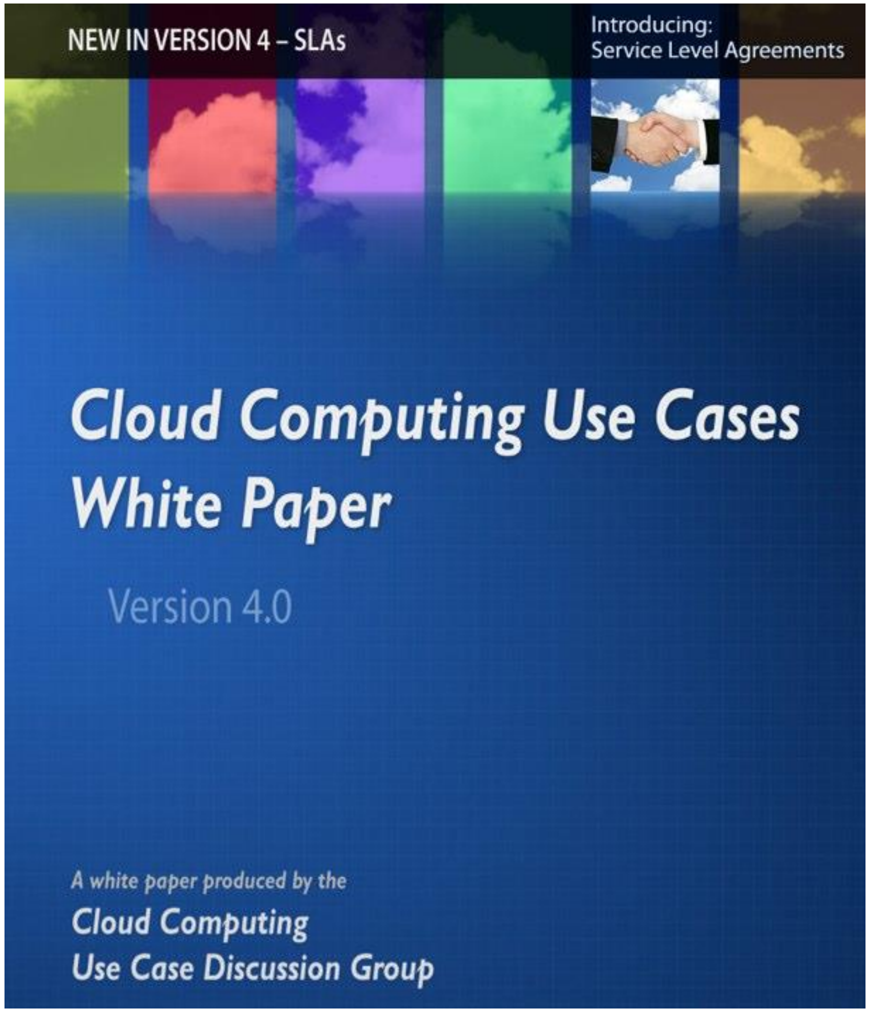


- **“Legacy” applications:**
  - Stateful
  - Monolitic, no orizontal scalability
  - Fail-over in the infrastructure
  - Scaling in the infrastructure

Fonte: VMware

Cloud use cases

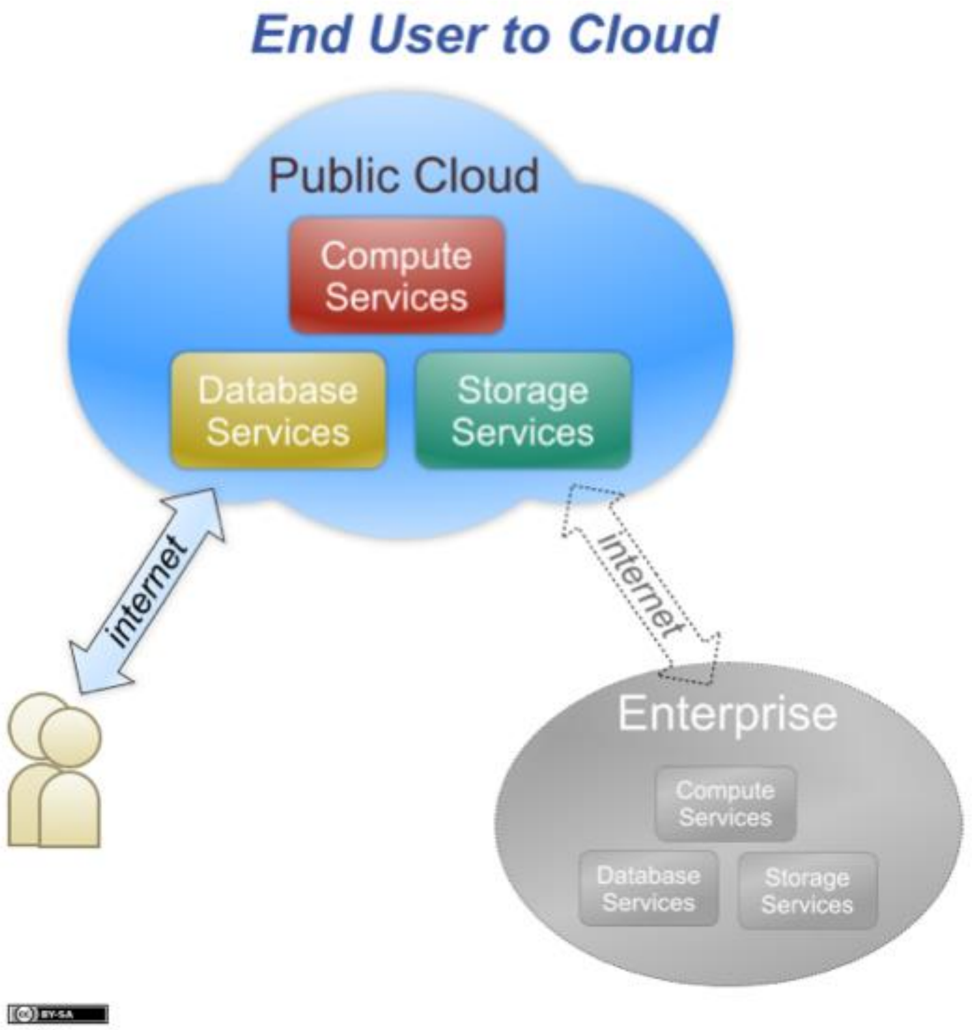
- https://goo.gl/qxRtrw
- 7 principal cases:
  - End user → Cloud
  - Enterprise → Cloud → end user
  - Enterprise → Cloud
  - Enterprise → Cloud → enterprise
  - Private Cloud
  - Changing Cloud vendors
  - Hybrid Cloud


NEW IN VERSION 4 – SLAs
Introducing: Service Level Agreements
Cloud Computing Use Cases White Paper
Version 4.0
A white paper produced by the Cloud Computing Use Case Discussion Group

Source: bit.ly/305Jfre

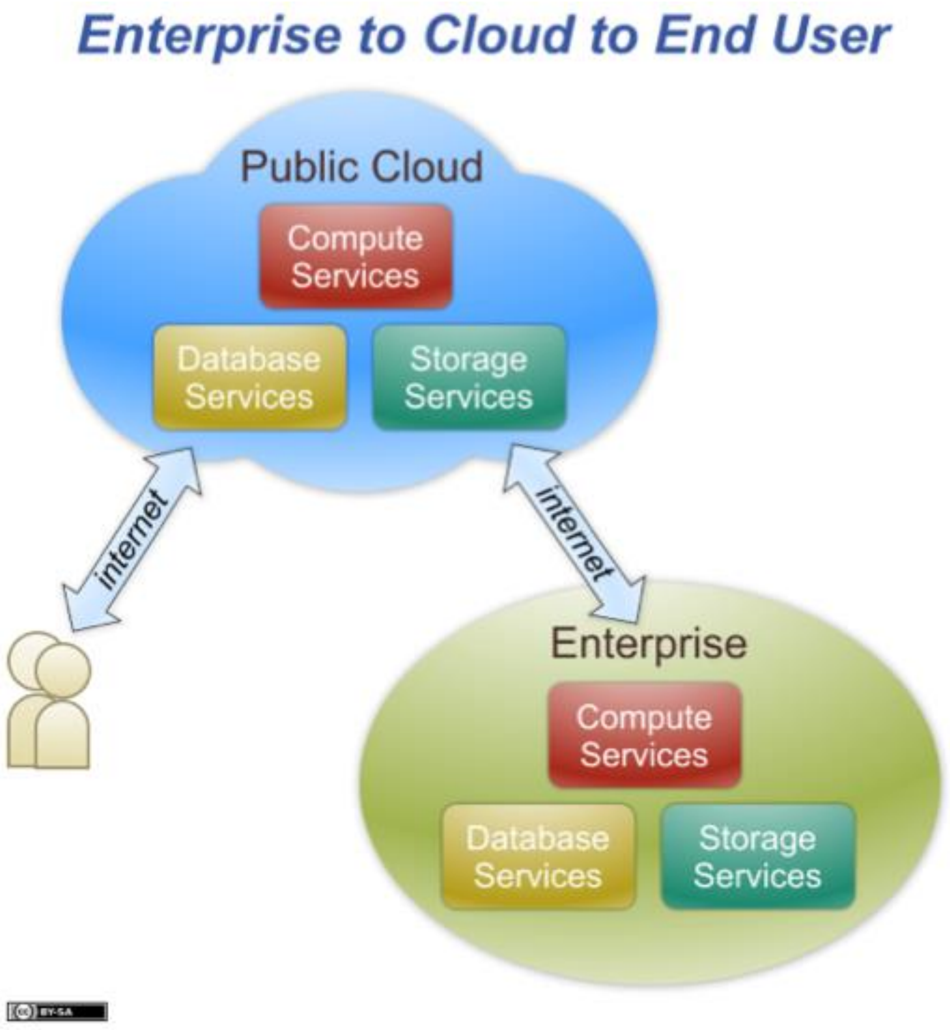
End user → Cloud

- The user accesses data or application into Cloud (e.g. email, social networks)
- Key points:
  - **Identity**
    - Authentication has to be provided
  - **Open client**
    - Access should not require particular technology
  - **Security/privacy**
  - **SLA are simpler than those with enterprise**



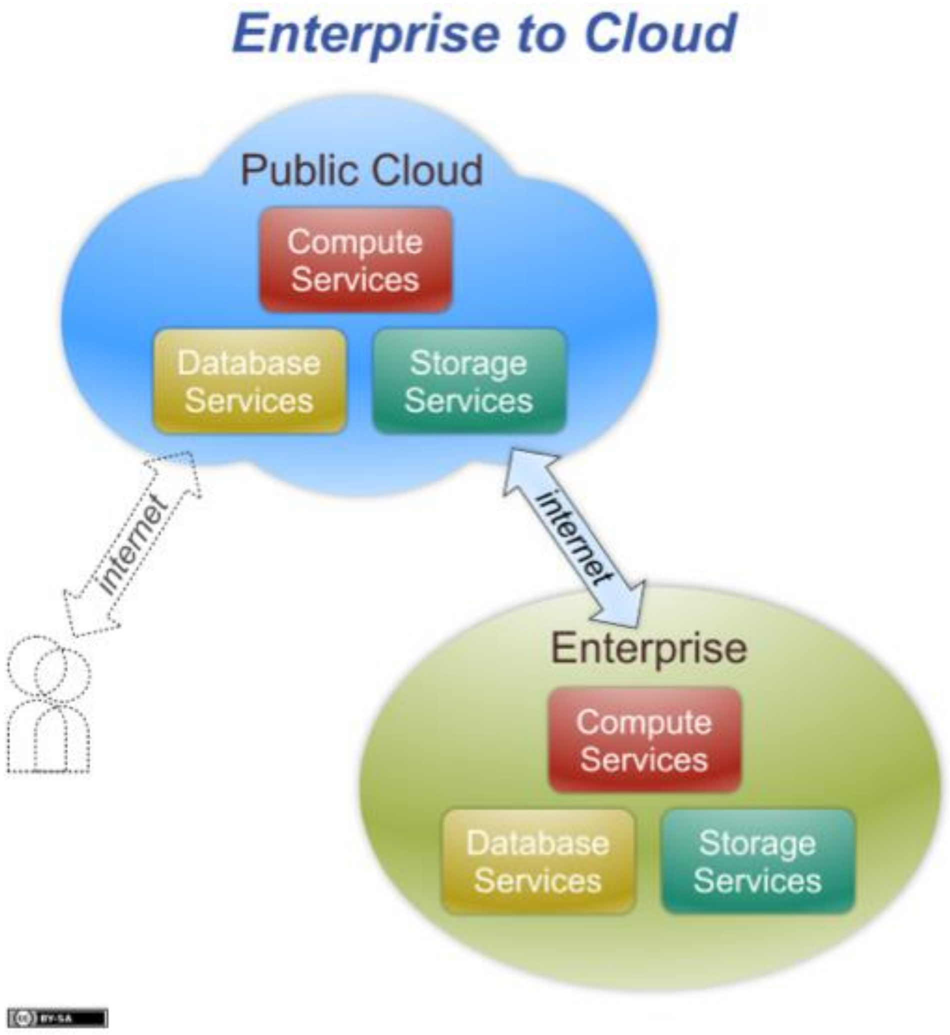
Enterprise → Cloud → End user

- An enterprise uses the Cloud to provide services to its users
- Key points:
  - Identity → federatation
    - an enterprise user is likely to have an identity within the enterprise
  - Location awareness (e.g. for legal issues)
  - Monitoring (for cost control)
  - Security
  - Common APIs (for different vendors)
  - SLA



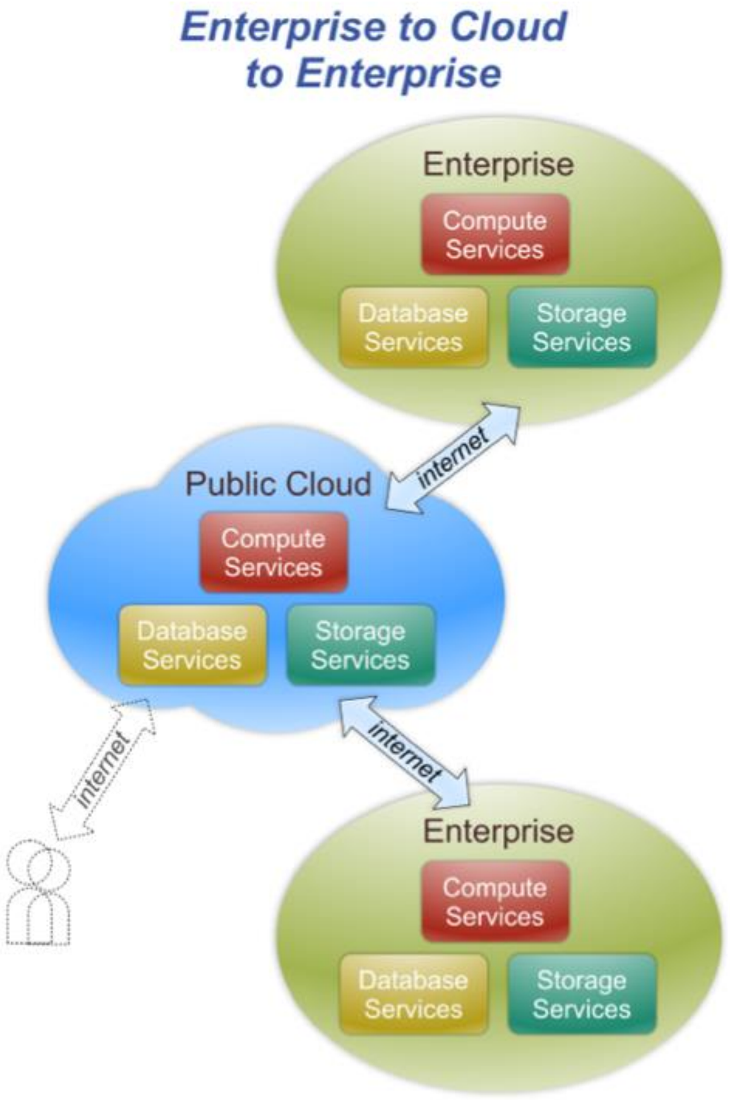
Enterprise → Cloud

- An enterprise uses the Cloud for its internal processes
- **Key points:**
  - Suppletive storage (e.g. for back-up)
  - "Cloud bursting" to supply peak demand
  - Cloud usage for some application (email, calendar, etc.)
  - Use of standards, avoiding vendor lock-in



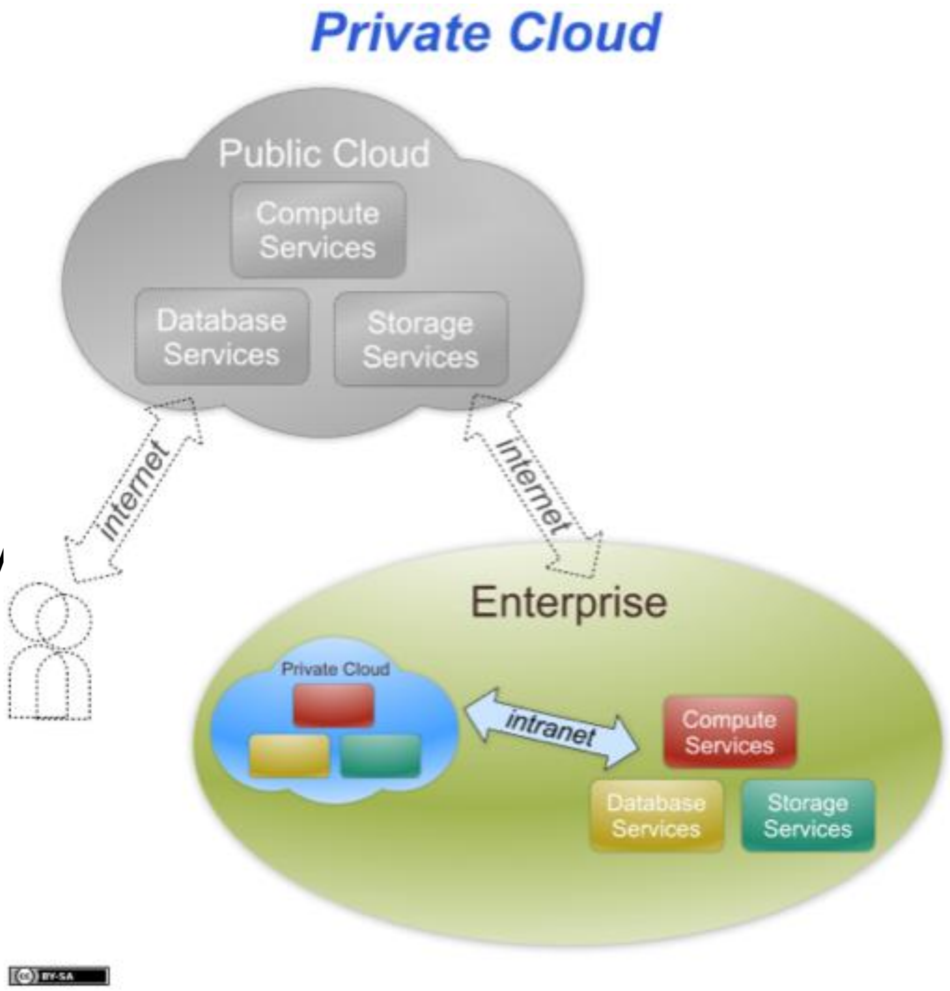
Enterprise → Cloud → Enterprise

- Two enterprises that use the same Cloud
- Key points:
  - Concurrency
    - For applications and data shared between different enterprises. If two enterprises are using the same cloud-hosted application, VM, middleware or storage, it's important that any changes made by either enterprise are done reliably



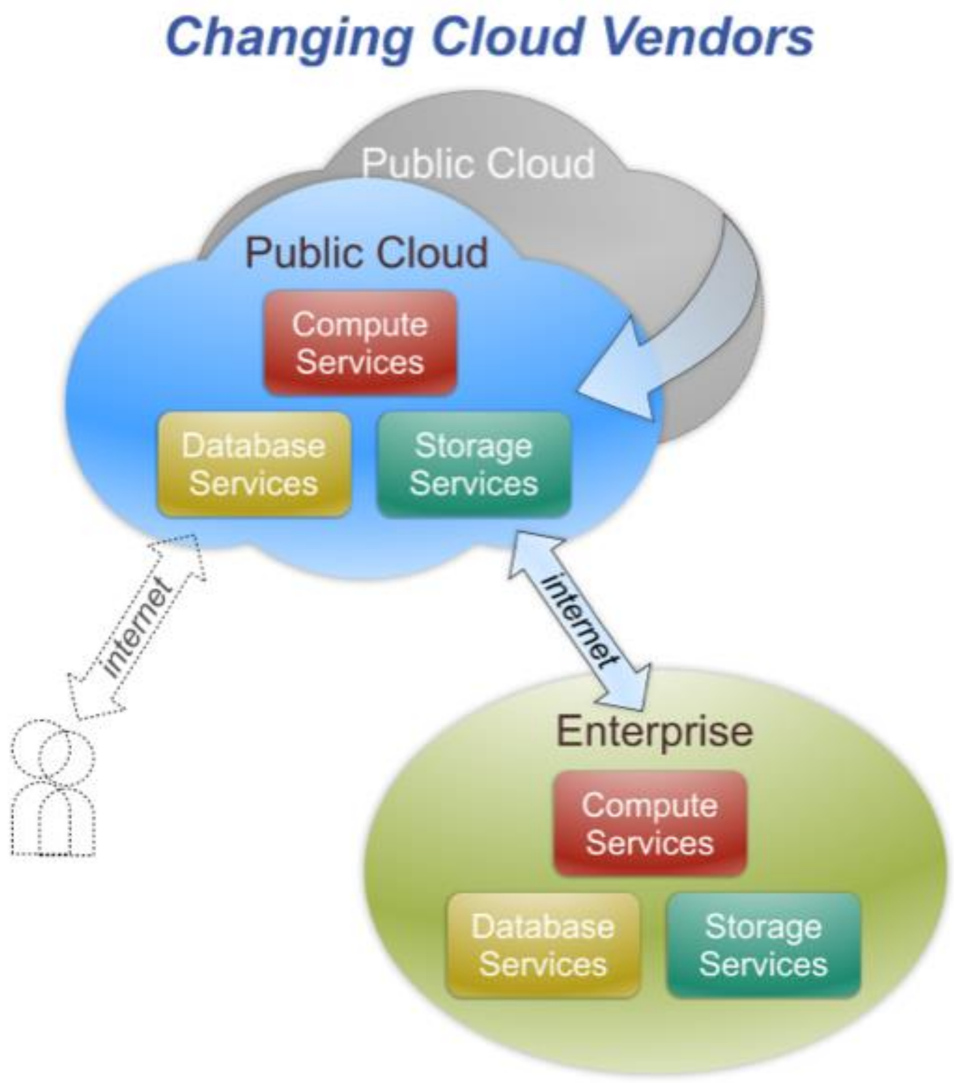
Private Cloud

- The cloud is contained within the enterprise
  - This is useful for large enterprises
- Does not require:
  - identity, federated identity location awareness, concurrency, industry standards, common APIs for Cloud middleware



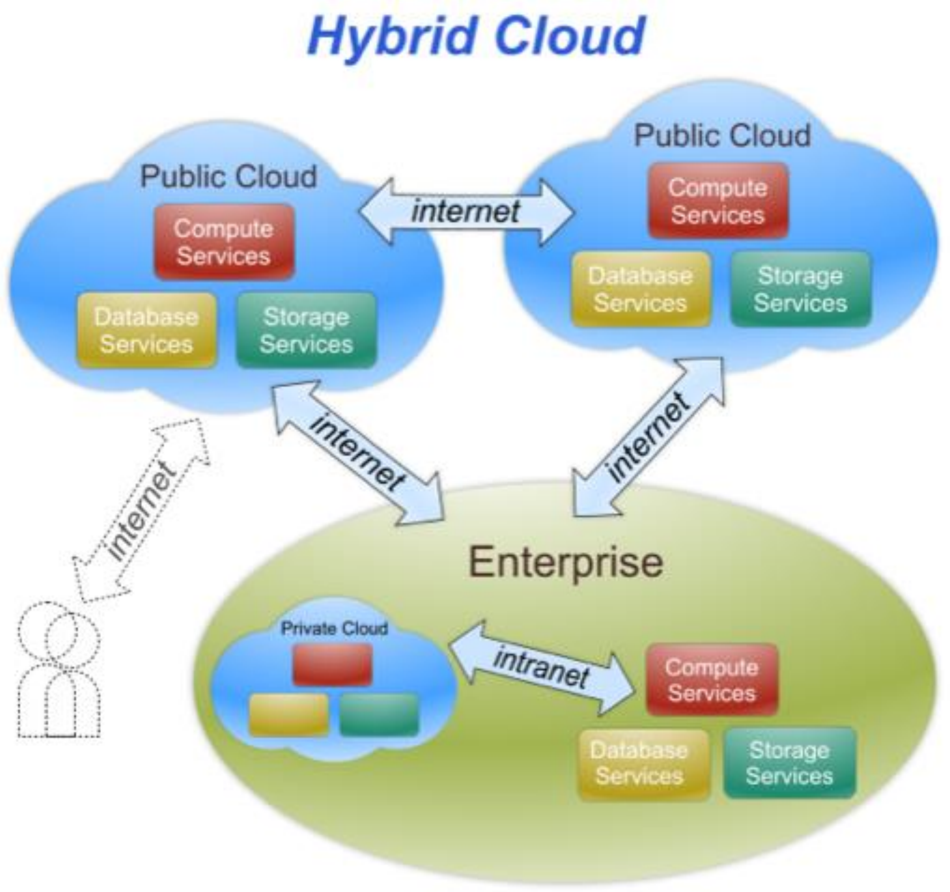
Changing Cloud vendors

- An enterpraise that want to change a Cloud vendor or add another
- Key point:
  - **Standardization**



Hybrid Cloud

- Using Cloud public and private together
- **Key point:**
  - For the end user this use case should be not different by the case End user → Cloud
  - The end user does not know the details of the underlying infrastructure



Service catalogue

http://www.eosc-hub.eu/catalogue

WP5

Open Collaboration services

- Applications Database
- Repositories

WP5

Federation services

- Accounting
- ARGO
- Check-in
- GGUS
- GOCDB
- Marketplace
- Operations Portal
- RC Auth
- SPMT
- DPMT
- B2ACCESS
- TTS
- SYMON

WP6

Basic infrastructure and added-value services

- EGI High-Throughput Compute
- EGI Cloud Compute
- EGI Cloud Container
- DIRAC4EGI
- EGI Online storage
- EGI DataHub
- B2HANDLE
- B2FIND
- B2DROP
- B2SAFE
- B2STAGE
- B2NOTE
- ETDR
- Sensitive Data Service
- Advanced IaaS
- TOSCA for Heat
- OPIE

WP7

Thematic services

- ECAS
- DARIAH Gateway
- OPENCoastS
- GEOSS
- EO Pillar
- WeNMR
- DODAS
- LifeWatch
- CMI

From month 19:

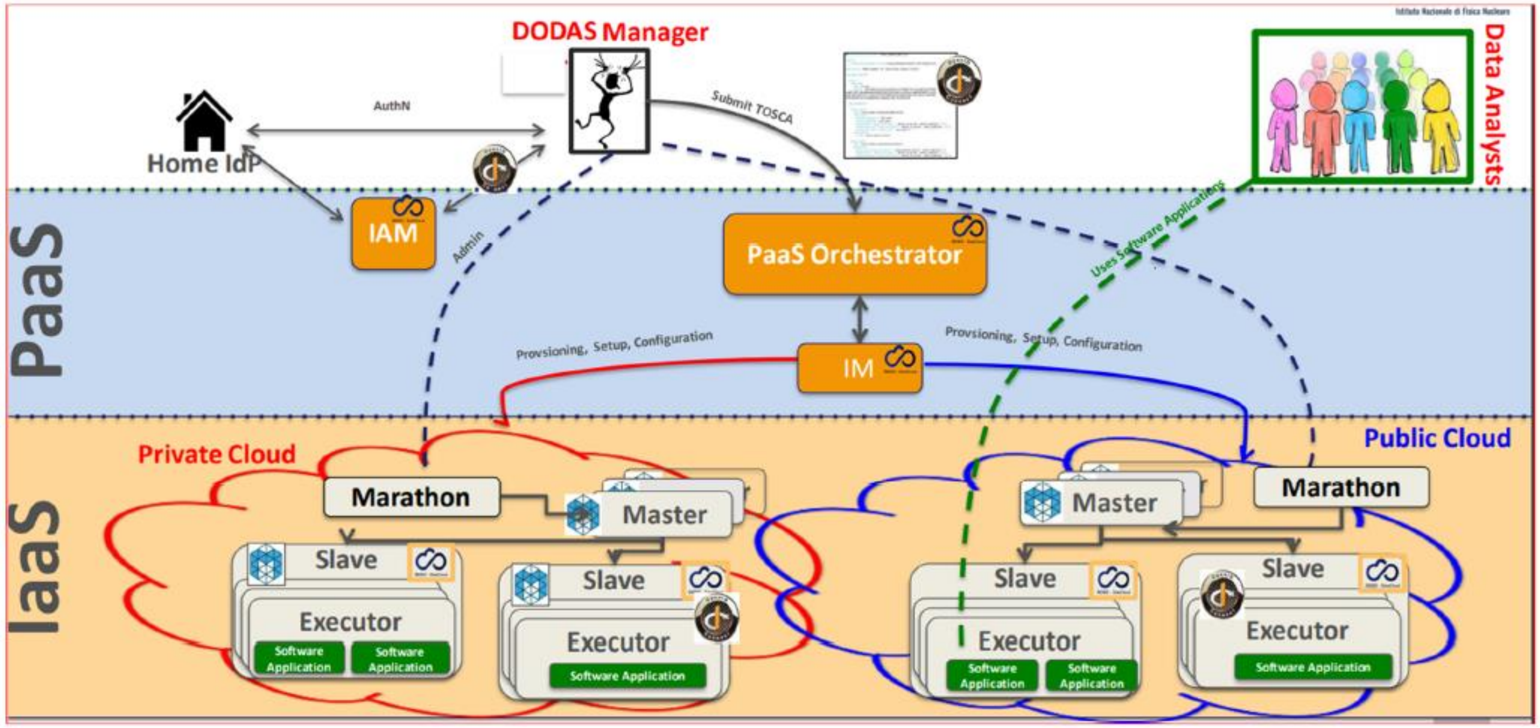
- IFREMER
- EISCAT_3D Portal

+ new service from outside the consortium

DODAS in a nutshell

DODAS: Dynamic On-Demand Analysis Service

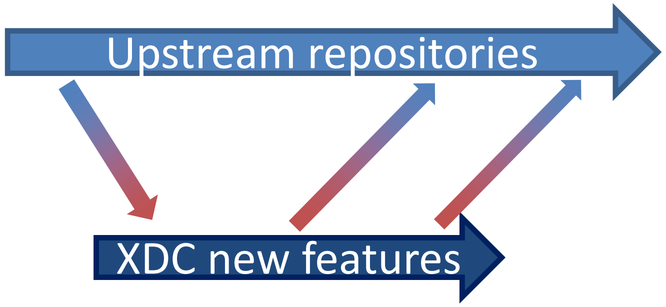
- A open source deployment manager
- **Allows on-demand creation and configuration of container based clusters** for data processing with **almost zero effort (HTCONDOR deployment)**
- **Support for hybrid clouds deployment**
- Based on “industry standards” to minimize code development and maintenance



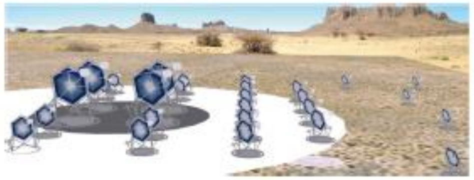
eXtreme DataCloud

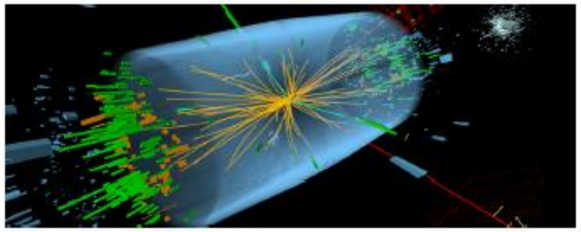
- The eXtreme DataCloud is a **software development** and integration project
- **Develops scalable technologies for federating storage resources** and **managing data** in **highly distributed** computing environments
  - Focus on efficient, policy driven and Quality of Service based DM
- The targeted platforms are the current and next generation e-Infrastructures deployed in Europe
  - European Open Science Cloud (EOSC)
  - The e-infrastructures used by the represented communities

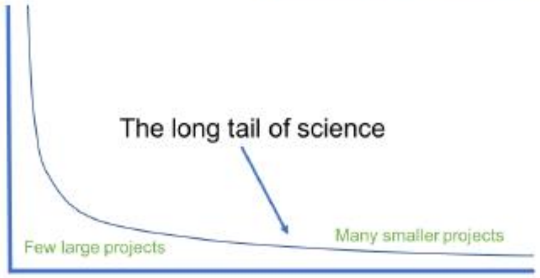
- **Improve already existing, production quality Data Management Services with new functionalities**
  - Intelligent & Automated Dataset Distribution
  - Data pre-processing during ingestion
  - Metadata management
  - Data management based on storage events
  - Smart caching
  - Sensitive data handling



XDC: A User Driven Project







DEEP HybridDataCloud

DEEP-Hybrid-DataCloud project aims to **promote the integration** of **specialized**, and **expensive**, hardware under a Hybrid Cloud platform, so it can be **used on-demand** by researchers of different communities.

- DEEPaaS to provide ML framework «as a service»
- Orchestration of long running services on containers
- Instantiation on GPU resources
- Different users' profiles served

User Driven Project

Citizen Science:

-- Plant classification

-- Image Classification

Earth Observation:

-- Satellite Imagery

Biological and Medical Science:

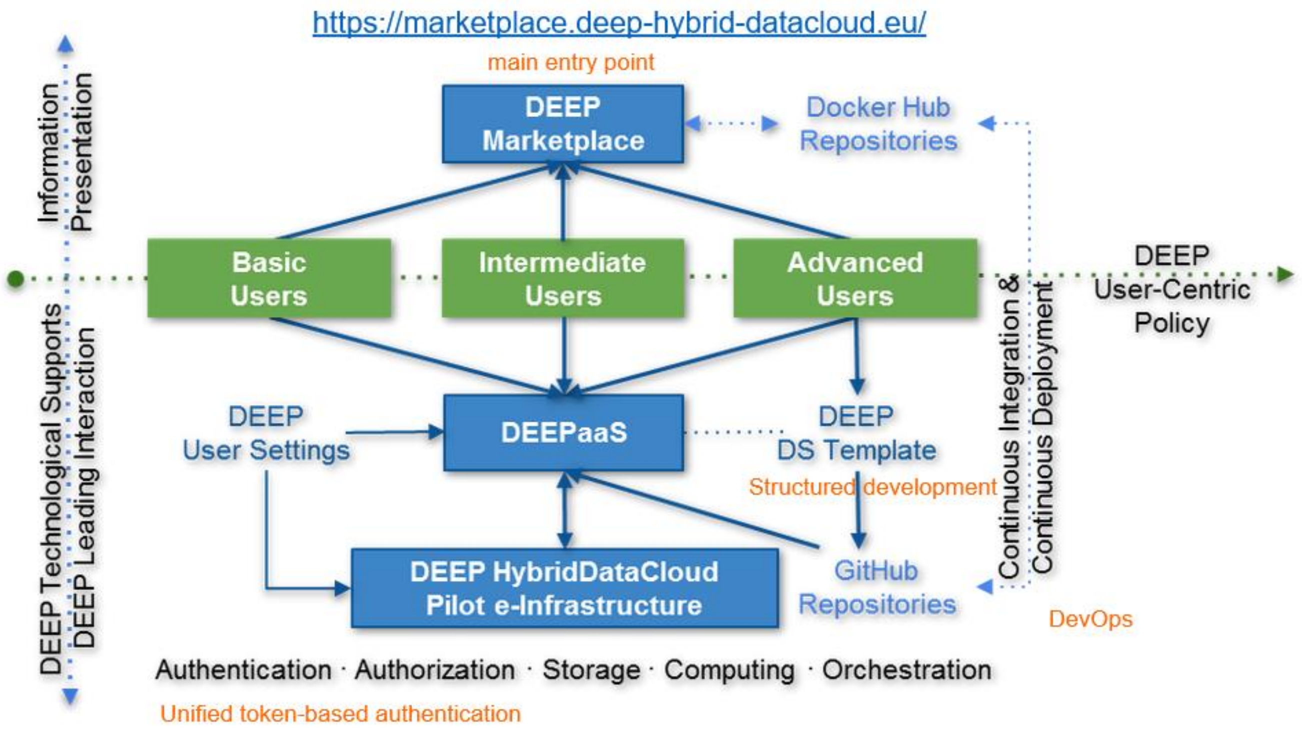
-- Retinopathy

Computing Security:

-- Massive Online Data Streams

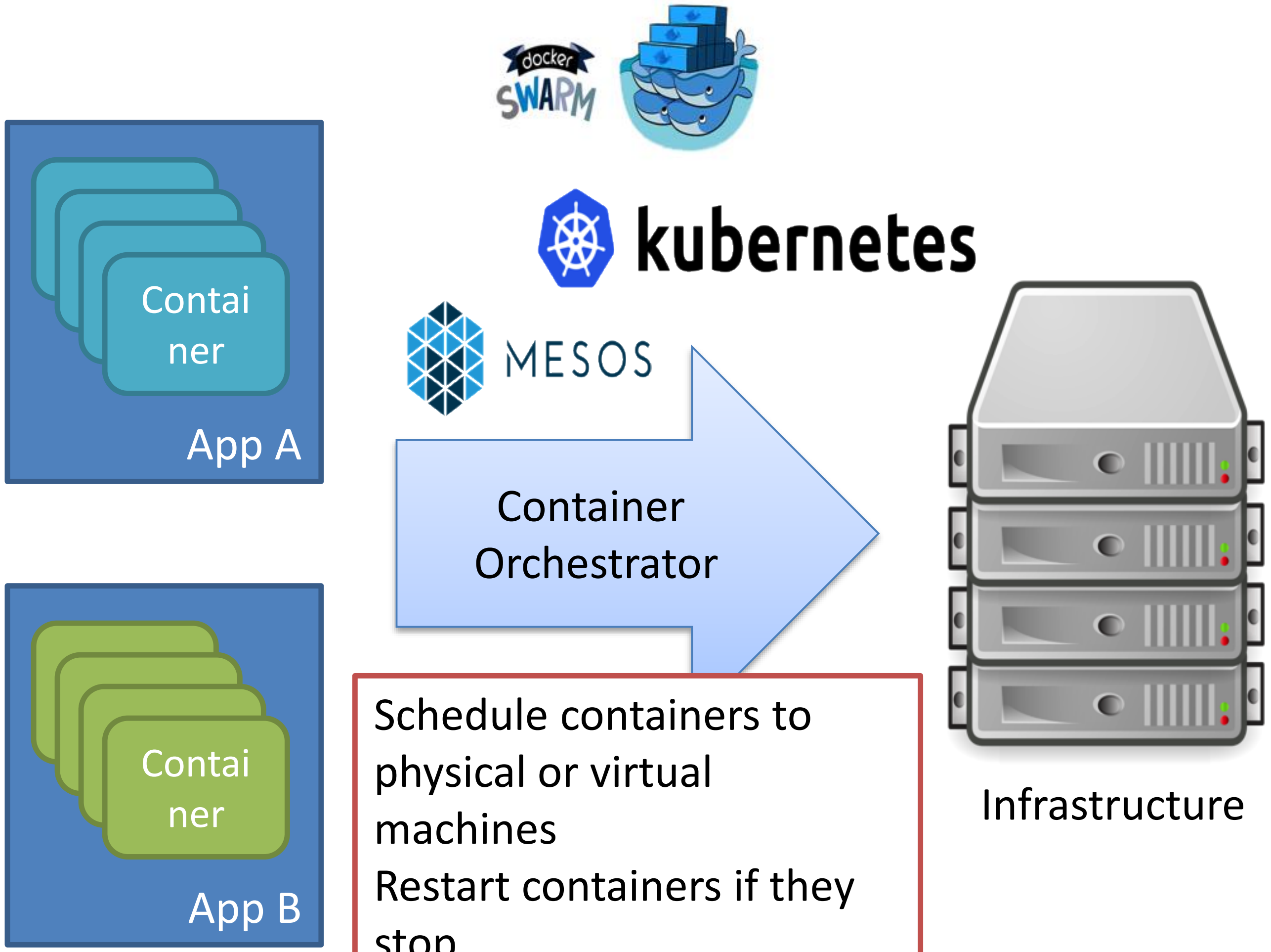
Physics:

-- Post-processing



https://marketplace.deep-hybrid-datacloud.eu/
main entry point
DEEP Marketplace
Docker Hub Repositories
Information Presentation
Basic Users
Intermediate Users
Advanced Users
DEEP User-Centric Policy
DEEP User Settings
DEEPaaS
DEEP DS Template
Structured development
DEEP HybridDataCloud Pilot e-Infrastructure
GitHub Repositories
DevOps
Continuous Integration & Continuous Deployment
DEEP Technological Supports
DEEP Leading Interaction
Authentication · Authorization · Storage · Computing · Orchestration
Unified token-based authentication

Container orchestration



Schedule containers to physical or virtual machines
Restart containers if they stop
Provide private container network
Scale up and down
Service discovery

Kubernetes

- Kubernetes is an *open-source platform for automating deployment, scaling, and operations of application containers across clusters of hosts, providing container-centric infrastructure.*
- Some concepts:
  - *Pod*: group of one or more containers, shared storage and options to run the containers
  - *Deployment* maintains the desired count of Pods all the time
  - *Service:* logical set of Pods and a policy by which to access them.
    - Exposed to the exterior of the Kubernetes cluster via mapping of ports and or Load Balancing
  - *Job*: A *job* creates one or more pods and ensures that a specified number of them successfully terminate.

Creative Commons License