Integrating and managing services for the European Open Science Cloud

Research Computational Infrastructures: Cloud Infrastructures

Alessandro Costantini

INFN

eosc-hub.eu

@EOSC_eu

EOSC-hub receives funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No. 777536.

EOSC-hub

About Visit Support us Women in Science at ICTP Marie Curie Library

The Abdus Salam International Centre for Theoretical Physics

IAEA

Research Scientific Calendar Programmes

Europe/Rome Administration

- Overview
- Program
- Speakers
- Practical info

Material

- General Information
- Poster
- Preliminary List of Participants

Go to day

Friday, 16 August 2019

08:30 - 13:00 **Data Stewards Track**

- 08:30 **Linked Data; Sparql queries** *2h0'*
  Speaker: Daniel Bangert (Göttingen University)
- 10:30 **Coffee Break** *30'*
- 11:00 **Linked Data; Sparql queries** *2h0'*
  Speaker: Daniel Bangert (Göttingen University)

08:30 - 13:00 **ECR's Track**

- 08:30 **Research Computational Infrastructure** *2h0'*
  Speaker: Rob QUICK (Indiana University), Alessandro COSTANTINI (INFN-CNAF)
- 10:30 **Coffee Break** *30'*
- 11:00 **Research Computational Infrastructure** *2h0'*
  Speaker: Rob QUICK (Indiana University), Alessandro COSTANTINI (INFN-CNAF)

13:00 - 15:00

- 13:00 **Lunch break** *1h0'*
- 14:00 **Closing ceremony** *1h0'*

The trainer

- alessandro.costantini@cnaf.infn.it
- National Institute for Nuclear Physics (INFN)
- Cloud expert
- 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. Hands-on with Cloud services
   - IaaS approach with OpenStack
   - SaaS approach with Jupyter Notebook service
3. See future possibilities with EOSC

Outline

PART 1 (8:30-10:30)

- Introduction to cloud computing (70')
- Introduction to EGI and related infrastructure (10')
- Identity and Access Management: 101 + Hands-on (20')
- Introduction to Hands-on: Explore Openstack IaaS (20')

BREAK (30')

PART 2 (11:00-12:45')

- Hands-on exercise 1 – deploy a VM in the IaaS environment (30')
- Introduction to Hands-on: exercise 2 (10')
- Hands-on exercise 2 – Jupyter Notebooks (30)
- The future of compute infrastructures in Europe: EOSC (20')

Feedback forms (15')

Introduction to cloud computing

Cloud Computing

Distributed computing

- “The Cloud” means essentially the use of distributed resources, which is certainly not a new thing

How grid computing helped CERN hunt the Higgs

FEATURE | AUGUST 15, 2012 | BY JOANNAH CABORN WENGLER

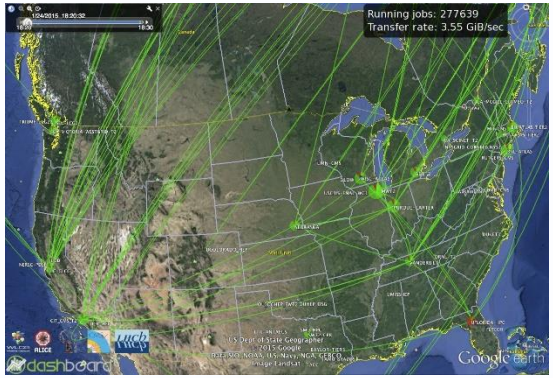
“As a layman, I’d say we have it.” It was with these words that CERN’s director general, Rolf Heuer, last month announced the discovery of a particle consistent with the Higgs boson, the long-sought-after corner stone of particle physics’ standard model. The scientific results upon which Heuer based his statement - taken from two experiments involved, ATLAS and CMS - are now set to be published in the upcoming issue of *Physics Letters B*. What many people outside of particle physics may not know is that distributed computing played a crucial role in the race towards this discovery.

Image courtesy ATLAS experiment © CERN

“Particle physics is nowadays an international and highly data-intensive field of science and it requires a massive international computing effort,” said Roger Jones, ATLAS physicist and collaboration board chair of the Worldwide LHC Computing Grid (WLCG), the organization that supplies this huge computing effort. Founded in 2002, today the WLCG involves the collaboration of over 170 computing centers in 36 countries, making it the largest scientific computing grid in the world.

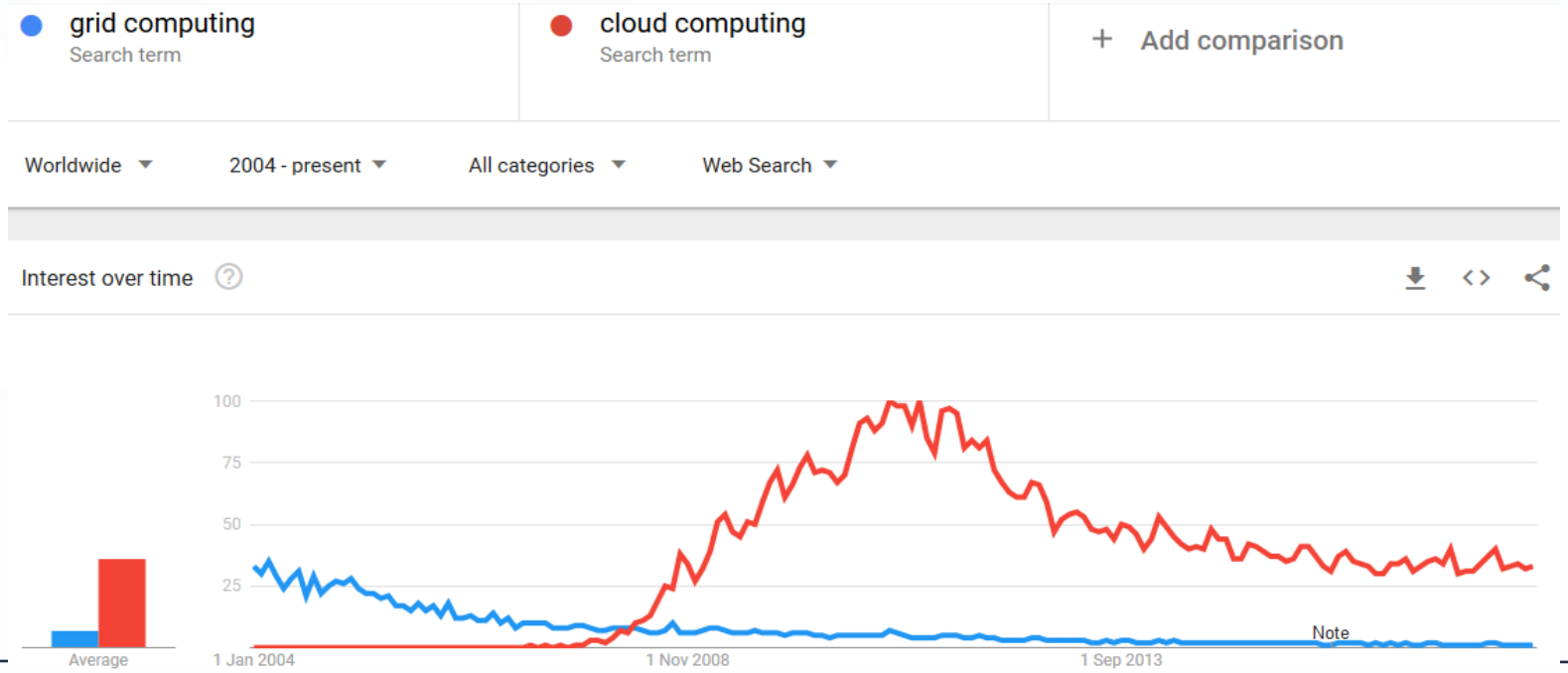
Grid Computing

- Provisioning and use of *massively distributed and federated resources* routinely happens since several years.
- An example of great success in the scientific world is given by **Grid Computing**.
- On the right:
  - Real-time situation of computation and data transfer for the physics experiments running at the CERN Large Hadron Collider (LHC).



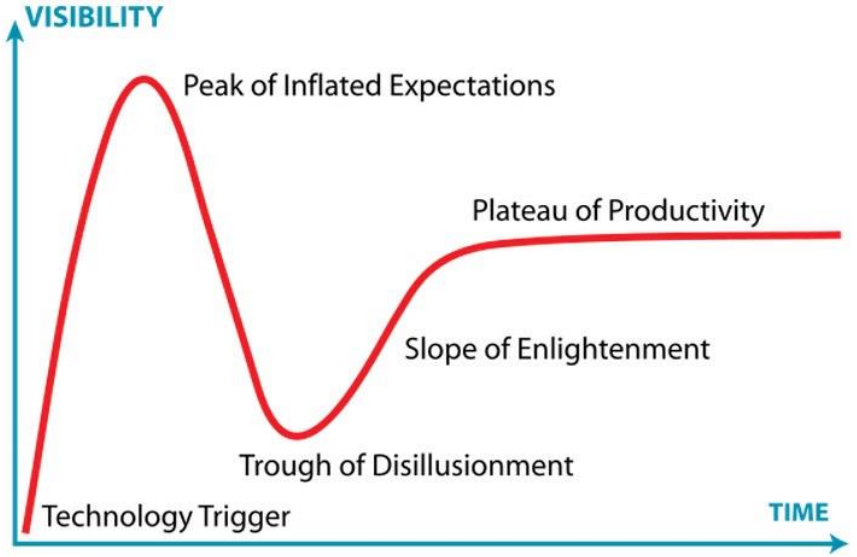
Cloud vs. Grid

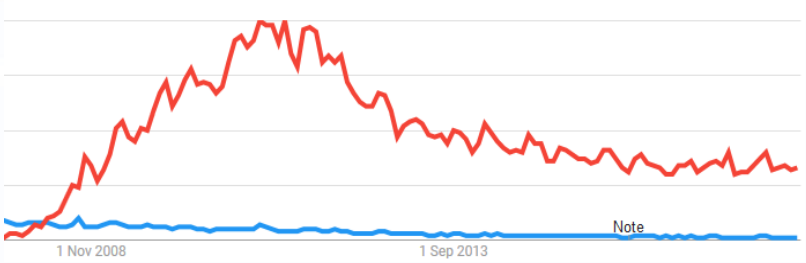
- Grid Computing, that is crucial for big scientific collaborations, never spread outside this field
- Google trends (http://www.google.com/trends/): Grid Computing vs. Cloud Computing



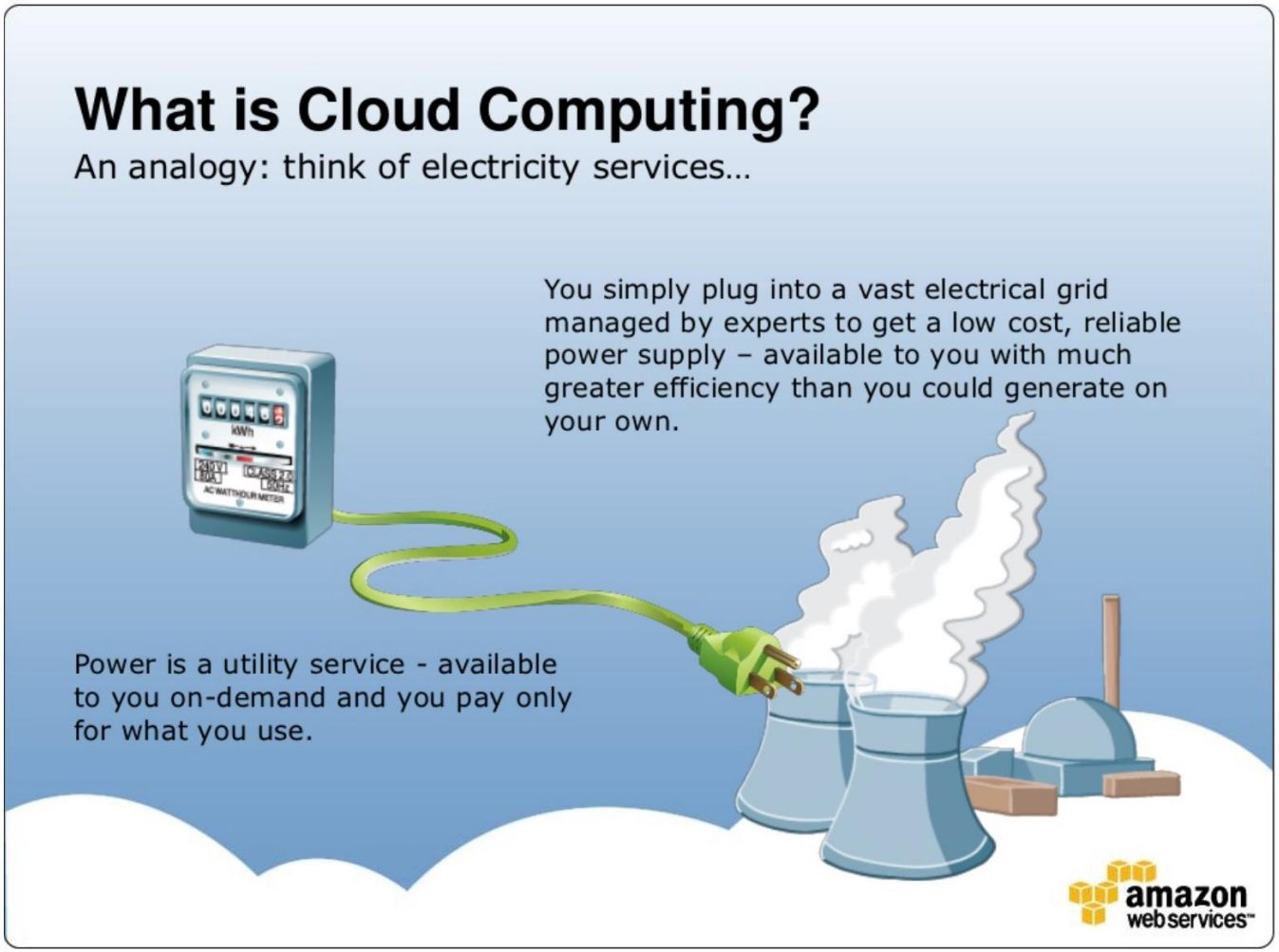
Cloud hype cycle

- The hype cycle is used to represent the maturity, adoption and social application of specific technologies, through five phases (bit.ly/2H4iX0N):
  - 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)





Cloud computing: Analogy



Source: bit.ly/2Z4kHNG

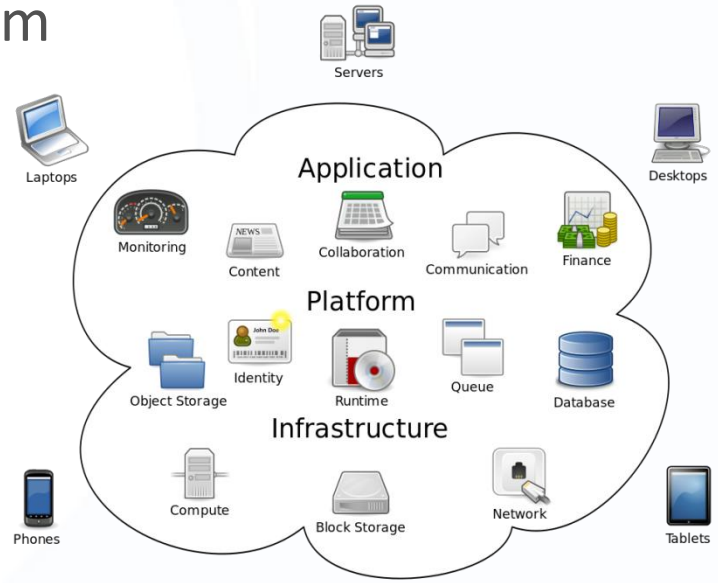
Cloud computing: Concept

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:



Cloud computing

Source: Wikipedia

1 Supplying
2 information and communication technologies
3 as a service

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

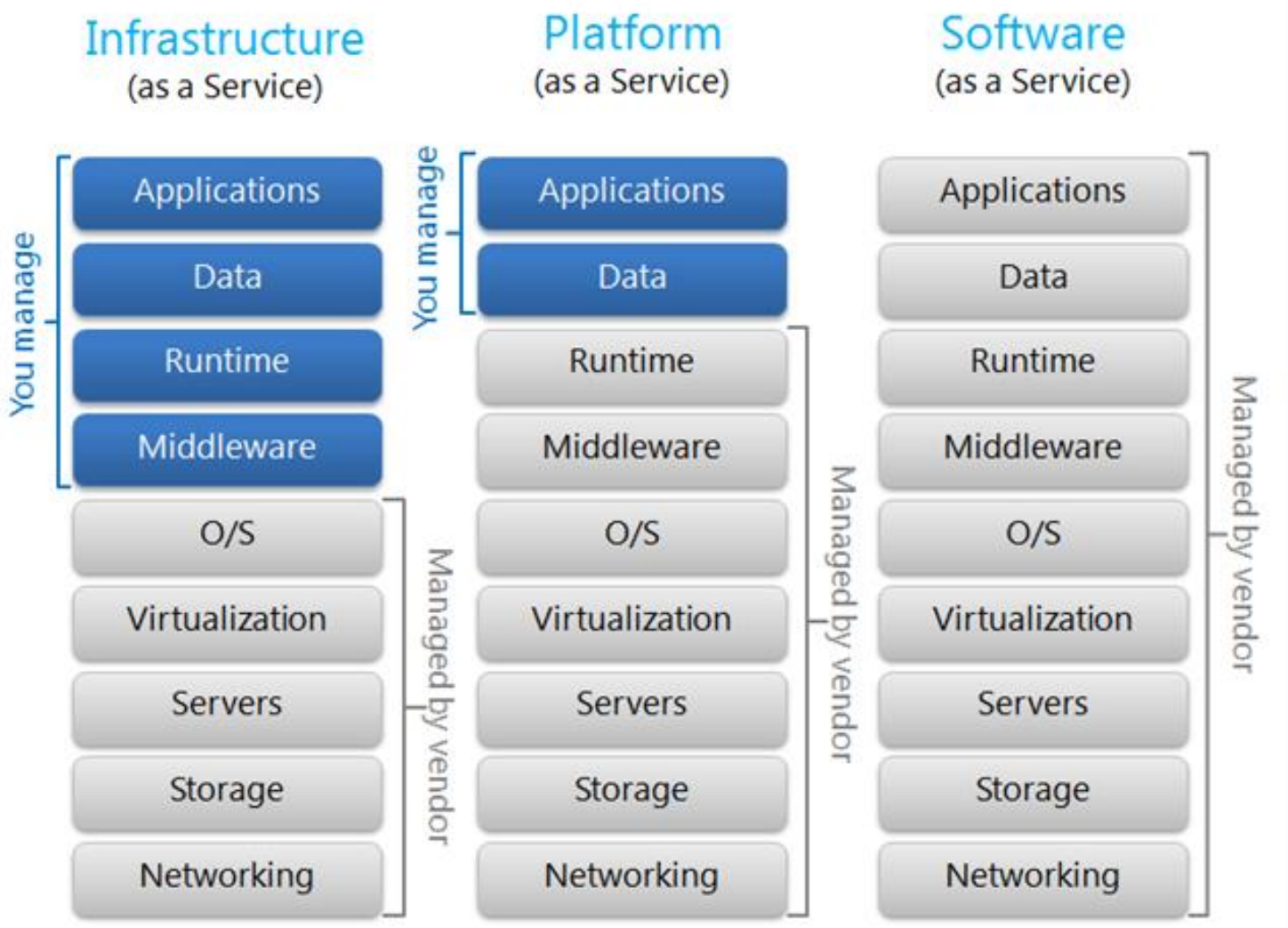
- **Self-service, on-demand**
  - Online or by telephone booking
- **Network**
  - Network of car rental all over the world
- **Resource pooling**
  - The car rental manages the availability of cars for customers
- **Elasticity**
  - The number of cars can vary depending on users demand
- **Pay-per-use**
  - 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.

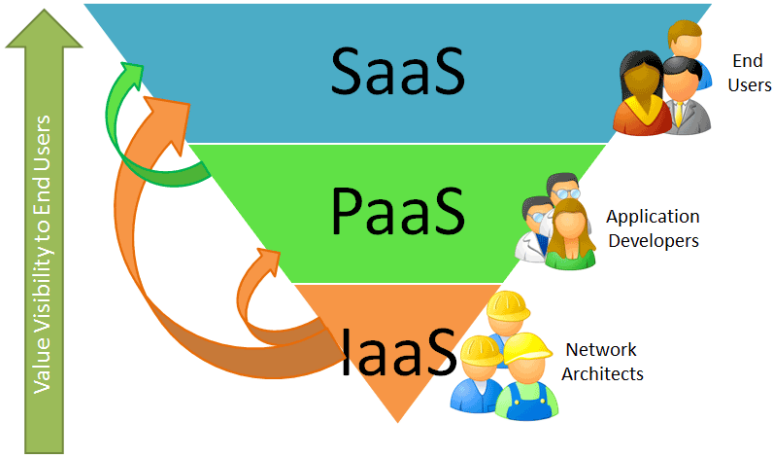
IaaS vs. PaaS vs. SaaS

	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!

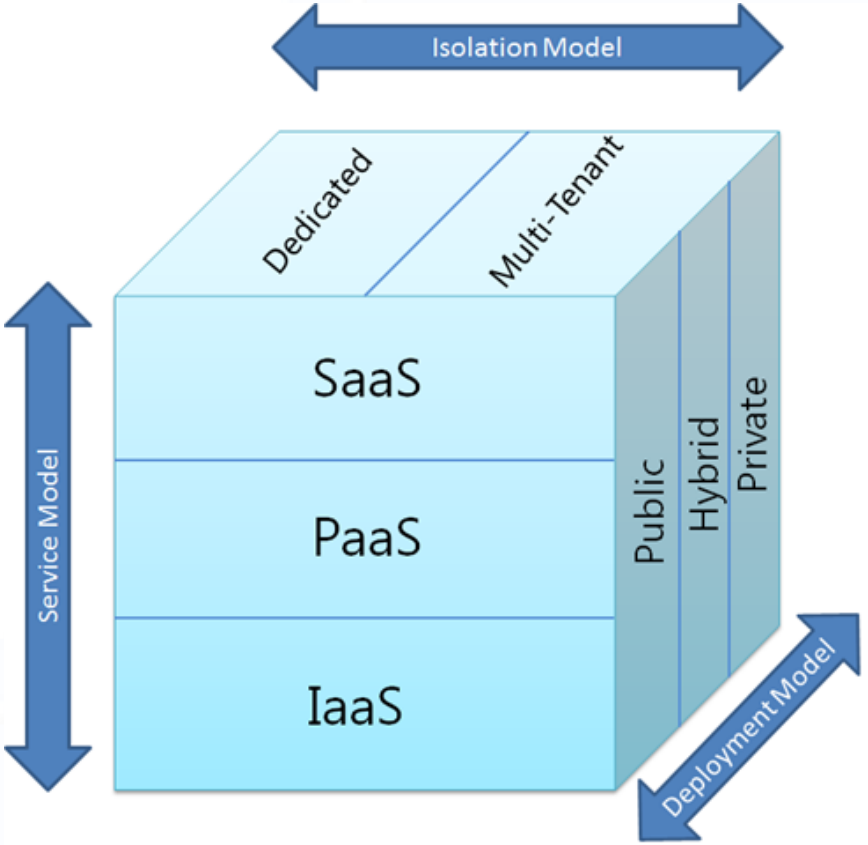


… however, **without Cloud providers (public or private)**, and without **efficient and effective** ways of managing distributed resources, applications cannot be deployed!
(rather obvious isn't it)

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

Deployment: “Cloud types”

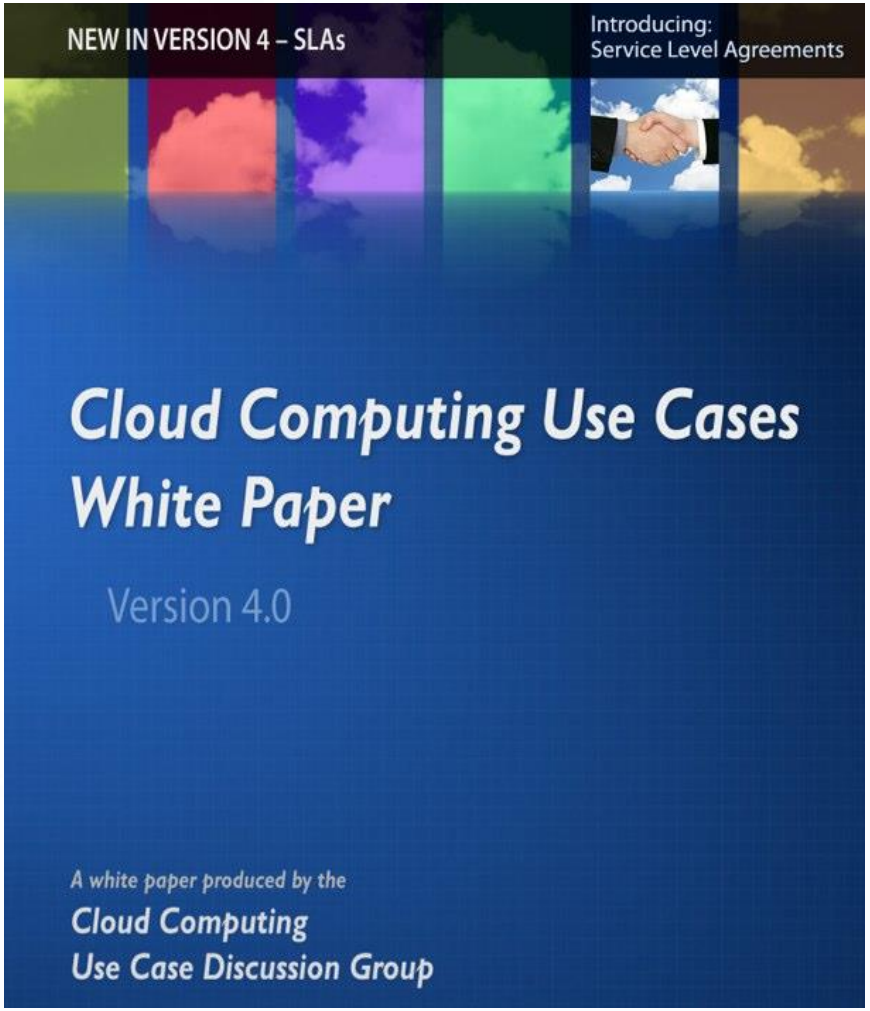
- **Private Cloud:**
  - The infrastructure is **procured for exclusive *use*** by a single organization. Management, operation, ownership, location of the private cloud, however, can be independent by the organization using it.
- **Community Cloud:**
  - The infrastructure is **available to a community** 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.

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

Cloud use cases

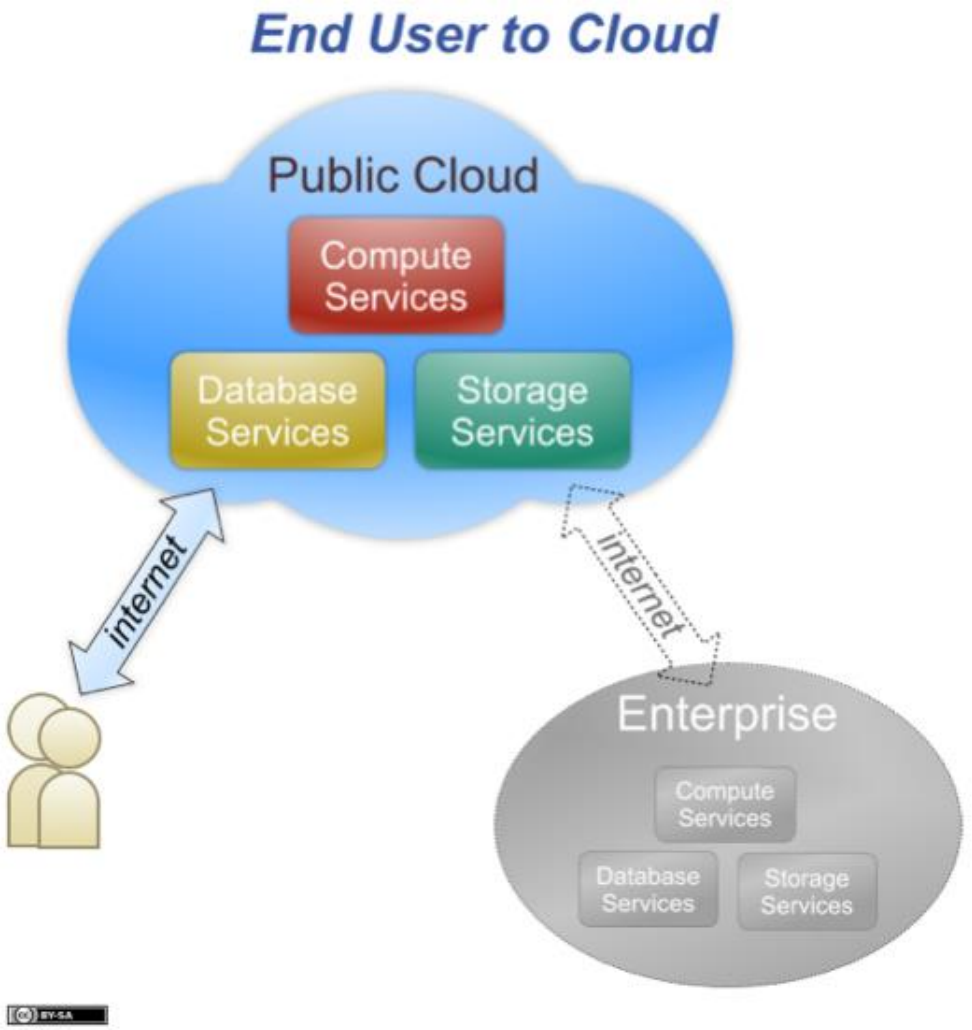
- https://goo.gl/qxRtrw
- 7 principal cases:
  - End user → Cloud
  - Enterprise → Cloud → end user
  - Enterprise → Cloud
  - Enterprise → Cloud → enterprise
  - Private Cloud
  - Changing Cloud vendors
  - Hybrid Cloud



Source: bit.ly/305Jfre

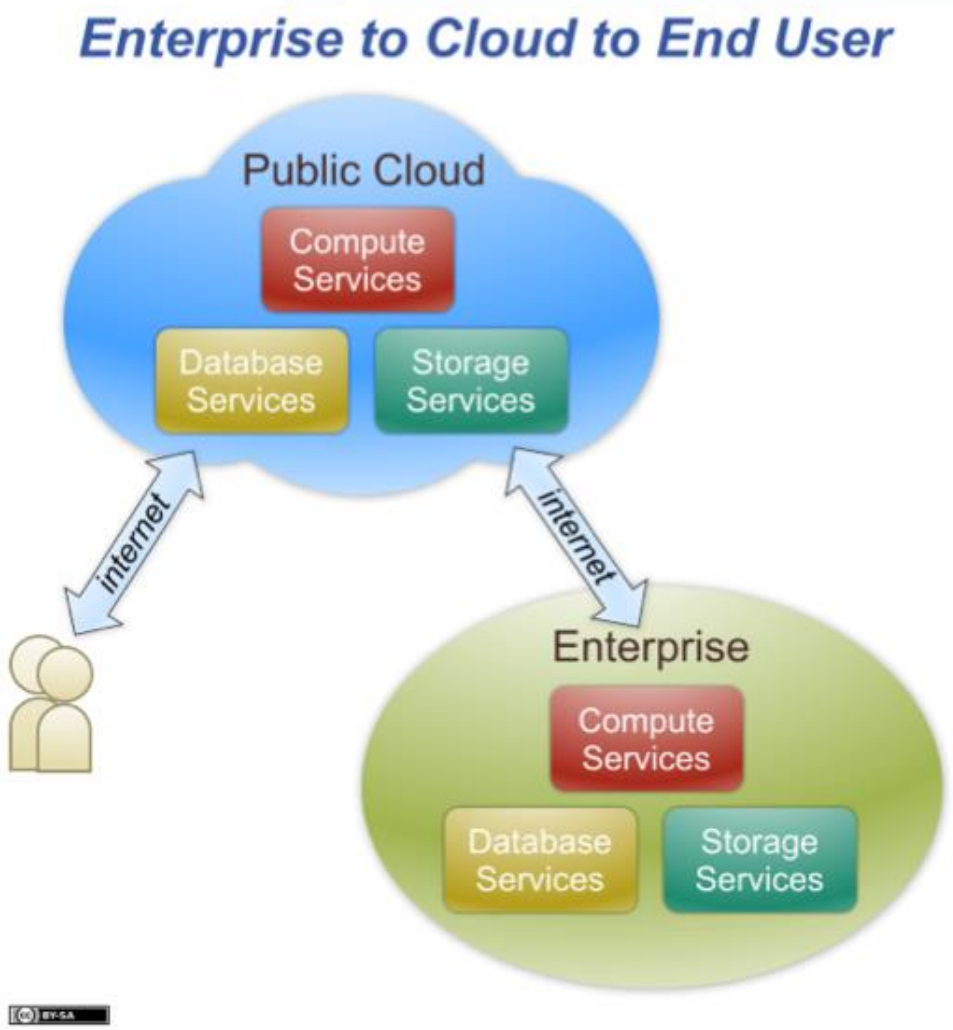
End user → Cloud

- 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**



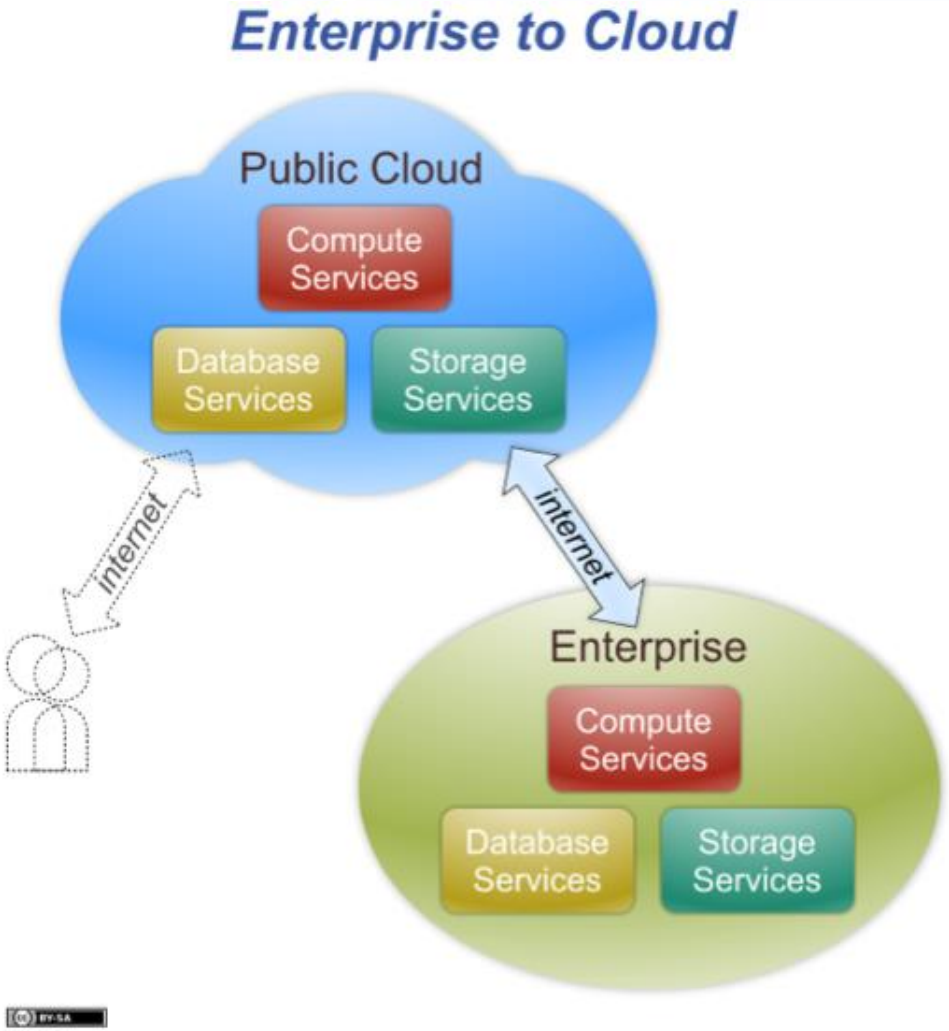
Enterprise → Cloud → End user

- 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



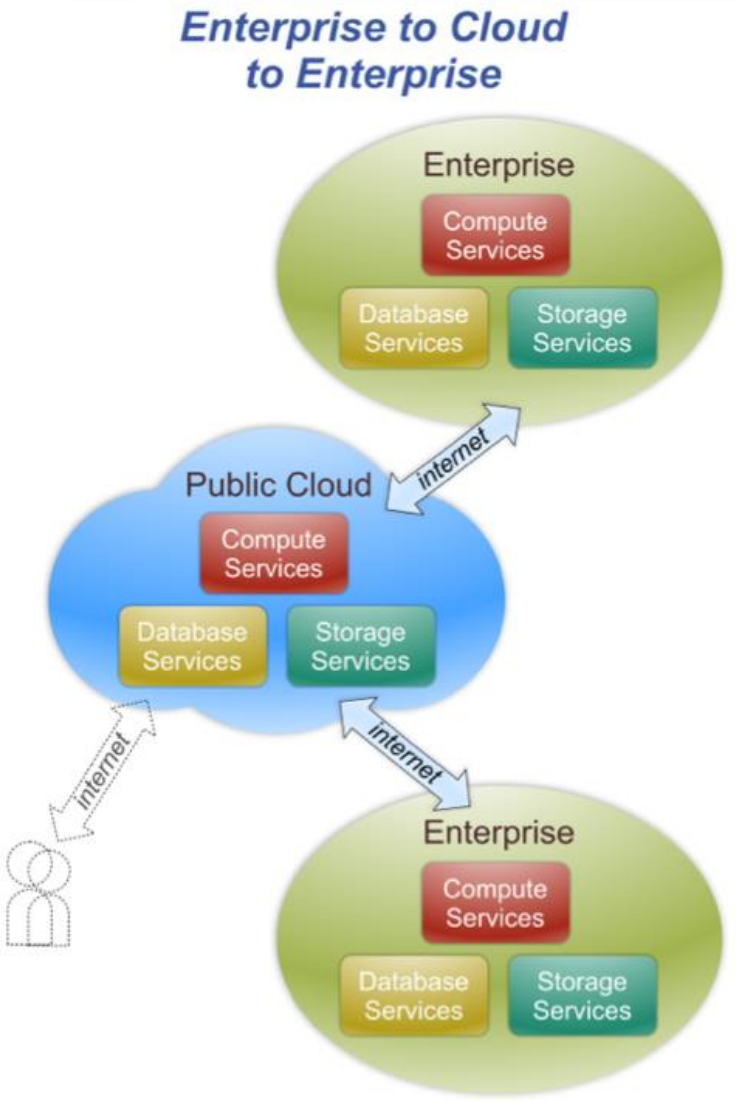
Enterprise → Cloud

- 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



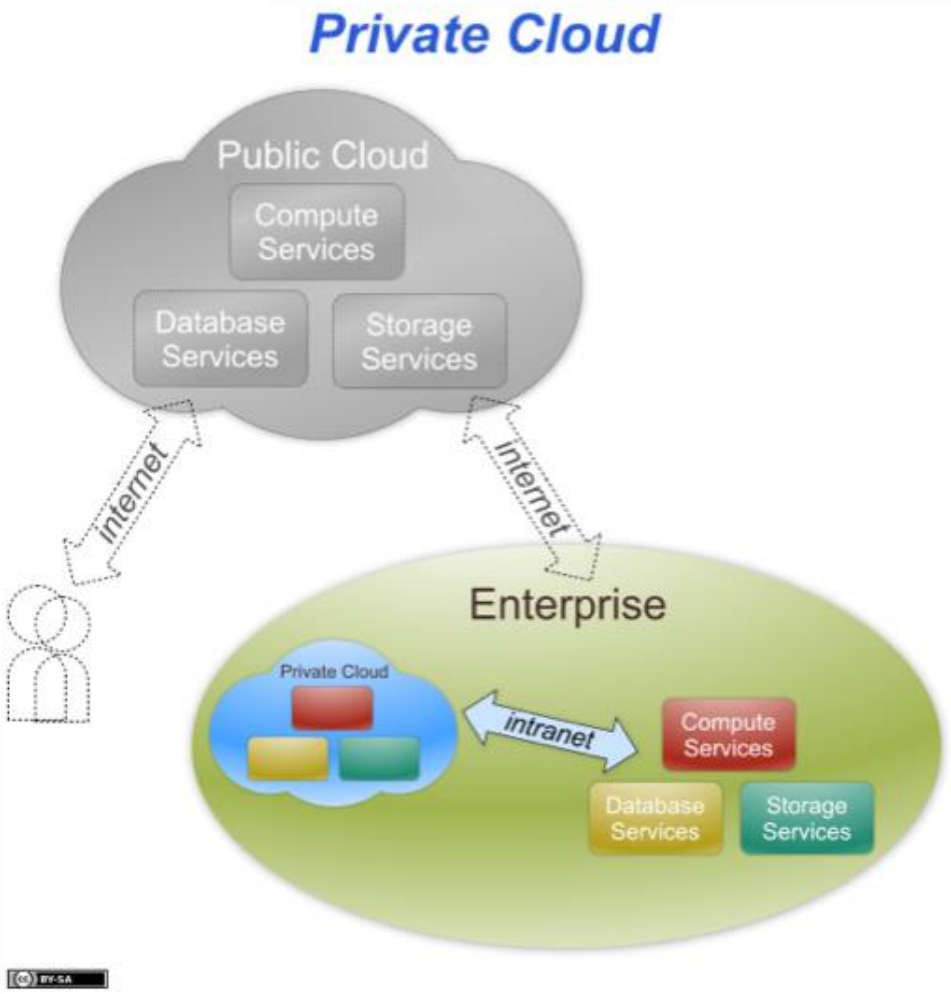
Enterprise → Cloud → Enterprise

- 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



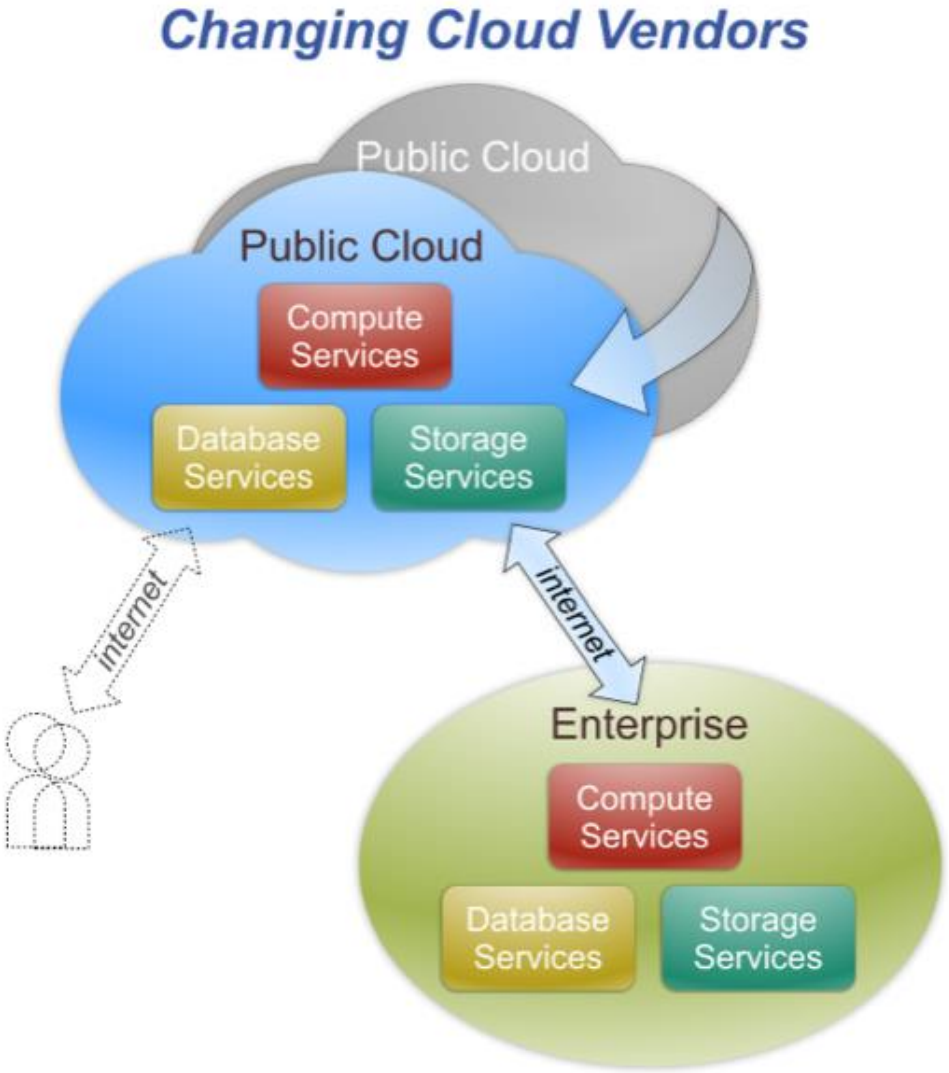
Private Cloud

- 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



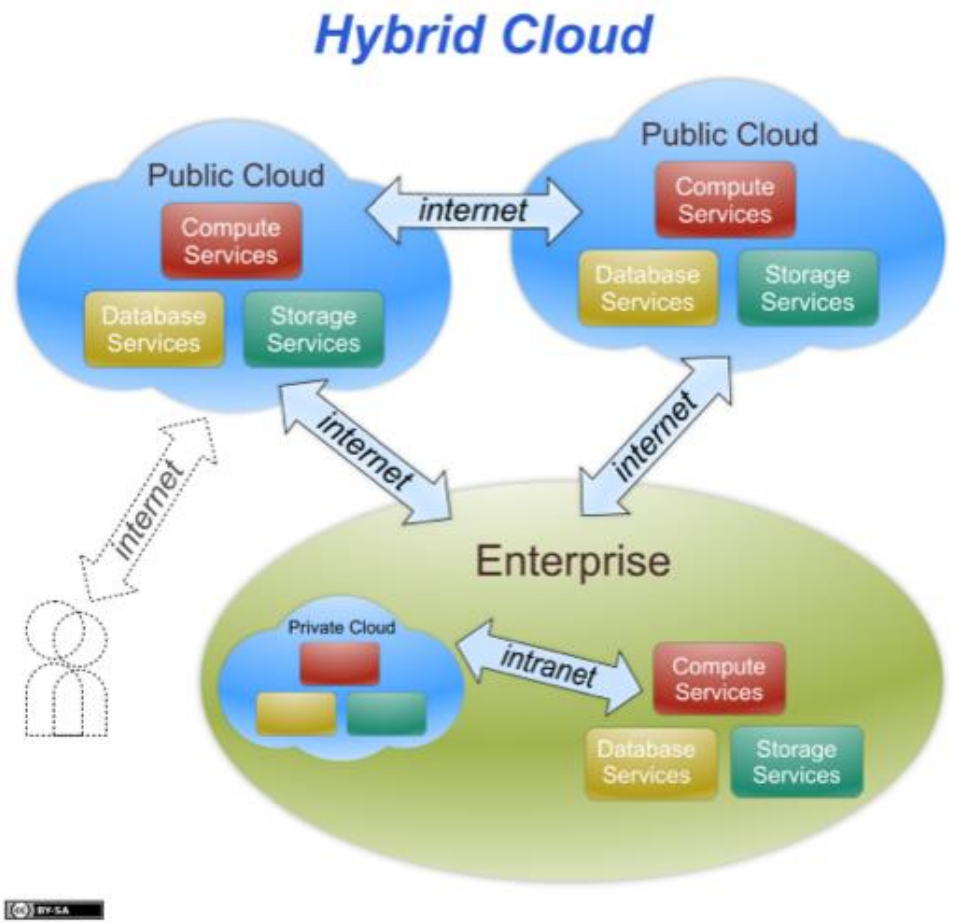
Changing Cloud vendors

- 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



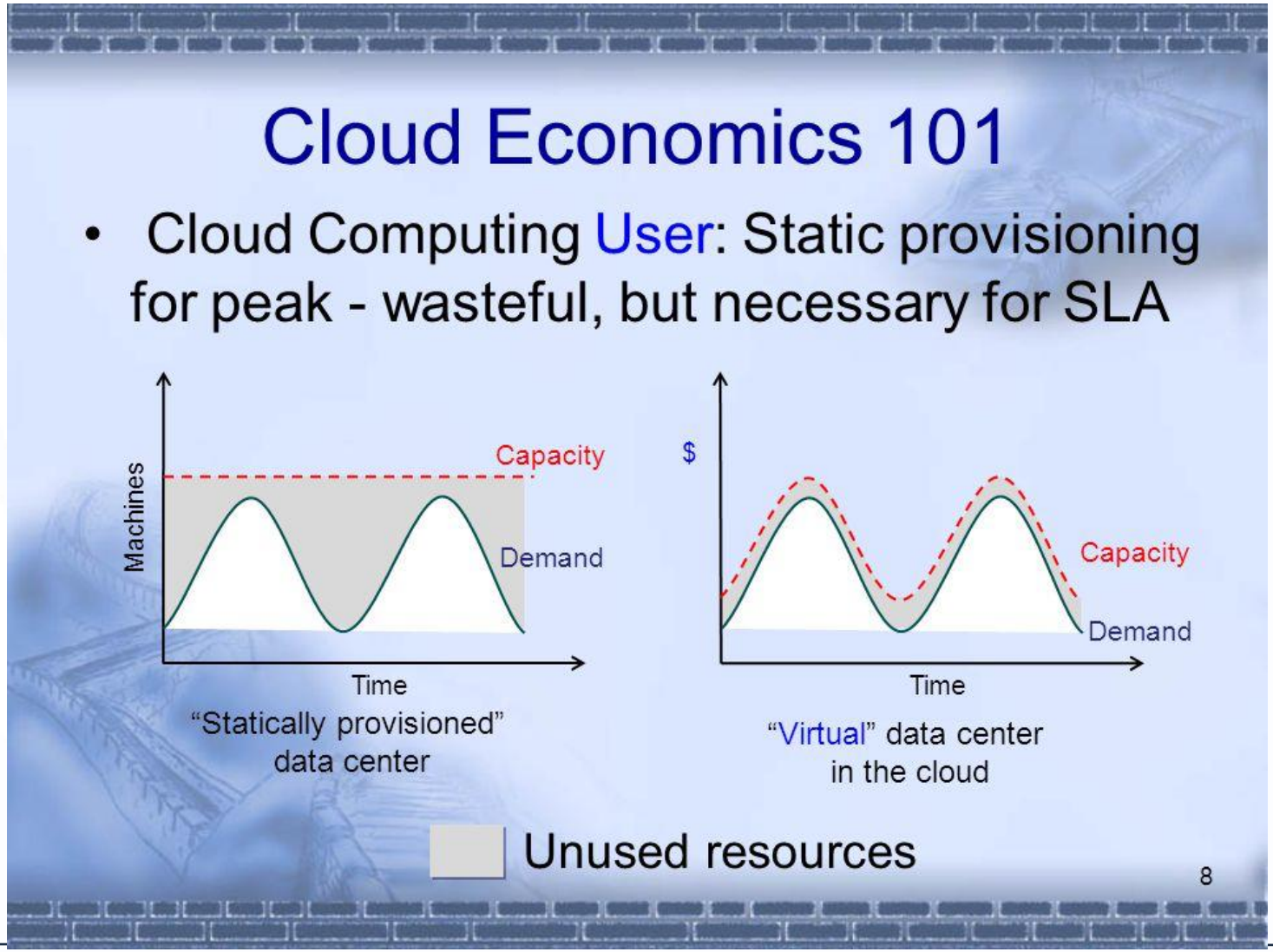
Cloud use cases: some examples

- **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.
- **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.
- **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

Static vs virtual resources

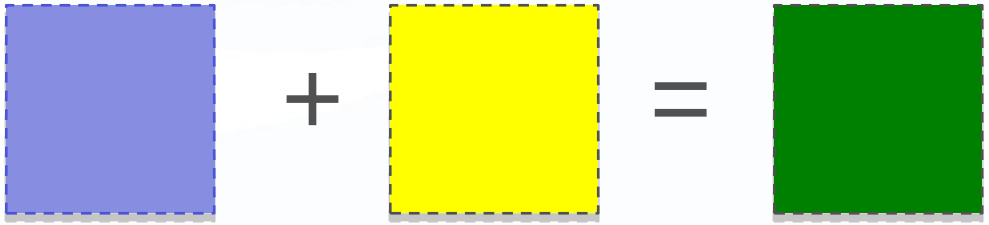


Source: http://slideplayer.com/slide/6085063/

Virtualization and Cloud

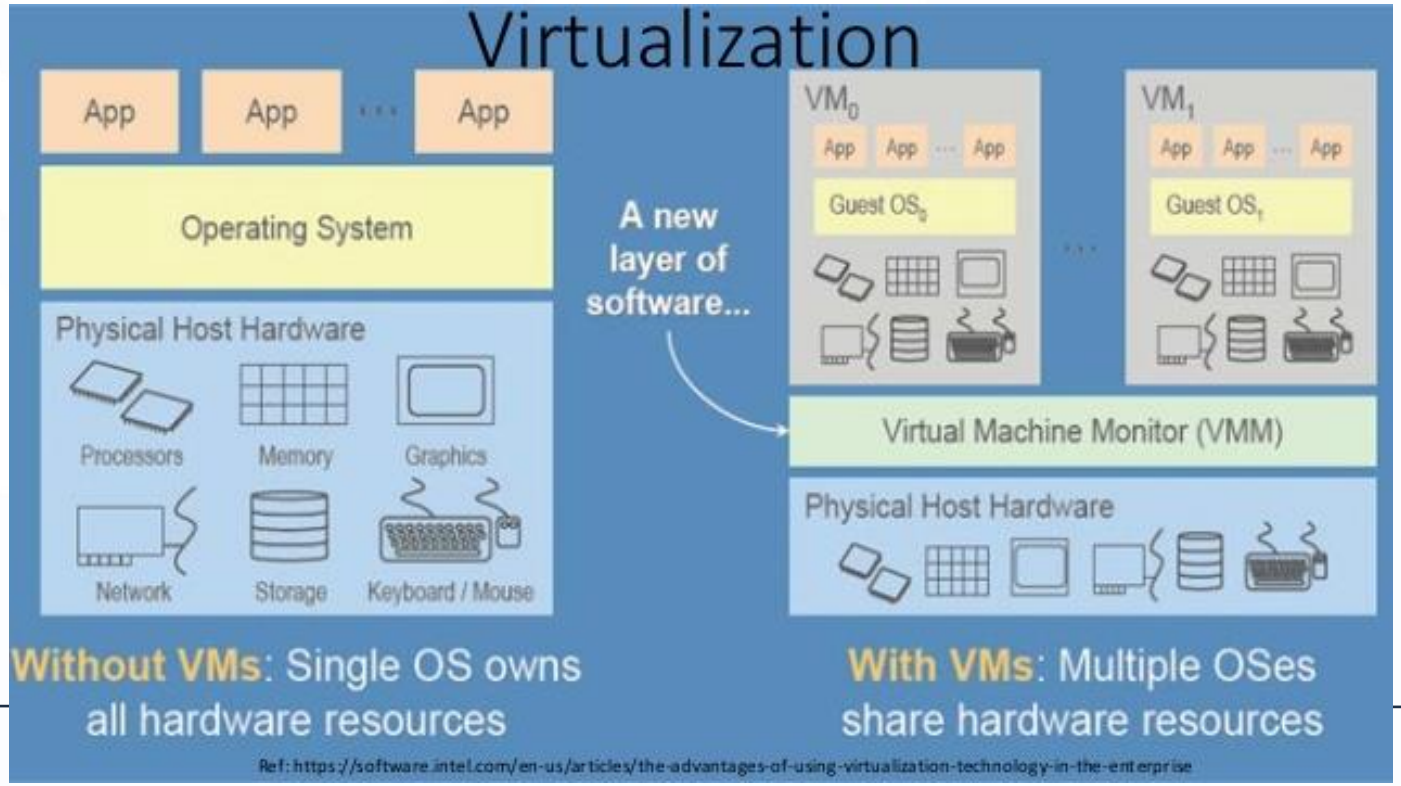
- Virtualization and Cloud computing

- 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.



Let's go virtual

Workloads without Virtualization

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

Workloads migrated To Virtual Machines Using Virtualization

- 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

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?

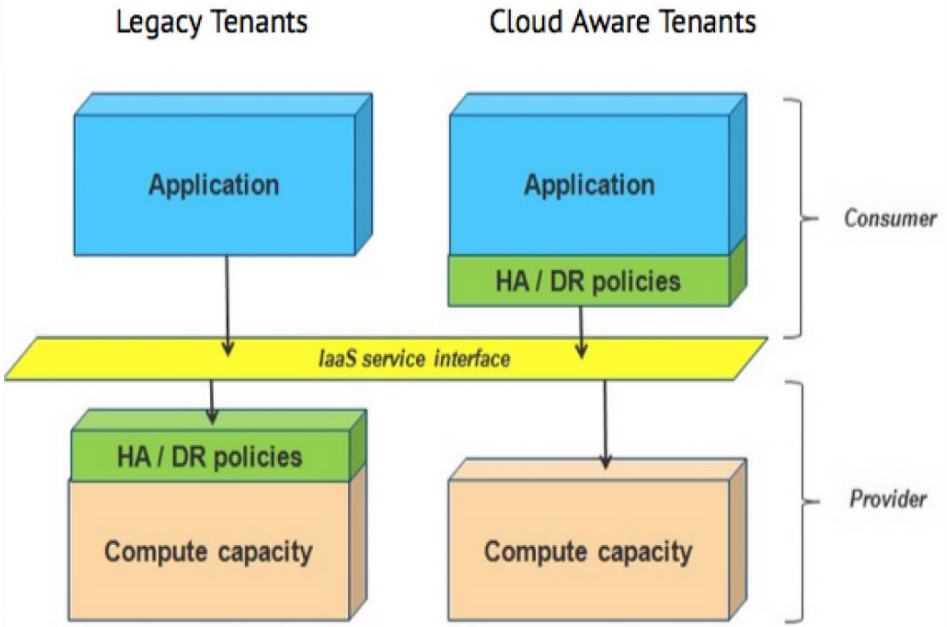
Stateless vs Stateful service

- 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 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

Cloud-related risks

Some Cloud-related risks

- **Liability**
- **Security and privacy**
- **Lock-in**
- **Insecure or incomplete data deletion**

- Some examples taken from a very popular public Cloud follow…

Amazon Web Services

- Amazon does not only sell books or general goods…

- Amazon Web Services:
  - on-demand cloud computing platforms to individuals, companies and governments, on a paid subscription basis.
  - Revenue: $17.5 billion (FY17)
  - Launched in March 2006

2017 Cloud Revenue	
Microsoft	$18.6B
Amazon	$17.5B
IBM	$17.0B
Salesforce.com	$9.92B (12 mos.ending Oct. 31)
Oracle	$5.6B (12 mos.ending Nov. 30)
SAP	$4.71B
Google	$3B (est.)

@bobevansIT Source: @bobevansIT

Source: bit.ly/2KHrwj5

Unavailability

- **Limitations of liability** in case of unavailability of data or services.
  - Due e.g. to power outages, system failures, or to any other service interruption.
  - Or due to unauthorized access, alteration, loss or failure to store any content in AWS.

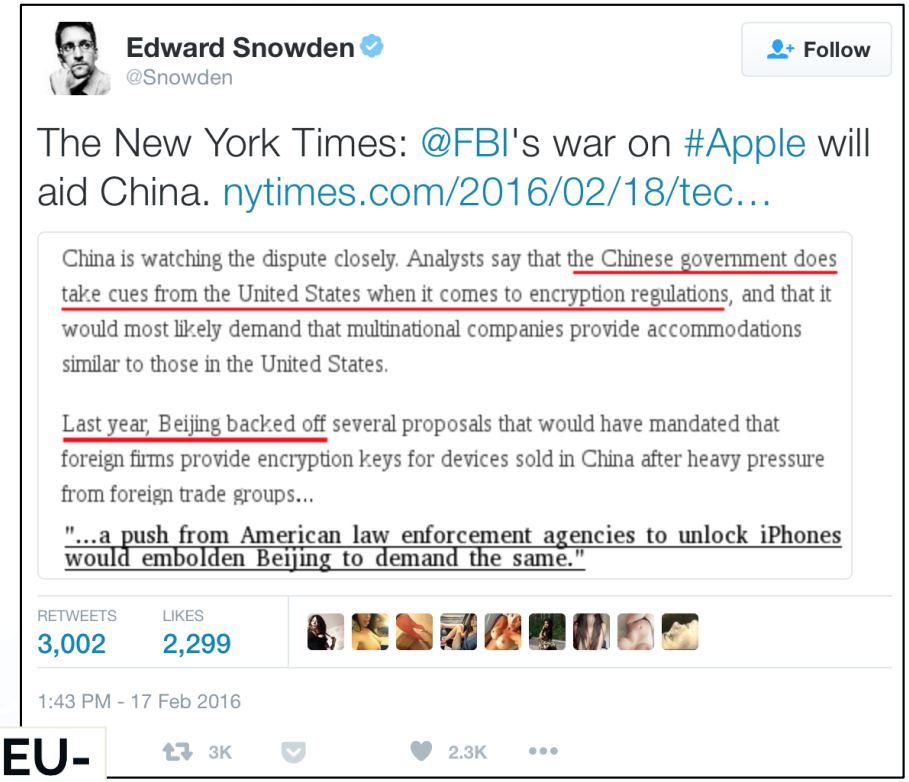
But you are responsible

- **You are responsible to make sure your data, code, etc. is safe, protected from unauthorized access, and *you are responsible for your own backup*** (again – with if it's in the order of several PB?)

> **4.2 Other Security and Backup.** You are responsible for properly configuring and using the Service Offerings and taking your own steps to maintain appropriate security, protection and backup of Your Content, which may include the use of encryption technology to protect Your Content from unauthorized access and routine archiving Your Content. AWS log-in credentials and private keys generated by the Services are for your internal use only and you may not sell, transfer or sublicense them to any other entity or person, except that you may disclose your private key to your agents and subcontractors performing work on your behalf.

Data property / privacy?

- When a contract with a Cloud provider gets cancelled, how can we make sure that **all our data is removed**?
- And how can I avoid ***vendor lock-in***?
- But where is my data? How about ***tapping***?



Edward Snowden
@Snowden

Follow

The New York Times: @FBI's war on #Apple will aid China. nytimes.com/2016/02/18/tec…

China is watching the dispute closely. Analysts say that the Chinese government does take cues from the United States when it comes to encryption regulations, and that it would most likely demand that multinational companies provide accommodations similar to those in the United States.

Last year, Beijing backed off several proposals that would have mandated that foreign firms provide encryption keys for devices sold in China after heavy pressure from foreign trade groups...

"...a push from American law enforcement agencies to unlock iPhones would embolden Beijing to demand the same."

RETWEETS 3,002 LIKES 2,299

1:43 PM - 17 Feb 2016

3K 2.3K

Microsoft admits Patriot Act can access EU-based cloud data

Microsoft's U.K. head admitted today that no cloud data is safe from the Patriot Act, and the company can be forced to hand EU-stored data over to U.S. authorities.

By Zack Whittaker for iGeneration | June 28, 2011 -- 08:10 GMT (09:10 BST) | Topic: Government : US

NSA infiltrates links to Yahoo, Google data centers worldwide, Snowden documents say

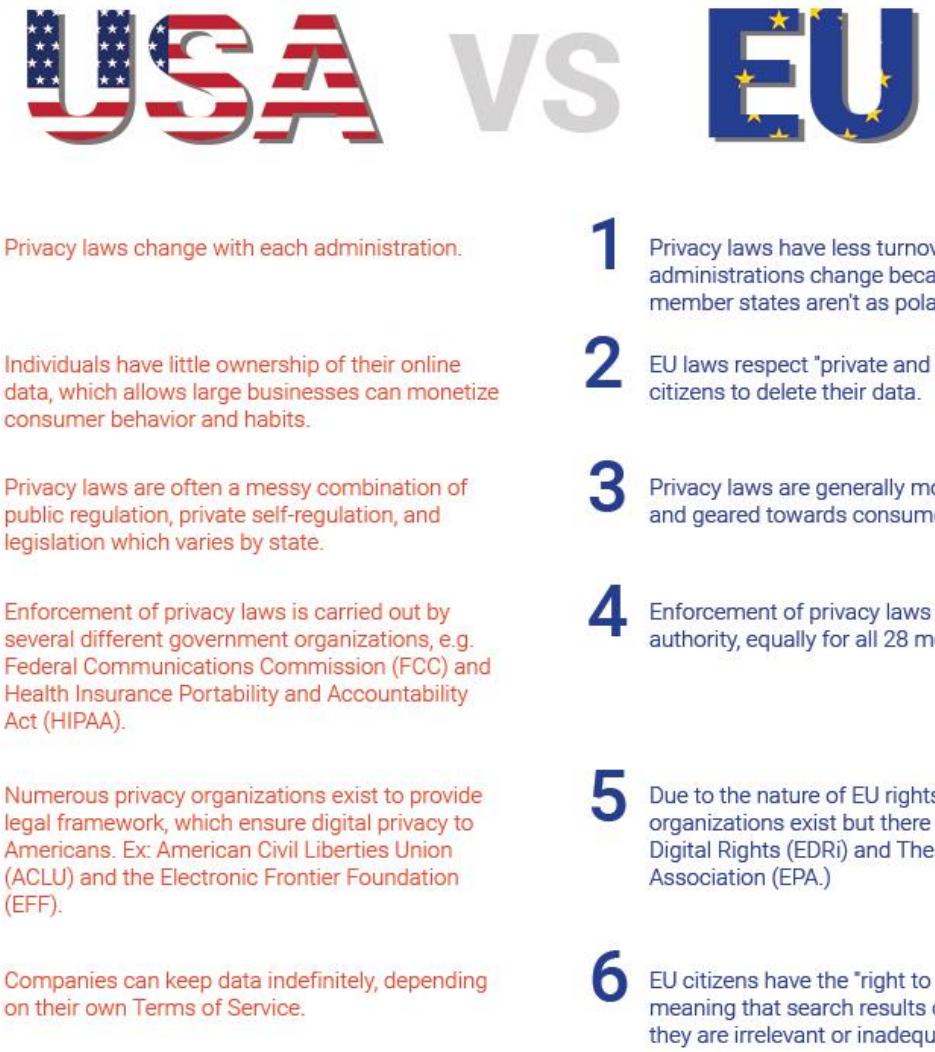
The CLOUD Act

- Clarifying Lawful Overseas Use of Data (CLOUD) Act was signed into law on March 23, 2018
- U.S. law enforcement officials at any level, from local police to federal agents, can force tech companies to turn over user data regardless of where the company stores the data
- Ability to enter into “executive agreements” with foreign nations, which could allow each nation to get its hands on user data stored in the other country, no matter the hosting nation’s privacy laws. These agreements don’t require congressional approval

Differences in Privacy

Source: https://blog.ur-browser.com/en/tag/europe-en/

See also “**10 Things You Need to Know About the EU General Data Protection Regulation”**, https://www.wordstream.com/blog/ws/2017/09/28/eu-gdpr



More on privacy in the EU

- For **personal data**: **GDPR** (General Data Protection *Regulation*) – see https://www.eugdpr.org/
  - "The EU General Data Protection Regulation (GDPR) is the most important change in data privacy regulation in 20 years"
  - GDPR enforcement: 25 May 2018
  - The aim of the GDPR is to **protect** all EU citizens from privacy and data breaches in an increasingly data-driven world
  - Organizations in breach of GDPR can be fined **up to 4% of annual global turnover** or €20 Million (whichever is greater)
- What is *personal data*?
  - Any information related to a natural person or 'Data Subject', that can be used to directly or indirectly identify the person. It can be anything from a name, a photo, an email address, bank details, posts on social networking websites, medical information, or a computer IP address

But for non-personal data?

- E.g. scientific data (non related to ”Data Subjects”)
  - See the draft “Regulation of the European Parliament and of the Council on a framework for the free flow of non-personal data in the European Union”, http://eur-lex.europa.eu/legal-content/EN/TXT/PDF/?uri=CELEX:52017PC0495&from=EN
- Key points:
  - Free movement of data within the Union (no strict data localization)
  - Data availability for competent authorities (across the Union)
  - Porting of data (no lock-in)

Miscellanea

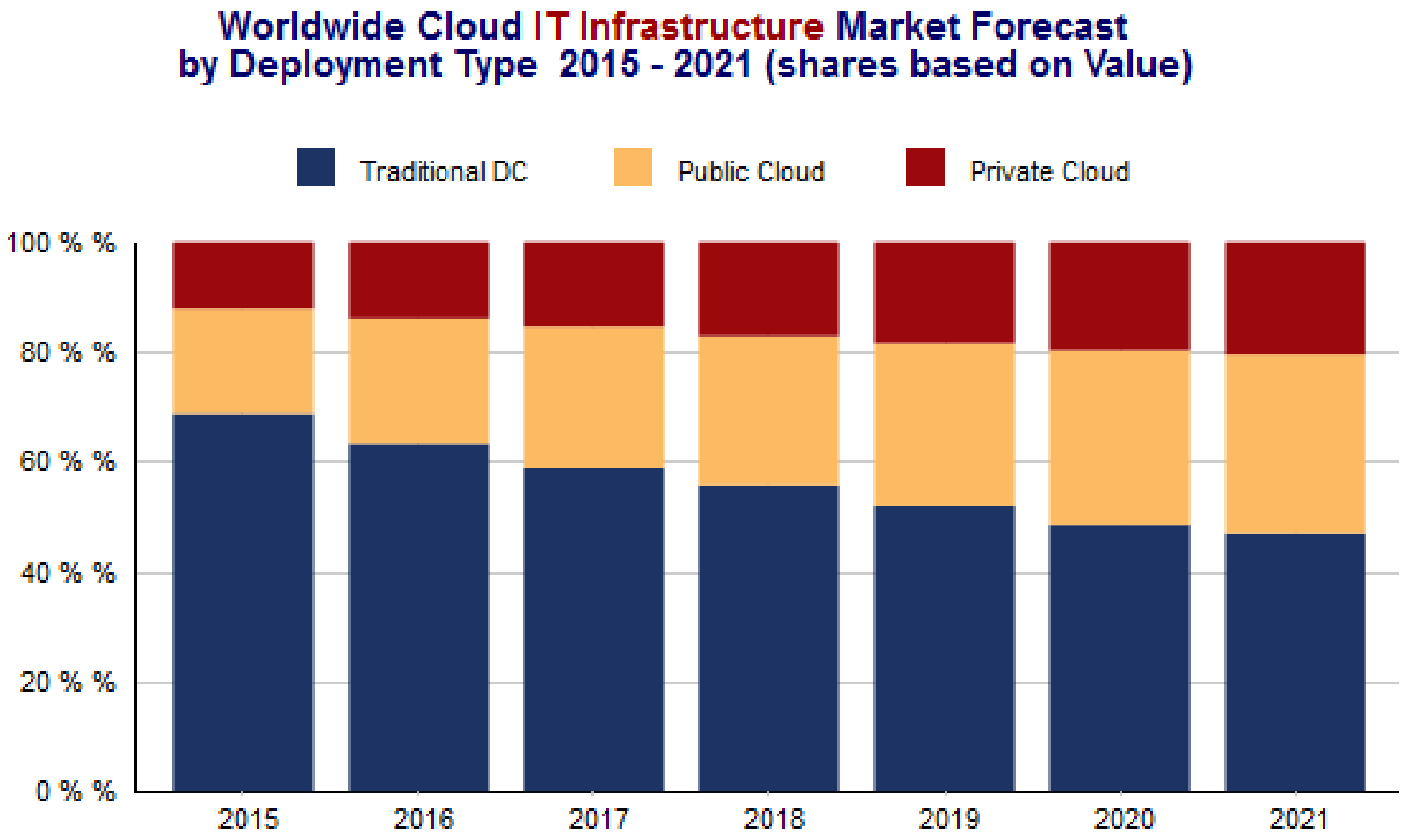
Last but not least, the big misunderstanding

- **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?

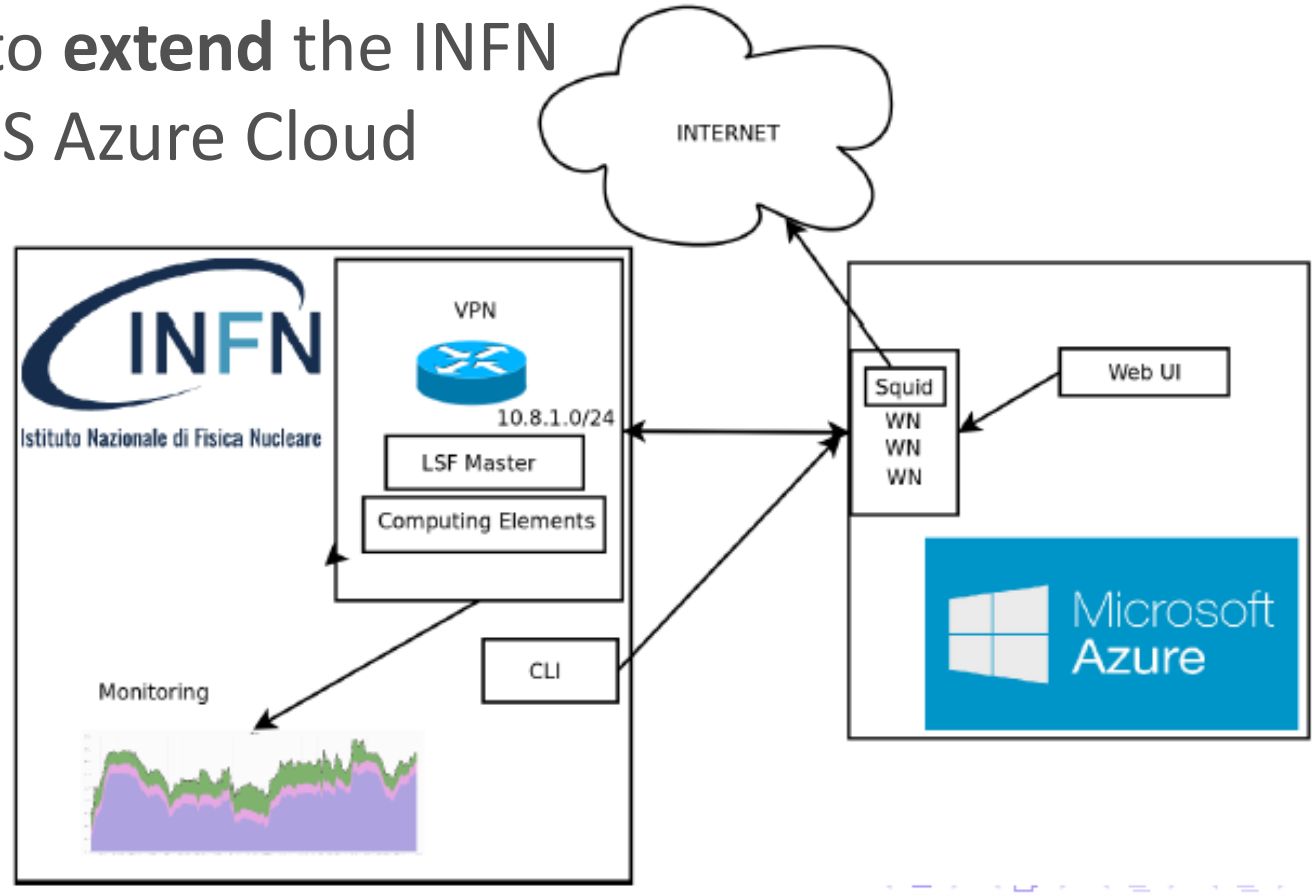
Cloud market Forecast



Source: bit.ly/2yZMrsp

CNAF extension on public Cloud

- The increasing demand of computing resources led to the investigation of several techniques to dynamically extend the existing farm
- An approach to **extend** the INFN farm to the MS Azure Cloud



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**

There is no Cloud,

it’s just someone else’s computer

Infrastructure for research @Europe

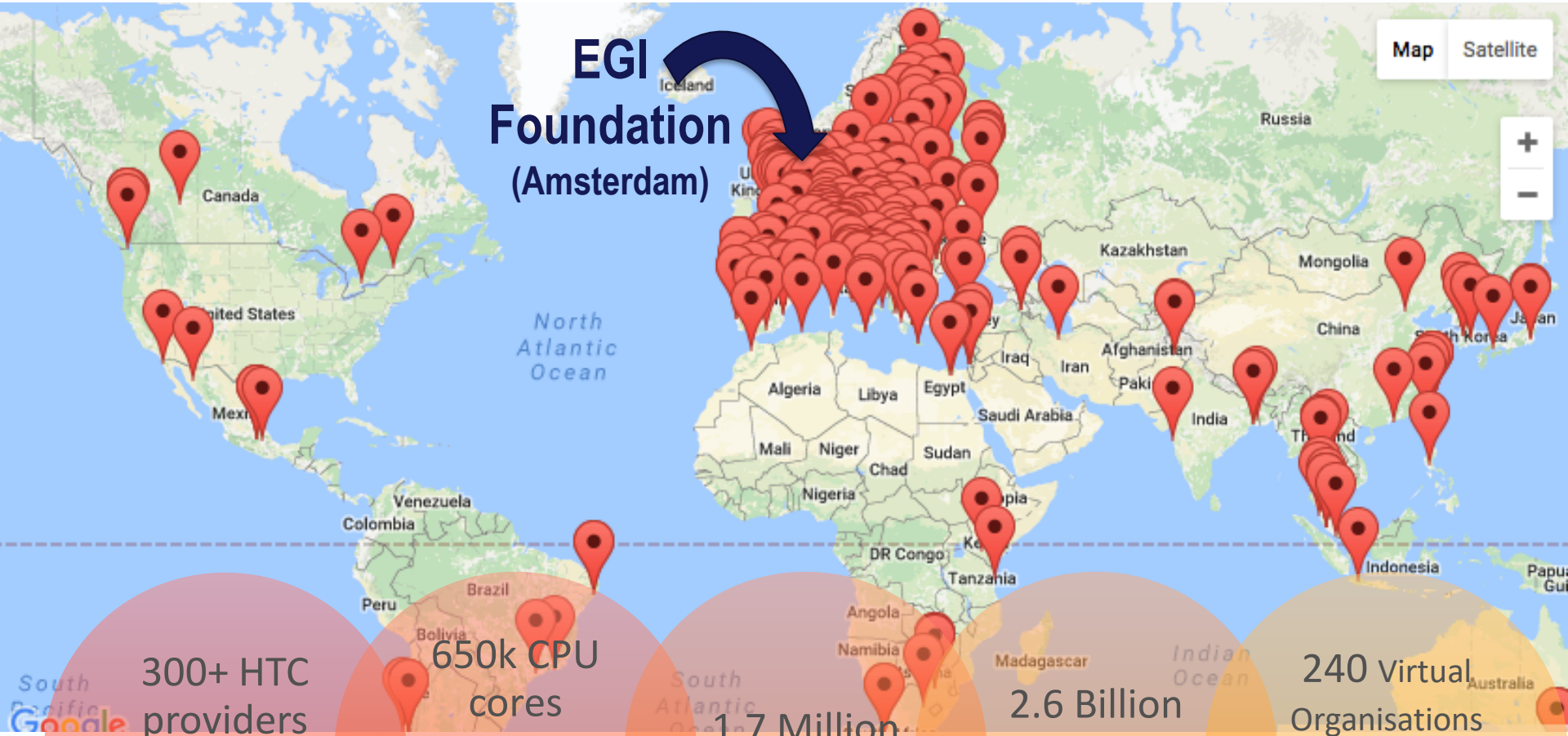
EGI: Federation of national e-infrastructures

- Established in 2010
  - EGI Foundation: Coordinator (Amsterdam, Science Park)
  - NGIs: National e-infrastructures (22 country + CERN)
- Membership fees sustain the federation; Projects innovate (e.g. EOSC-hub)
- EGI = Compute, Storage, Data, Training, Consultancy services

Institutional representatives

EGI Federation

Largest distributed compute e-Infrastructure of the world



300+ HTC providers

25 Cloud providers

650k CPU cores

285 PB online storage

1.7 Million jobs/day

2.6 Billion CPU hours/year

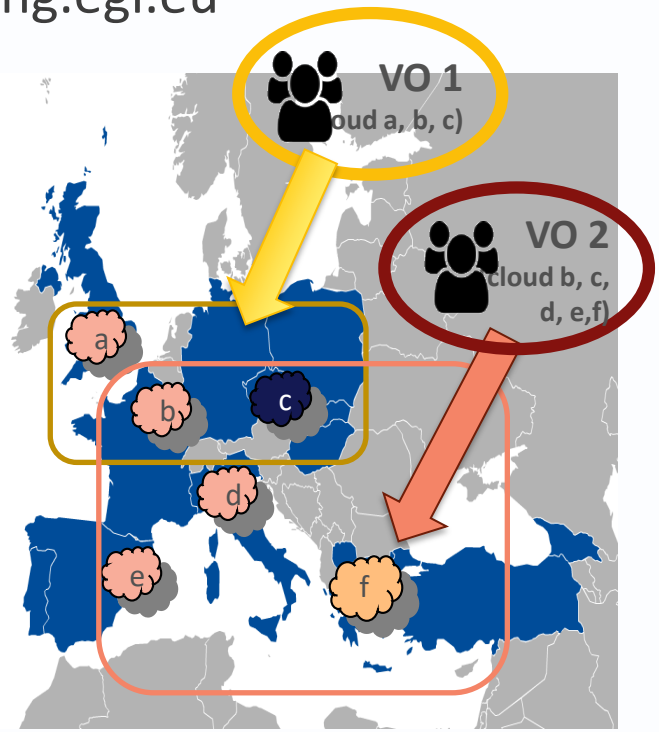
240 Virtual Organisations

>48 000 users

International Partnerships

EGI Cloud Compute IaaS

- Run Virtual Machines on demand
  - Similar to AWS EC2/EBS or GCP Compute Engine
- Access is based on 'Virtual Organisations'
  - VO = group of users + cloud providers supporting them
    - Community-specific VOs – e.g. CHIPSTER, EISCAT, etc.
    - Generic VOs – e.g. fedcloud.egi.eu, training.egi.eu
- Diverse providers with common:
  - AuthN and AuthZ
  - VM Image catalogue (applications)
  - Information discovery
  - Accounting
  - Monitoring
  - GUI dashboard



Summary – Comparing compute infrastructures

Commercial clouds (e.g. Azure, AWS, GCE)	Academic clouds (e.g. EGI federated cloud, INFN, others)
• For interactive and batch computing • For service hosting • Flexible OS and application use • **Typically one provider is used** • **Pay-as-you-go** • **User support as paid service**	• For interactive and batch computing • For service hosting • Flexible OS and application use • **Single or multiple providers** • **Free at the point of use (for research)** • **With local user support**

Federated access:
Identity and Access Management

Authentication and Authorization

- Typically, each of your applications and each of the resources that you want to access have some form of «**Identity Management System**»

Local username and password

- The simplest form of authentication to a resource is to have local usernames and passwords.
- **Local ->** every system or application should store its own set of credentials.
  - This way, if you connect for example to 10 different machines (or to 10 web applications), those 10 machines or applications should all have a local file storing your credentials (username and password), as well as those of everybody else allowed to connect to them.
    - In Linux systems, this file is typically */etc/passwd*.
- Clearly, this is **not a scalable** method. So, we need something more sophisticated.

Some IDP solutions

- **L**ightweight **D**irectory **A**ccess **P**rotocol (LDAP) to access (browse) the X.500 Directory via TCP/IP.
  - https://ldap.com
- **R**emote **A**uthentication **D**ial-**I**n **U**ser **S**ervice (RADIUS) is an intermediate service that can be used to connect to various services related to authentication and authorization.
  - https://tools.ietf.org/html/rfc2865
- **Kerberos:** a service based on shared keys offering strong authentication, often combined with LDAP to offer single sign-on for users.
  - https://web.mit.edu/kerberos/
- **X.509 certificates:** a solution used to overcome the issues of shared keys, widely used e.g. to securely identify web servers or Grid users.
  - https://en.wikipedia.org/wiki/X.509
- **S**ecurity **A**ssertion **M**arkup **L**anguage (SAML): a standard used to implement single sign-on for Web applications.
  - http://bit.ly/2sggEAk

And the Cloud?

- All the mechanisms and protocols we have seen so far about Authentication and Authorization are useful and fine.
- However, we still have not solved the problem of how to generally authenticate and authorize services in a Cloud.
- This is where two newer protocols come into play: **OAuth** and **OpenID-Connect**.

Oauth and OpenID-Connect

- **OAuth** is an authorization framework.
  - It deals with **Authorization**.
  - OAuth was designed to handle the Authorization of *generic applications or resources* (such as accounts, files, etc.) on the Internet.
    - https://oauth.net
- **OpenID-Connect (OIDC)** protocol
  - Is a simple identity layer on top of the OAuth framework, handle **Authentication**
  - Gives information about **who the user is** and **how it was authenticated** via an additional **ID token (JSON Web Token, or JWT)**
    - http://openid.net/connect/

A Cloud-friendly AAI solution: INDIGO IAM

INDIGO - DataCloud

https://www.indigo-datacloud.eu

- The INDIGO Identity and Access Management (**INDIGO-IAM**) service provides a layer where identities, enrolment, group membership, attributes and policies to access distributed res ources and services can be managed in a homogeneous and interoperable way, supporting **the federated authentication mechanisms** (SAML, OpenID Connect and X.509) behind the INDIGO AAI.
- The IAM service provides user identity and policy information to services so that consistent authorization decisions can be enforced across distributed services.
  - https://indigo-iam.github.io/docs/v/v1.4.0/

A Cloud-friendly AAI solution: INDIGO IAM

Flexible authentication support

- (SAML, X.509, OpenID Connect, username/password, …)

Account linking

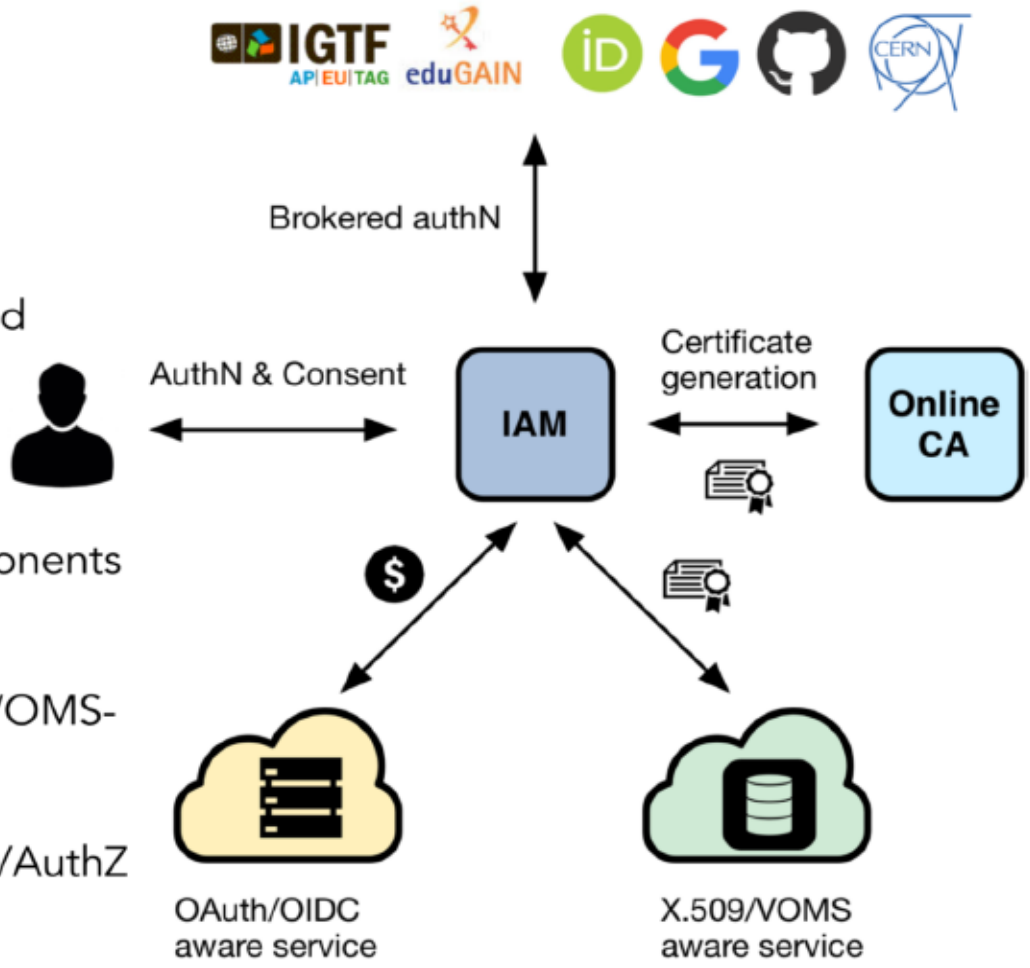
Registration service for moderated and automatic user enrollment

Enforcement of AUP acceptance

Easy integration in off-the-shelf components thanks to **OpenID Connect/OAuth**

VOMS support, to integrate existing VOMS-aware services

Self-contained, comprehensive AuthN/AuthZ solution



Demo and click-through

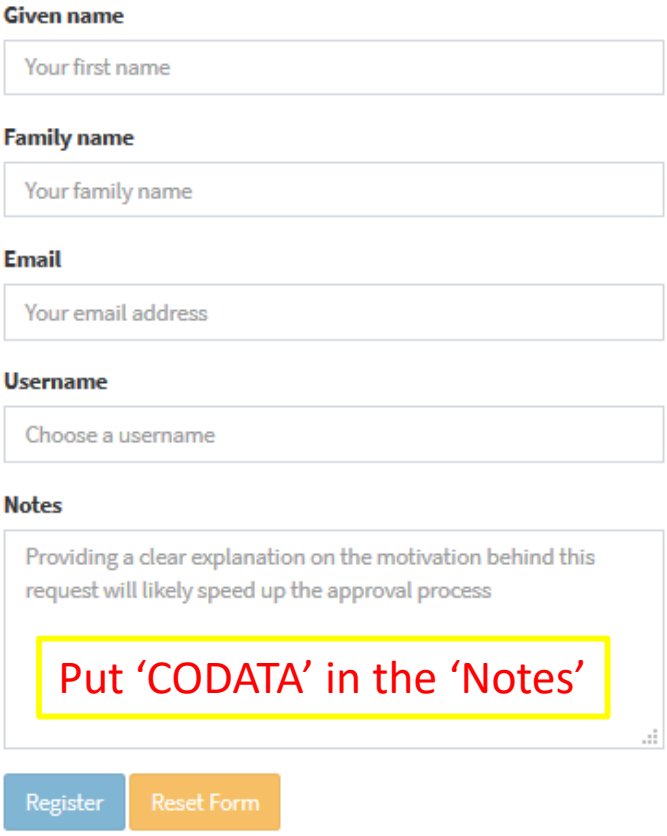
Register to the **iam-demo** service

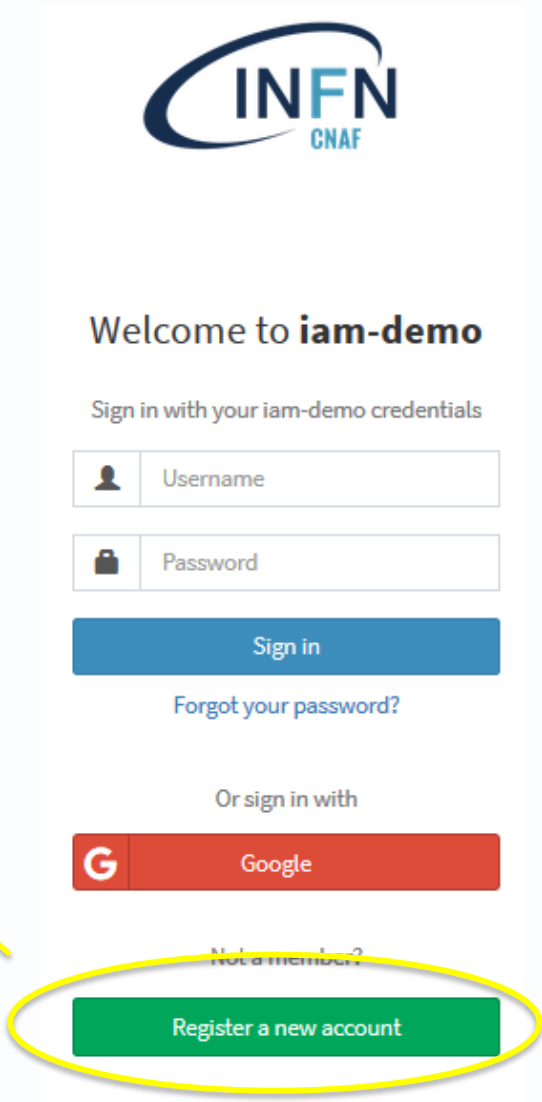
Account will be valid for the duration of the course

Register new account into iam-demo

Register a new account into the

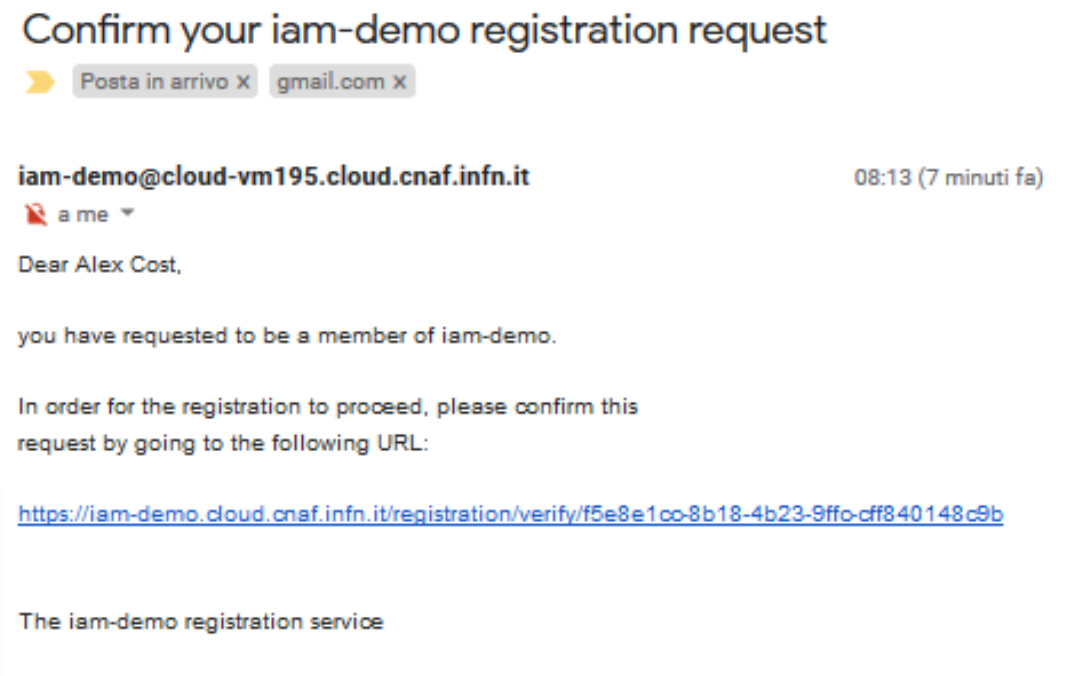
- https://iam-demo.cloud.cnaf.infn.it

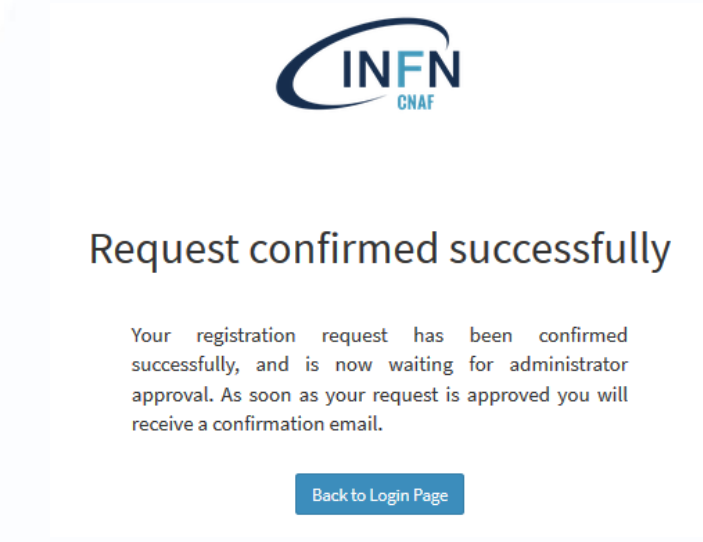




Register new account into iam-demo

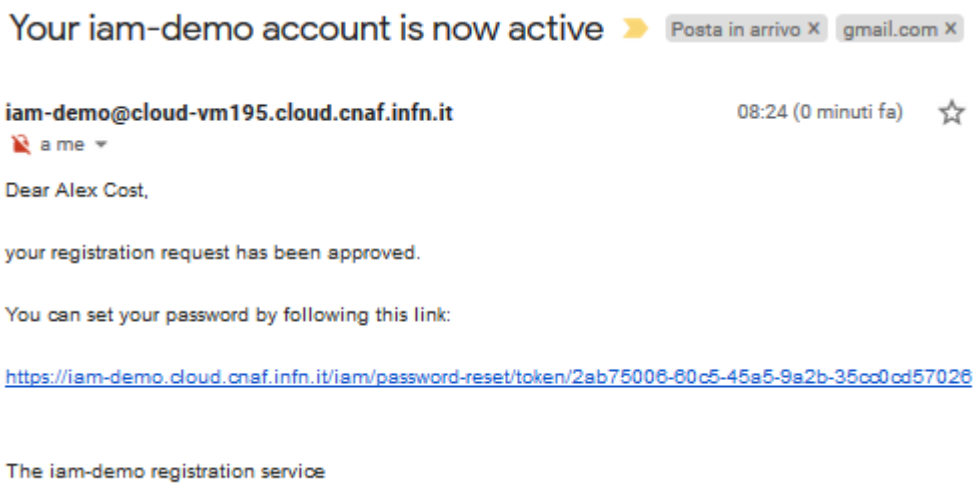
- You will receive a confirmation email, go to the provided link

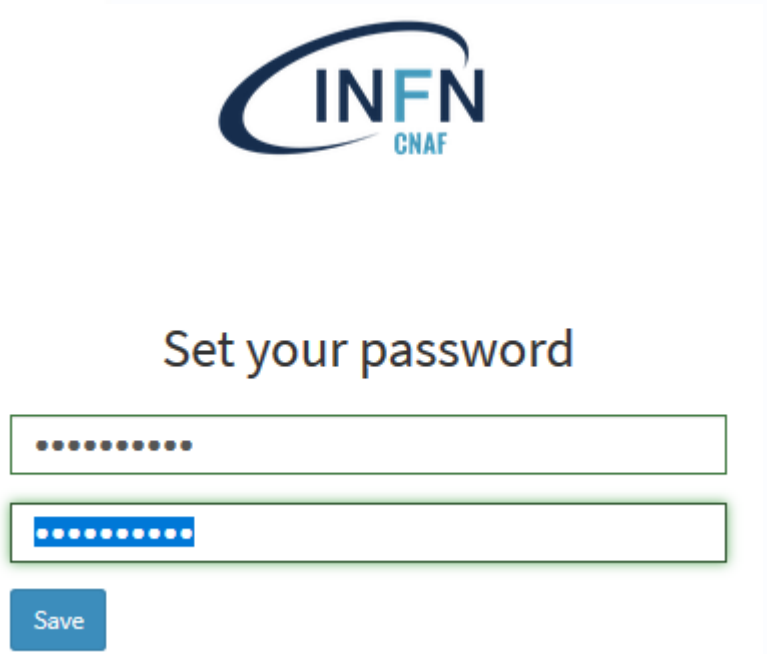




Register new account into iam-demo

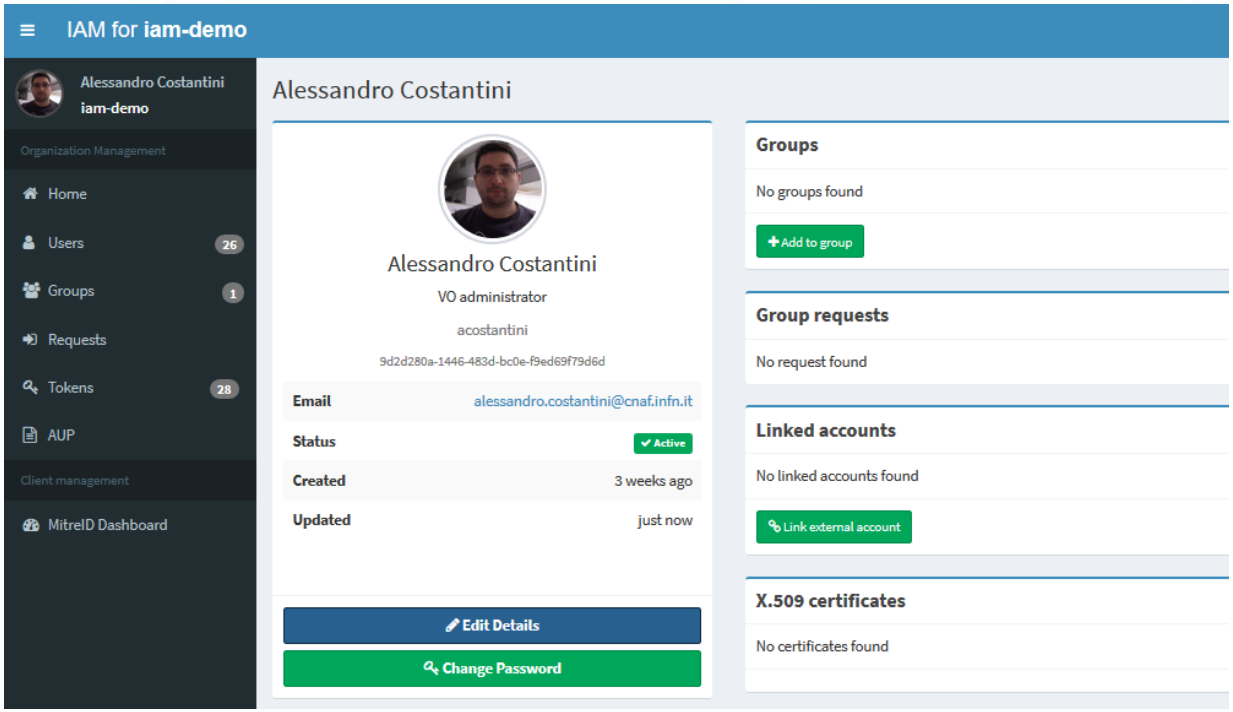
- Wait for the confirmation mail and set your password

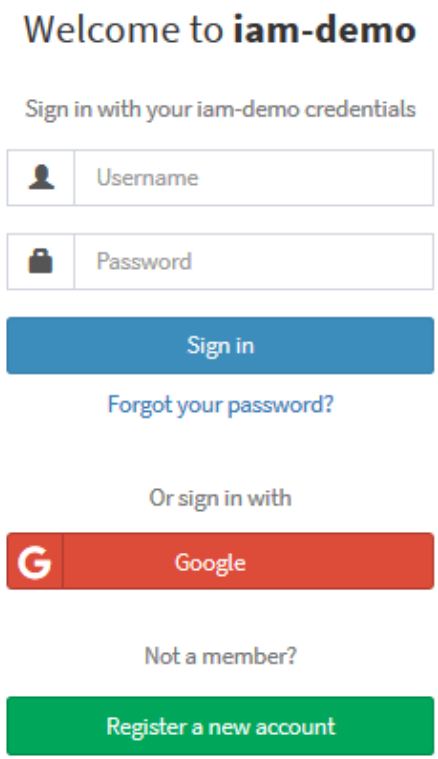




Explore iam-demo

- Use your credentials
  - https://iam-demo.cloud.cnaf.infn.it





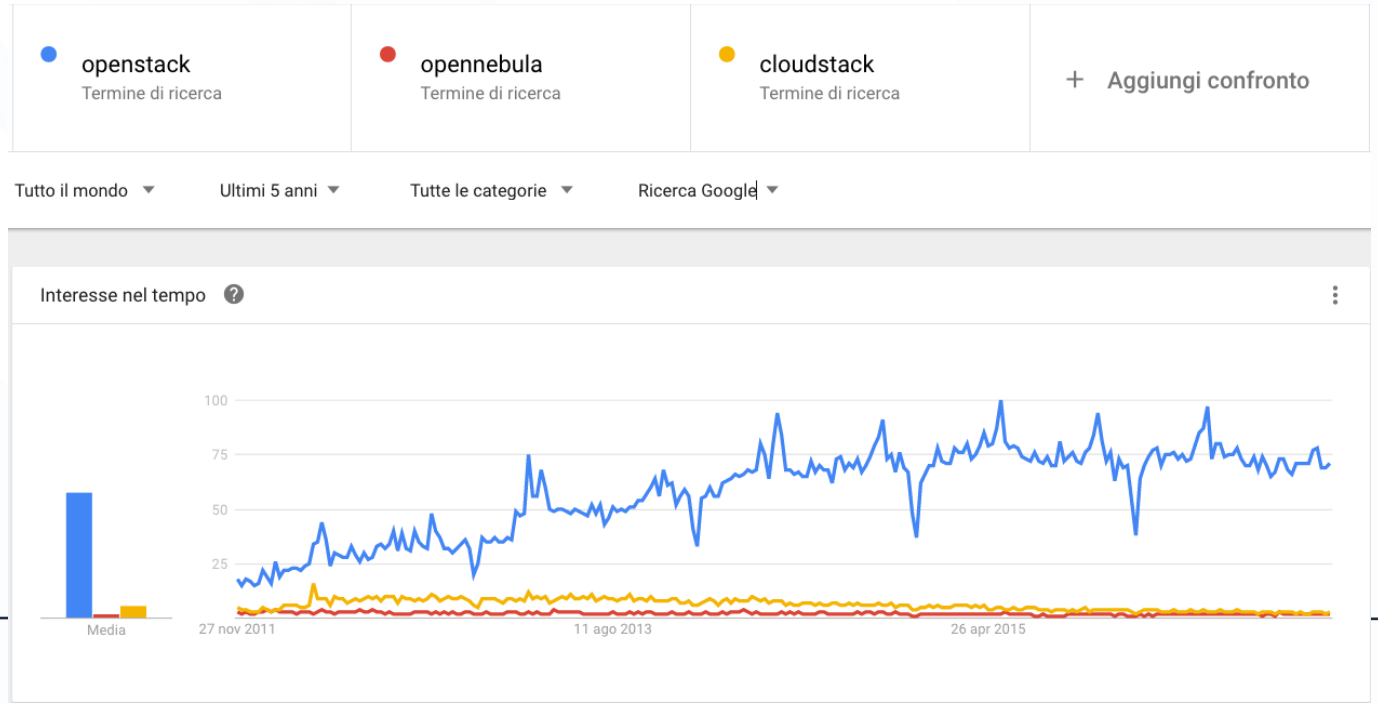
Hands on
Dealing with IaaS:
The OpenStack implementation

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 1/3

OpenStack is

- ***Open source***
  - Fully Functional Open Source
    - Pluggable functionalities
  - Acceptable Licensing - Apache License, 2.0
  - Dependencies and Optional Modules - need to be vetted in the global requirements

OpenStack principles 1/3

OpenStack is

- ***Openly designed***
  - Common development cycle – (most) release every 6-months
    - Common cycle with development milestones
    - Common cycle with intermediary releases
    - Trailing the common cycle
    - Independent release model
  - Design summit (~ Ubuntu Devels Summit)
    - Beginning devel. Cycle
    - Open to public
    - Feedback on new features & final implementation, contributors, testers

OpenStack principles 2/3

OpenStack is

- *Openly developed*
  - Engage larger communities, boarder group of members
  - Project Team Leader (PTL), Code Reviewers
  - Specifications - http://specs.openstack.org/
- by an ***Open community***
  - Public Meetings on IRC - limited number of meeting channels
  - Project IRC (Internet Relay Chat) Channels – meetings logged
  - Mailing Lists
  - Community Support Channels
    - Bugs on Launchpad
    - Mailing list requests
    - IRC message requests
    - ask.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

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

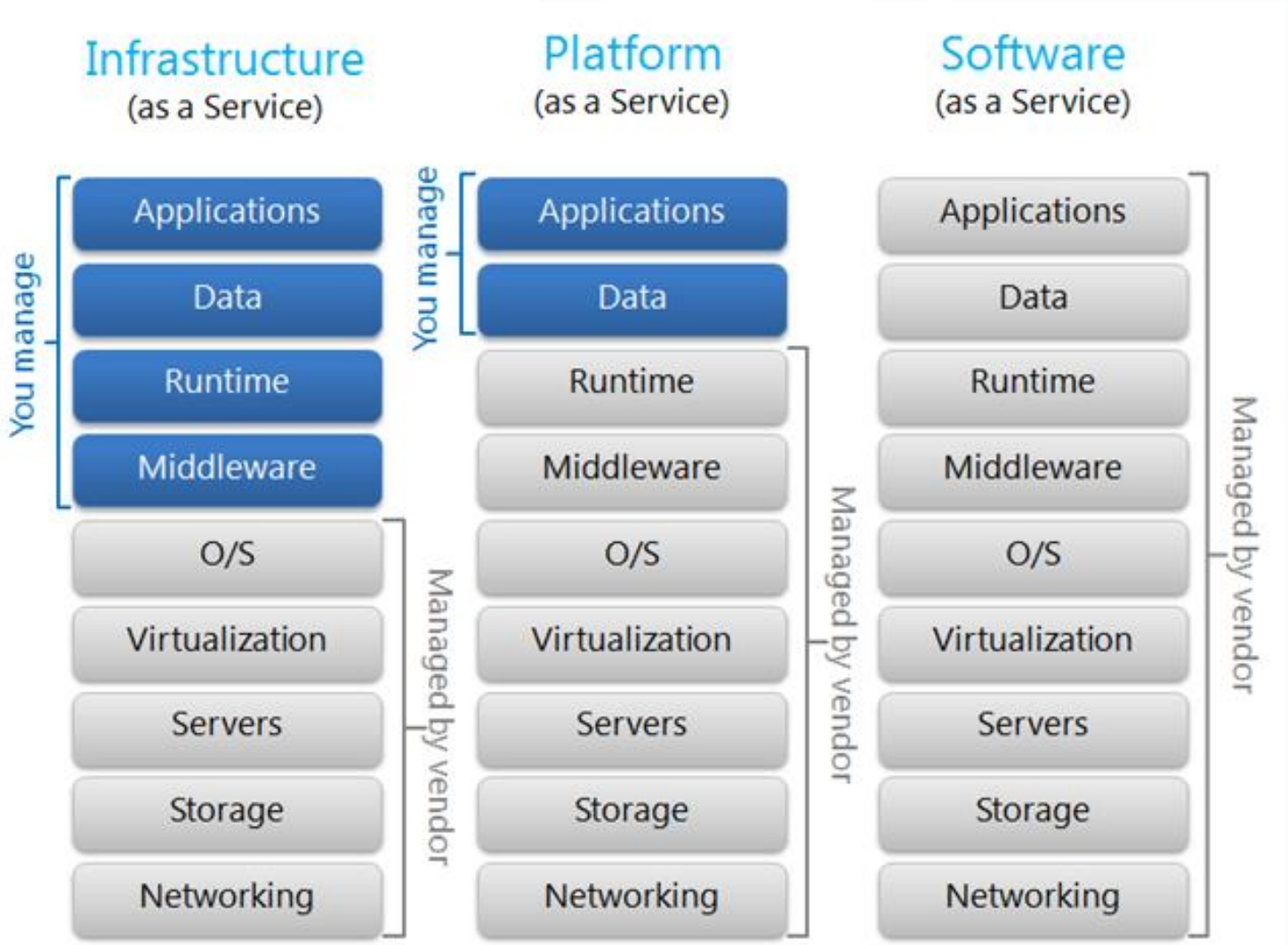
20 gold ($50K/y – 200K/y)

Release

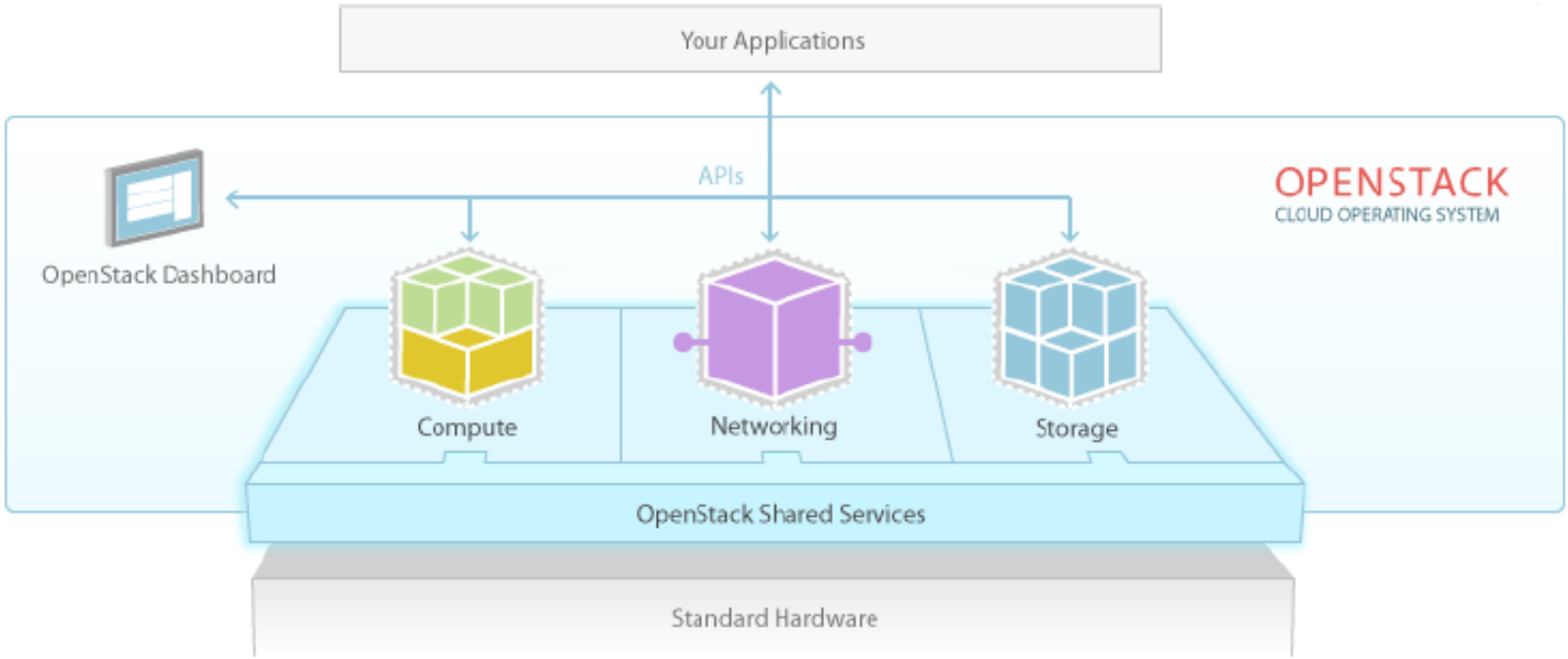
- OpenStack is developed and released around 6-month cycles.

Series	Status	Initial Release Date	Next Phase	EOL Date
Train	Development	2019-10-16 *estimated* (schedule)	Maintained *estimated 2019-10-16*	
Stein	Maintained	2019-04-10	Extended Maintenance *estimated 2020-10-10*	
Rocky	Maintained	2018-08-30	Extended Maintenance *estimated 2020-02-24*	
Queens	Maintained	2018-02-28	Extended Maintenance *estimated 2019-10-25*	
Pike	Extended Maintenance	2017-08-30	Unmaintained *estimated TBD*	
Ocata	Extended Maintenance	2017-02-22	Unmaintained *estimated 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

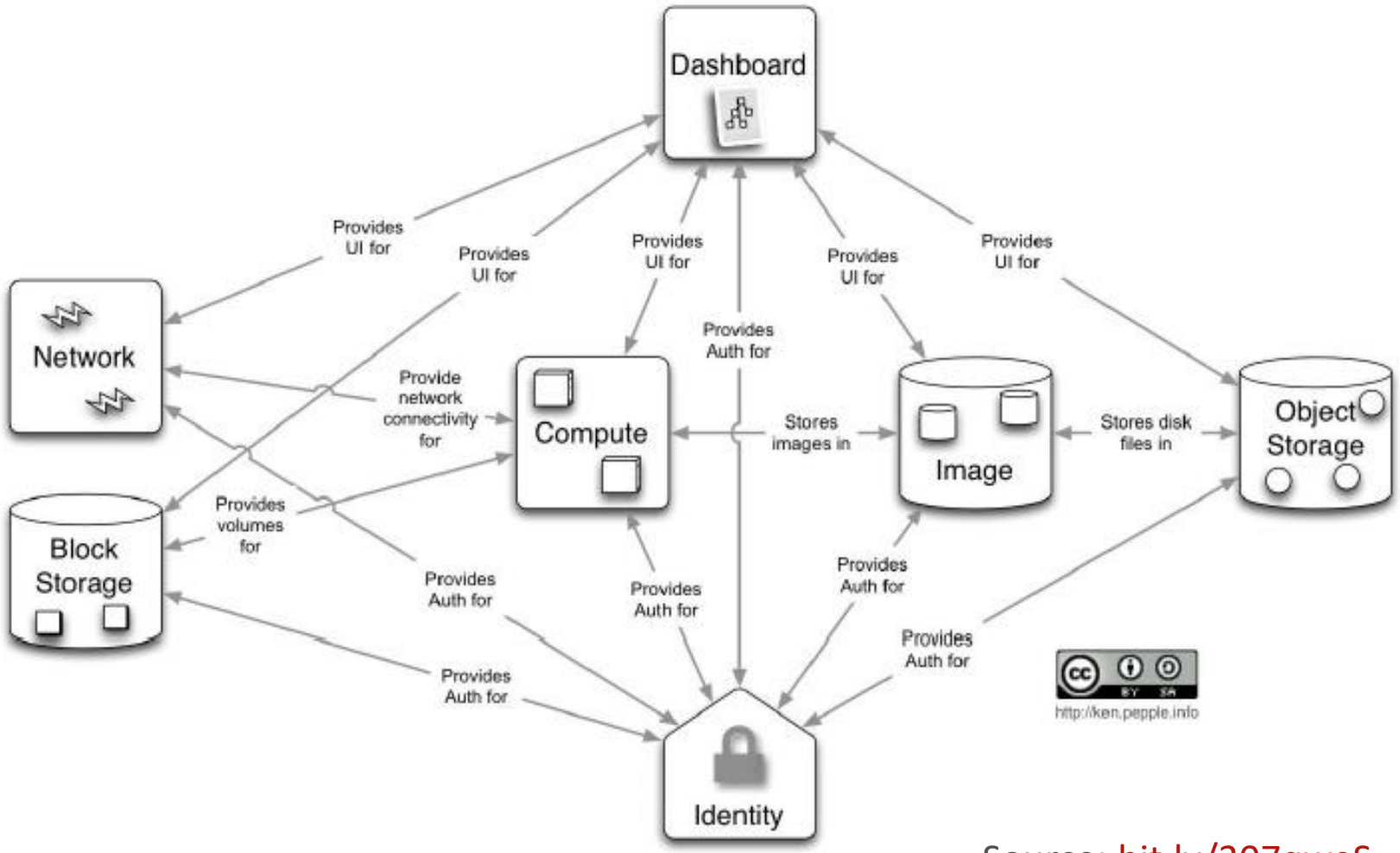


OpenStack: HL overview



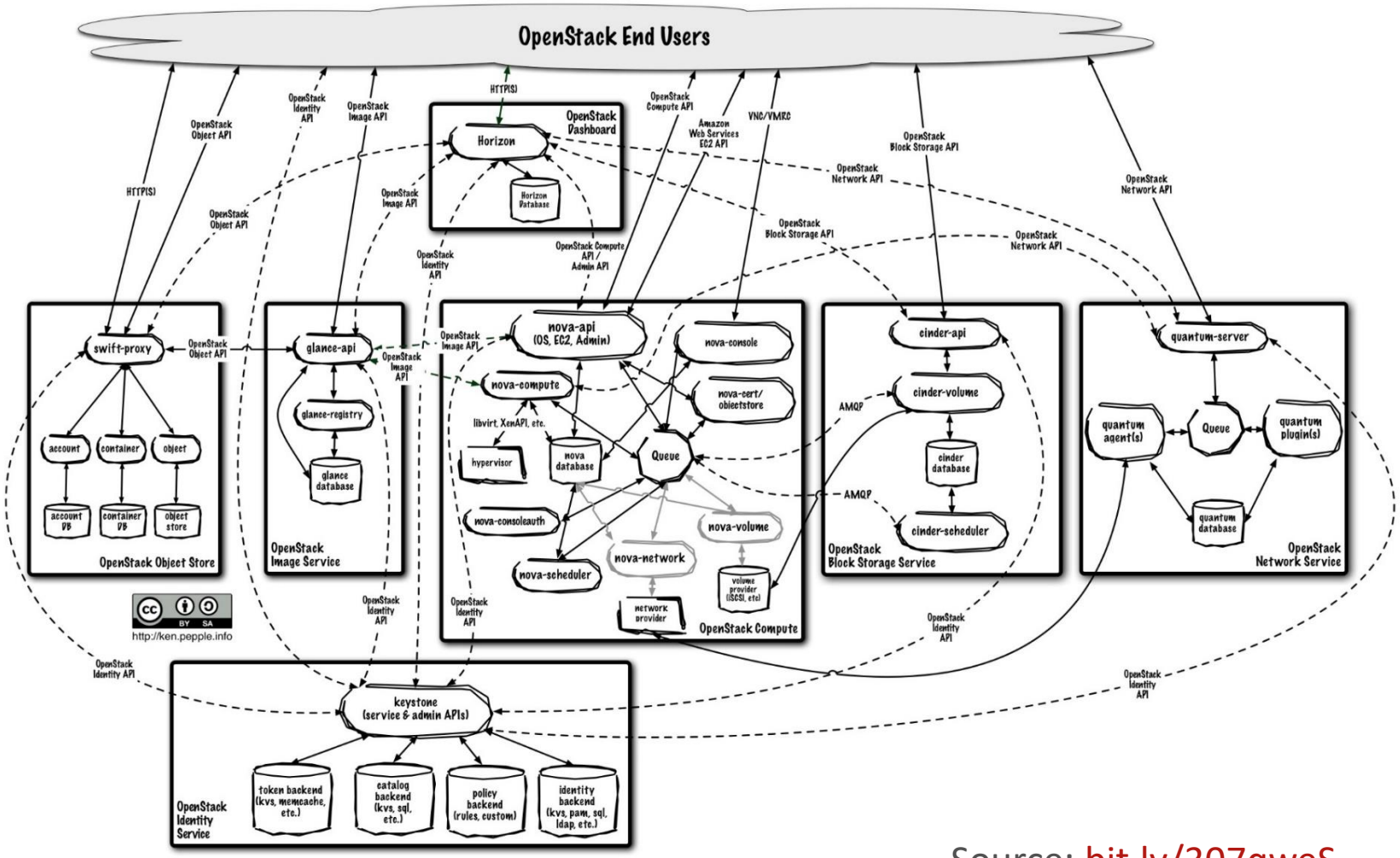
Source: OpenStack

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

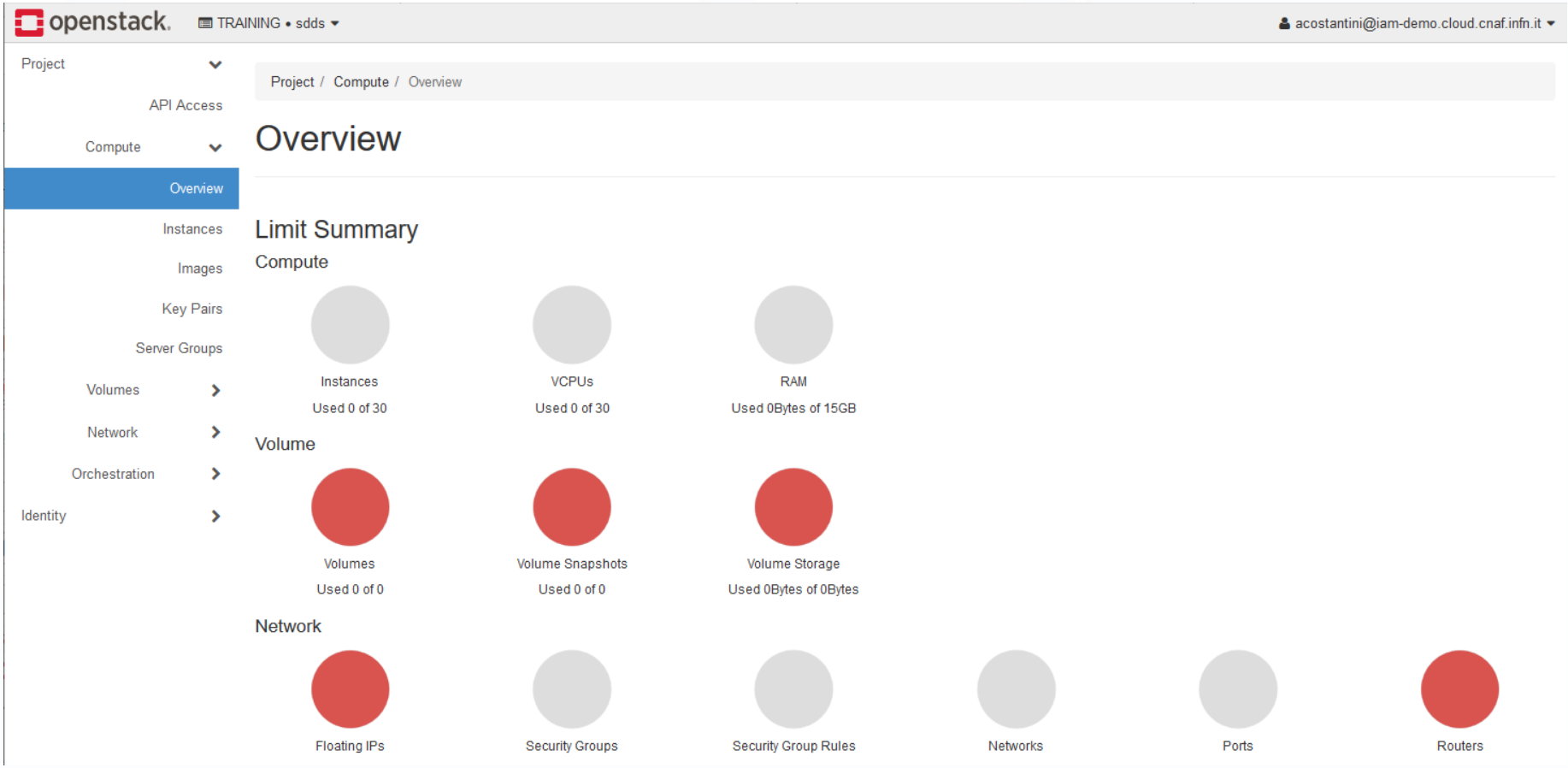
Domain, Project, Domain, Users, Roles

- **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

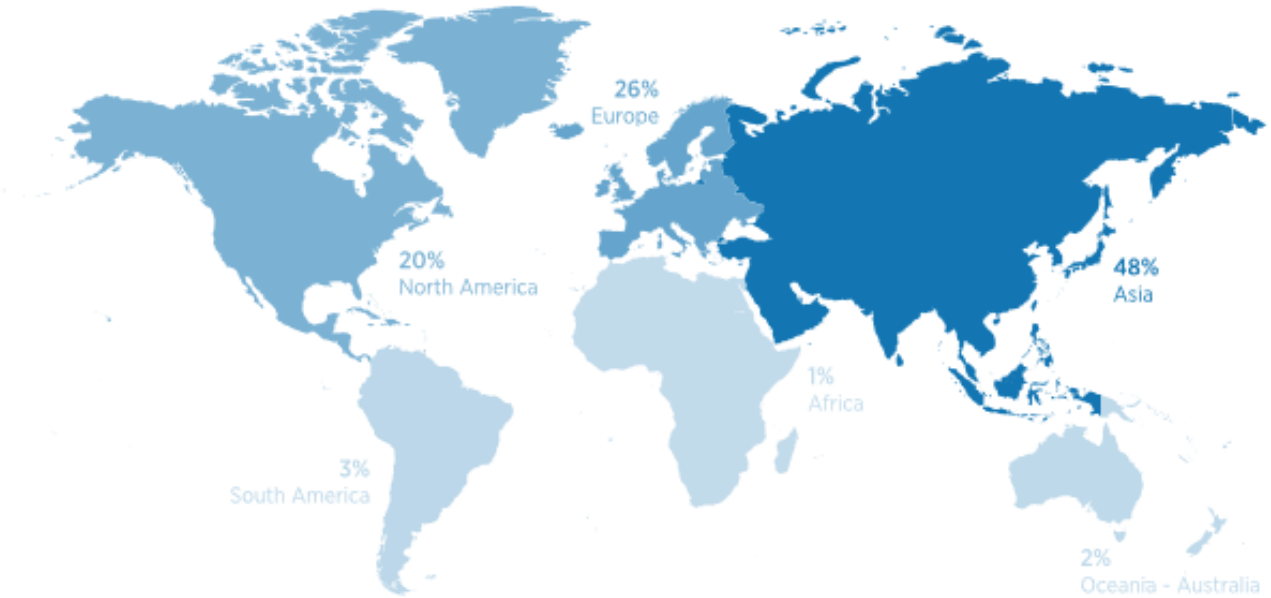
Promoted by OpenStack Foundation

Survey Snapshot

- Data set time frame: August 2017 through August 2018
- Deployments: 858
- Unique organizations represented: 441
- Respondents with more than one deployment: 115

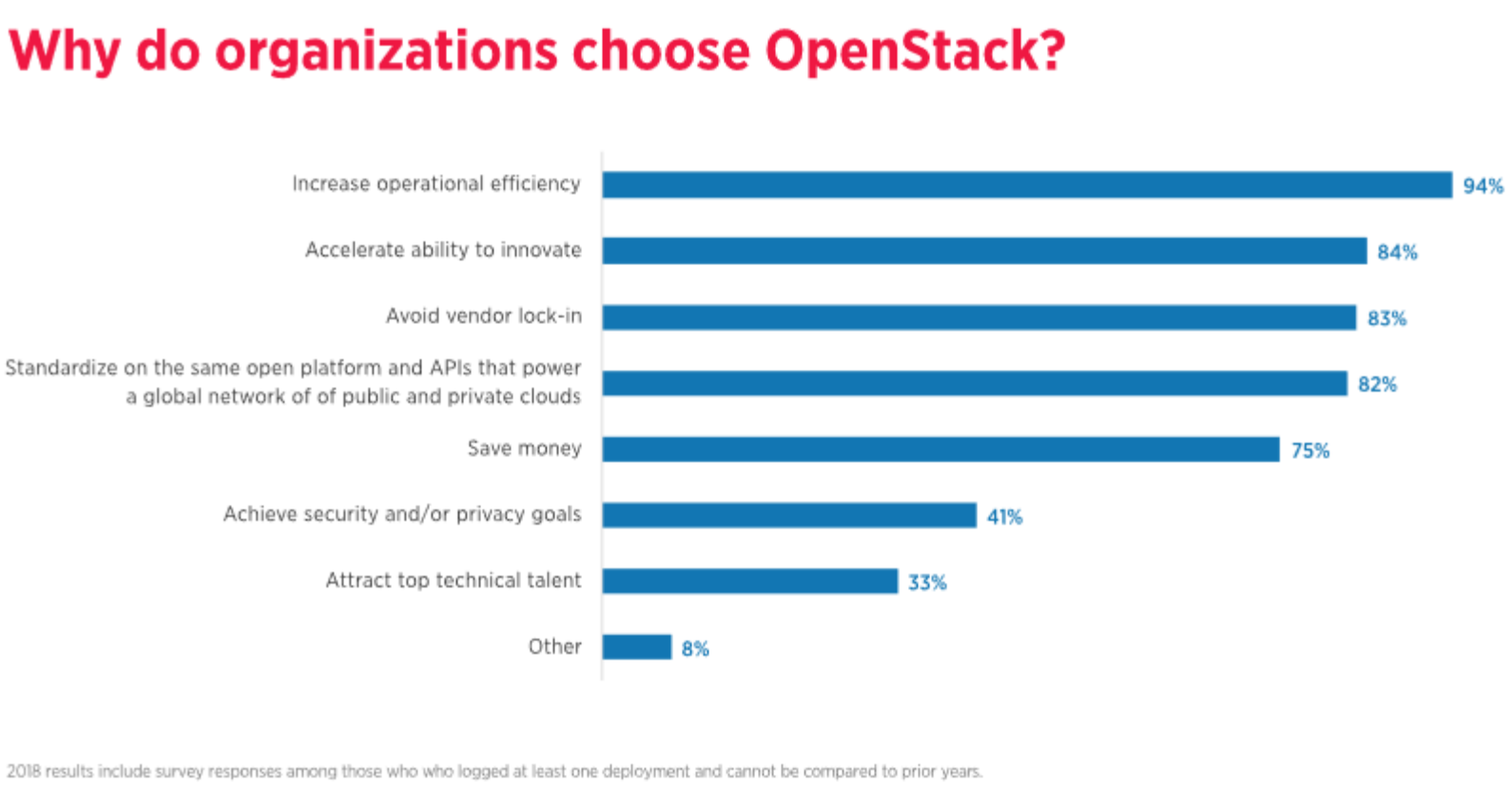
OpenStack survey

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 exercise 1

Deploy a VM in the OpenStack cloud infrastructure

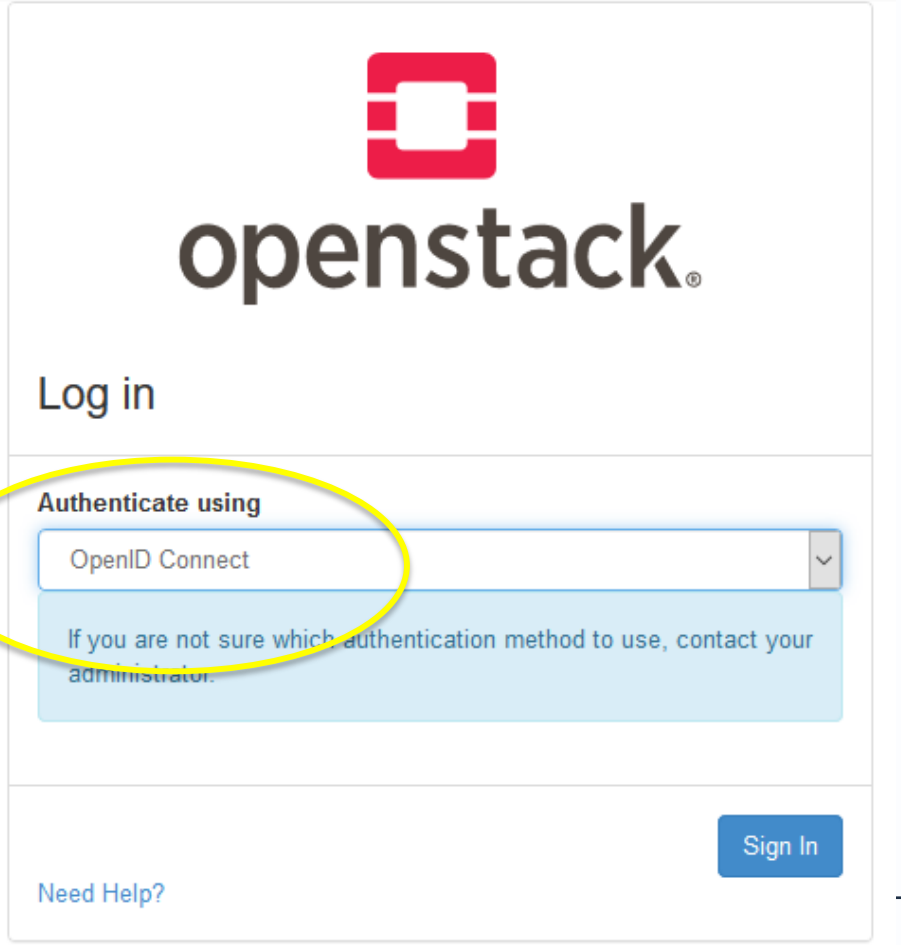
Suggestion: work in groups made by 2 or more people

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

Deploy your first VM

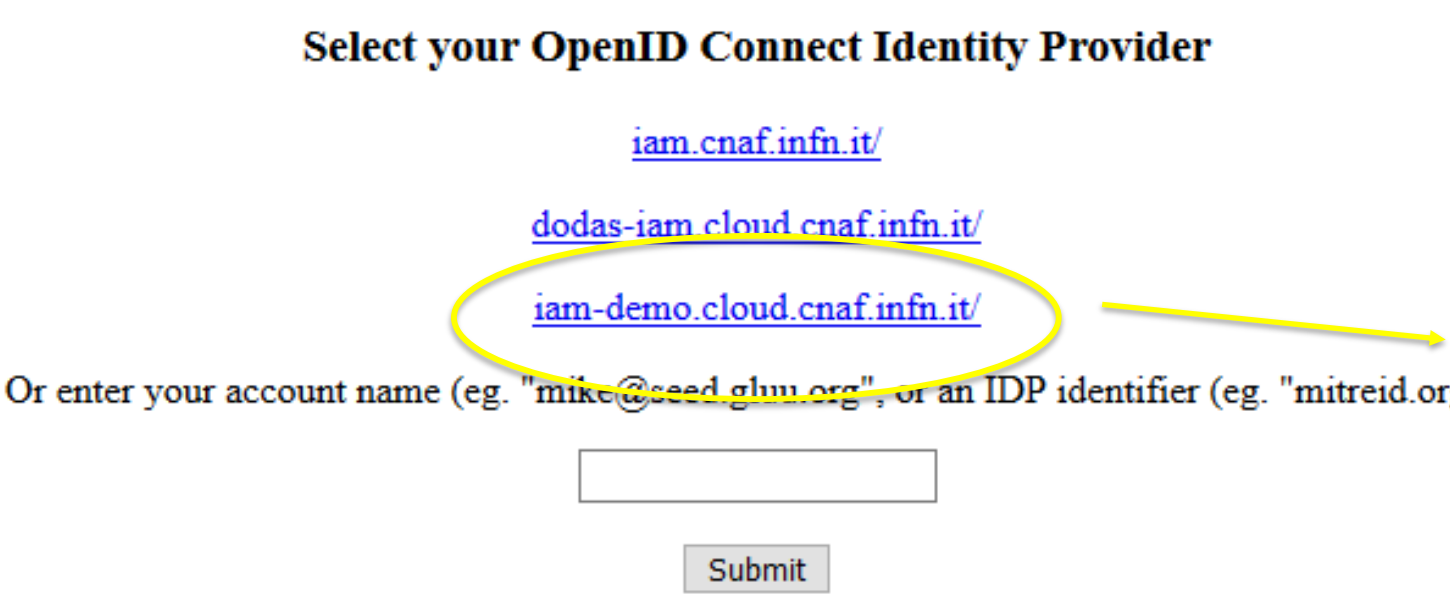
Access the Openstack infrastructure

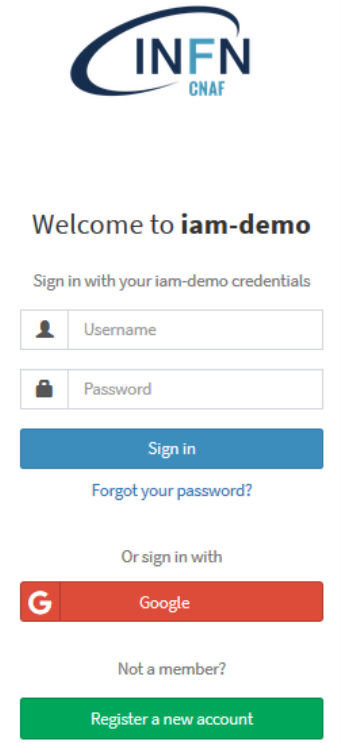
- https://cloud-dashboard.cnaf.infn.it/dashboard



Deploy your first VM

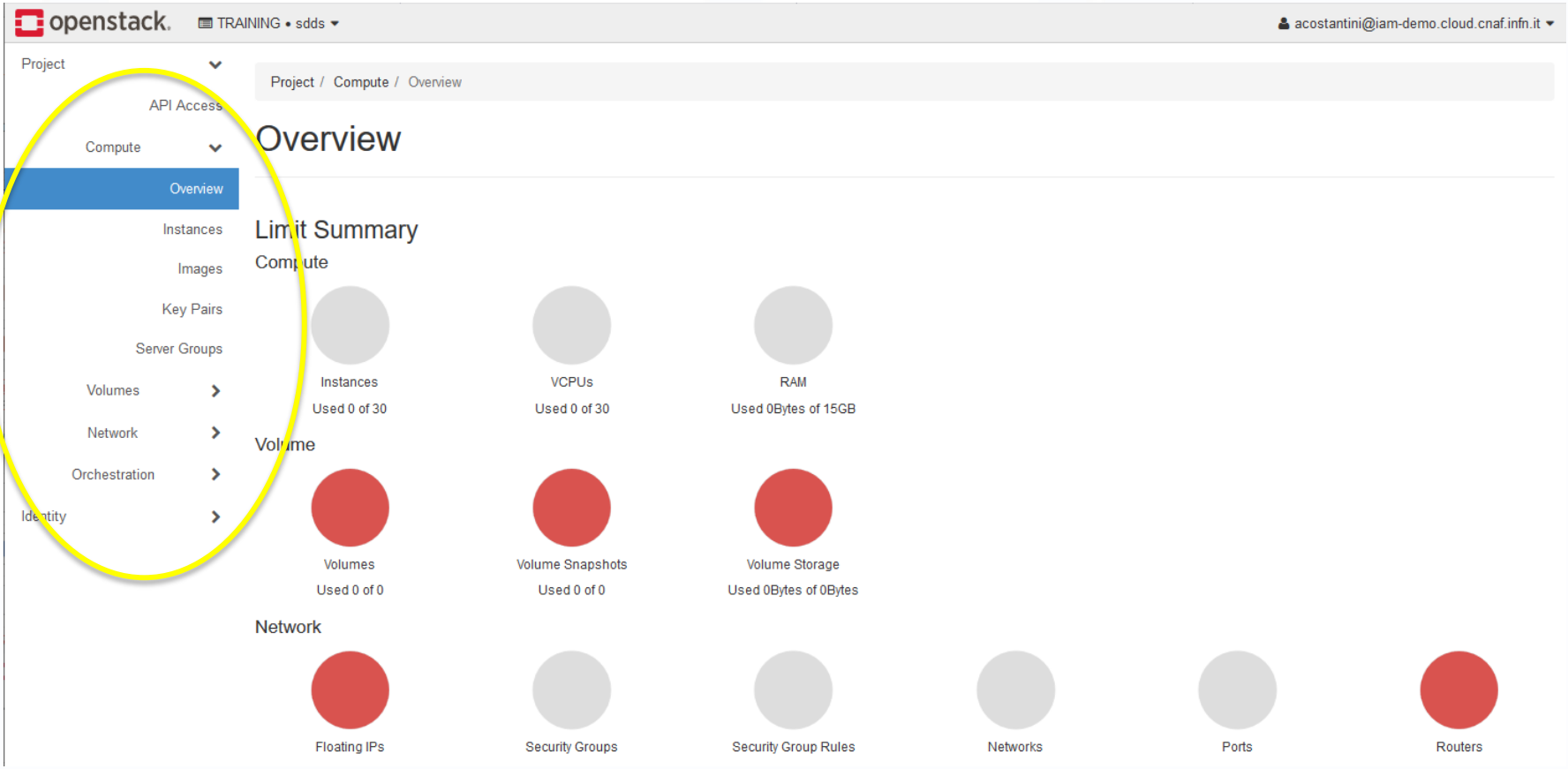
- Select **iam-demo** IDP





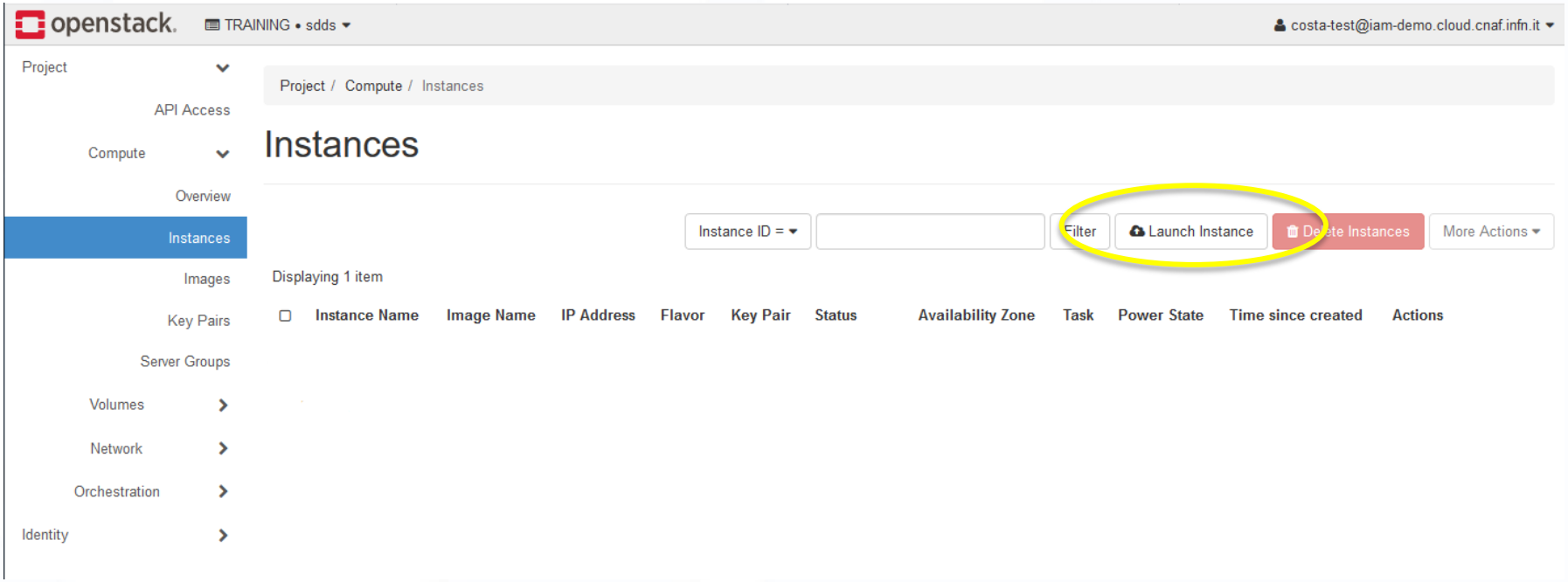
- **Authorize the Client**

Resources overview

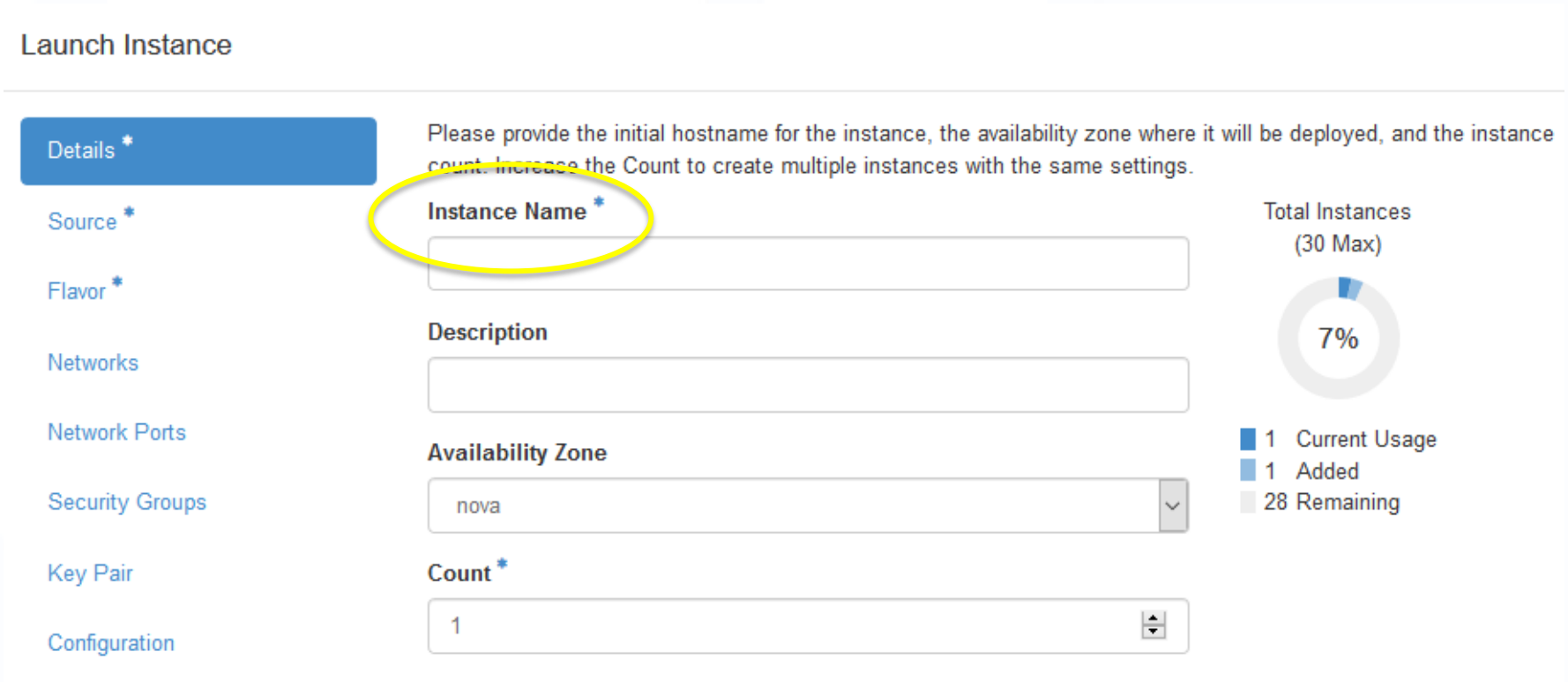


Instance

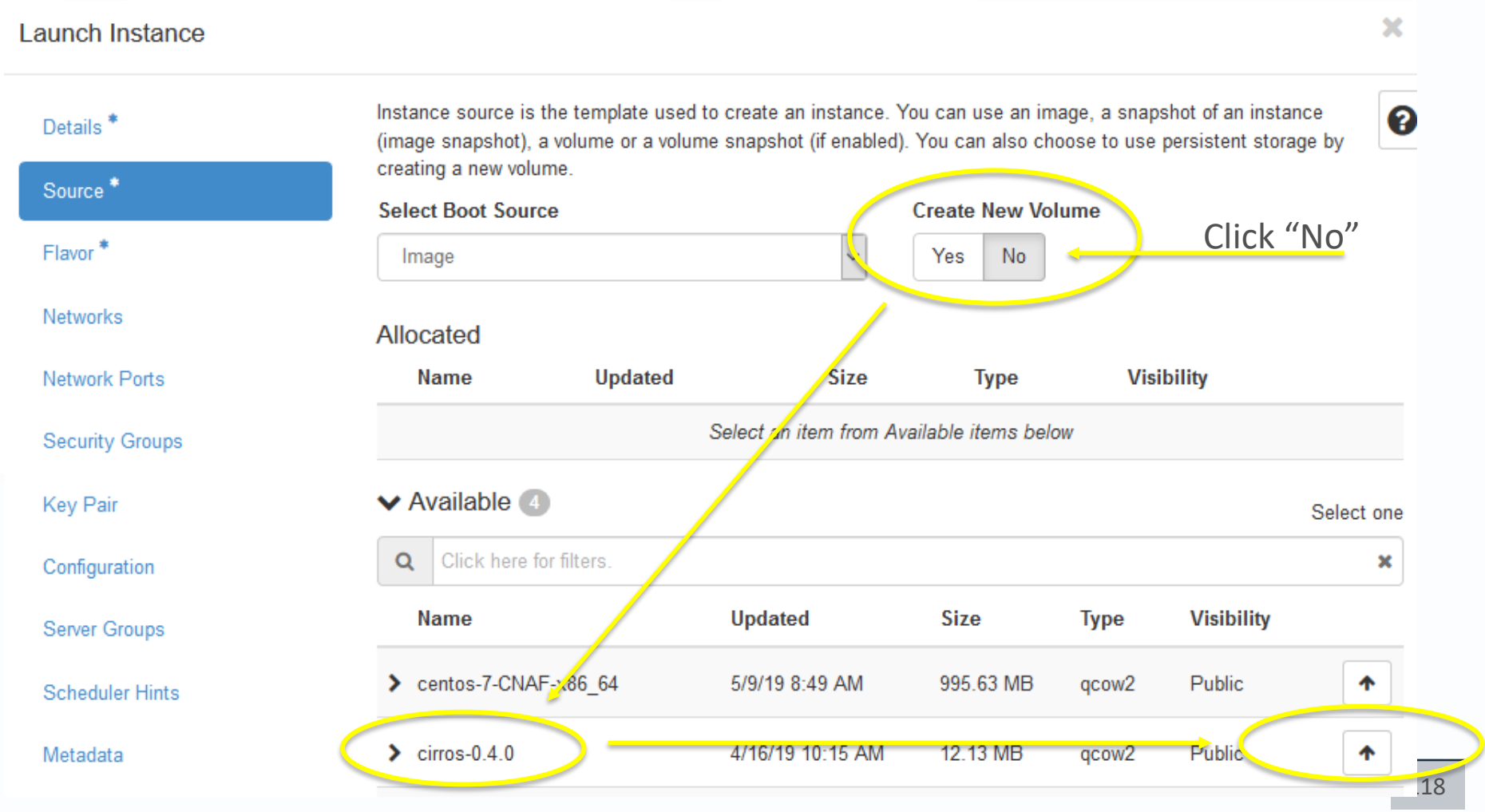
- Deploy new instance



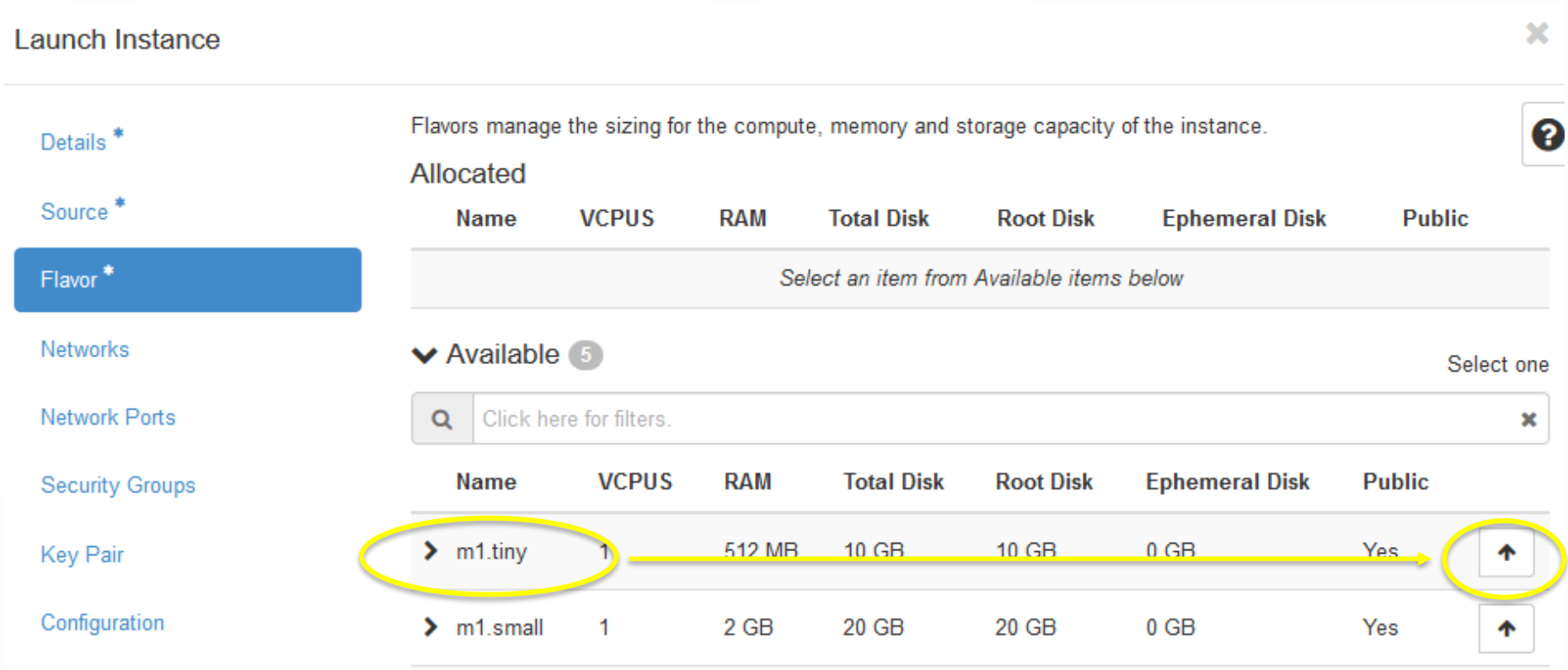
Define Instance name



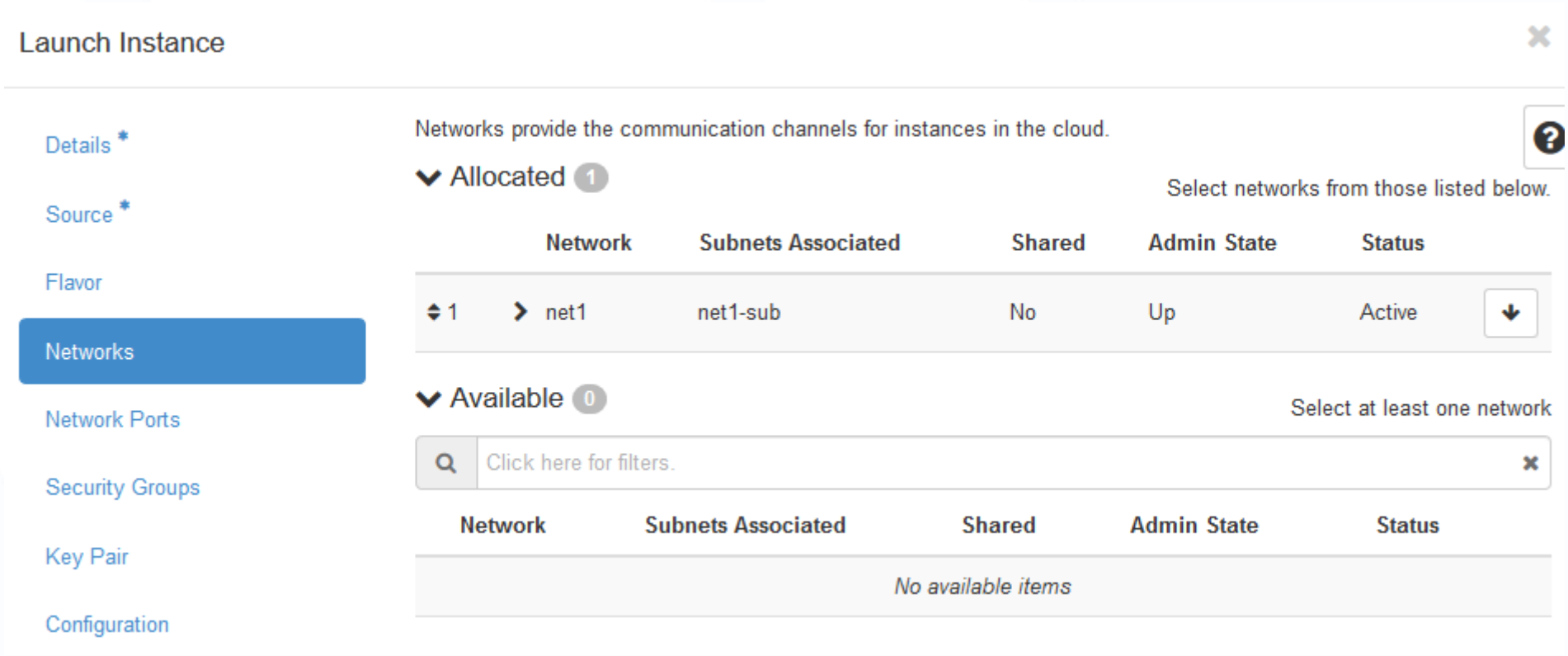
Select Image (cirros)



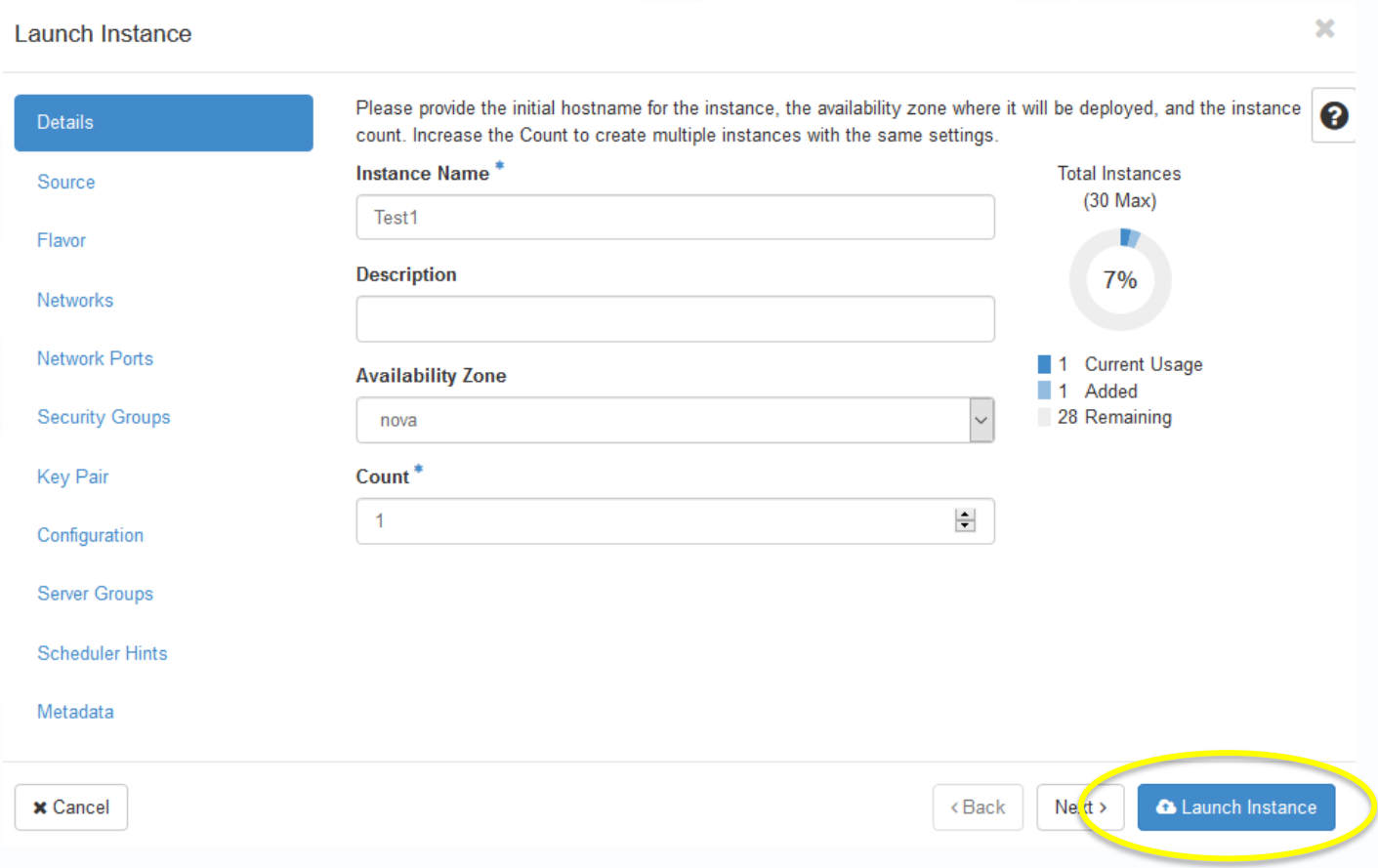
Select Flavour (m1.tiny)



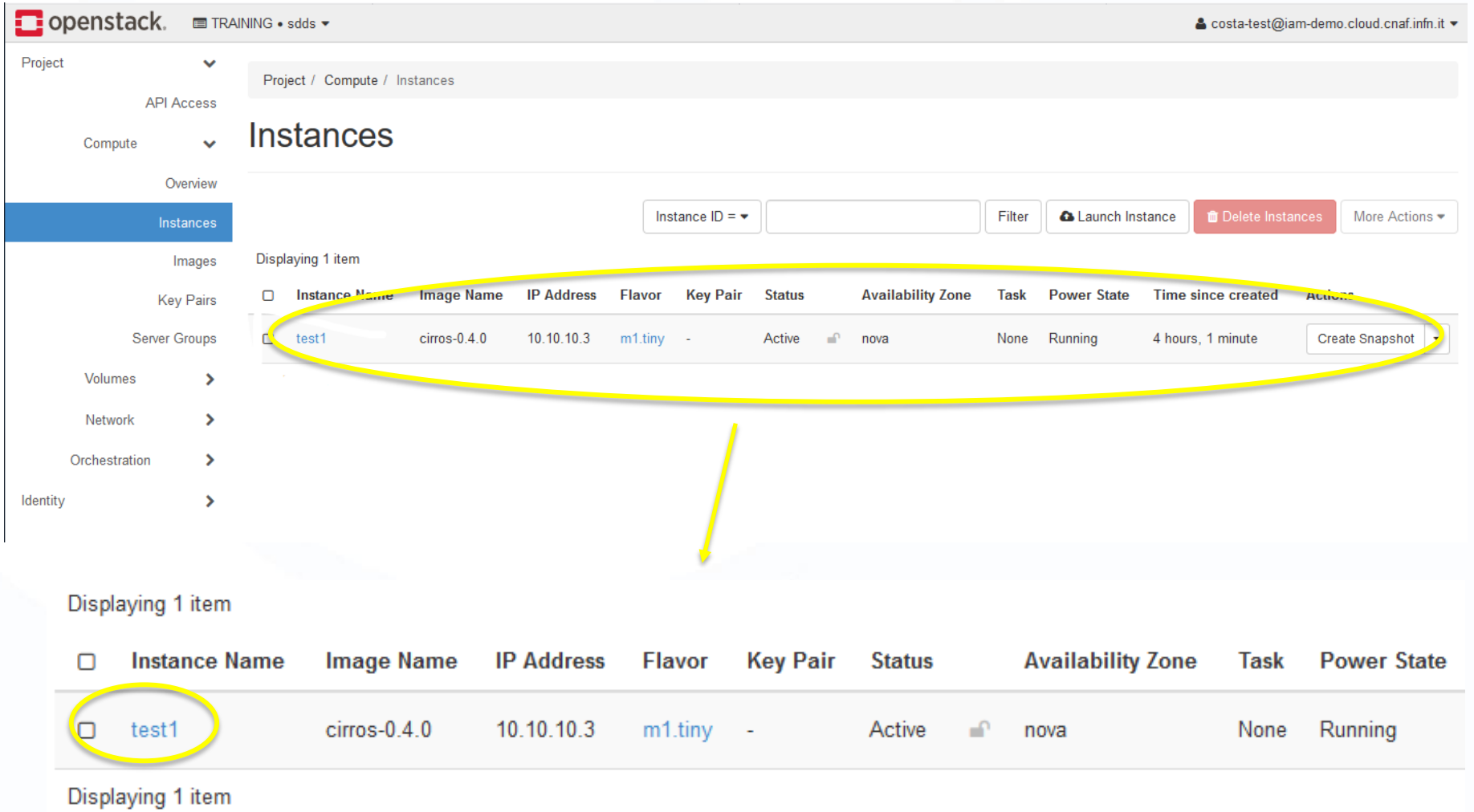
Select Network (net1)



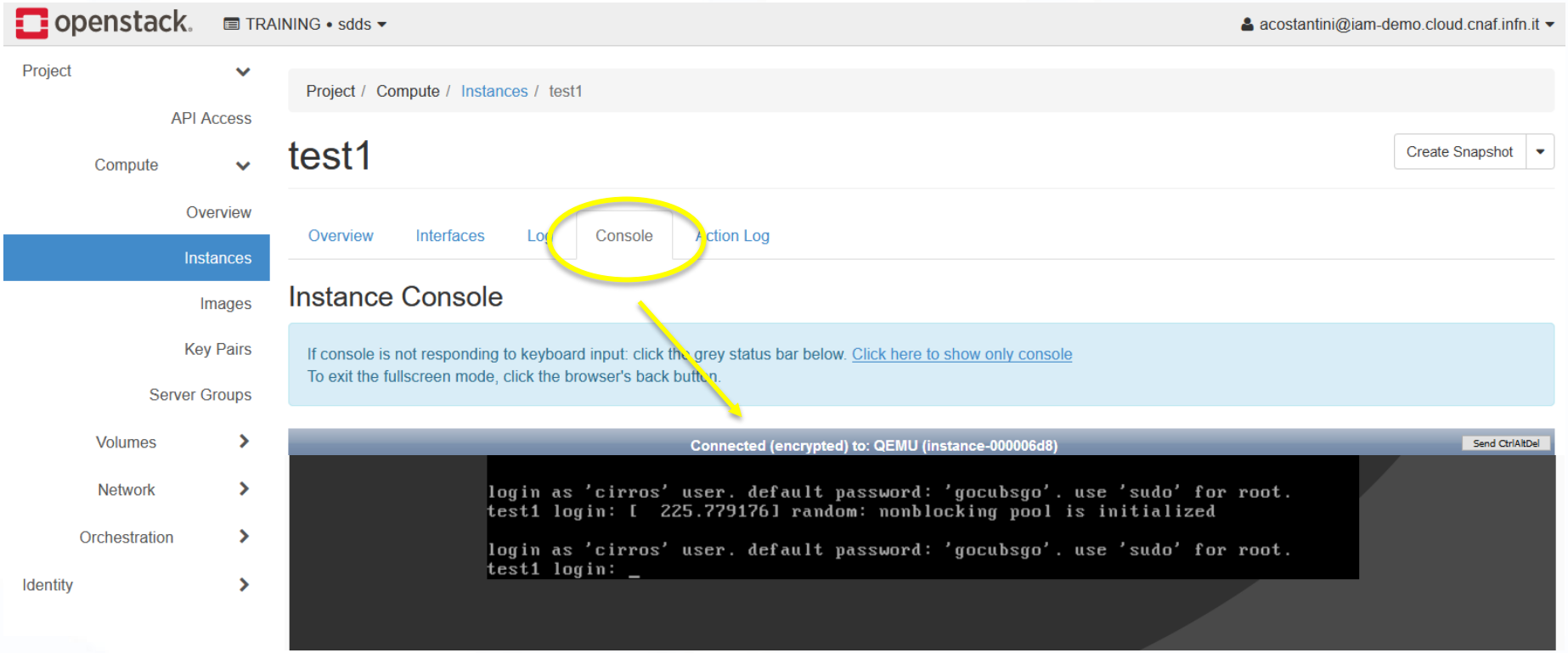
Launch Instance



Explore your Instance



Explore your Instance



Login
User: cirros
Password: gocubsgo

Jupyter Notebooks

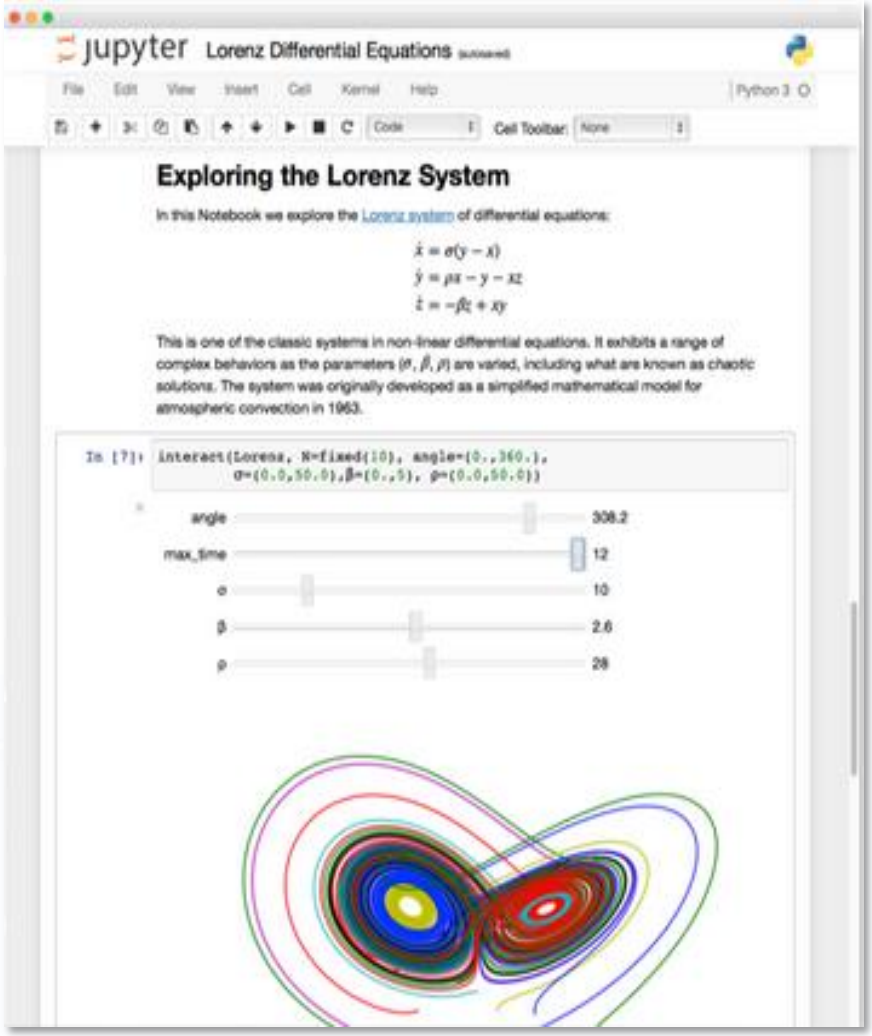
Jupyter as a Service in the Cloud@CNAF

Presentation

Hands-on

The Jupyter Notebook in a nutshell

- Non-profit, open-source, interactive platform for Data Science born out of the iPython project in 2014
- Released under the BSD license
- Notebooks can be shared with others using email, Dropbox, GitHub
- Interactive widgets



software carpentry

DATA CARPENTRY

MAKING DATA SCIENCE MORE EFFICIENT

Some key features

Language of choice

The Notebook has support for over **40** programming languages, including Python, R, Julia and Scala

Share notebooks

Notebooks can be shared with others using email, Dropbox, GitHub and the Jupyter Notebook Viewer

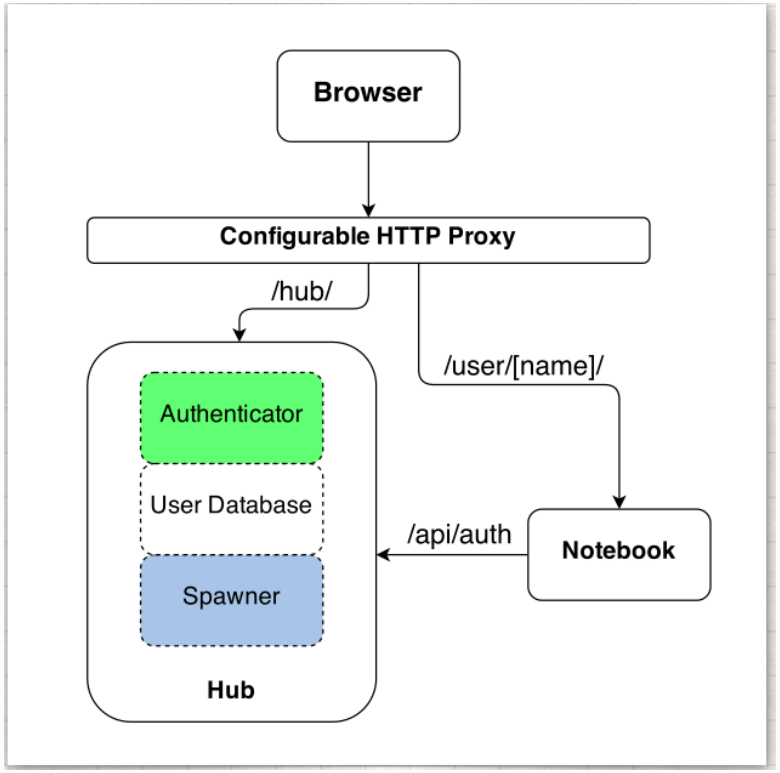
Interactive output

Your code can produce interactive output: HTML, images, videos, LaTeX, and custom MIME types

Big data integration

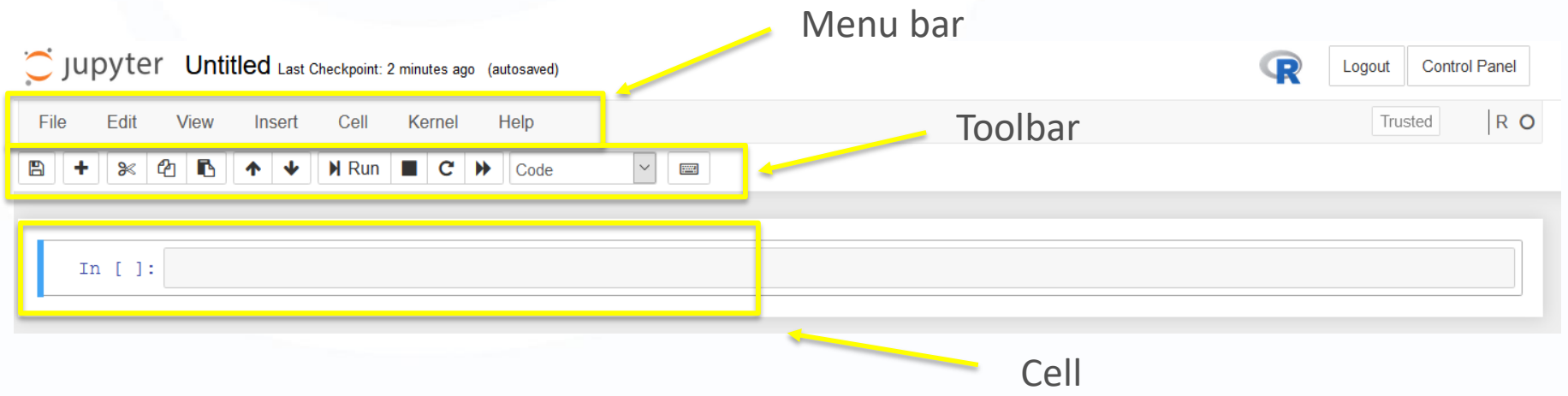
Leverage big data tools, such as Apache Spark for Python, R and Scala.

- Jupyter is single user by design
- JupyterHub is a multi-user version of notebook designed for companies, classrooms and research labs
- JupyterHub capabilities:
  - Manages Authentication
  - Spawns single-users notebooks servers on-demand
  - Gives each user a complete Jupyter server



The Jupyter notebook control panel

- **Menu bar**: The menu bar presents different options that may be used to manipulate the way the notebook functions.
- **Toolbar**: The tool bar gives a quick way of performing the most-used operations within the notebook, by clicking on an icon.
- **Cell**: the notebook cell



Structure of a notebook

- The notebook consists of a sequence of cells.
  - A cell is a **multiline text input field**
  - The execution behaviour of a cell is determined by the cell's type.

- There are three types of cells: **Code**, **Markdown**, and **Raw** cells.

Code cells allow you to edit and write new code, with full syntax highlighting and tab completion.
The programming language you use depends on the kernel

Markdown cells allow to alternate descriptive text with code

Raw cells provide a place in which you can write output directly. Raw cells are not evaluated by the notebook

Links

- JupyterHub tutorial
  - https://github.com/jupyterhub/jupyterhub-tutorial
- Documentation of JupyterHub | PDF (latest)
  - https://jupyterhub.readthedocs.io/en/latest/
- Documentation of JupyterHub's REST API
  - http://petstore.swagger.io/?url=https://raw.githubusercontent.com/jupyter/jupyterhub/master/docs/rest-api.yml
- Project Jupyter website
  - https://jupyter.org/

Cloud@CNAF Jupyter Notebook

CODATA-RDA 2019

Lab session

- Download datasets (e.g. temperature) from the Climate Change Knowledge Portal and calculate and plot the average monthly temperature

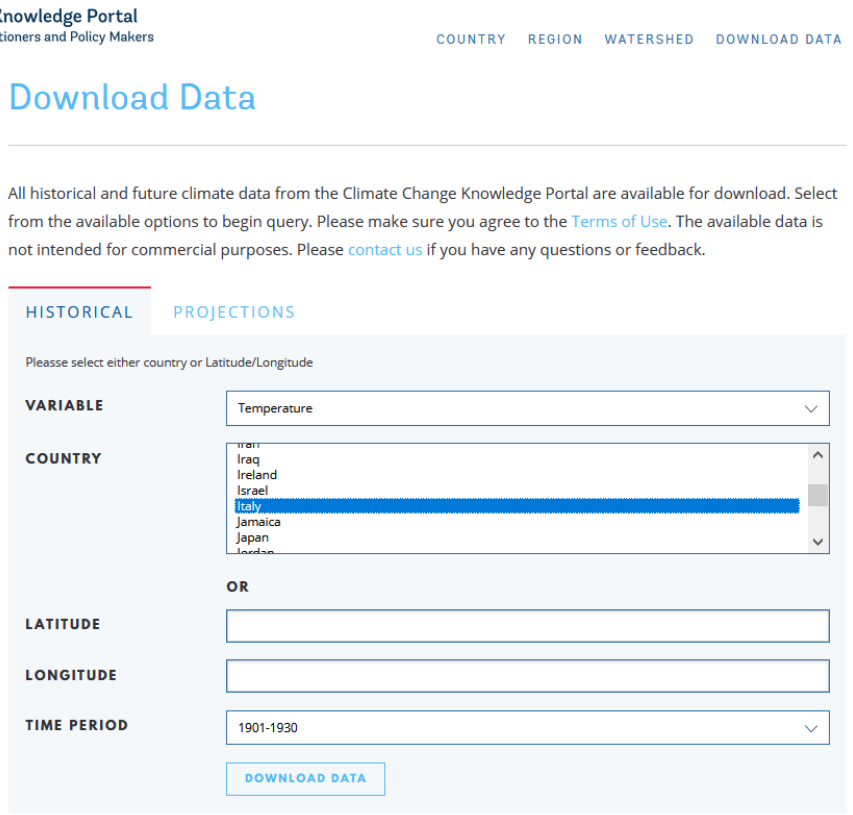
The access to the resources will be valid for the duration of the course

Download dataset from the Climate Change Knowledge portal

1. Visit:
   - https://climateknowledgeportal.worldbank.org/download-data
2. Select ‘Temperature’, the country and the time period you are interested
3. Click ‘Download Data’, and save the CSV file on your computer as **Temperatures.csv**

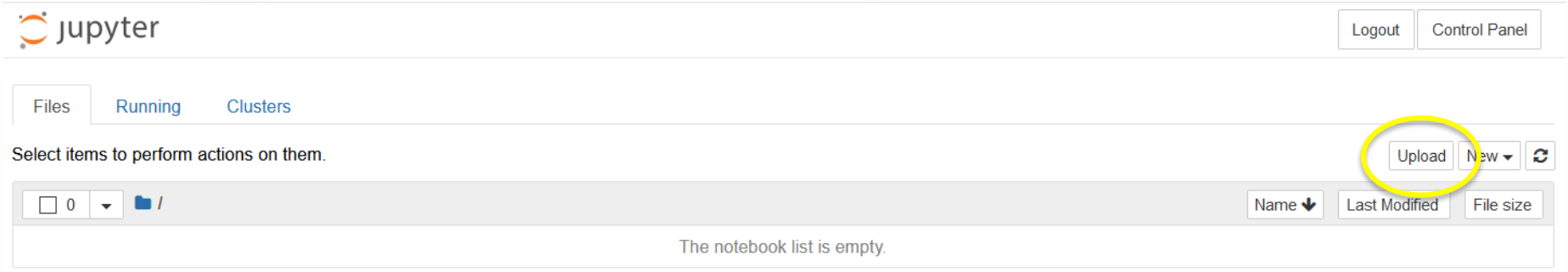
Temperatures

	A	B	C	D	E
1	Temperature - (Celsius), Year, Statistics, Country, ISO3				
2	1.54109, 1901, Jan Average, Italy, ITA				
3	0.35607, 1901, Feb Average, Italy, ITA				
4	6.06705, 1901, Mar Average, Italy, ITA				
5	10.2529, 1901, Apr Average, Italy, ITA				
6	13.4703, 1901, May Average, Italy, ITA				
7	18.587, 1901, Jun Average, Italy, ITA				
8	20.4537, 1901, Jul Average, Italy, ITA				
9	20.0018, 1901, Aug Average, Italy, ITA				
10	17.2653, 1901, Sep Average, Italy, ITA				



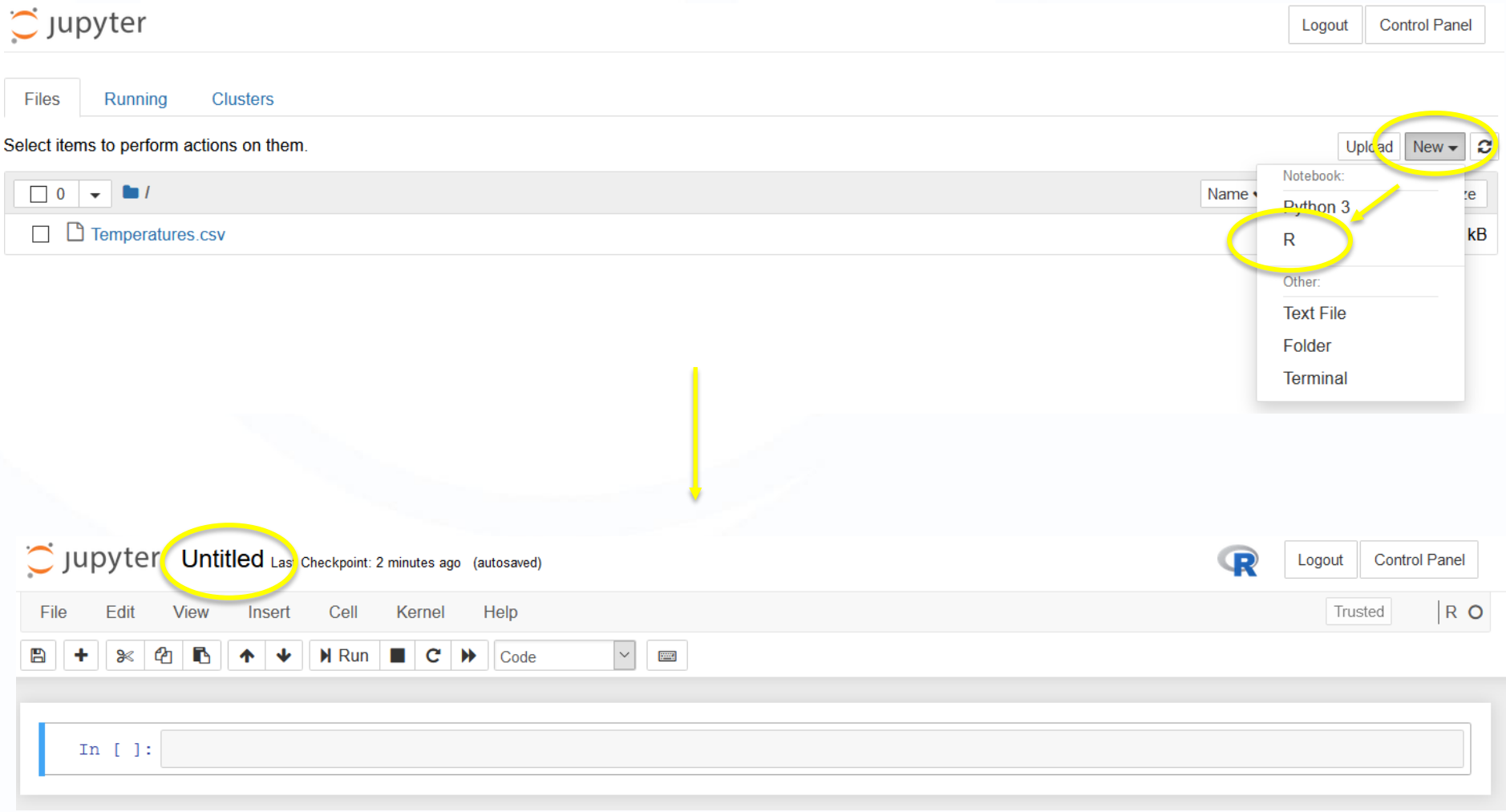
Connect to Cloud@CNAF Jupyterhub

4. Go to the training instance of the Clod@CNAF Jupyter Notebook service:
   - https://notebooks.cloud.cnaf.infn.it:8888
5. Click on 'Sign in with iam-demo' and use your account
6. Wait for your Jupyter server to boot up
7. Upload **Temperatures.csv**



Work with Notebooks

8. Open a new R Notebook and save it under a new name



Works with Notebooks

9. Use the gdata() library and the read.csv() method to read the CSV file and press “Run”

```
In [ ]: library(gdata)
        temp <- read.csv(file="Temperatures.csv", header=TRUE, sep=",")
```

Don’t worry about the warnings ☺

10. Use the head() method to display the first few rows of the imported dataset (press “Run”):

```
In [2]: head(temp)
```

A data.frame: 6 x 5

Temperature....Celsius.	Year	Statistics	Country	ISO3
<dbl>	<int>	<fct>	<fct>	<fct>
1.54109	1901	Jan Average	Italy	ITA
0.35607	1901	Feb Average	Italy	ITA
6.06705	1901	Mar Average	Italy	ITA
10.25290	1901	Apr Average	Italy	ITA
13.47030	1901	May Average	Italy	ITA
18.58700	1901	Jun Average	Italy	ITA

Works with Notebooks

11. use the aggregate() function to group temperatures per Year, and to calculate the mean for each year (press “Run”):

```
In [ ]: datasets = aggregate(temp[, 1:2], list(temp$Year), mean)
```

12. Print the average mean temperature per year (press “Run”):

```
In [ ]: print(datasets)
```

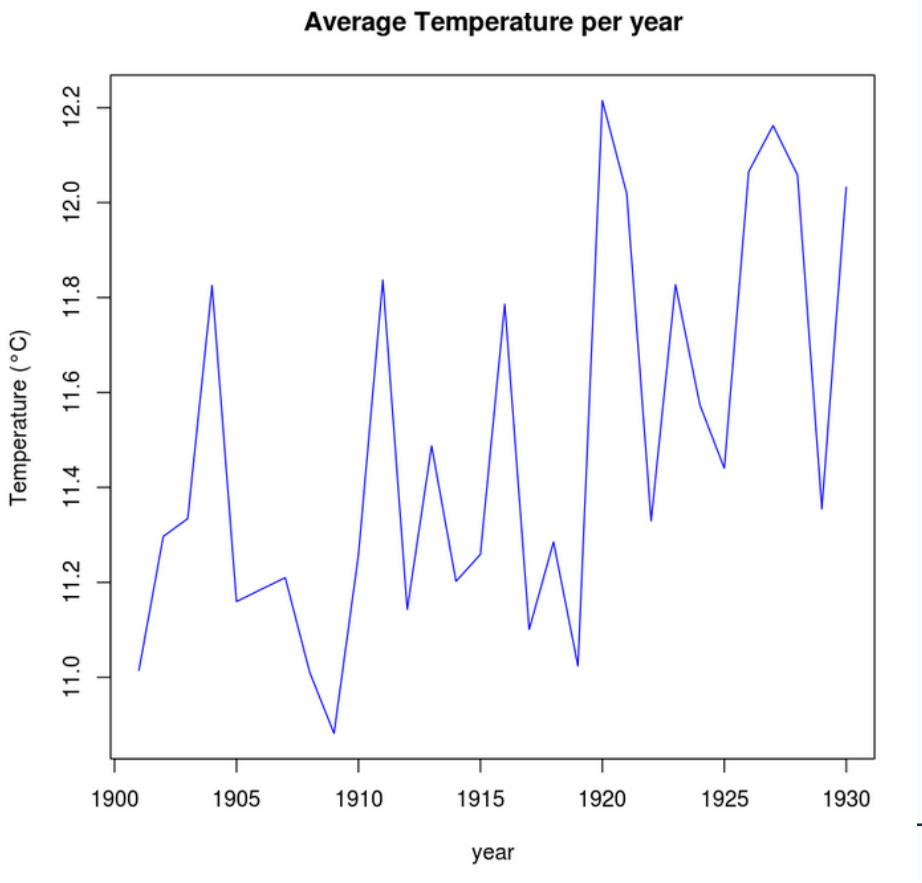
```
In [4]: print(datasets)
   Group.1 Temperature....Celsius. Year
1     1901                 11.01464 1901
2     1902                 11.29684 1902
3     1903                 11.33422 1903
4     1904                 11.82562 1904
5     1905                 11.15955 1905
6     1906                 11.18498 1906
7     1907                 11.20992 1907
8     1908                 11.01100 1908
9     1909                 10.88168 1909
10    1910                 11.25716 1910
```

Works with Notebooks

13. Drop the duplicate column (Year) and plot results (press “Run”):

```
In [ ]: plot (datasets[-3], type="l", col="blue", main="Average Temperature per year", xlab="year", ylab="Temperature (°C)")
```

It is a lower case “L”



Cloud@CNAF Jupyter Notebook

CODATA-RDA 2019

Lab session

- Add the second datasets (e.g. rainfall) and calculate and plot the average monthly rainfall

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

The future of research compute infrastructructures in Europe

The European Open Science Cloud (EOSC)

EOSC - Problem statement

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…"

What happened since then…

- EC activities:
  - EOSC summits (2017, 2018)
  - 2nd EOSC HLEG (draft report bit.ly/2OSJk0b)
  - EOSC implementation Roadmap (April 2018) (bit.ly/2KH56i6)
    - 6 action lines (Archi., Data, Services, Access & Interf., Rules, Governance)
  - Consultation with member states
- H2020 projects:
  - eInfraCentral (2017-19)
  - **EOSCpilot (2017-19)** https://www.eoscpilot.eu
  - **EOSC-hub (2018-2020)** https://www.eosc-hub.eu
  - **eXtremeDataCloud (2018-2020)** www.extreme-datacloud.eu
  - **DEEP Hybrid-DataCloud (2018-2020)** https://deep-hybrid-datacloud.eu
  - **…**

EOSCpilot: High Level Aims

The *EOSCpilot* project supports the first phase in the development of the EOSC. It will between 2017-2019

- **Establish the governance framework** for the EOSC and contribute to the development of European open science policy and best practice;

Scientific demonstrators

- **Develop a number of demonstrators** functioning as high-profile pilots that integrate services and infrastructures to show interoperability and its benefits in a number of scientific domains;

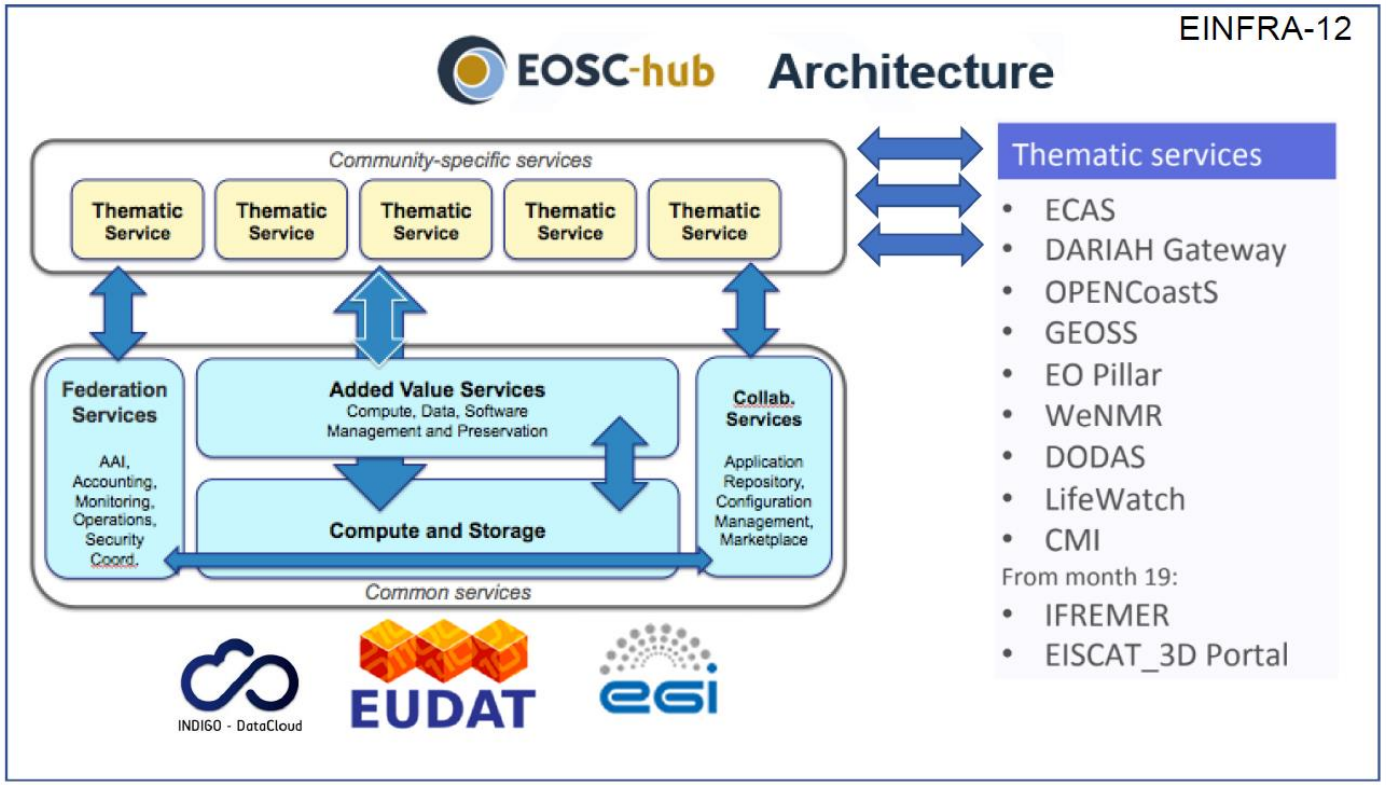
5x3 demonstrators
1FTE for 1 year / demonstrator
One 'shepherd' / demonstrator

- **Engage with a broad range of stakeholders**, crossing borders and communities, to build the trust and skills required for adoption of an open approach to scientific research.

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

EINFRA-21

Hybrid DataCloud

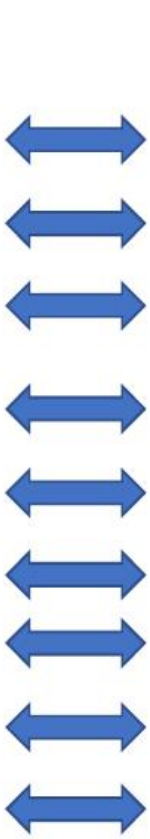
eXtreme DataCloud

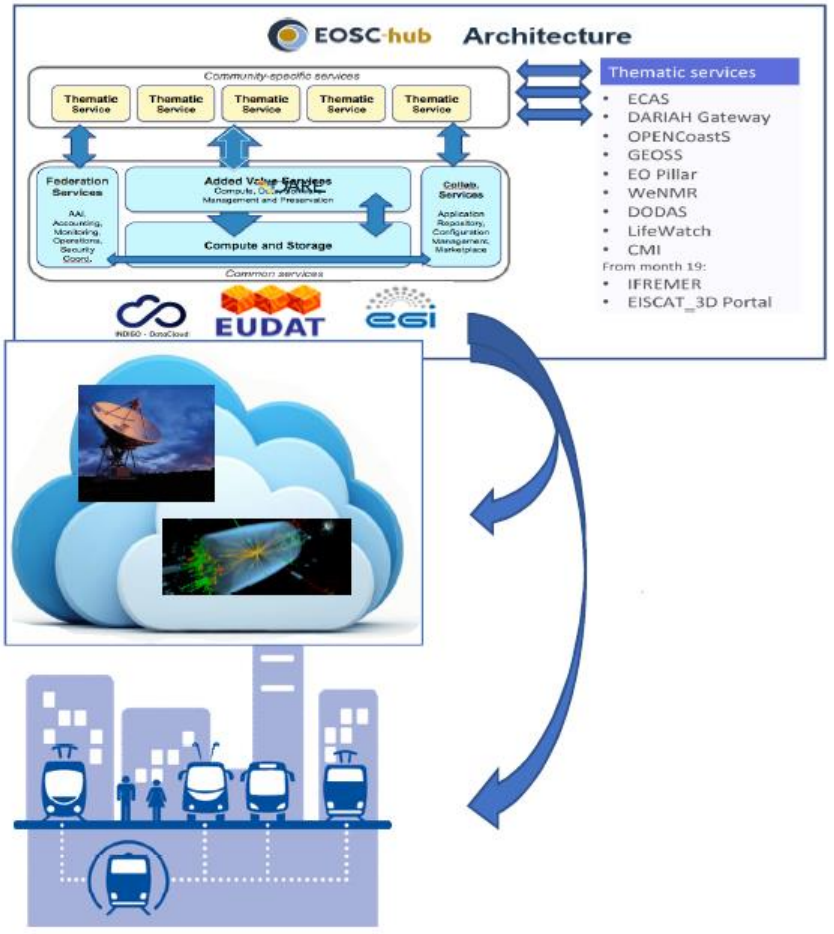
DARE

EUXDAT

PROCESS

TROPOMI





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:

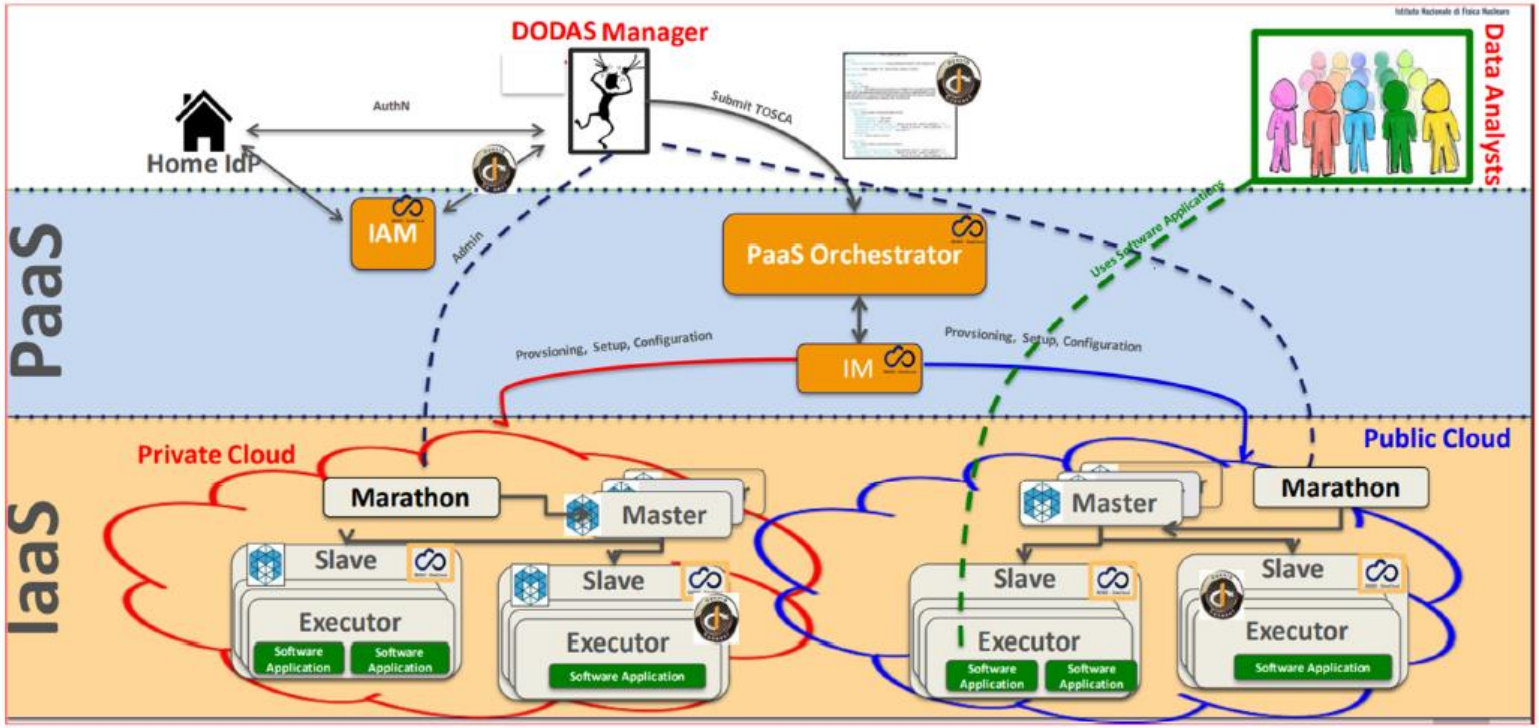
- IFREMER
- EISCAT_3D Portal

+ new service from outside the consortium

DODAS in a nutshell

DODAS: Dynamic On-Demand Analysis Service

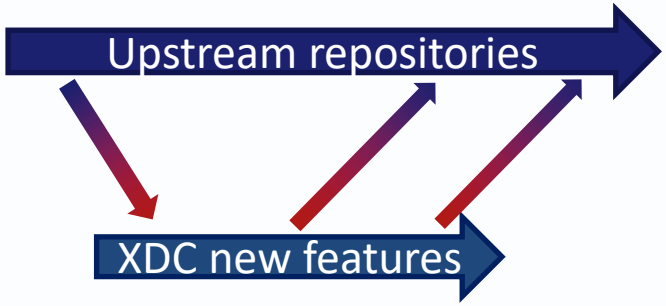
- A open source deployment manager
- **Allows on-demand creation and configuration of container based clusters** for data processing with **almost zero effort (HTCONDOR deployment)**
- **Support for hybrid clouds deployment**
- Based on “industry standards” to minimize code development and maintenance



eXtreme DataCloud

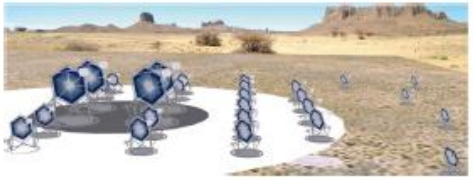
- The eXtreme DataCloud is a **software development** and integration project
- **Develops scalable technologies for federating storage resources** and **managing data** in **highly distributed** computing environments
  - Focus on efficient, policy driven and Quality of Service based DM
- The targeted 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



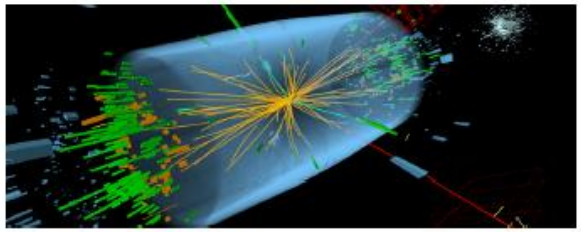
XDC: A User Driven Project

cherenkov telescope array



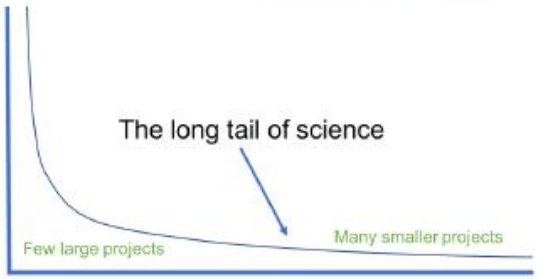
European XFEL

WLCG
Worldwide LHC Computing Grid



LifeWatch ERIC

ECRIN
EUROPEAN CLINICAL RESEARCH INFRASTRUCTURE NETWORK



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:

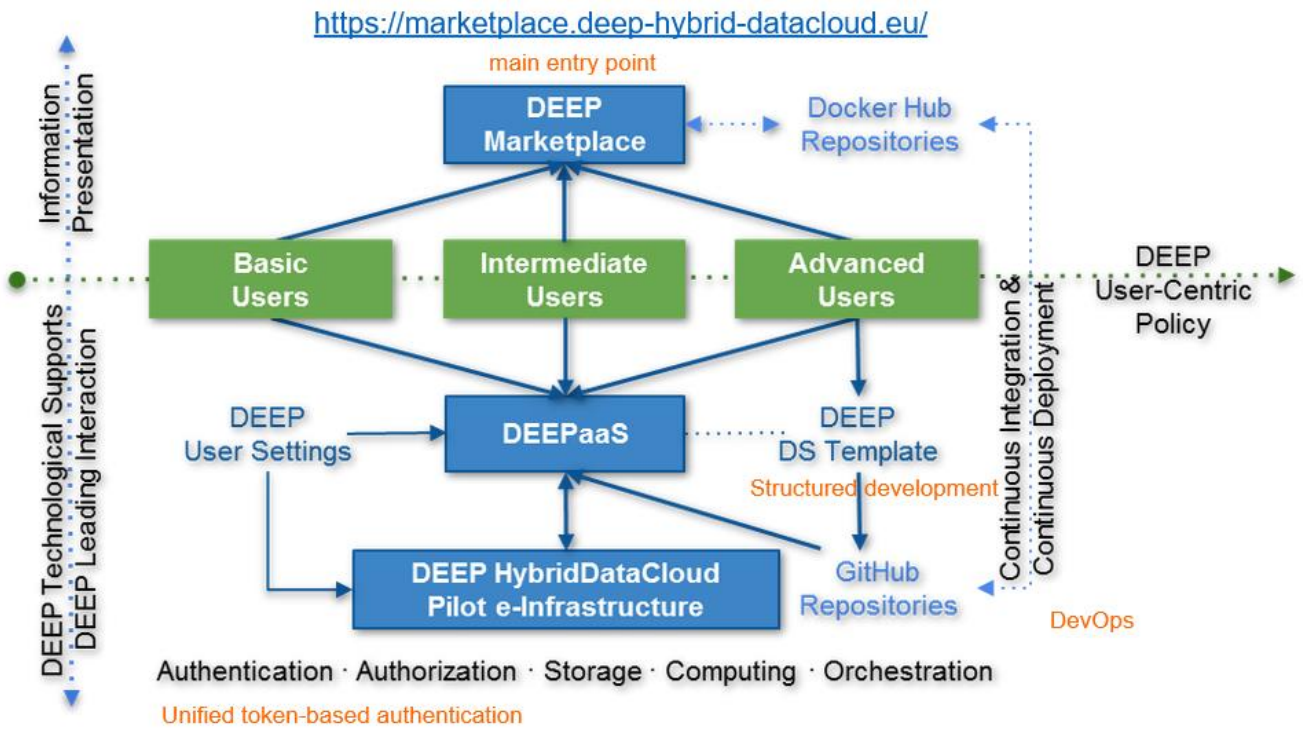
-- Retinopathy

Computing Security:

-- Massive Online Data Streams

Physics:

-- Post-processing



What's next?

- 2019:
  - DEEP-HDC All Hands Meeting
    - Poznan, Poland, 11-13 September
  - IBEGRID 2019/EOSC Synergy KOM
    - Santiago de Compostela, Spain, 23-26 September
  - Workshop on Data Management Solutions for the European Research Communities: The eXtreme-DataCloud Project
    - Helsinki, Finland 22nd October
- 2019-20:
  - Eosc-hub Week 2020
    - Karlsruhe, Germany, 18-10 May 2020
  - Focus on engaging with external communities
    - New INFRAEOSC-* projects: Research Infrastructure clusters; Commercial procurement; Governance; National services; etc.

Aknowledgements and Material

@INFN

Cristina Duma, Davide Salomoni, Diego Michelotto, Daniele Cesini, Andrea Ceccanti, Daniele Spiga

@EGI

Gergely Sypos, Giuseppe La Rocca

EOSC-hub

eosc-hub.eu @EOSC_eu

Thank you for your attention!

Questions?

Contact:
www.eosc-hub.eu
Alessandro.Costantini@cnaf.infn.it

EOSC-hub

EXTRA SLIDES

EOSC-hub

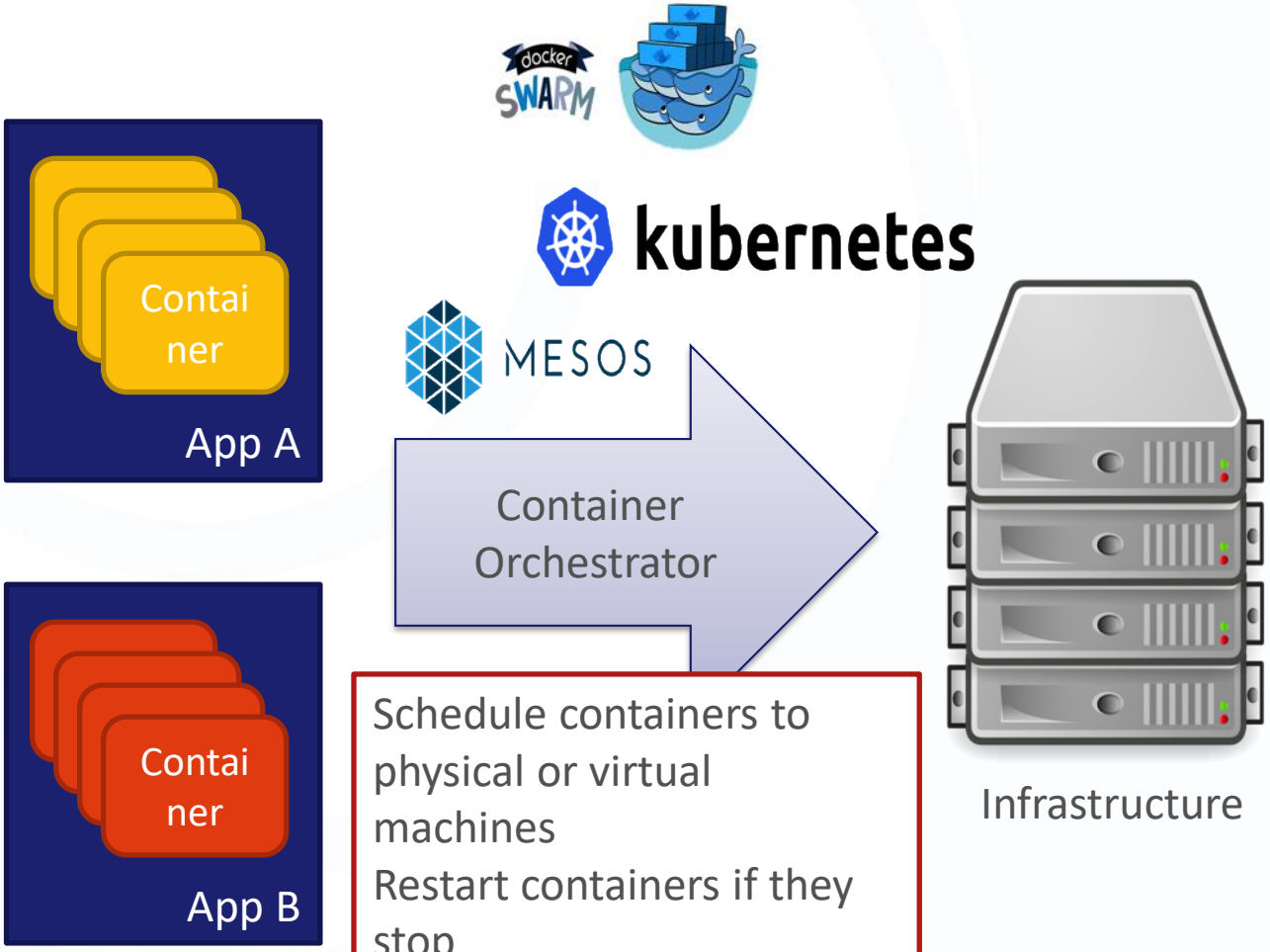
eosc-hub.eu @EOSC_eu

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

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.