Architettura di INFN Cloud

Giacinto DONVITO

INFN-Bari

Agenda

- Concetti di cloud computing
- Concetti di cloud federate
- Un po’ di storia
- Il progetto INFN Cloud e le funzionalità
- Architettura di INFN Cloud
  - Proprietà
  - Pro & Cons

Definizione

- La definizione classica di riferimento è quella del National Institute of Standards and Technology (NIST) USA (http://goo.gl/eBGBk)

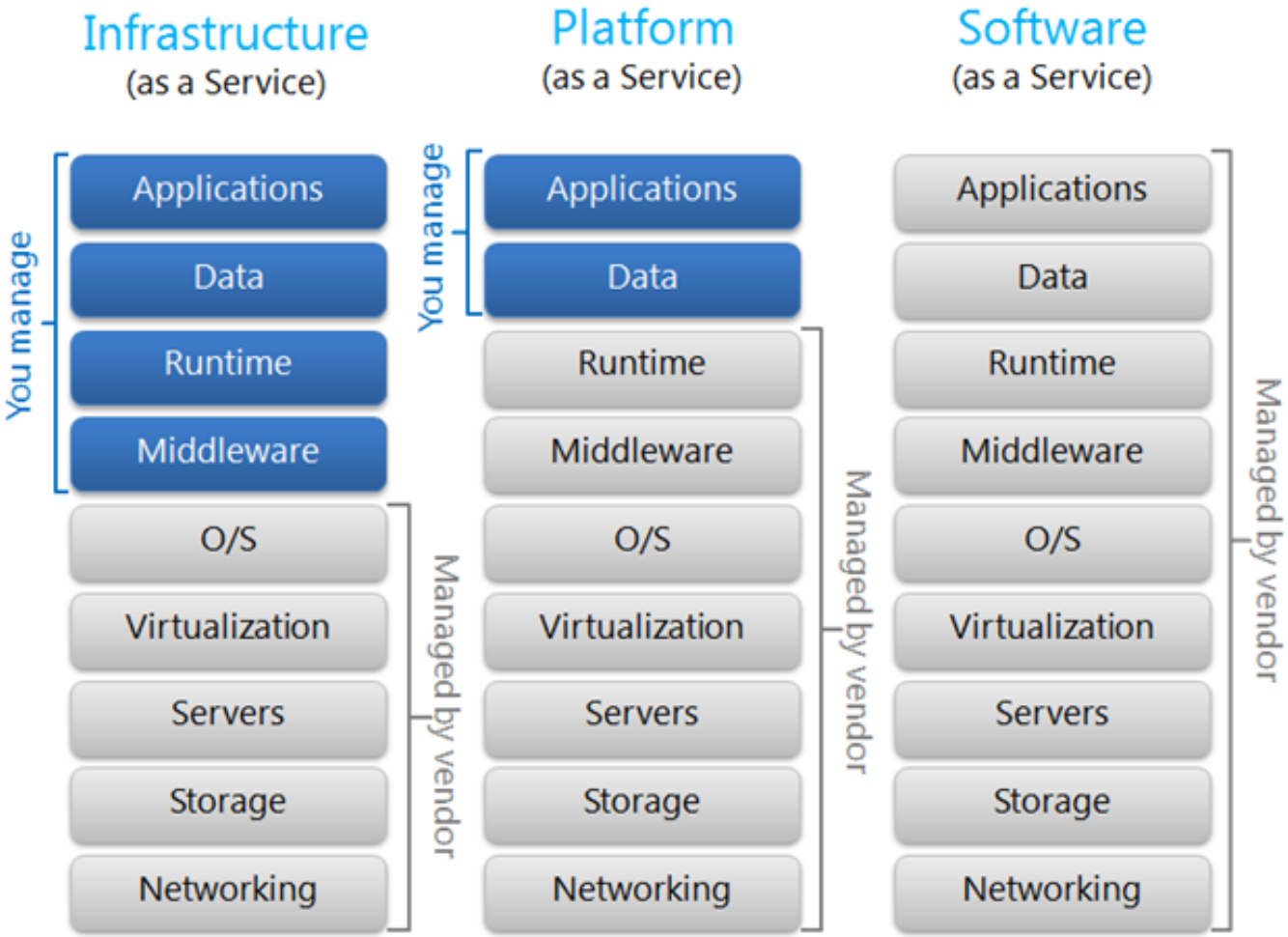
- In sintesi il Cloud computing si occupa di:

Fornitura di tecnologia di informazione e comunicazione (ICT) come servizio

CLOUD COMPUTING?

- **On-demand self-service.**
  - A consumer can unilaterally provision computing capabilities, such as server time and network storage, as needed automatically without requiring human interaction with each service provider.
- **Broad network access.**
  - Capabilities are available over the network and accessed through standard mechanisms that promote use by heterogeneous thin or thick client.
- **Resource pooling.**
  - Computing resources are pooled to serve multiple consumers using a multi-tenant model, with different physical and virtual resources dynamically assigned and reassigned according to consumer demand.
- **Rapid elasticity.**
  - Capabilities can be elastically provisioned and released, in some cases automatically, to scale rapidly outward and inward commensurate with demand.
- **Measured service.**
  - Cloud systems automatically control and optimize resource use by leveraging a metering capability at a level of abstraction appropriate to the type of service.

Chi fa cosa?


Infrastructure
(as a Service)
Platform
(as a Service)
Software
(as a Service)
You manage
Applications
Data
Runtime
Middleware
O/S
Virtualization
Servers
Storage
Networking
Managed by vendor

SaaS:
Software as a Service

What is Software as a Service? (SaaS)

- SaaS is a software delivery methodology that provides licensed multi-tenant access to software and its functions remotely as a Web-based service.
  - Usually billed based on usage
  - Usually multi tenant environment
  - Highly scalable architecture

What is Software as a Service? (SaaS)

- Cloud application services or “Software as a Service” (SaaS) are probably the most popular form of cloud computing and are easy to use. SaaS uses the Web to deliver applications that are managed by a third-party vendor and whose interface is accessed on the clients’ side. Most SaaS applications can be run directly from a Web browser, without any downloads or installations required. SaaS eliminates the need to install and run applications on individual computers. With SaaS, it’s easy for enterprises to streamline their maintenance and support, because everything can be managed by vendors: applications, runtime, data, middleware, O/S, virtualization, servers, storage, and networking. Gmail is one famous example of an SaaS mail provider.

SaaS Examples

PaaS:

Platform as a Service

Platform as a Service (PaaS)

- PaaS provides all the facilities required to support the complete life cycle of building and delivering web applications and services entirely from the Internet.
  - Typically applications must be developed with a particular platform in mind
  - Multi tenant environments
  - Highly scalable multi tier architecture

Platform as a Service (PaaS)

- The most complex of the three, cloud platform services or “Platform as a Service” (PaaS) deliver computational resources through a platform. What developers gain with PaaS is a framework they can build upon to develop or customize applications. PaaS makes the development, testing, and deployment of applications quick, simple, and cost-effective, eliminating the need to buy the underlying layers of hardware and software. One comparison between SaaS vs. PaaS has to do with what aspects must be managed by users, rather than providers: With PaaS, vendors still manage runtime, middleware, O/S, virtualization, servers, storage, and networking, but users manage applications and data.
- PaaS provides the computing infrastructure, the hardware, and the platforms that are installed on top of the hardware. Similar to the way that you might create macros in Excel, PaaS allows you to create applications using software components that are controlled by a third-party vendor. PaaS is highly scalable , and users don’t have to worry about platform upgrades or having their site go down during maintenance. Users who benefit most from PaaS include companies who want to increase the effectiveness and interactivity of a large staff. For the needs of larger companies and independent software vendors, Apprenda is one provider of a private PaaS for .Net business-application development and deployment.

PaaS Examples


IaaS:
Infrastructure as a Service

Infrastructure as a Service (IaaS)

- IaaS is the delivery of technology infrastructure as an on demand scalable service
  - Usually billed based on usage
  - Usually multi tenant virtualized environment
  - Can be coupled with Managed Services for OS and application support

IaaS: Infrastructure as a Service

- Cloud infrastructure services, known as “Infrastructure as a Service” (IaaS), deliver computer infrastructure (such as a platform virtualization environment), storage, and networking. Instead of having to purchase software, servers, or network equipment, users can buy these as a fully outsourced service that is usually billed according to the amount of resources consumed. Basically, in exchange for a rental fee, a third party allows you to install a virtual server on their IT infrastructure. Compared to SaaS and PaaS, IaaS users are responsible for managing more: applications, data, runtime, middleware, and O/S. Vendors still manage virtualization, servers, hard drives, storage, and networking. What users gain with IaaS is infrastructure on top of which they can install any required platforms. Users are responsible for updating these if new versions are released.

Infrastructure as a Service (IaaS)

INFN CLOUD
Istituto Nazionale di Fisica Nucleare

- Advantages
  - Customized environment with “root” access
  - Easy access to scalable resources
- Disadvantages
  - Variety of APIs and interfaces
  - VM image creation is difficult and time-consuming
- Trends
  - Lots of specialized cloud providers appearing
  - Orchestration pushing into PaaS space

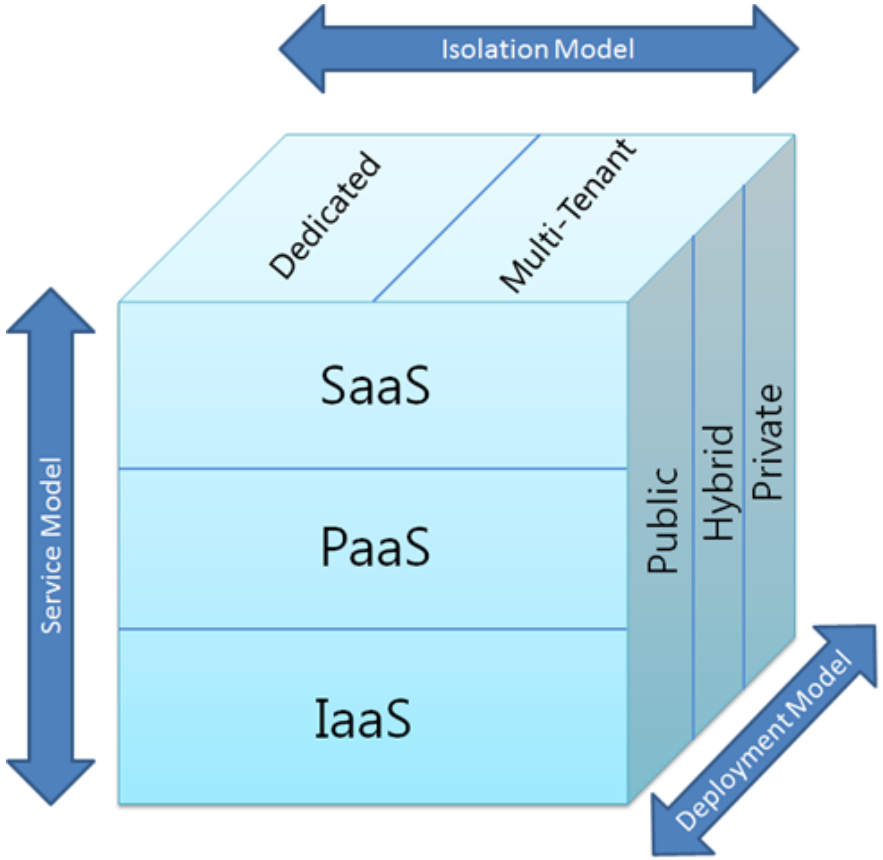
IaaS is not Managed Hosting

- Traditional managed hosting is a form of web hosting where a user chooses to lease entire server(s) housed in an off-site data center.
  - Term based contracts based on projected resource requirements

IaaS Examples

Aggiungiamo dimensioni

- Oltre i modelli di *servizio*, parti importanti per definire e capire il Cloud computing sono i modelli di:
  - *deployment* (dove distribuisco i servizi)
  - *isolamento* (come isolo i servizi)



Deployment Models

Public cloud

- *Public cloud* (off-site and remote) describes cloud computing where resources are dynamically provisioned on an on-demand, self-service basis over the Internet, via web applications/web services, open API, from a third-party provider who bills on a utility computing basis.

Private cloud

- A *private cloud* environment is often the first step for a corporation prior to adopting a public cloud initiative. Corporations have discovered the benefits of consolidating shared services on virtualized hardware deployed from a primary datacenter to serve local and remote users.

Hybrid cloud

- A *hybrid cloud* environment consists of some portion of computing resources on-site (on premise) and off-site (*public cloud)*. By integrating public cloud services, users can leverage cloud solutions for specific functions that are too costly to maintain on-premise such as virtual server disaster recovery, backups and test/development environments.

Community cloud

- A *community cloud* is formed when several organizations with similar requirements share common infrastructure. Costs are spread over fewer users than a *public cloud* but more than a single tenant.

Isolamento

- I modelli di isolamento nel Cloud (spesso ignorati) sono importanti e si dividono in:
  - Infrastrutture dedicate
  - Infrastrutture “multi-tenant” (con diversi [tipi di] clienti)
- Il tipo di isolamento è importante per molti aspetti, come:
  - Segmentazione delle risorse
  - Protezione dei dati
  - Sicurezza delle applicazioni
  - Auditing
  - Disaster recovery

Cloud Federation

Cloud Federation

- Cloud federation, which enables cloud providers and IT companies to collaborate and share their resources, is associated with many portability and interoperability issues.

- Cloud developers and researchers have proposed or implemented numerous federation architectures, including ***cloud bursting, brokering, aggregation,*** and ***multitier.***

- These architectures can be classified according to the level of coupling or interoperation among the cloud instances involved, ranging from ***loosely coupled*** (with no or little interoperability among cloud instances) to ***tightly coupled*** (with full interoperability among cloud instances).

Loosely Coupled Federation

INFN CLOUD
Istituto Nazionale di Fisica Nucleare

- This scenario is formed by independent cloud instances—for example, a private cloud complementing its infrastructure with resources from an external commercial cloud—with limited interoperation between them.

- A cloud instance has little or no control over remote resources (for example, decisions about VM placement are not allowed), monitoring information is limited (for example, only CPU, memory, or disk consumption of each VM is reported), and there is no support for advanced features such as cross-site networks or VM migration.

Partially Coupled Federation

- This scenario typically consists of various partner clouds that establish a contract or framework agreement stating the terms and conditions under which one partner cloud can use resources from another.

- This contract can enable a certain level of control over remote resources (for example, allowing the definition of affinity rules to force two or more remote VMs to be placed in the same physical cluster); can agree to the interchange of more detailed monitoring information (for example, providing information about the host where the VM is located, energy consumption, and so on); and can enable some advanced networking features among partner clouds (for example, the creation of virtual networks across site boundaries).

Tightly Coupled Federation

INFN CLOUD
Istituto Nazionale di Fisica Nucleare

- This scenario usually includes clouds belonging to the same organization and is normally governed by the same cloud OS type.

- In this scenario, a cloud instance can have advanced control over remote resources—for example, allowing decisions about the exact placement of a remote VM—and can access all the monitoring information available about remote resources.

- In addition, it can allow other advanced features, including the creation of cross-site networks, cross-site migration of VMs, implementation of high availability techniques among remote cloud instances, and creation of virtual storage systems across site boundaries.

Cloud Federation Architectures

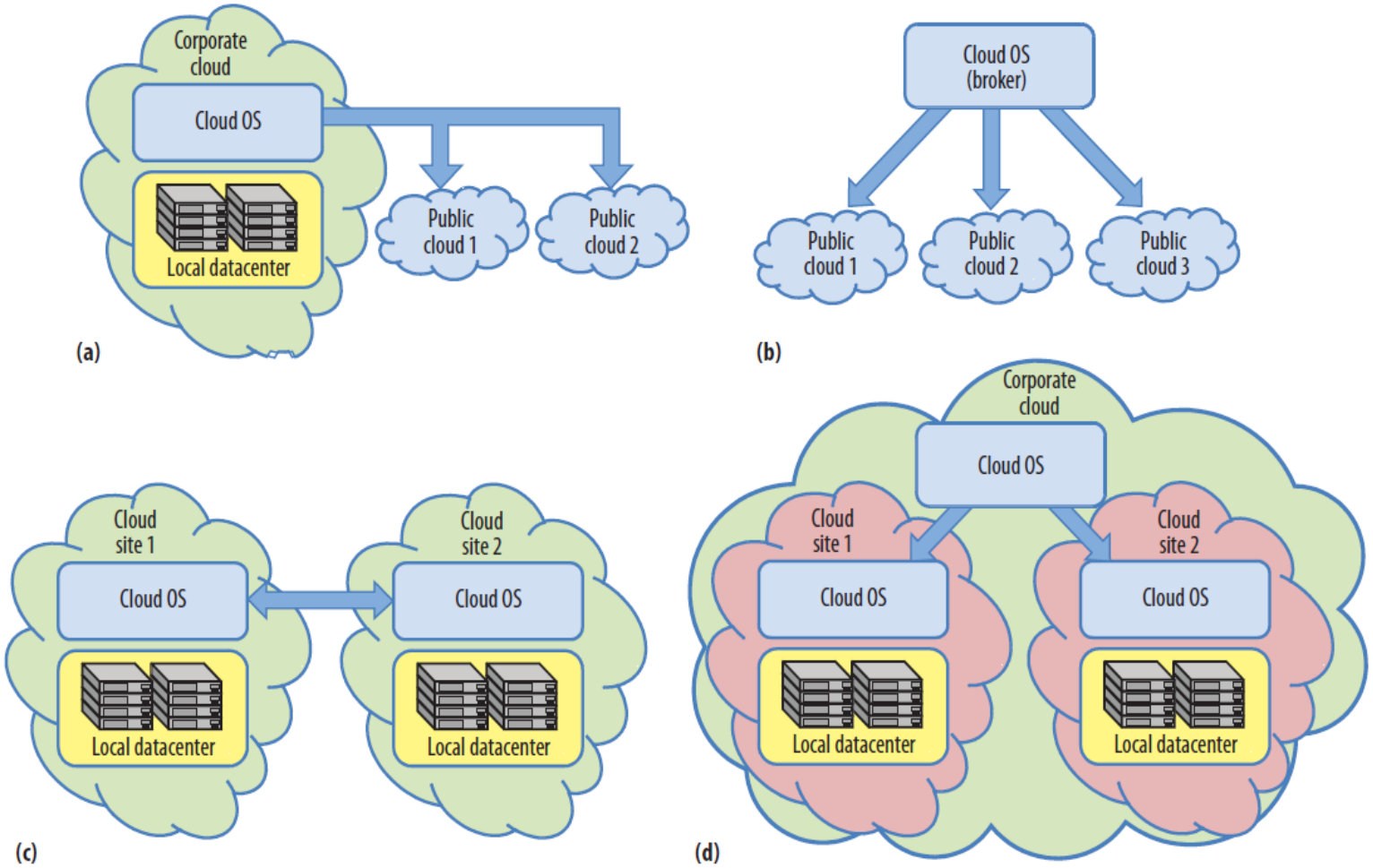


Figure 2. Cloud federation architectures: (a) bursting (hybrid), (b) broker, (c) aggregated, and (d) multitier.

Bursting (Hybrid) Architecture

INFN CLOUD
Istituto Nazionale di Fisica Nucleare

- Cloud bursting or hybrid architecture combines the existing on-premise infrastructure (usually a private cloud) with remote resources from one or more public clouds to provide extra capacity to satisfy peak demand periods.

- Because the local cloud OS has no advanced control over the virtual resources deployed in external clouds beyond the basic operations the providers allow, this architecture is loosely coupled. Most existing open cloud managers support the hybrid cloud architecture and is used in infrastructures such as StratusLab (http://stratuslab.eu).

Broker Architecture

- The central component of the broker architecture is a broker that serves various users and has access to several public cloud infrastructures. A simple broker should be able to deploy virtual resources in the cloud as selected by the user.

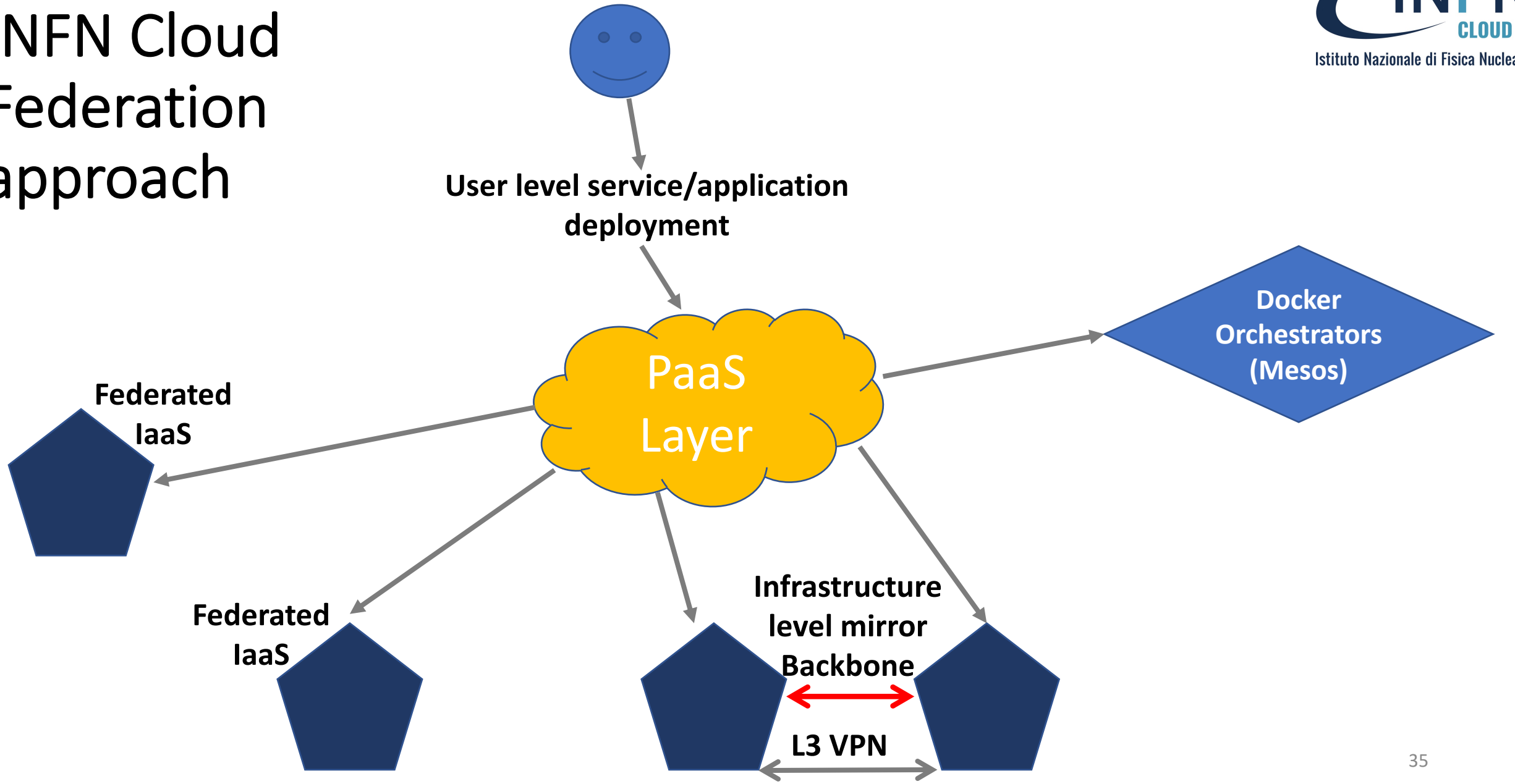
- An advanced broker offering service management capabilities could make scheduling decisions based on optimization criteria such as cost, performance, or energy consumption to automatically deploy virtual user service in the most suitable cloud, or it could even distribute the service components across multiple clouds. This architecture is also loosely coupled since public clouds typically do not allow advanced control over the deployed virtual resources.

- Brokering is the most common federation scenario. Examples include BonFIRE (www.bonfire-project.eu), Open Cirrus, and FutureGrid (http://futuregrid.org).

INFN Cloud

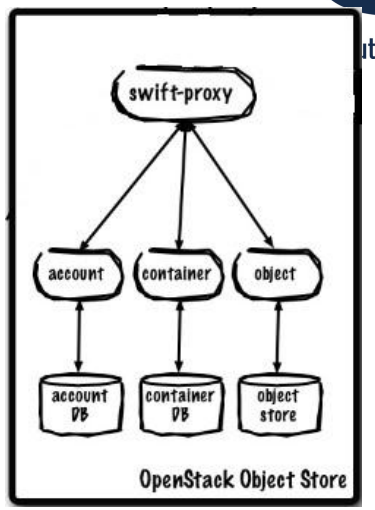
- È un mix (la somma) di due approcci diversi:
  - Tightly Coupled Federation
    - Lo chiameremo INFN-Backbone
  - Loosely Coupled Federation
    - Cloud Federate via Layer PaaS

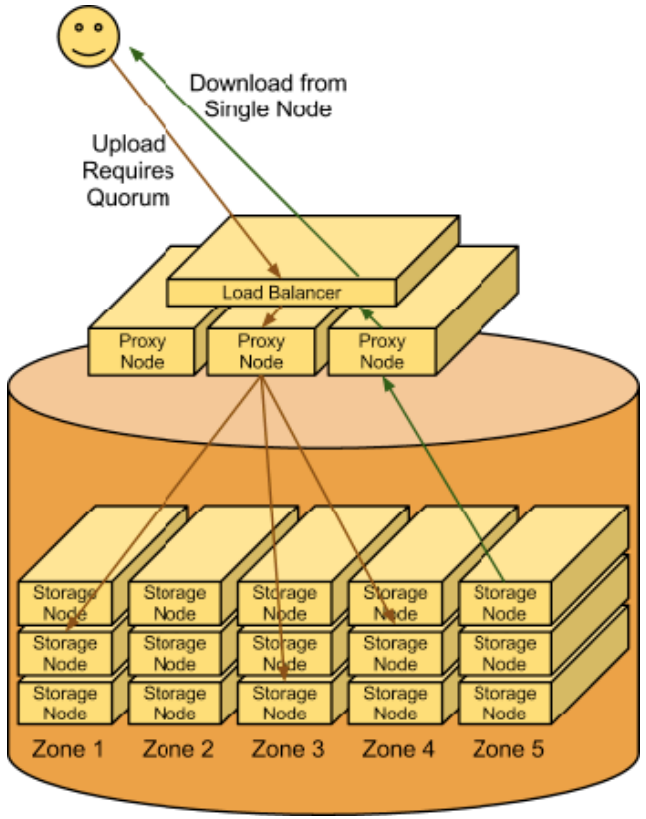
INFN Cloud Federation approach

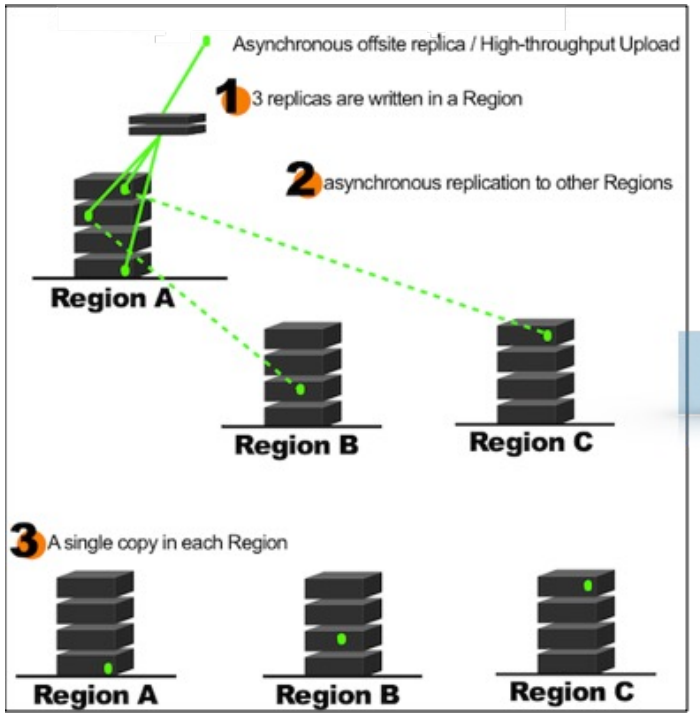


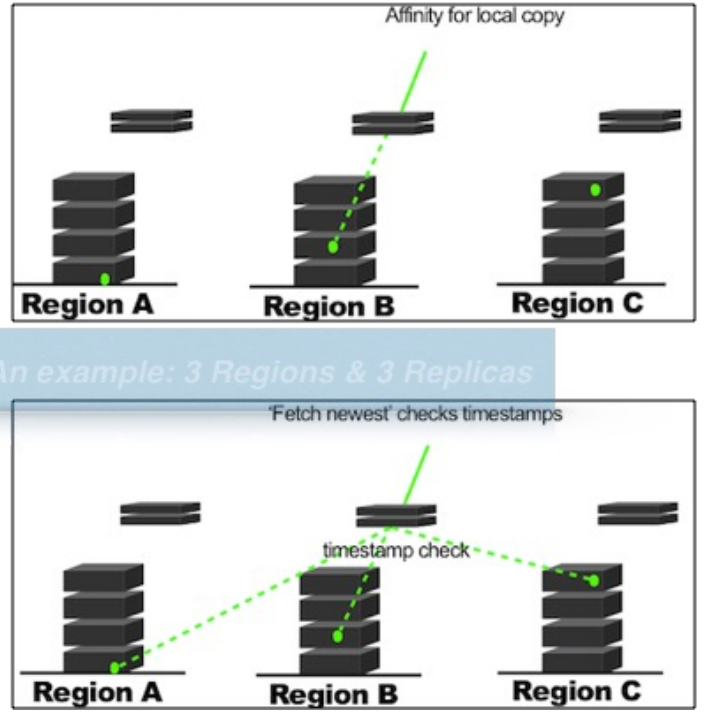
INFN Cloud Federation approach

Swift highlight









An example: 3 Regions & 3 Replicas

INFN Cloud: modello di Federazione

- La federazione «Loosely Coupled» permette di:
  - Avere una buona indipendenza nei singoli siti per quanto riguarda la gestione di hw e sw stack di cloud
    - Siamo riusciti a federare non solo OpenStack, ma anche OpenNebula, AWS, Azure, etc insieme a Mesos, etc.
  - È sufficiente individuare attentamente le APIs e le configurazioni supportate (e richieste) dal layer di federazione

- È però necessario fare attenzione alle seguenti criticità (gestite a livello di PaaS e anche di IaaS):
  - La presenza di una buona gestione delle SLAs
  - Un buon monitoring della risorse da federare

INFN Cloud: modello di Federazione

- La federazione «Tightly Coupled» permette di:
  - Avere una replica di dati e servizi a livello di infrastruttura senza richiedere lavoro a livello applicativo
  - Non si aggiungono layer aggiuntivi richiesti per il funzionamento
  - Le operazioni possono essere gestite in modo distribuito fra i due team di lavoro

- È però necessario fare attenzione alle seguenti criticità (gestite a livello di PaaS e anche di IaaS):
  - I siti devono avere un elevato livello di trust
  - È necessario che il sw stack sia perfettamente allineato
  - I team che si occupano delle risorse devono avere un buon coordinamento

INFN Cloud: Modello di PaaS

- SDK-like: PaaS come un toolkit in cui sviluppare codice in una serie di linguaggi supportati dal cloud provider

- automation-like: PaaS con un set di servizi già pronti che possono essere usati:
  - Per sviluppare applicazioni
  - Per deployare servizi
  - Per automatizzare il deployment di servizi e applicazioni

INFN Cloud: Caratteristiche dei modelli di PaaS

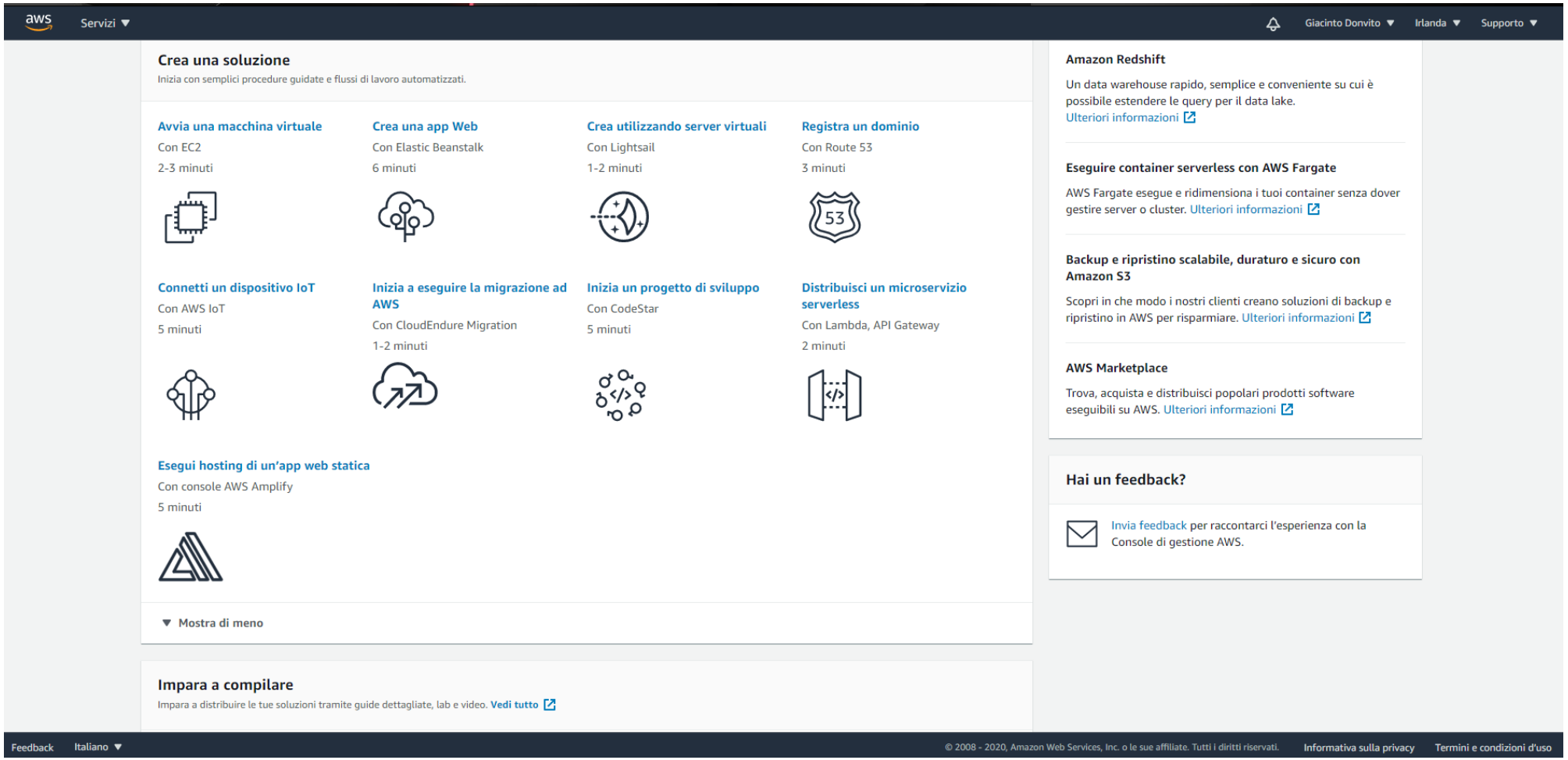
- SDK-like:
  - PRO:
    - Se il linguaggio è quello dell'applicazione di interesse, il porting è facile è la piattaforma pensa a tutto
  - Contro:
    - Spesso sono funzionalità/librerie proprietarie da usare per ottenere il massimo dalla piattaforma.
    - Se l'applicazione usasse un linguaggio diverso da quello supportato non esiste modo di fare il porting, l'applicazione va riscritta.

- automation-like:
  - PRO:
    - È molto semplice aggiungere linguaggi framework non già supportati dalla piattaforma
    - È possibile orchestrare servizi già disponibili senza bisogno di pensate code-rafactoring.
    - È semplice fare il porting su altri provider cloud, perché non si usano librerie o funzionalità specifiche
  - Contro:
    - Meno trasparente per dell'altro approccio: richiedere uno studio preventivo dello use case per trovare l'implementazione migliore per il determinato problema

INFN Cloud: Caratteristiche dei modelli di PaaS

- INFN Cloud ha adottato un approccio a orchestrazione/automazione
- I servizi sono definiti da «TEMPLATE TOSCA»
  - Che descrivono:
    - i componenti
    - Le relazioni fra i componenti
    - Come installare/configurare i componenti

INFN Cloud: «Marketplace»



Se ASW ha la sua dashboard…

cosa abbiamo in INFN Cloud??

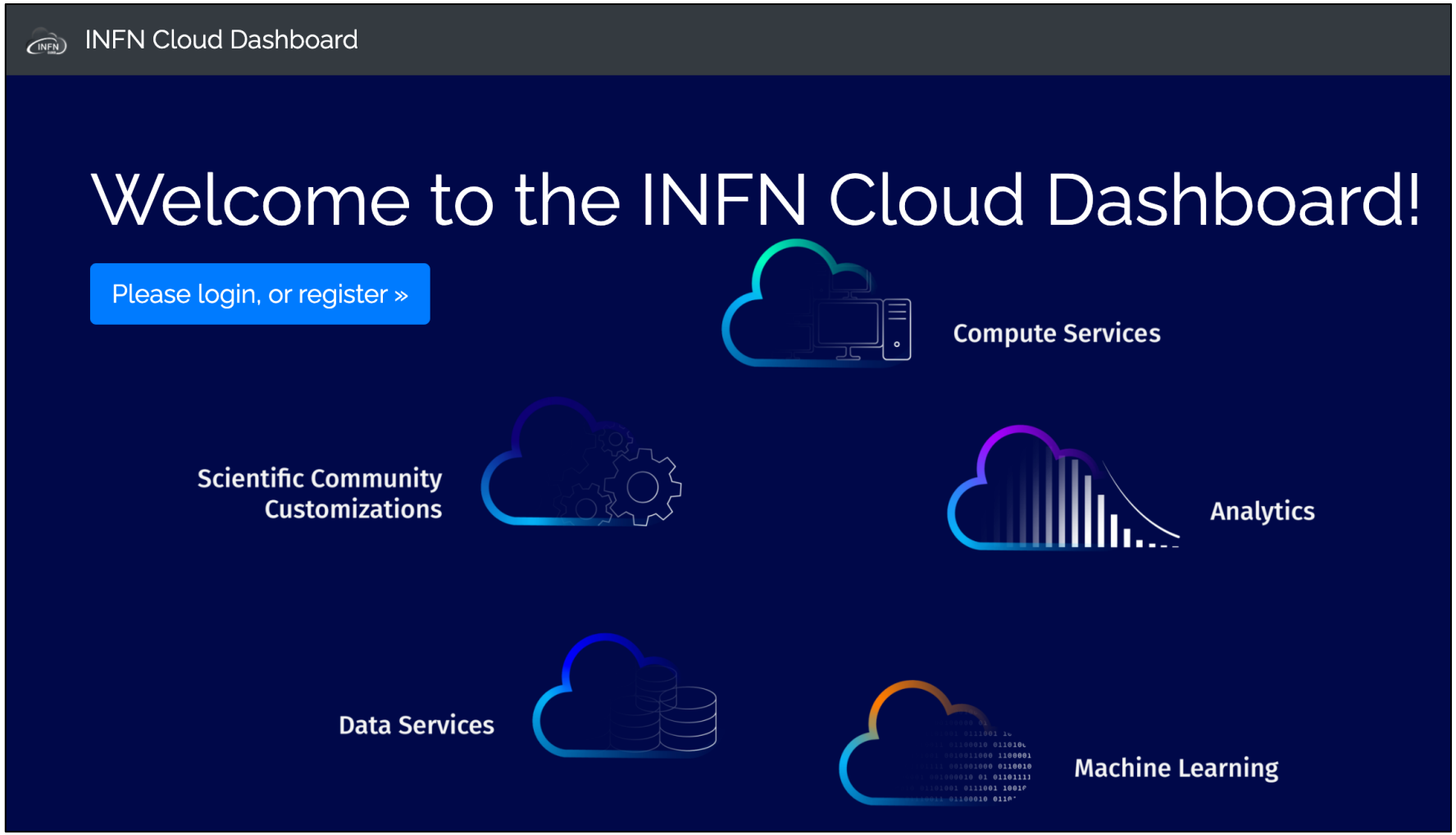
Struttura organizzativa

- INFN Cloud è internamente organizzato in **5 Work Packages**, ai quali partecipano dipendenti INFN di O(10) sedi in modo totalmente distribuito:
  - WP1: Architecture, Operations and Service Portfolio
  - WP2: Documentation, User Support, Communication and Training
  - WP3: Monitoring and Accounting
  - WP4: Security, Policies and Rules of Participation
  - WP5: Service Evolution and New Developments
- La gestione *ad interim* (in attesa di una definizione più strutturata all'interno di una possibile nuova organizzazione del calcolo, attualmente in discussione con il gruppo coordinato da Gianpaolo Carlino) è fornita dal **INFN Cloud Management Board** , composto dal coordinatore di INFN Cloud (Davide Salomoni) e dai WP leader (2 per ogni WP).

Che cosa fornisce INFN Cloud?

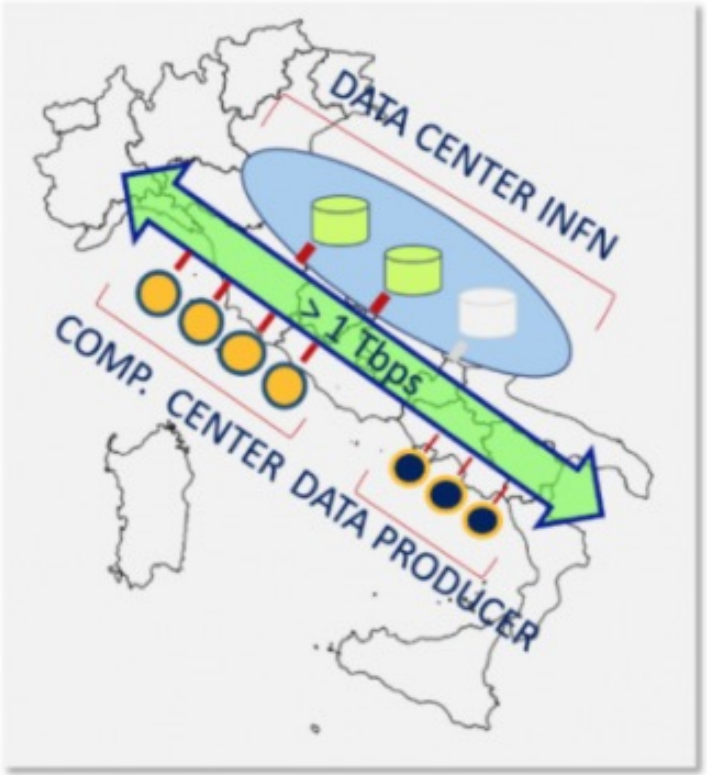
- Un «portafoglio» di servizi che include la possibilità di istanziare:
  - **Macchine virtuali** (VM) di differenti dimensioni e tipi, con o senza volumi esterni.
  - **Docker containers** oppure **applicazioni a container multipli** definite attraverso file `docker-compose`.
  - **Cluster basati su orchestratori di container come Mesos e Kubernetes**. Un utente può cioè chiedere autonomamente ad esempio un «cluster Kubernetes as a service» e poi utilizzarlo per istanziarvi le proprie applicazioni.
  - **Ambienti pre-configurati per data analytics** (che usino es. Spark e/o ElasticSearch e Kibana, R, etc.).
  - Soluzioni di **storage a oggetti e posix**, anche connesse a servizi applicativi ad alto livello; ad esempio, **Jupyter Notebooks dotati di storage permanente e automaticamente replicato**.
  - **Cluster *dinamici* realizzati secondo specifiche di esperimento o collaborazione**; ad esempio, cluster con un HTCondor batch system, ambienti ottimizzati per ML con GPU, notebook pre-configurati con simulatori di Quantum Computing, etc.).
  - Servizi che necessitino di **user-level encryption di dischi** (sull'intera infrastruttura INFN Cloud) o dedicati al **trattamento di dati sensibili** (per ora solo sulla infrastruttura Cloud certificata IEC/ISO 27001 presso il CNAF).
- **Il portafoglio di INFN Cloud può venire facilmente esteso** a fronte di nuove esigenze, grazie ad un linguaggio ad alto livello che ci consente di riusare e comporre una serie di moduli standard («service composition»).

La nuova Dashboard: https://my.cloud.infn.it



L’architettura fisica di INFN Cloud

1. **Un *backbone*** che include i due centri di calcolo più grandi dell’INFN (CNAF and Bari).
   - In ciascuno dei due siti esiste una “INFN Cloud backbone infrastructure”, connessa ad alta velocità con l’altro sito.
   - Il backbone è utilizzato per ospitare gli **INFN Cloud core services**, come i servizi PaaS, il sistema di DNS interno, i servizi di monitoraggio e logging. Ospita anche servizi utenti che devono sfruttare le caratteristiche intrinseche del backbone, come la **replica automatica** di storage ad oggetti tra i due siti.
2. **Un insieme di infrastrutture cloud distribuite**, connesse e federate con il backbone. Attualmente le infrastrutture Cloud di CNAF e Bari (che *non coincidono* con le relative infrastrutture di backbone) sono già connesse al backbone di INFN Cloud. Diversi altri siti INFN sono nella «pipeline» per la connessione al backbone (il prossimo sarà probabilmente la parte INFN di Cloud Veneto).
   - Questa federazione consente agli utenti INFN di sfruttare tutte le risorse e i servizi resi disponibili da INFN Cloud (vedi dopo).



Altre comunità «in Cloud»…

- Ci sono anche diversi altri esperimenti che già usano risorse fornite in Cloud e magari *non lo sanno / non lo vedono*. **Non c'è in questi casi ancora esplicita connessione a INFN Cloud** ma per tutti prevediamo la possibilità di utilizzare in futuro il portafoglio di INFN Cloud. Alcuni *esempi* su Cloud@CNAF*:*
  - **Neutrino/Dune** (gruppo 2) → interactive node (16CPU) for analysis with scratch space on cloud volumes - archive storage on t1 storage – graphical access via x2go.
  - **AMS e Darkside** (gruppo 2) → ancillary services (monitoring information).
  - **Virgo** (gruppo 2) → ancillary services (monitoring information), Kubernetes cluster (5 nodes, it might become a Kubernetes as a Service with INFN Cloud for low-latency analysis (to avoid batch system queues delay), plus several powerful user interfaces for interactive access (currently only virtualized but not cloudified yet).
  - **Icarus** (gruppo 2) → interactive node (16CPU), for analysis with scratch space on cloud volumes.
  - **JUNO** (gruppo 2) → ancillary services for Testbeds (RUCIO+FTS) and infrastructure developments plus INDIGO-IAM test instance (to be moved to production).
  - **FAZIA, N_TOF** (gruppo 3) → small clusters (4/5 nodes) for production/analysis, with scratch space on Cloud volumes, instantiated on demand.
  - **ASFIN** (gruppo 3) → powerful (16 cores) node for interactive/graphical HPC jobs – performance issues more cores needed, with scratch space on cloud volumes.
- Vogliamo dunque discutere se e in che modo integrare o espandere i servizi di INFN Cloud per queste comunità e le loro applicazioni, eventualmente portate in modalità *cloud-native*.

Il **backbone** di INFN Cloud

Il backbone di INFN Cloud si basa sulle istanze OpenStack ex INFN-CC in esecuzione a Bari ed al CNAF e sui servizi di supporto realizzati per quel progetto

- ospita i servizi necessari al funzionamento di INFN Cloud
- ospita risorse IaaS e PaaS istanziate dagli utenti
- offre servizi di supporto agli use case degli utenti
- si caratterizza per la facilità di implementarvi servizi ed applicazioni in HA geografica

INFN Cloud: la vision

- **INFN Cloud offre un catalogo di servizi in continua evoluzione**
  - fornendo supporto tecnico per il porting di nuove applicazioni in cloud
- **INFN Cloud è una federazione di infrastrutture cloud esistenti**
  - Il backbone, fatto di 2 siti: CNAF e Bari
  - Un set di infrastrutture cloud distribuite geograficamente (p.e. RECAS-BARI, CLOUD@CNAF, CLOUD-Veneto)
- **La federazione delle risorse** è abilitata tramite
  - uno stesso layer di autenticazione/autorizzazione basato su **INDIGO-IAM**
  - un set di policy consistenti per la gestione degli utenti, delle risorse (reti, compute, storage), etc.
  - orchestrazione trasparente e dinamica delle risorse su tutte le infrastrutture federate tramite l'**INDIGO PaaS Orchestrator**