Storage solutions for a production-level cloud infrastructure

Giacinto DONVITO
INFN-Bari

Outline

- Use cases
- Technologies evaluated
- Implementation (hw and sw)
- Problems and optimization
- Services/features implemented and usage statistics
- Users feedback
- Conclusions

Use cases

- PON PRISMA project:
  - PRISMA is a **Industrial Research** project, founded by European Community, Italian Minister of Research and Italian Minister of Economical Development.
    - A 32 months project **ending in June 2015** with an overall budget of about **20M€**
  - Joining together IaaS, PaaS and SaaS development
  - Both public research institution and private company
  - The aim is to implement an Open Source **IaaS+PaaS platform** that could allow development of SaaS for both **Public Administration** and **Scientific applications**

Use cases

- Scientific data analysis:
  - Providing support for interactive physics data analysis
    - **Virtual Analysis Facilities, User interfaces as a Service, Batch System as a Service, etc.**
  - Providing support for other science data analysis
    - **Bioinformatics**, Seismic Risk, **astrophysics**, etc.
  - **Data preservation** use-cases
    - Emulating very old Operating System
  - Supporting **tutorial and courses**
    - Providing virtual environment in a transparent

SW Technologies evaluated

- Both for PRISMA and scientific use cases, we always look for **Open Source solution/software** well supported by communities and **widely used**
- With the aim of implementing a complex infrastructure base on the collection of software and **solution** coming from **different communities**
  - One of the major effort is understanding/**optimizing the configuration** and the interaction among software

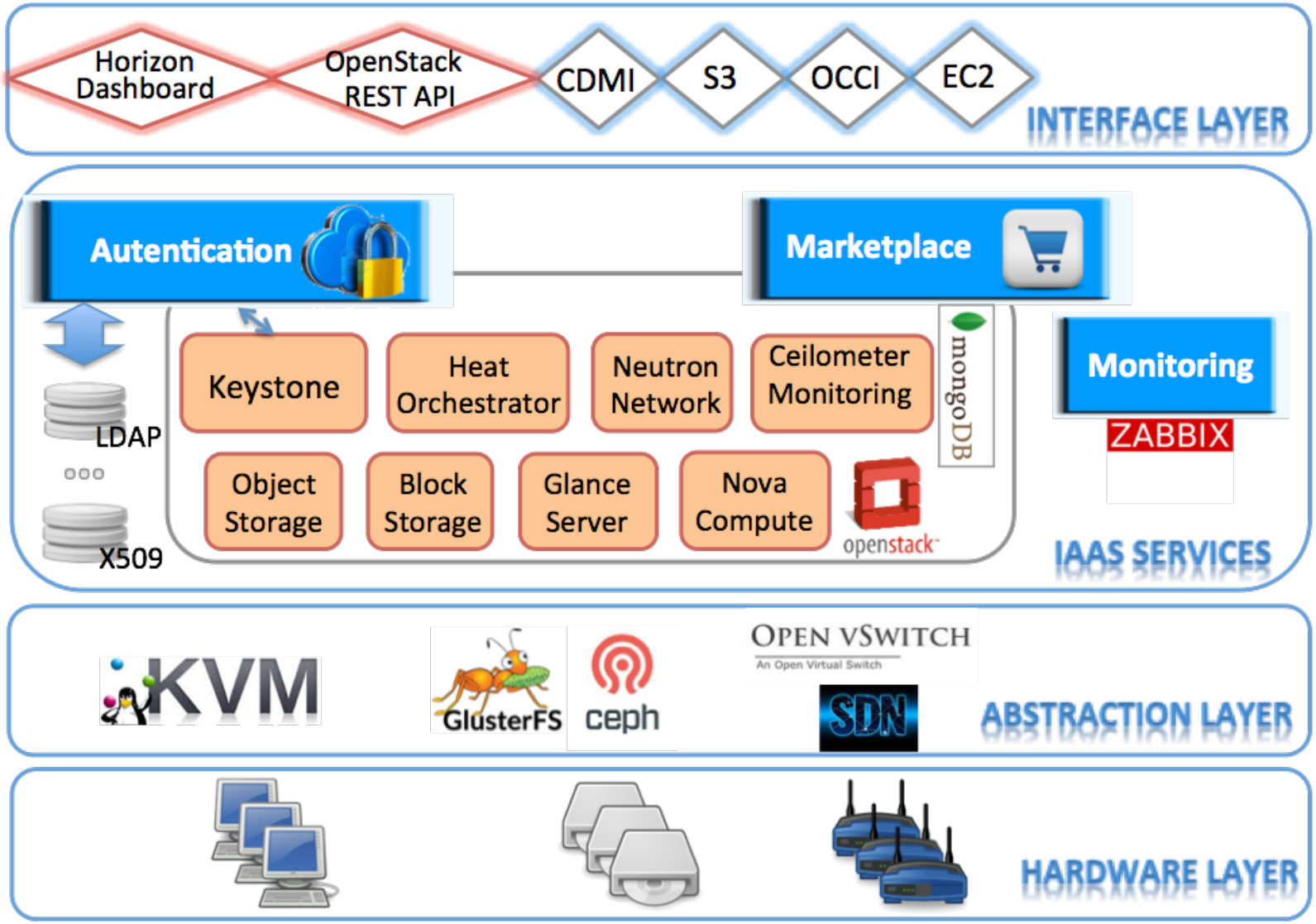
SW Technologies evaluated

- **Cloud IaaS Solution**:
  - **OpenStack** (Icehouse at the moment)
    - KVM based virtualization
- **Storage**:
  - **GlusterFS** 3.4 (replica 2 and 3) both posix and iSCSI export
  - **CEPH** Firefly release (replica 3)
  - **Swift**: Supported both S3 and CDMI interface
- **Network**:
  - **Open vSwitch**, pfSense, OpenVPN,
- **Monitoring**:
  - **Ceilometer + Zabbix** 2.2.2
- **Operating System**:
  - **Ubuntu** 12.04 LTS

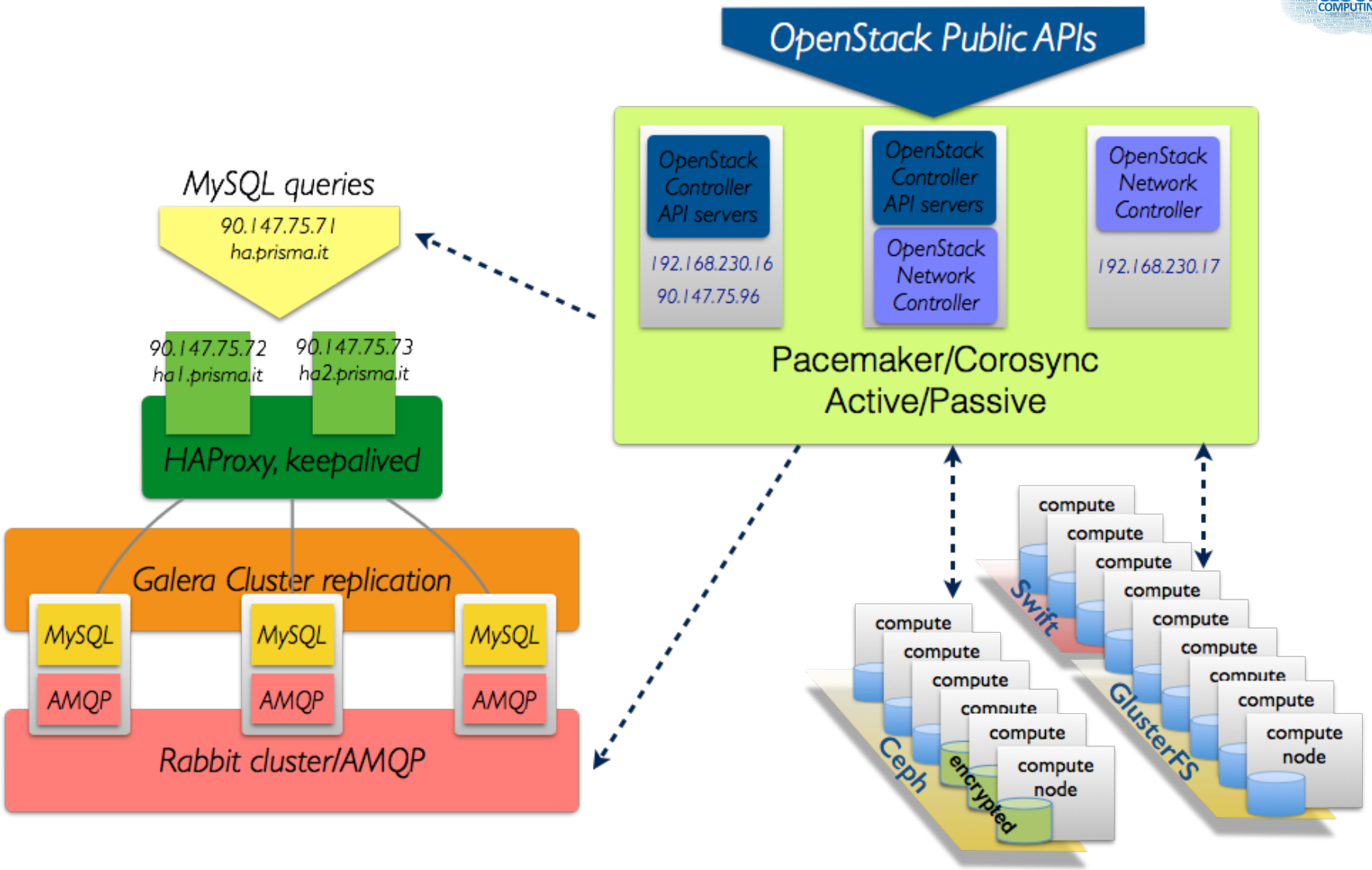
HW configuration

- **Base services**:
  - MySQL + RabbitMQ
    - 3 hosts with 8 Cores and 18GB of RAM each
- **Core services**:
  - 3 hosts with 24 Cores and 80GB of RAM each
- **Compute Node**:
  - 12 nodes with 32 Cores and 256 GB of RAM each + 15 nodes with 24 Cores and 80GB of RAM each
    - About **700 cores** and **4TB of RAM**
- **Network:**
  - Each physical hosts has 1x10Gbit/s and 2x1Gbit/s network connection. All wire-speed guaranteed bandwidth
- **Storage:**
  - ~150 disks, for a total of ~**470TB** of overall storage

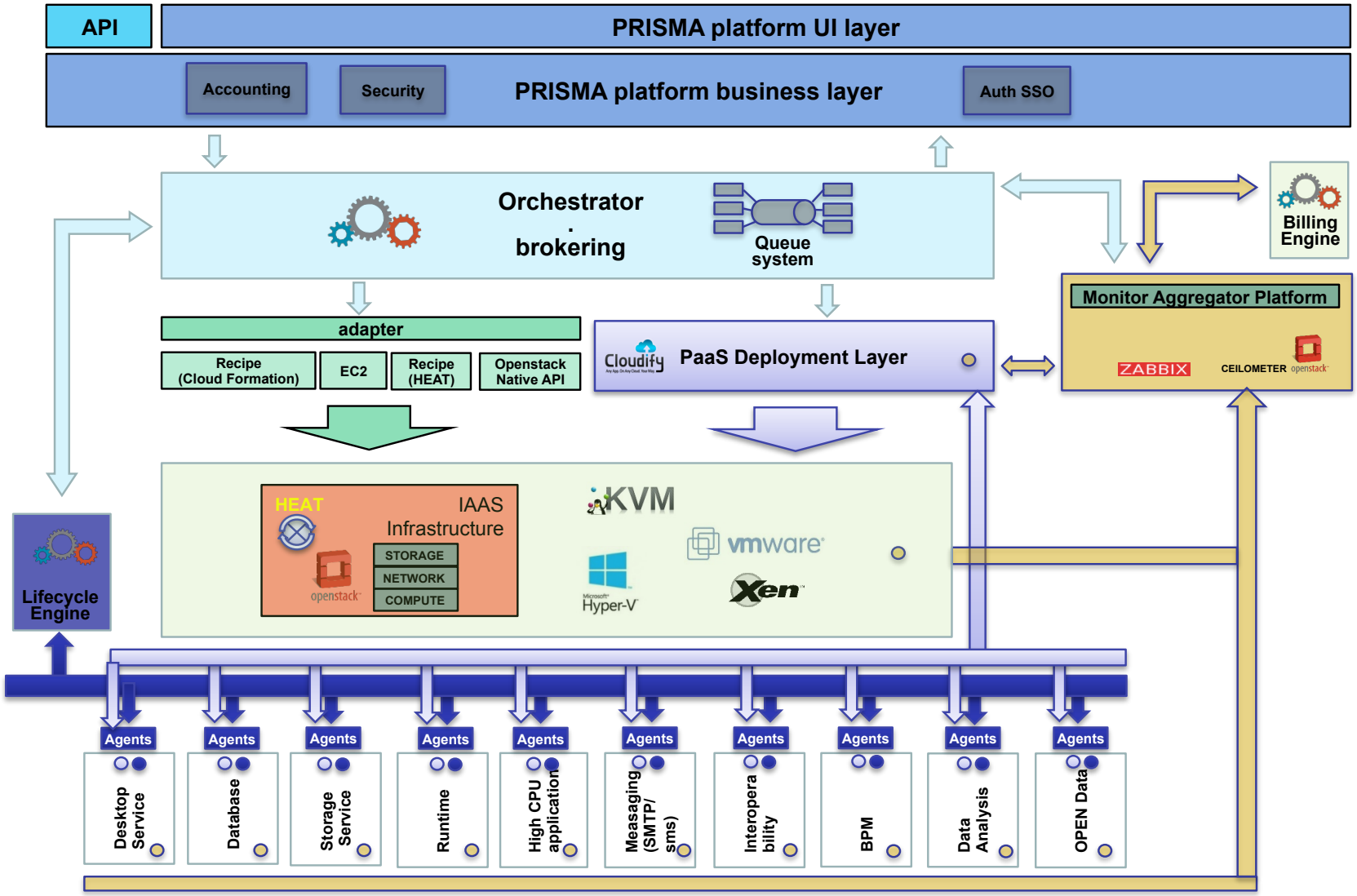
IaaS implementation



IaaS implementation



IaaS + PaaS platform



Problems and optimization

- **Live migration**:
  - It is important to provide fast, reliable and scalable, storage for the running instances
- **Block Storage**:
  - For **I/O demanding services** we will provide block storage facility with good scalability, reliability and performance
  - **Different use cases** (RDBMS, data analysis, test environment, etc) require different storage performance, size, **QoS**, etc.

Problems and optimization

- **Scalability**:
  - This infrastructure is intended to be scaled out in the next few months up to about 5000 CPU cores and about 2 Pb of storage
- **Reliability**:
  - The infrastructure should be able to overcome the failure of a fraction of node (hypervisors)

Problems and optimization

- **GlusterFS for the live-migration export mounted posix (native protocol)**:
  - We started with GlusterFS 3.3 and replica 2
    - We experienced frequently problem on the file-system that lead to file-system corruption on the VM or lost of files.
  - We tried also with GlusterFS 3.4 and replica 3
    - The system is more stable: no file lost.
    - Still experiencing quite few file-system corruption on the VMs

Problems and optimization

- We started using **GlusterFS** for hosting a **loopback devices** exploiting **Cinder LVM driver**:
  - This is the most complete driver when we start looking into OpenStack (2012) but, soon:
    - The performances hits a limitation
    - The HA for block accessing is quite hard to achieve
- Actually we are exploiting **CEPH** for providing the **block storage** implementation
  - This provide good results in terms of **availability** and reliability
  - Better **scalability**
  - Easy to set-up pool of storage for different QoS

Problems and optimization

- We are providing **different pools** of **CEPH** storage to **implement** different **QoS**:
  - Small and fast disks for DB
  - Large and slower disks for the data analysis services
- We are also testing the usage of **SSD for Tiered Storage**:
  - Both writing and reading operation are cached within the SSD and than staged on the slower disks.
  - Depending on the use cases, this increase noticeably the performances
- Using the **CEPH encrypted pool** it is possible to provide the VM on OpenStack with a “secure storage solution”

Problems and optimization

- We are in the last phase of testing the **RBD** support for **running the VM** and supporting live-migration.
  - In Icehouse it work quite well
    - There is still a bug on disk resize
- Both CEPH and OpenStack are able to support the **geographical distribution** of resources:
  - Using “availability zone” and CEPH “pools”
  - We tested it successfully ☺
- We are exploiting **Swift** not only to provide **Object Storage** to end-users
  - But also to provide high-available back-end for glance
    - Storing images, and **backup** (and **disaster recovery**) of the VM

Services implemented

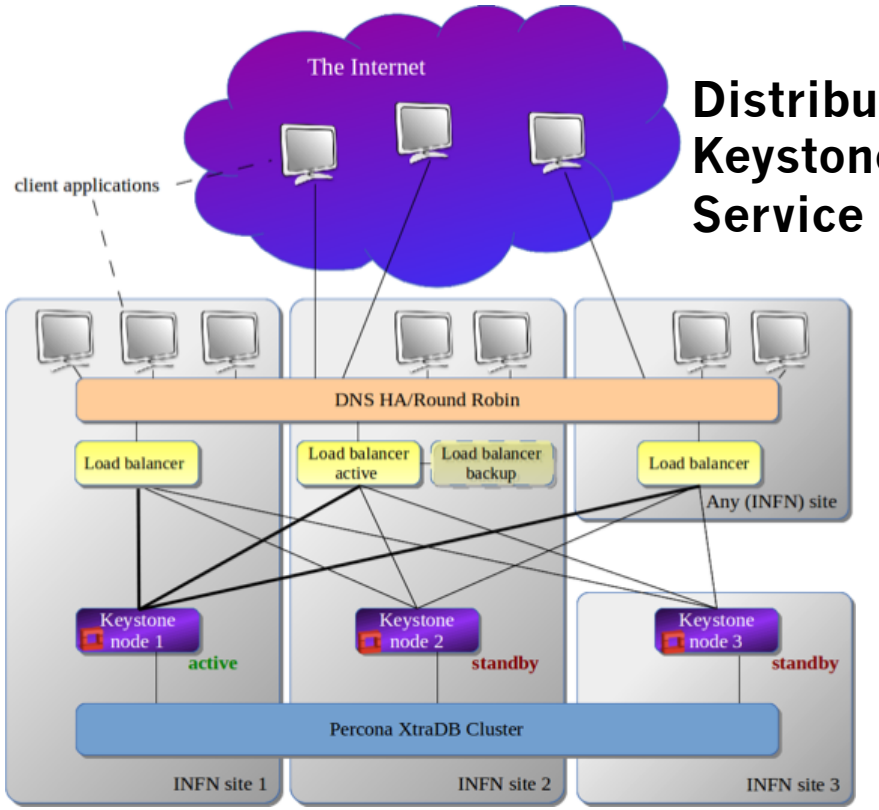
- In the contest of PaaS in PRISMA, we are developing a **DB as a Service**:
  - Based on **HEAT** and **CEPH**
  - We have developed a template that is able to provide automatically all the needed resources:
    - A dedicated machine,
    - a dedicated block device, that could be **encrypted** or not
  - The PaaS orchestrator could **automatically** ask for vertical **scaling** of the services (CPU, Ram, disk) if the monitoring system reveals an **overload**
  - It is very important in this context to have a flexible, scalable and powerful RDBMS solution

Services implemented

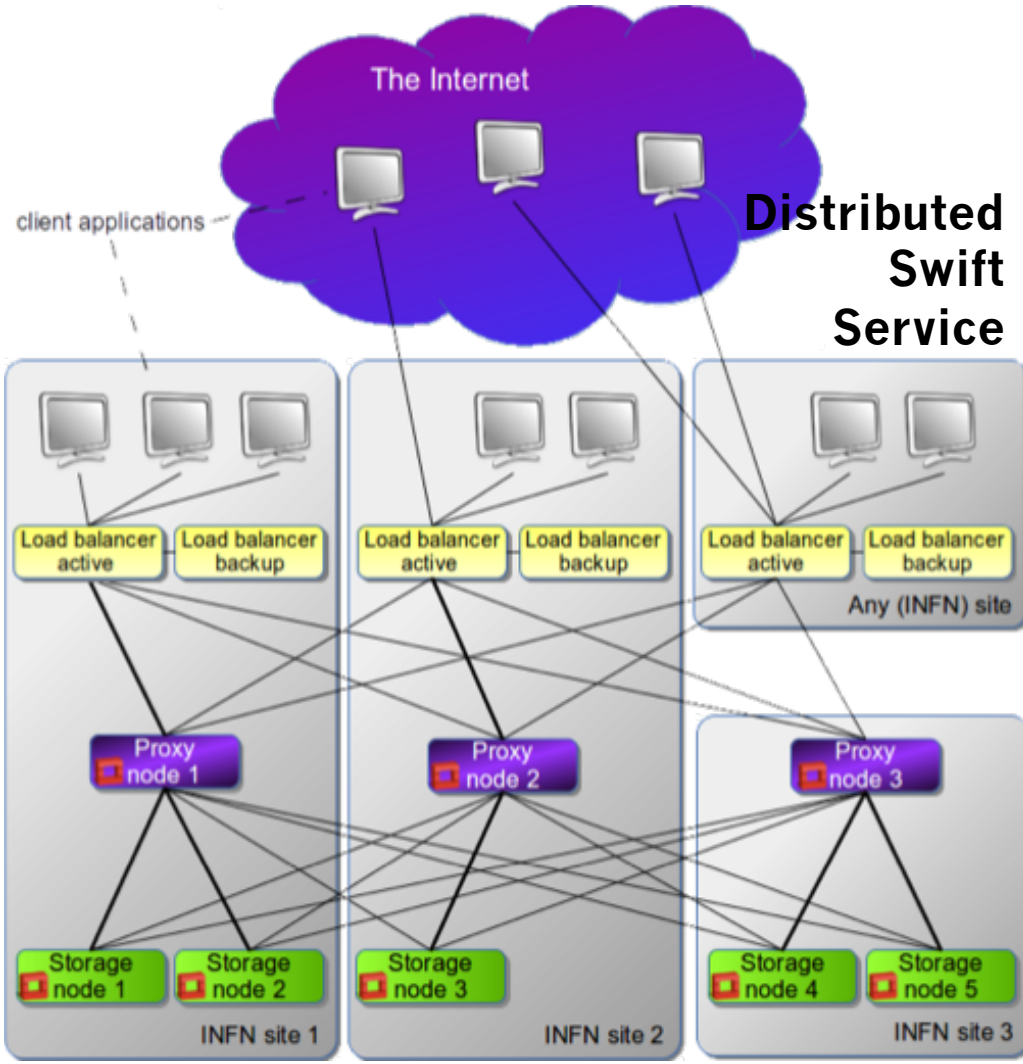
- In the contest of PaaS in PRISMA, we are developing a **Personal Storage as a Service**:
  - Based on **ownCloud** and cloud storage
  - We are in the phase of testing both: ownCloud + RBD over a dedicated machine, or ownCloud + Object storage (Swift)
  - The users (or group) could ask for a **dedicated service**, where they can put their personal data
    - This approach is specifically required when the privacy of data (and the encryption) is a crucial aspect

Services implemented

- A **geographically distributed Swift** storage service
  - Exploiting a **geographically** distributed Keystone service
  - It is possible to store data "**eventually consistently**" across many sites over a WAN connection
  - The data reading is performed exploiting the **data locality** in order to improve users experience
- This looks a good solution for **disaster recovery** of data across several sites.



Distributed Keystone Service



Distributed Swift Service

- Slides from **Stefano Stalio** @INFN-LNGS
  - Based on activities carried on by **4 INFN centres** (Lngs, Bari, Padua, Rome)

Services implemented

- One of the most interesting use-case for Cloud and Cloud Storage is: Disaster Recovery
  - The use-case is surviving also to a complete site failure
  - We are testing two models:
    - Cold (backup based) migration
      - This is already successfully tested:
      - Using Swift ad repository for the VM snapshot
      - It is possible to start the snapshot of the VM on another OpenStack Region
    - ~Live (distributed storage based) migration

Services implemented

- In the contest of PaaS in PRISMA, we are developing a **Desktop as a Service**:
  - Based on **X2Go**
  - The end-users could have a powerful and “always-on” machine dedicated to their needs.
  - Storing data on performant “drive” is quite important
    - Many non-HEP users need this environment for analysing data
  - We are providing this service to UNIBA PhDs with good results.
- Some users are looking for more advanced solutions:
  - **Hadoop as a Service**, or **Cluster as a Service**
    - In this case the scalability of the storage performance is one of the most important parameters

Service summary

Status:	Beta
Number of users (current, target):	~200
Default and Maximum quota:	1—2 TB
Linux/Mac/Win user ratio:	90/8/2
Desktop clients/Mobile Clients/Web access ratio:	90/5/5
Technology:	DB||Personal Storage||Analisys Facility
Target communities:	University PhD || researchers in physics and bioinformatics || Private company
Integration in your current environment (examples):	Push the data into the system via webdav or synchronizing with ownCloud
Risk factors:	Data loss || system un-availability
Most important functionality:	Storage QoS, reliability, scalability
Missing functionality (if any):	

User feedback

- Reliability
  - Reliability problem and data loss with GlusterFS
    - This is quite bad from the users point of view
  - We are now migrating everything to CEPH,
    - It seems far better, but we still need to gather statistics
- Performance and scalability
  - This is perceived, by the end-users, as quite important in order to use those services in a real production environment

Conclusions

- INFN is deeply exploring cloud concepts and technologies in order to fulfill the new uses requirements.
- This effort is not only related to funded projects, but there is INFN internal “Cloud working group” that is actively working on those items in order to provide cloud based solution for the end-users.
- Those solutions
  - would be very useful to increase the flexibility in using resources
  - Would give the possibility to provide disaster recovery solution.