The 'Cloud Area Padovana': lessons learned after two years of a production OpenStack-based IaaS for the local INFN user community

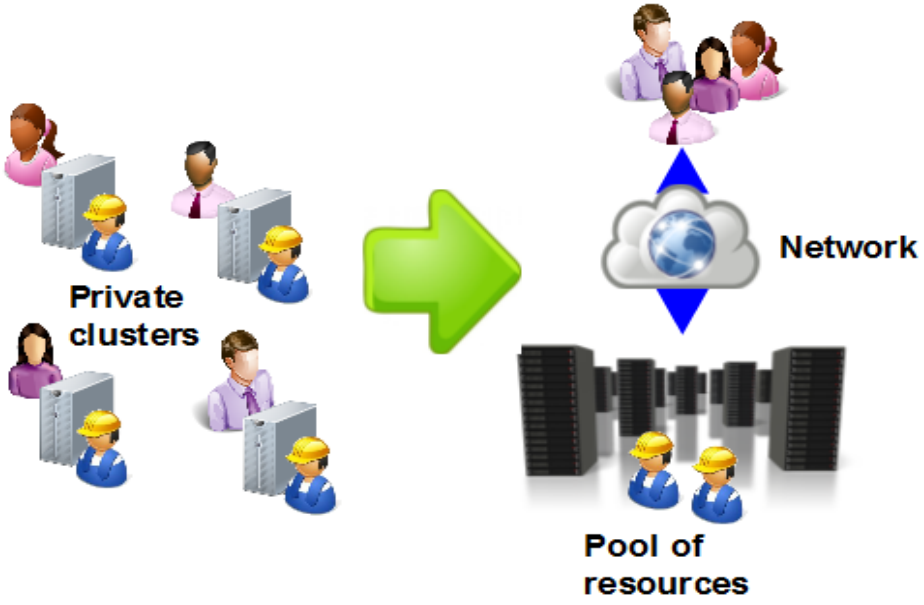
International Symposium on Grids and Clouds (ISGC) 2017
Academia Sinica, Taipei, Taiwan, 5-10 March 2017

Marco Verlato - on behalf of Cloud Area Padovana team

INFN (National Institute of Nuclear Physics)
Division of Padova
Italy
marco.verlato@pd.infn.it

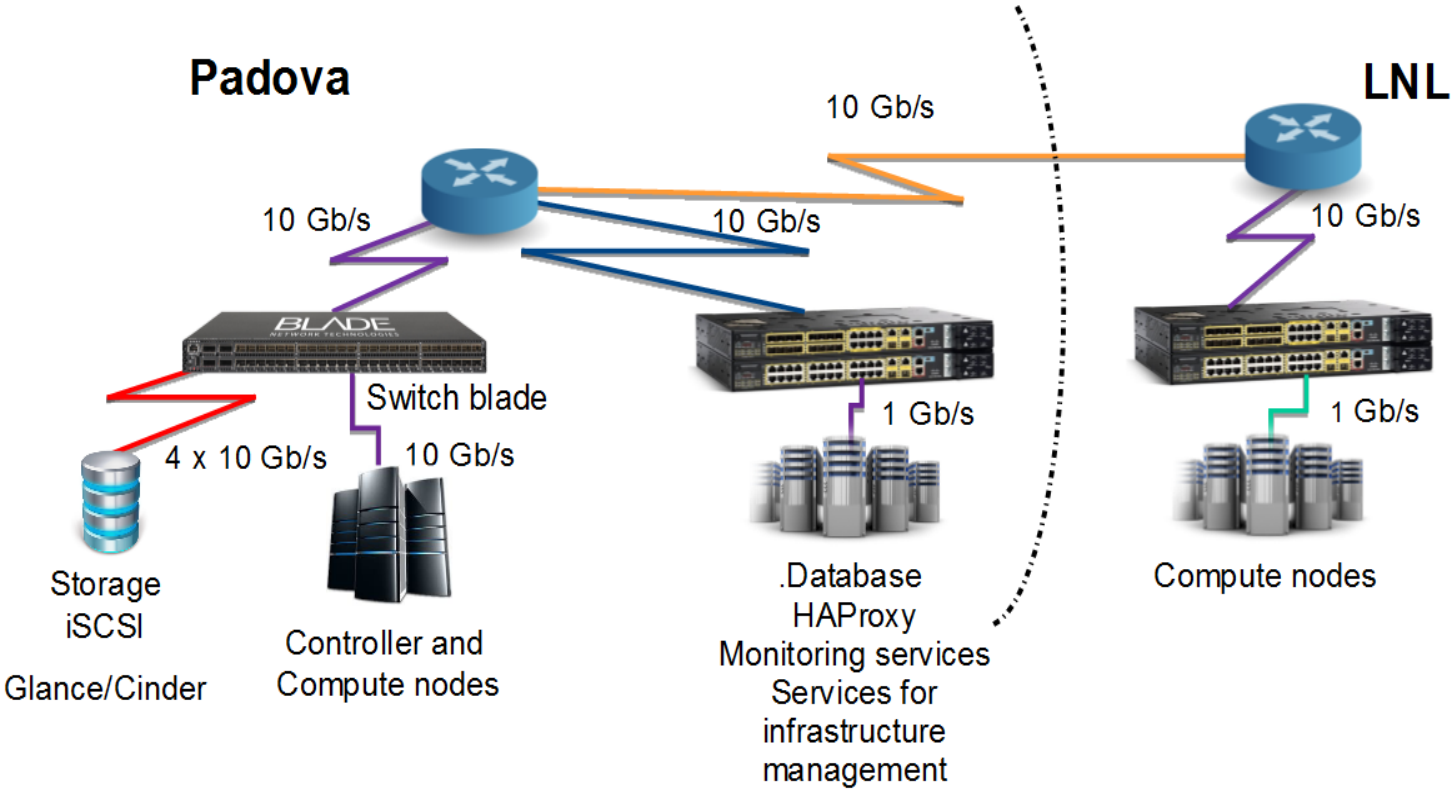
A distributed cloud

- **Cloud Area Padovana** is a OpenStack based distributed IaaS cloud designed at the end of 2013 by INFN Padova and INFN LNL units
  - ✓ To satisfy computing needs of the local physics groups not easily addressed by the grid model
  - ✓ To limit the deployment of private clusters
  - ✓ To provide a pool of resources to easily share among stakeholders
- Sharing of infrastructure, hardware and human resources



Cloud Area Padovana layout

- Based on the longstanding collaboration as LHC Grid Tier-2 for ALICE and CMS experiments:
  - ✓ resources distributed in two data centers connected with a dedicated 10 Gbps network link
  - ✓ INFN-Padova and Legnaro National Labs (LNL) ~10 km far away



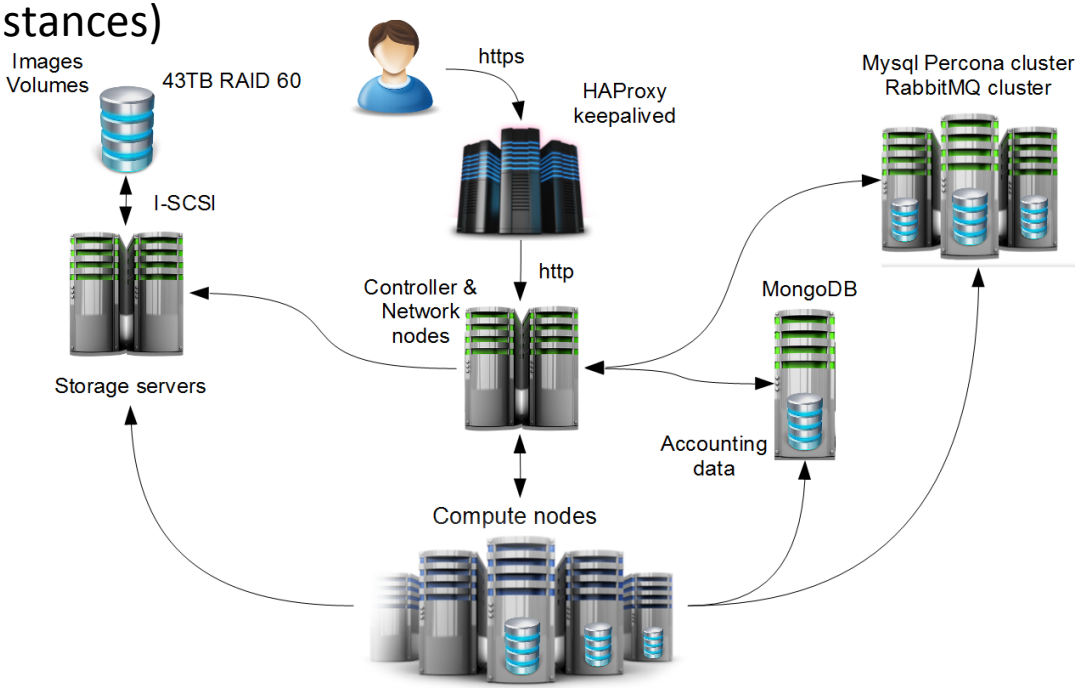
Cloud Area Padovana current status

- Service declared production ready at the end of 2014, now ~100 registered users, ~30 projects
- Physics groups planning to buy new hardware are invited to test the cloud, and if happy, their hardware joins the pool

Location	# servers	# cores (HT)	Storage (TB)
Padova	15	656	43 (img+vols)
LNL	13	416	
Total	28	1072	

Cloud Area Padovana architecture

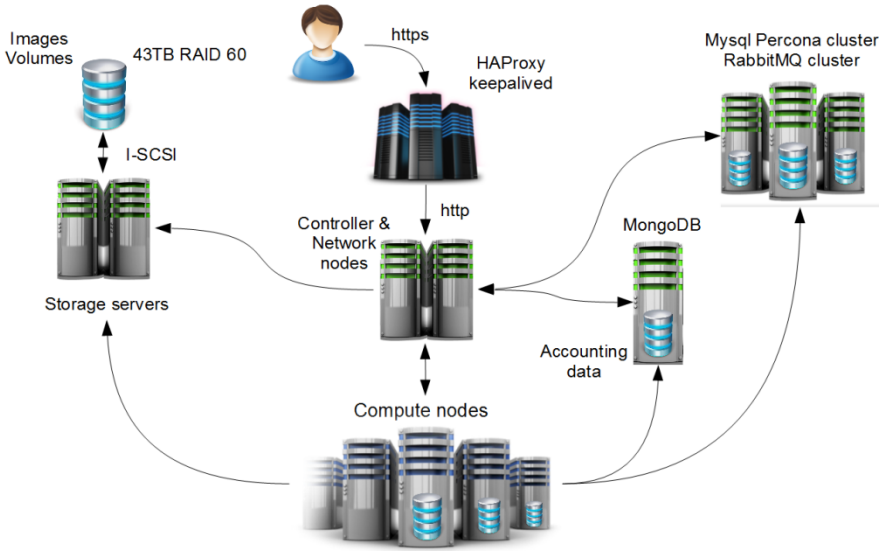
- OpenStack Mitaka version currently installed
- A OpenStack update per year (skipping one release)
  - ✓ Right balance for having last fix/functionalities with limited manpower
- Services configured in High Availability (active/active mode)
  - ✓ OpenStack services installed on 2 controller/network nodes
  - ✓ HAProxy/KeepAlived cluster (3 istances)
  - ✓ Mysql Percona XtraDB cluster (3 istances)
  - ✓ RabbitMQ cluster (3 istances)
- Core services installed:
  - ✓ Keystone (Identity)
  - ✓ Nova (Compute)
  - ✓ Neutron (Networking)
  - ✓ Horizon (Dashboard)
  - ✓ Glance (Images)
  - ✓ Cinder (Block storage)



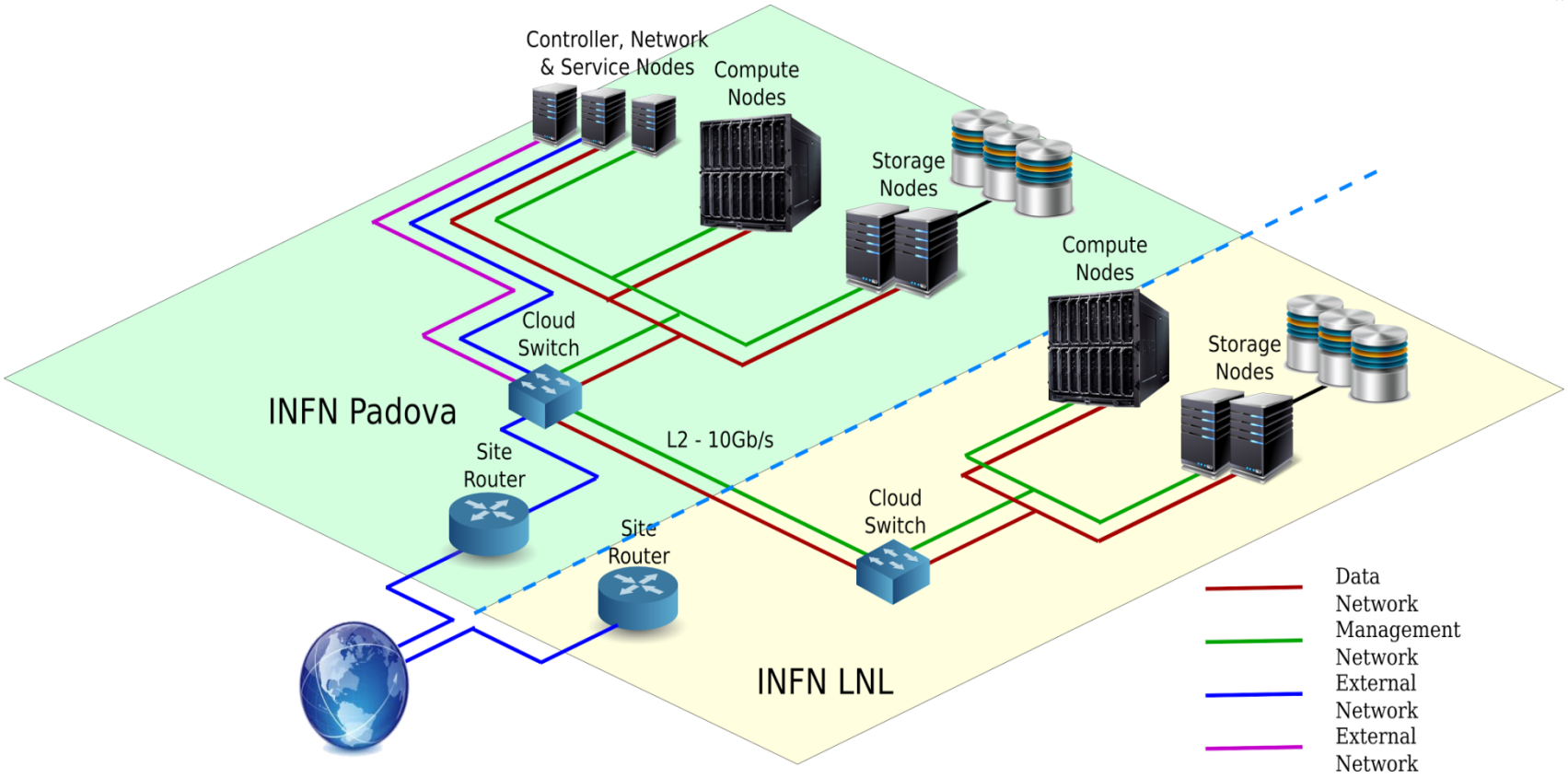
Additional services installed

- OpenStack optional services
  - ✓ Heat (Orchestration engine)
  - ✓ Ceilometer (Resource usage accounting)
  - ✓ EC2 API (to provide Amazon EC2 compatible interface)
  - ✓ Nova-docker (to manage Docker containers)
    - Recently deprecated, maintained by INDIGO-DataCloud project (github.com/indigo-dc/nova-docker)
    - OpenStack Zun being evaluated as replacement

- Home-made developments integrated:
  - ✓ Integration with Identity providers (INFN-AAI and UniPD SSO) for user authentication
  - ✓ User registration service
  - ✓ Accounting information service
  - ✓ Fair-share scheduling service



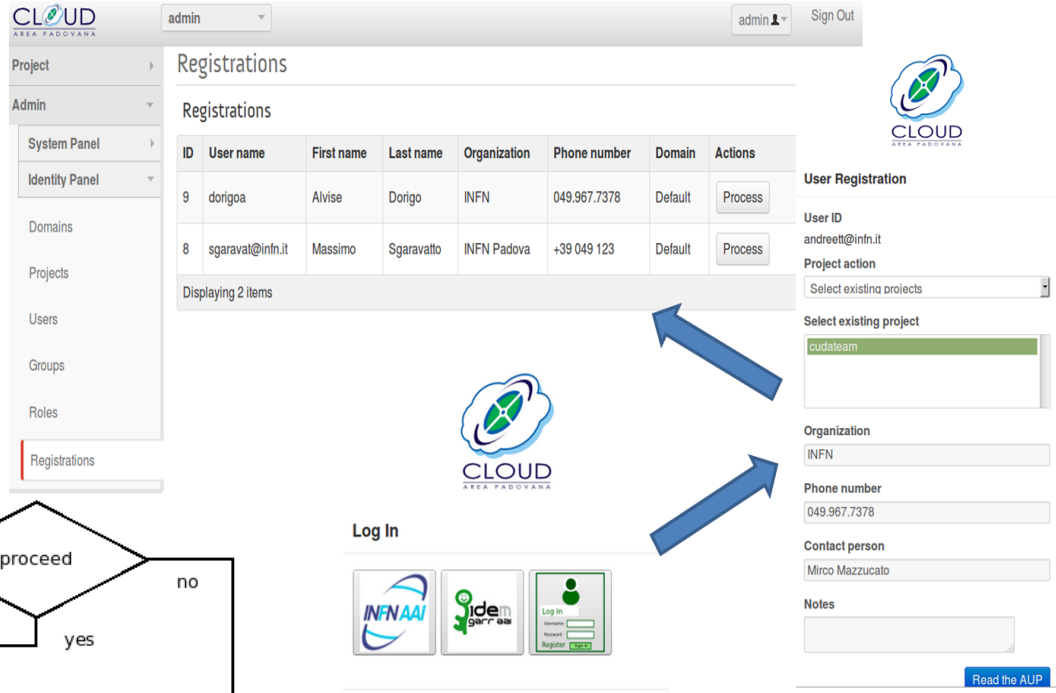
Network layout

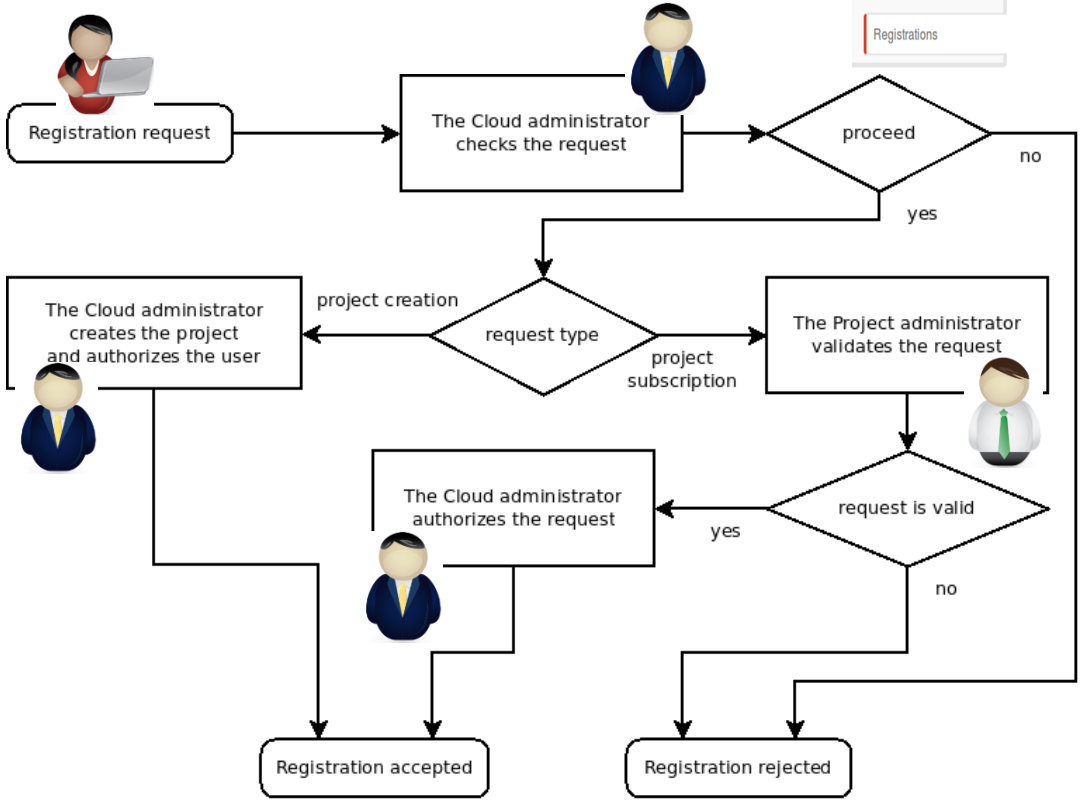


- Neutron with Open vSwitch/GRE configuration
- Two virtual routers with external gateways on public and LAN networks
- GRE tunnels among Compute nodes and Storage servers to allow high performance storage access (via e.g. NFS) from VMs

Identity and access management

- OpenStack Keystone Identity service and Horizon Dashboard extension:
  - ✓ to allow authentication via SAML based INFN-AAI Identity Provider, and the IDEM Italian Federation

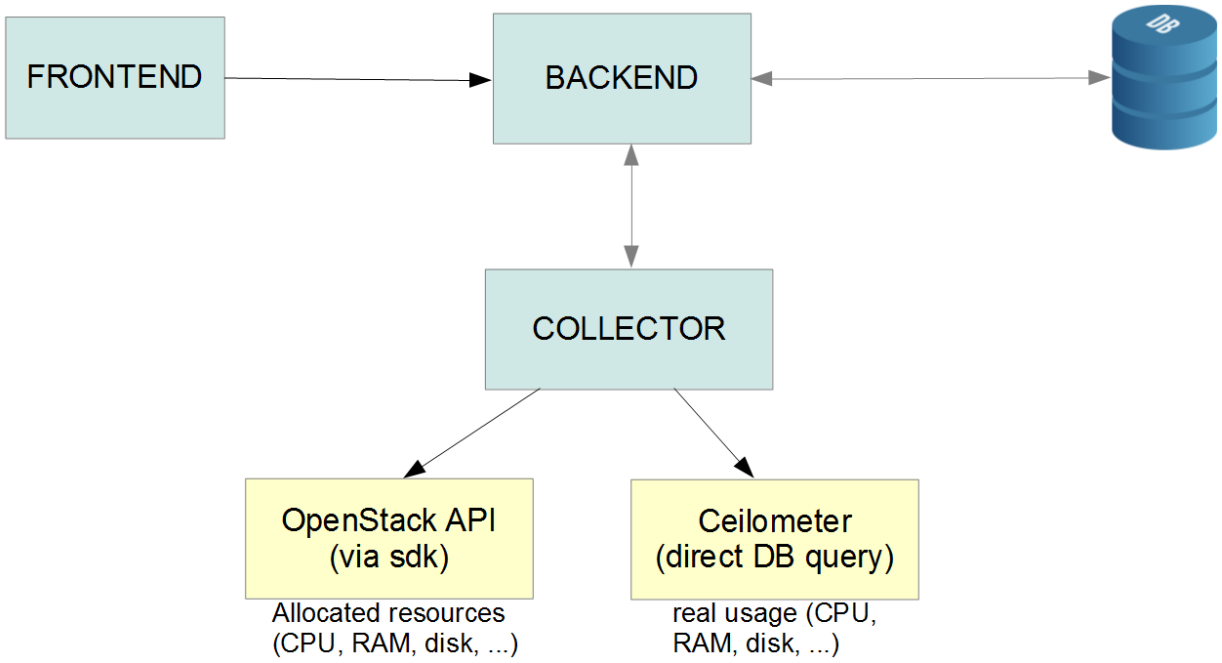




- ✓ to manage user and project registrations
  - a registration workflow (involving the cloud administrator and the project manager) was designed and implemented for authorizing users

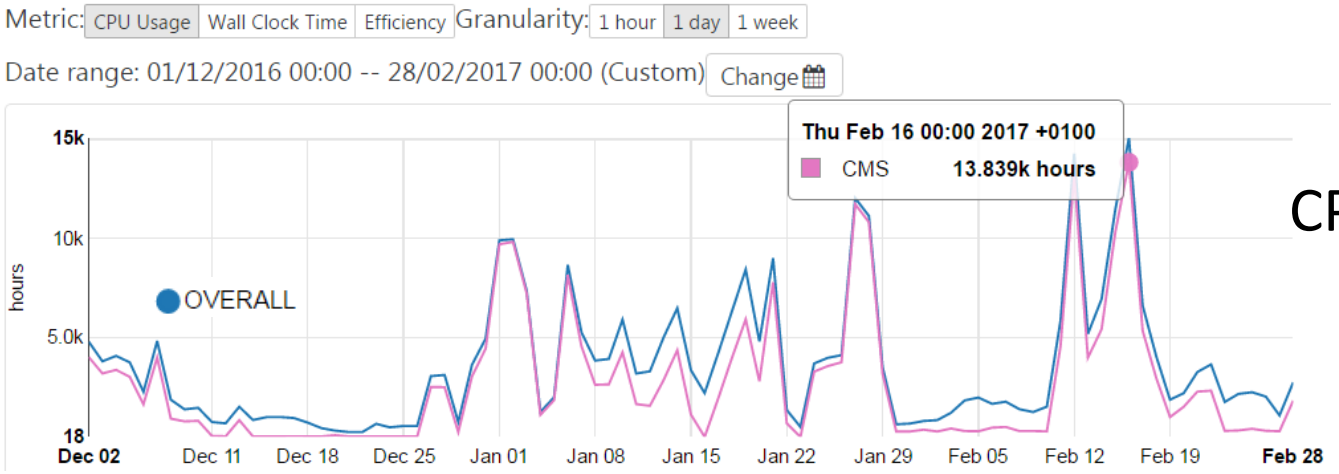
CAOS/1

- Accounting information are collected by Ceilometer service and stored in a single MongoDB instance
- Ceilometer APIs have well-known scalability and performance problems
- Data retrieval implemented through an in-house developed tool: CAOS
- CAOS extracts information directly from OpenStack API and MongoDB database

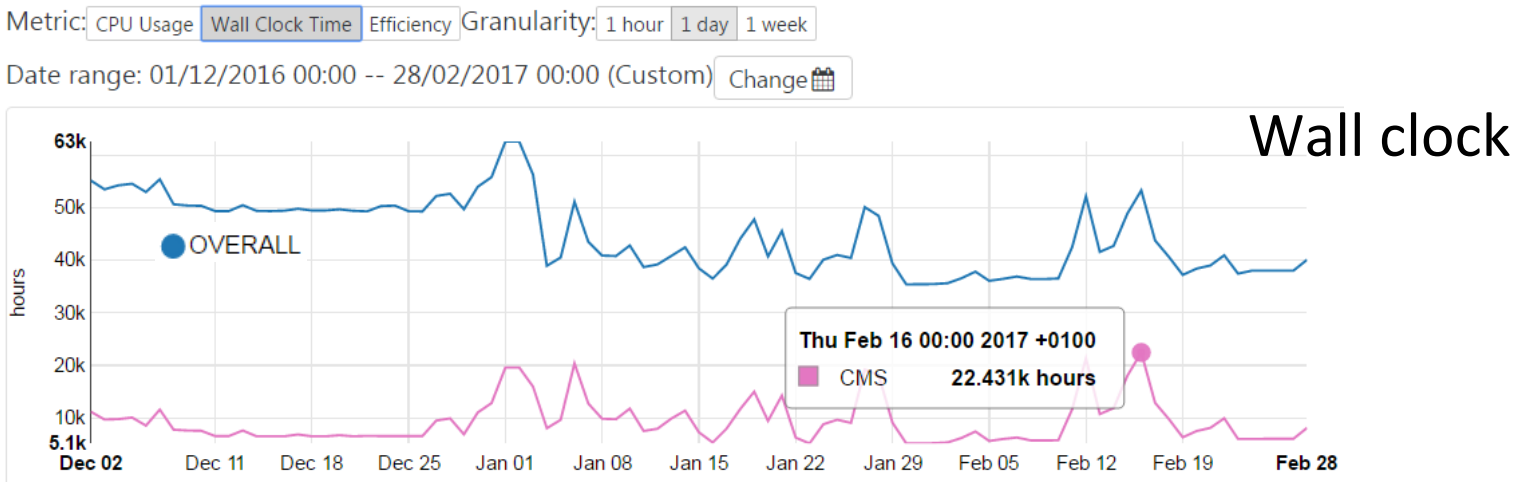


CAOS/2

- CAOS manages accounting data presentation
  - ✓ e.g. to show CPU time and Wall clock time consumed by each project vs time



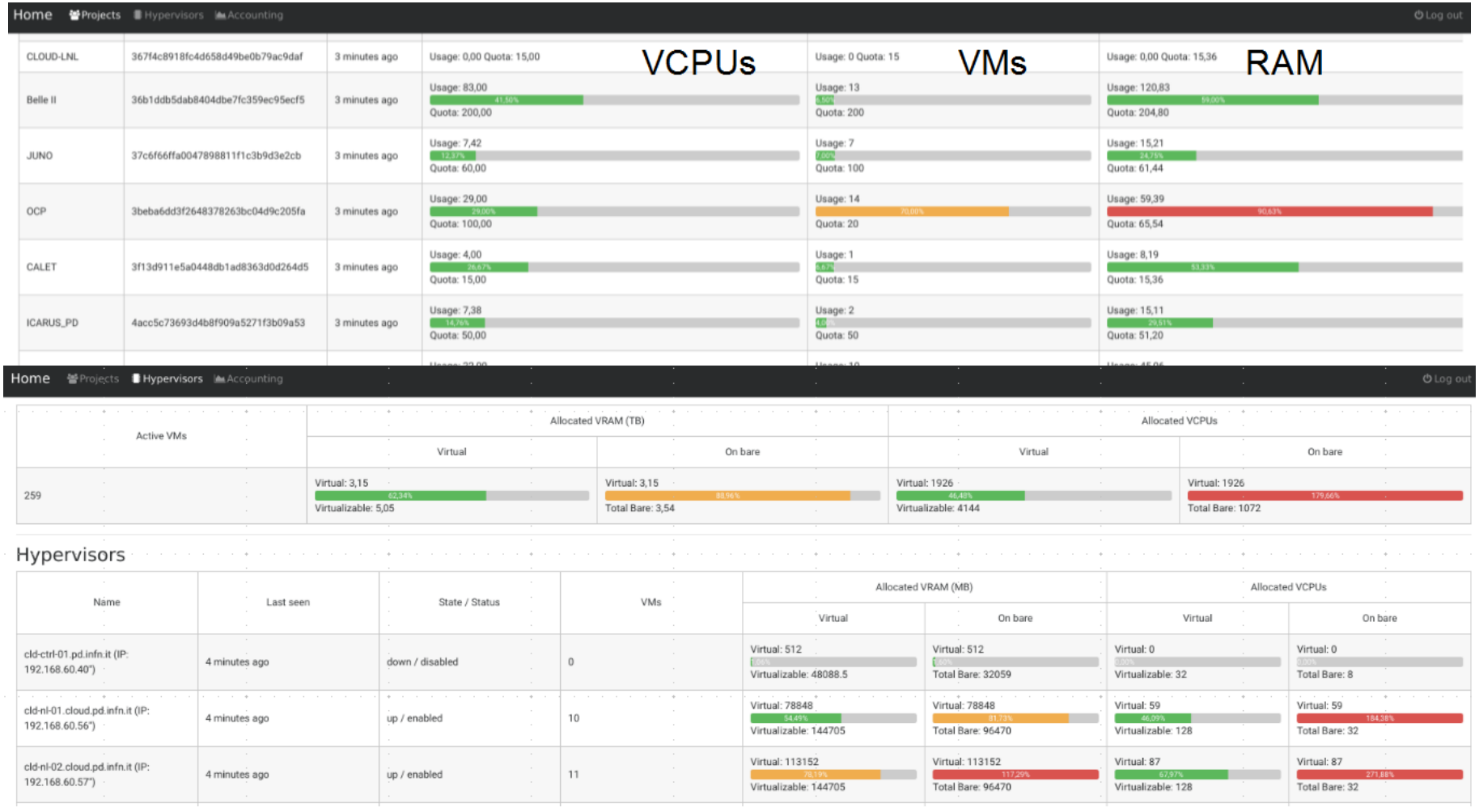
CPU



Wall clock

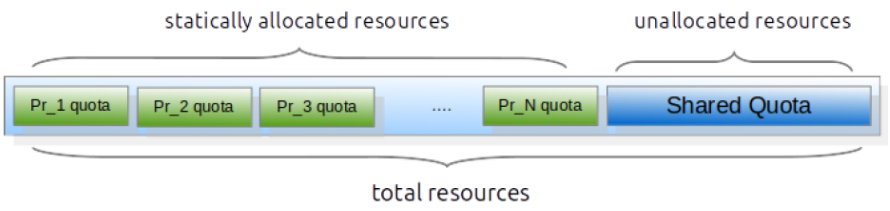
CAOS/3

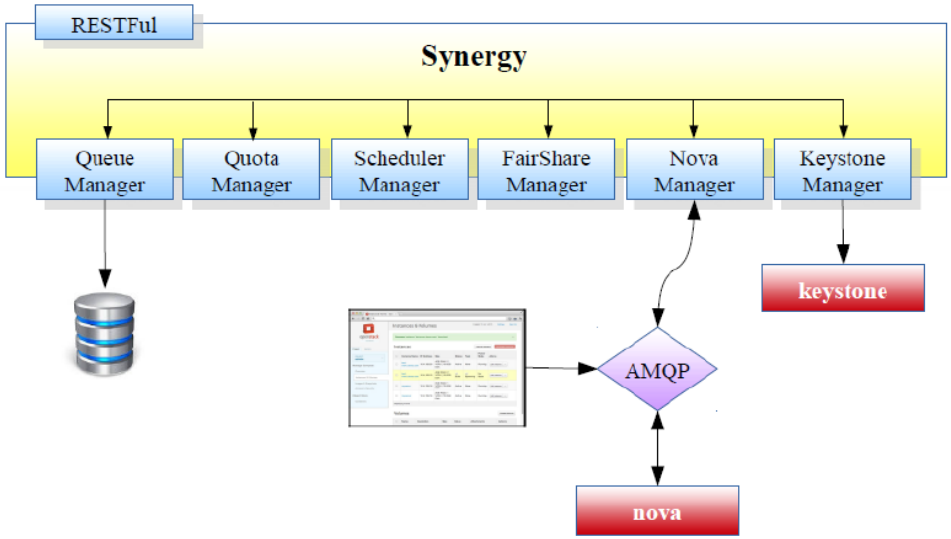
- CAOS also monitors:
  - ✓ resource quota usage per project
  - ✓ resource usage per node



Fair-share scheduling

- Static partitioning of resources in OpenStack limits the full utilization of data center resources
  - ✓ A project cannot exceed its quota even if another project is not using its own
  - ✓ Traditional batch systems addressed the problem via advanced scheduling algorithms, allowing the provision of average computing capacity over a long period (e.g. 1 year) to user groups sharing resources
- In cloud environment, the problem is addressed by Synergy
  - ✓ A service implementing fair-share scheduling over a shared quota
  - ✓ See next talk of Lisa Zangrando





Cloud Area Padovana usage

- ~ 100 registered users grouped in ~30 projects
- Each project maps to an INFN experiment/research group
  - ✓ ALICE, CMS, LHCb, Belle II, JUNO, CUORE, SPES, CMT, Theoretical group, etc.
- Different usage patterns:
  - ✓ Interactive access (analysis jobs, code development & testing, etc.)
  - ✓ Batch mode (job run on clusters of VMs)
  - ✓ Web services
- Current main customers are CMS and SPES experiments

CMS use case/1

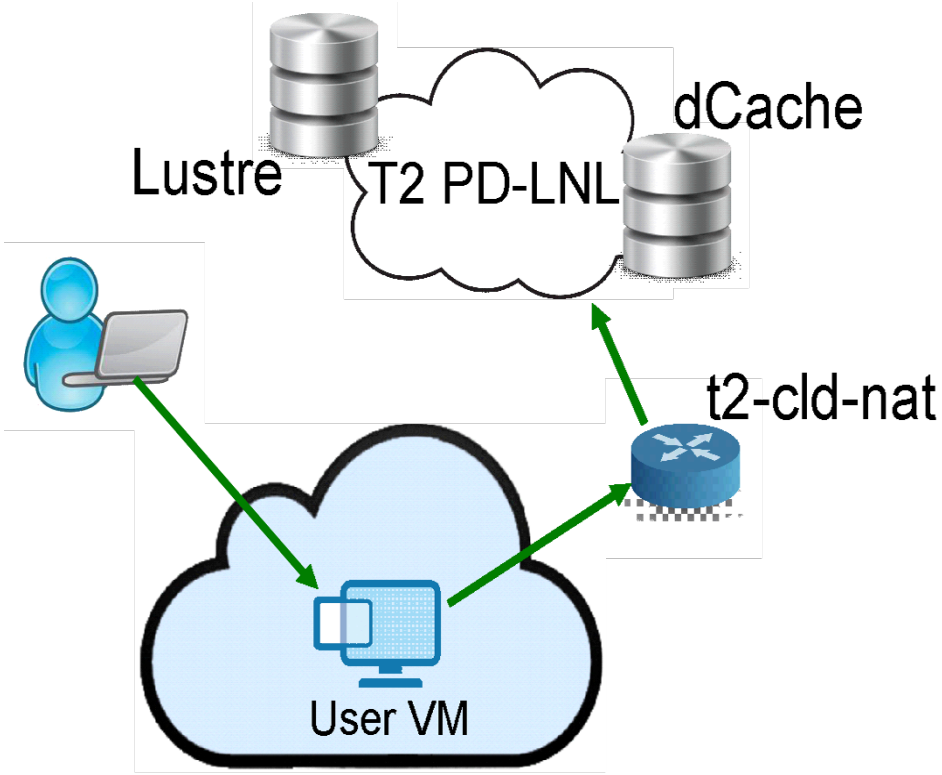
- Interactive usage:

✓ Each user instantiate his own VM for:

- code development and build
- ntuple productions
- end-user analysis
- grid user Interface

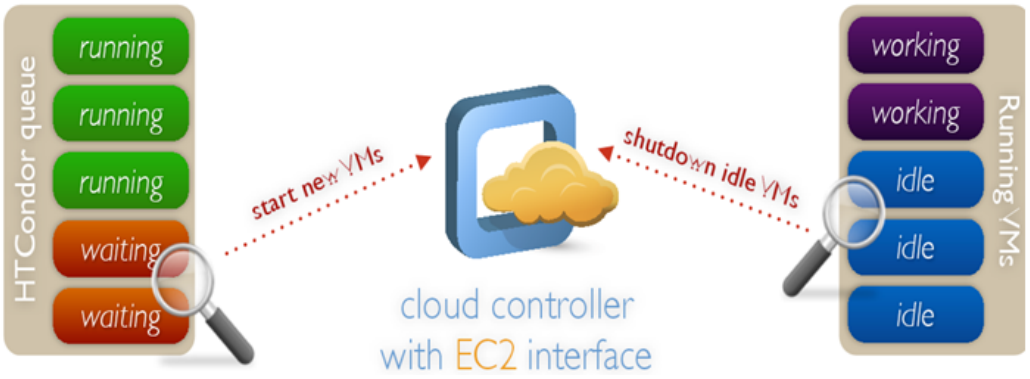
✓ VMs can access the local Tier-2 network

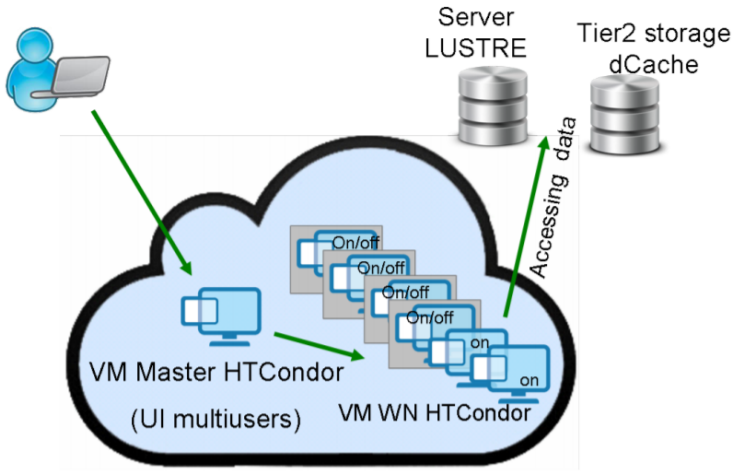
- dCache storage system (> 2 PB) and Lustre file system (~ 80 TB)



CMS use case/2

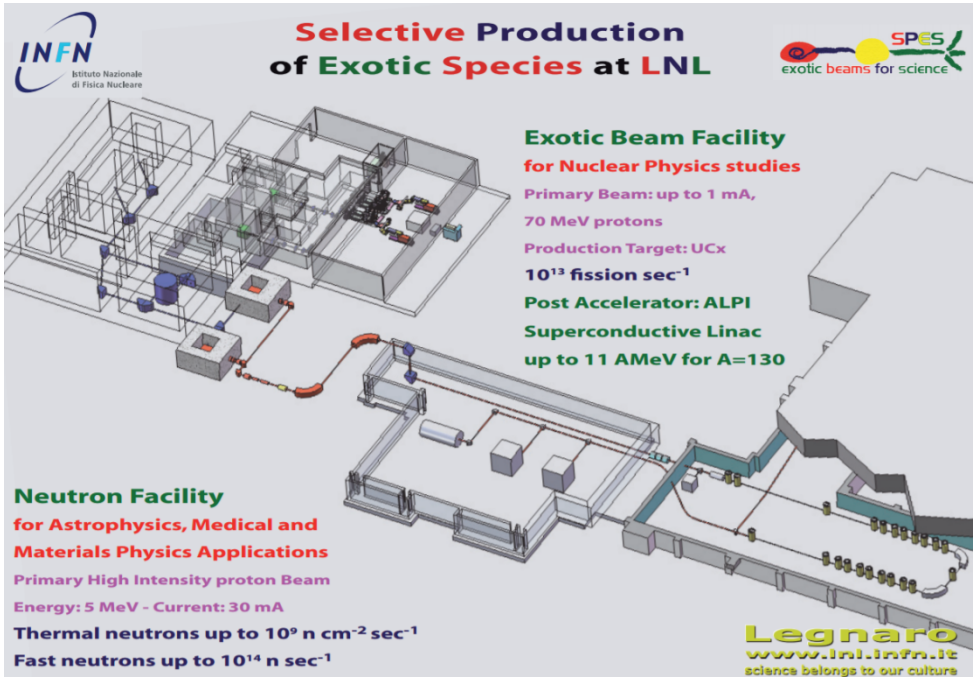
- Batch usage:
  - ✓ Elastic HTCondor cluster created and managed by ***elastiq***
    - lightweight Python daemon that allows a cluster of VMs running a batch system to scale up and down automatically
    - Scale up: if too many jobs are waiting, it requests new VMs
    - Scale down: if some VMs are idle for some time, it turns them off
  - ✓ Used to generate 50k toy Monte Carlo followed by unbinned ML fits for the study of $B_0 \rightarrow K^*\mu\mu$ rare decay
    - ~ 50k batch jobs in the HTCondor elastic cluster
    - up to 750 simultaneous jobs on VMs with 6 VCPUs

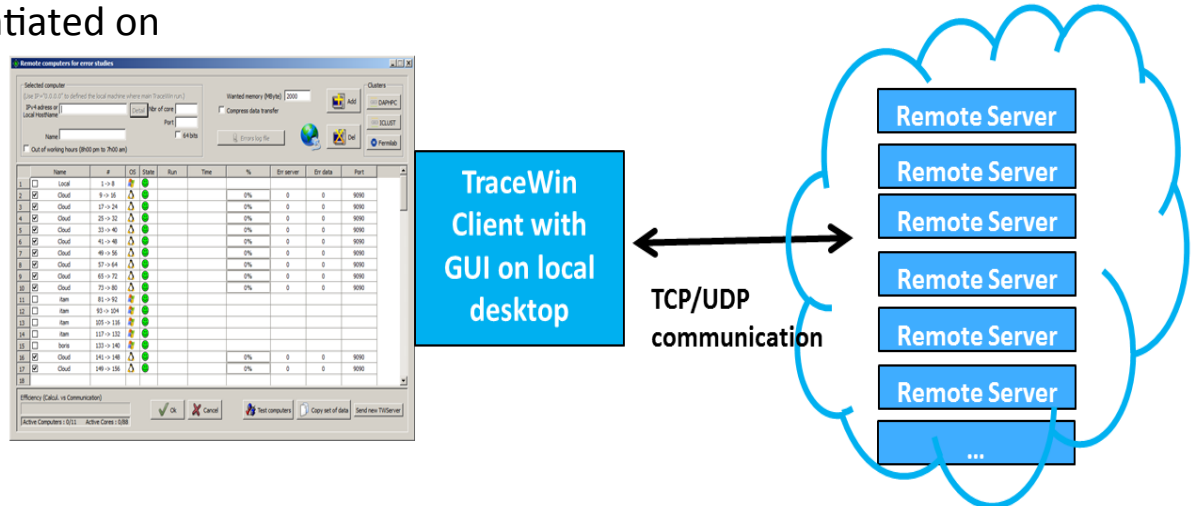




SPES use case

- Beam Dynamics characterization of the European Spallation Source - Drift Tube Linac (ESS-DTL)
- Monte Carlo simulations of 100k different DTL configuration, each one with 100k macroparticles
  - ✓ Configurations split in groups of 10k
  - ✓ For each group 2k parallel jobs running on the cloud in batch mode
  - ✓ TraceWin client-server framework
  - ✓ TraceWin clients elastically instantiated on the cloud receive tasks from the server
  - ✓ Up to 500 VCPUs used simultaneously
  - ✓ Results obtained on the cloud reduced the design time of a factor 10





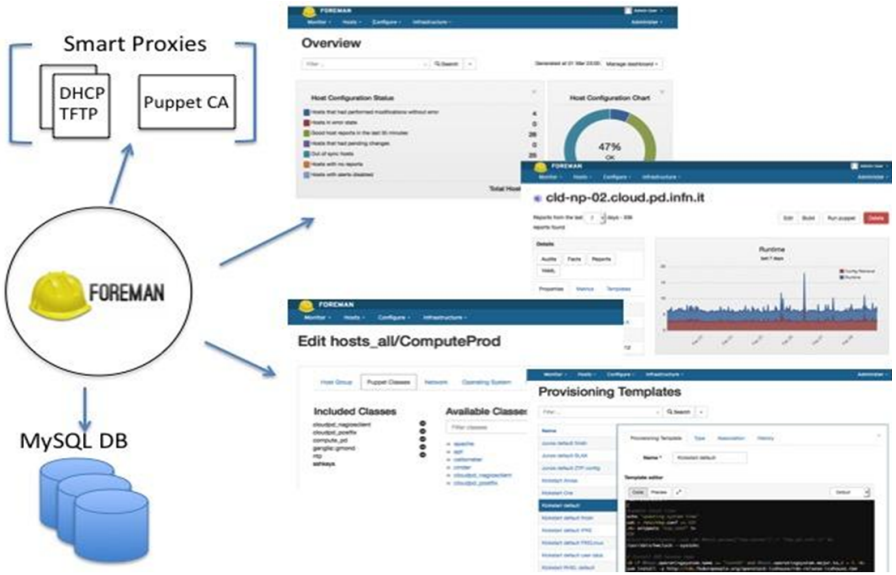
Lessons learned/1

- Properly evaluate where to deploy the services
  - ✓ in particular don't mix storage servers with other services
  - ✓ initial configuration:
    - 2 nodes configured as controller nodes
    - 2 nodes configured as network nodes + storage (Gluster) servers
  - ✓ current deployment:
    - 2 nodes configured as controller nodes + network nodes
    - 2 nodes configured as storage (Gluster) servers
- Database is a critical component
  - ✓ started with Percona cluster deployed on 3 VMs, then moved to physical machines for performance reasons
  - ✓ using different primary servers for different services (e.g. glance, cinder)

Lessons learned/2

- Evaluate pros and cons of live migration
  - ✓ scalability and performance problems found by using a shared file system (GlusterFS) to enable live migration
  - ✓ however live migration is really a must only for few of our applications
  - ✓ Moved a different set up:
    - Most compute nodes use their local storage disks for Nova service
    - Only a few nodes use a shared file system → targeted to host critical services, and exposed in a ad-hoc availability zone

- Any manual configuration should be avoided
  - ✓ combined use of Foreman + Puppet as infrastructure manager
  - ✓ not only to configure OpenStack, but also the other services (e.g. ntp, nagios probes, ganglia, etc)



Lessons learned/3

- Monitoring is crucial for a production infrastructure
  - ✓ based on Nagios, Ganglia and Cacti
  - ✓ in particular Nagios heavily used to prevent/early detect problems
    - Sensors to test all OpenStack services, registration of new images, instantiation of new VMs and their network connectivity, etc.
    - Most sensors available on internet, some other more specific of our infrastructure were implemented in-house

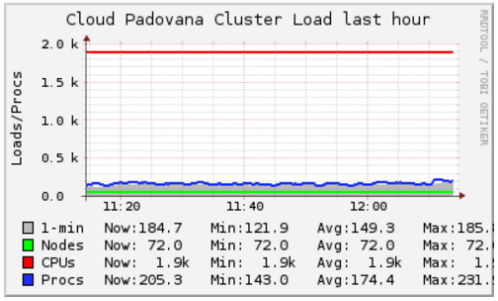
cld-nl-02	KVM ?	OK	03-01-2017 16:00:01	103d 7h 14m 2s	1/1	hosts:12 OK:12 WARN:0 CRIT:0 - instance-0000449a:running instance-000393cf:running instance-0007ceb1:running instance-0007cef9:running instance-000734e5:running instance-0005e90e:running instance-0005eb06:running instance-0009aabb:running instance-0009ab7e:running instance-000aba38:running instance-000ac42b:running instance-000acefc:running
	Nova Compute	OK	03-01-2017 16:13:37	23d 5h 11m 1s	1/4	PROCS OK: 1 process with command name 'nova-compute'
	Nova Partition	OK	03-01-2017 16:13:42	105d 0h 55m 30s	1/4	DISK OK - free space: /var/lib/nova/instances 1827669 MB (95% inode=99%):
	OpenvSwitch ?	OK	03-01-2017 16:10:02	103d 7h 44m 1s	1/1	• openvswitch.service - Open vSwitch: Loaded: loaded (/usr/lib/systemd/system/openvswitch.service: enabled: vendor preset: disabled): Active: active (exited) since Wed 2016-11-16 15:18:41 CET: 3 months 13 days ago: Process: 1677 ExecStart=/bin/true (code=exited, status=0/SUCCESS): Main PID: 1677 (code=exited, status=0/SUCCESS): CGroup: /system.slice/openvswitch.service::Nov 16 15:18:41 cld-nl-02.cloud.pd.infn.it systemd[1]: Starting Open vSwitch...:Nov 16 15:18:41 cld-nl-02.cloud.pd.infn.it system
	PING	OK	03-01-2017 16:09:09	11d 0h 50m 37s	1/2	PING OK - Packet loss = 0%, RTA = 0.70 ms
	Root Partition	OK	03-01-2017 16:12:55	105d 0h 54m 34s	1/4	DISK OK - free space: / 255587 MB (99% inode=99%):
	SSH	OK	03-01-2017 16:13:42	105d 4h 0m 0s	1/2	SSH OK - OpenSSH_6.6.1 (protocol 2.0)
	VM network	OK	03-01-2017 11:45:45	42d 4h 28m 16s	1/2	VM cld-nl-02.cloud.pd.infn.it-2017-03-01-11:45:48 successfully created, pinged and deleted

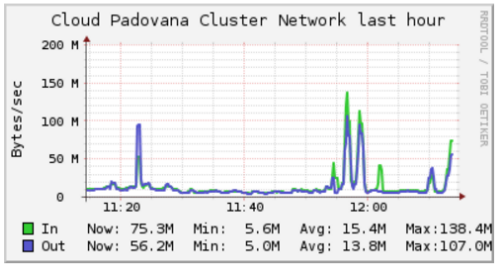
Infrastructure monitoring

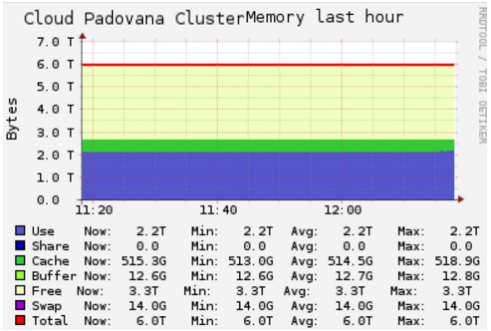
Cloud Padovana (physical view)

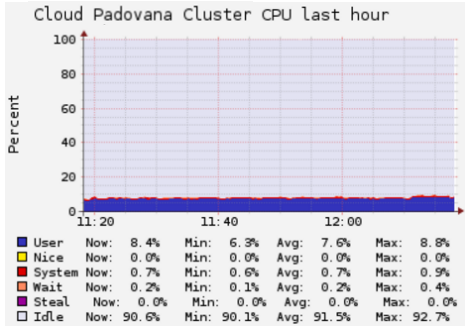
CPUs Total:	**1906**
Hosts up:	**72**
Hosts down:	**0**

Current Load Avg (15, 5, 1m):
8%, 9%, 10%
Avg Utilization (last hour):
8%
Localtime:
2017-03-03 11:14







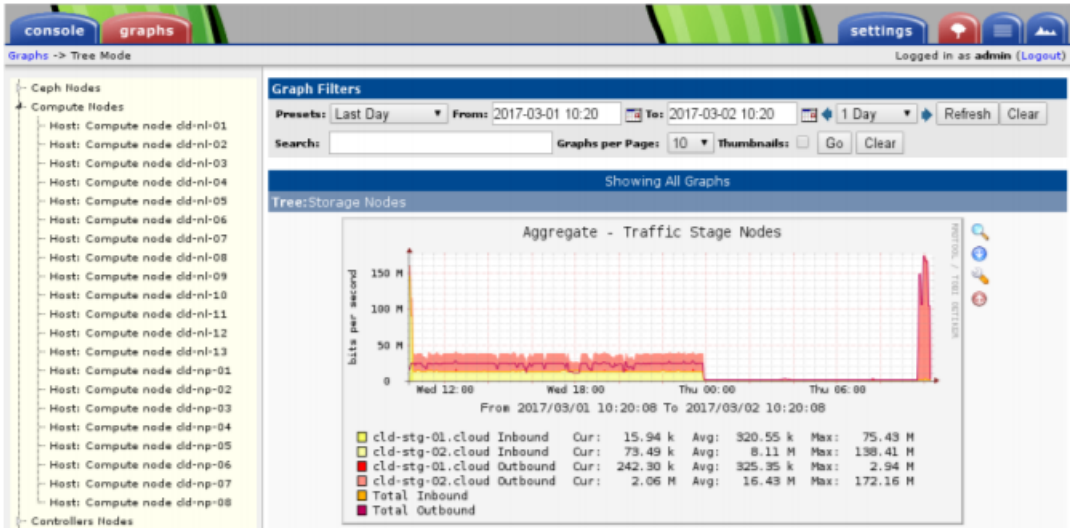


Ganglia

✓ For CPU, memory, disk space, network usage of all physical and virtual servers

✓ Specific for network related information

cacti



Lessons learned/4

- Security auditing is challenging in cloud environment
  - ✓ Even more complex for our peculiar network set up
  - ✓ Typical security incident: something bad originated from IP a.b.c.d at time YY:MM:DD:hh:mm
  - ✓ A procedure was defined to manage security incidents:
    - Given the IP a.b.c.d, to find the VM private IP
    - Given the VM private IP, to find the MAC address
    - Given the VM MAC address, to find the UUID
    - Given the VM UUID, to find the owner
  - ✓ The above workflow is possible by using specific tools (netfilter.org ulogd, CNRS os-ip-trace) and archiving all the relevant log files
  - ✓ It allows to trace any internet connection initiated by a VM on the cloud, even if in the meantime it was destroyed

Lessons learned/5

- OpenStack updates must be properly managed
  - ✓ Every change done in the production cloud is first tested and validated on a dedicated testbed
  - ✓ This is a small infrastructure resembling the production one:
    - two controller/network nodes where service are deployed in HA
    - a Percona cluster
    - Nagios monitoring sensors active to immediately test the applied changes
  - ✓ We are currently running OpenStack Mitaka version (EOL 2017-04-10)
  - ✓ Plans for updating to Ocata version by the end of 2017 (skipping the Newton release)
  - ✓ Choice made for keeping the right balance between offering the latest features and fixes and the need of limiting the manpower effort

Next steps/1

- The Cloud Area Padovana keeps evolving in terms of provided resources and offered services
- Foreseen future activities:
  - ✓ Simplify authentication by integrating IdPs through OS-Federation
  - ✓ Adding support for user account renewal (per project)
  - ✓ To deploy a CEPH based storage service, to be used for all cloud needs
  - ✓ To deploy Synergy service, to allow efficient resource sharing among user groups limiting the need of static partitioning (→ see next talk)
  - ✓ To integrate Cloud Area Padovana with the Cloud infrastructure owned by the University of Padova (CED-C) → **cloudveneto.it**

Next steps/2

- CED-C is in production since November 2015
- Is hosted at INFN Padova data center besides CAP
  - ✓ 50+ users grouped in 26 projects from 10 University departments
  - ✓ 240 physical cores → 480 cores in HT → 1920 VCPUs available for VMs (overcommitment = 4)
  - ✓ 68 TB available for permanent storage volumes
  - ✓ 19 TB for ephemeral VM storage and VM images
- The unified cloud aims to become a reference infrastructure for scientific computing at regional level

cloudveneto.it

Thanks for your attention.

Questions?

The Cloud Area Padovana Team

INFN-Padova

Paolo Andreetto
Fabrizio Chiarello
Fulvia Costa
Alberto Crescente
Alvise Dorigo
Federica Fanzago
Ervin Konomi
Matteo Segatta
Massimo Sgaravatto
Sergio Traldi
Nicola Tritto
Marco Verlato
Lisa Zangrando

INFN-LNL

Sergio Fantinel