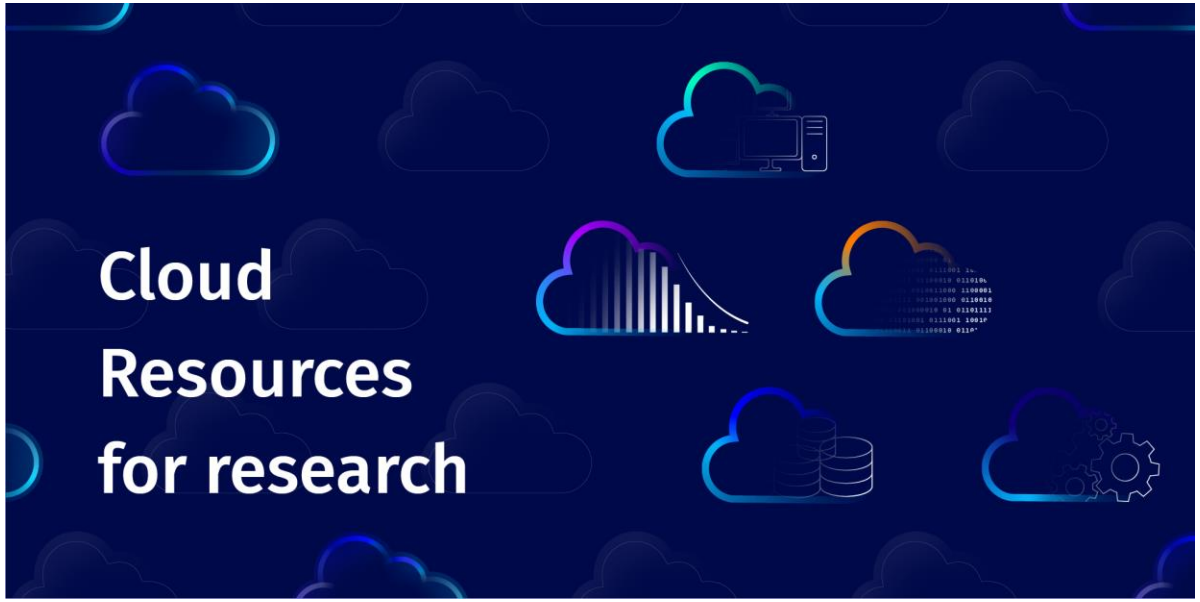
INFN Cloud Monitoring and Accounting services

Alessandro Costantini (acostantini@infn.it)
Vincenzo Spinoso (spinoso@infn.it)
On behalf of the **INFN Cloud Team** (cloud@lists.infn.it)

Highlights

- INFN Cloud Monitoring
  - Objective
  - Infrastructure level services
  - User level services
- INFN Cloud Accounting
  - Objective
  - Solutions and services
- What next



INFN Cloud Monitoring

Objectives

- Store information related to monitored services
- Identify anomalies with respect to desired situations
  - And perform related actions and countermeasures
- Keep a history
  - to detect recurring anomalies
  - allow a retrospectively data analysis
- Representing performance metrics

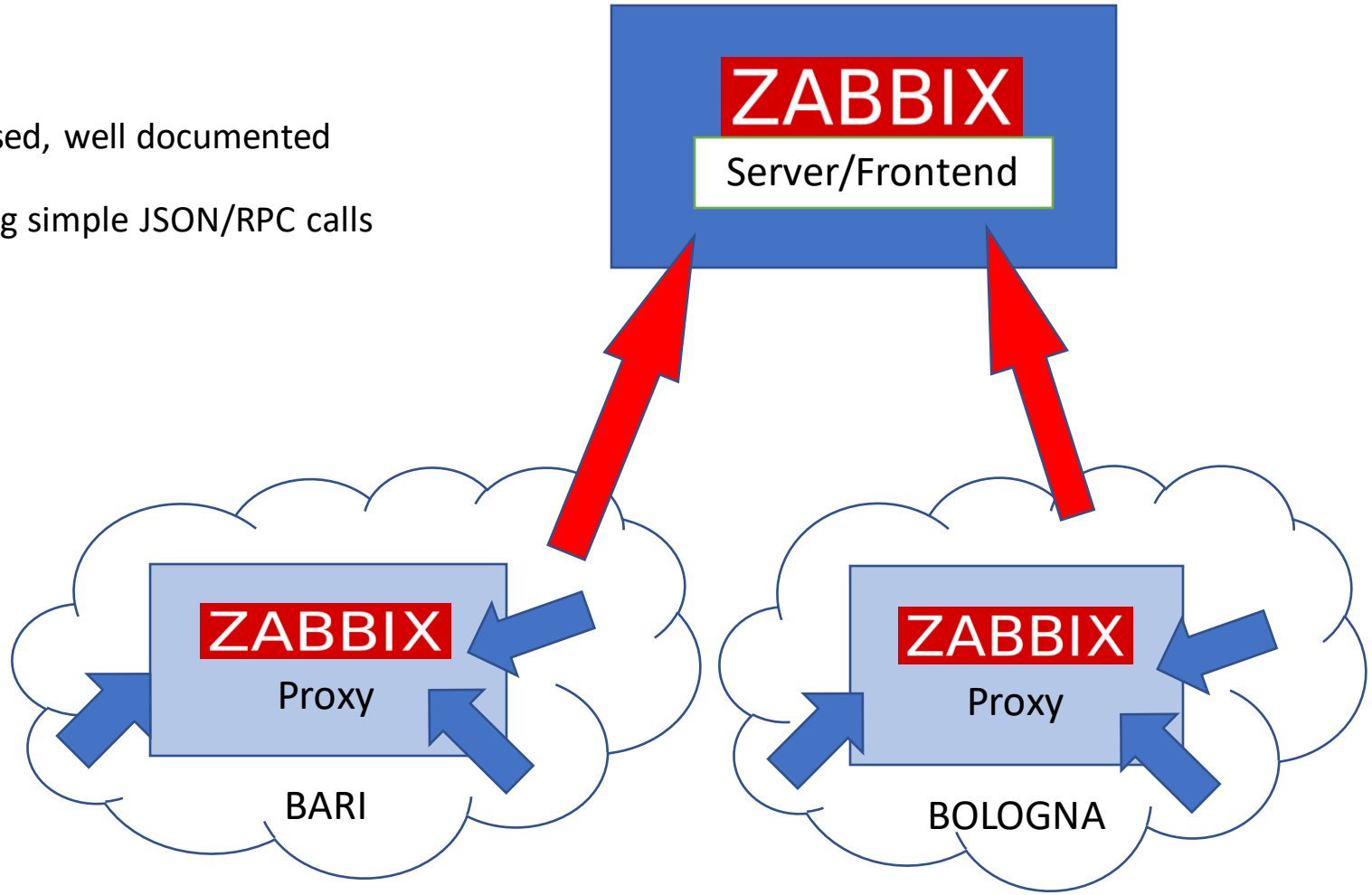
Backbone monitoring

Zabbix

- Robust, supported, mature, reliable, widely used, well documented
- Zabbix provides APIs
  - Monitoring can be easily developed using simple JSON/RPC calls
- Agents integrated with Puppet/Foreman

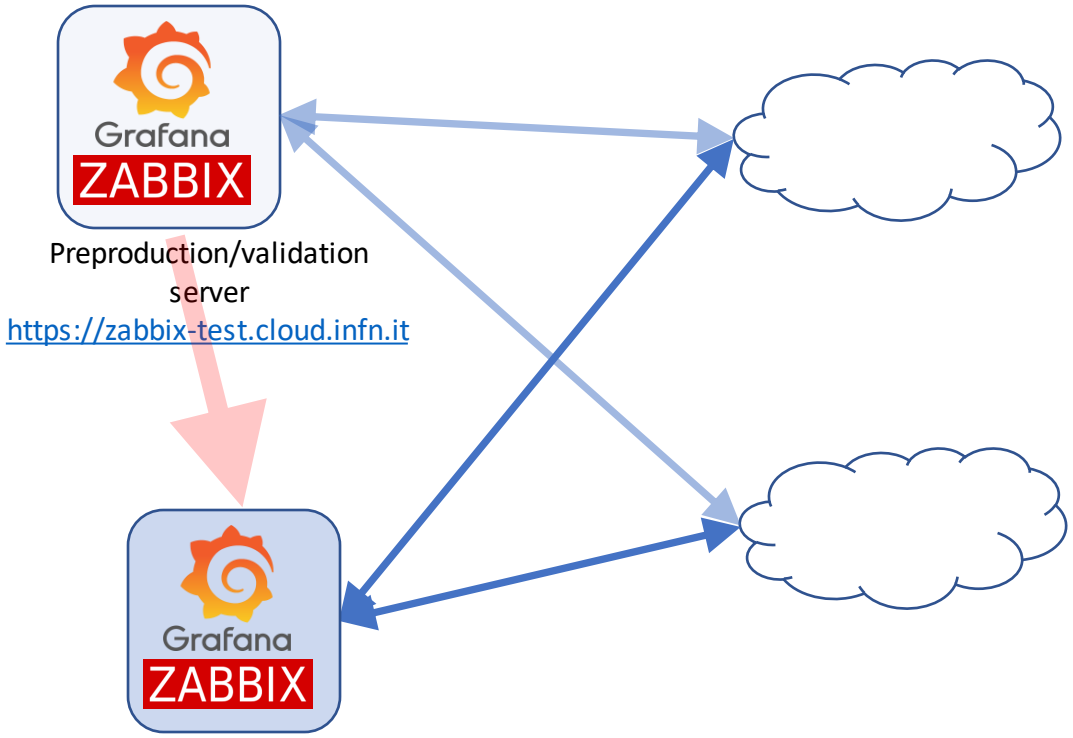
Architecture

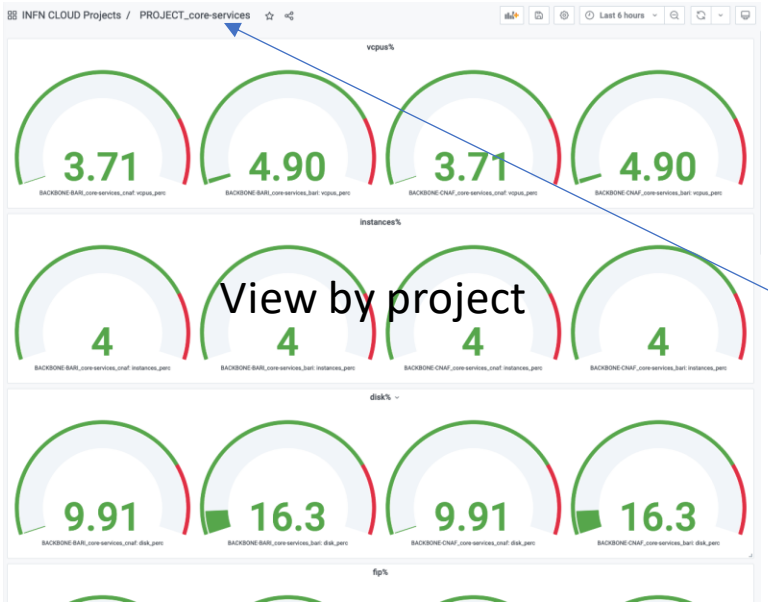
- Proxies send the metrics to the central server, exposing the Zabbix frontend
- Notifications via mailing list and messaging systems (Microsoft Teams as of now)
- Host metrics (CPU, memory, disks, network…) and service metrics (CEPH, DNS, other basic services)
- Some other metrics in progress



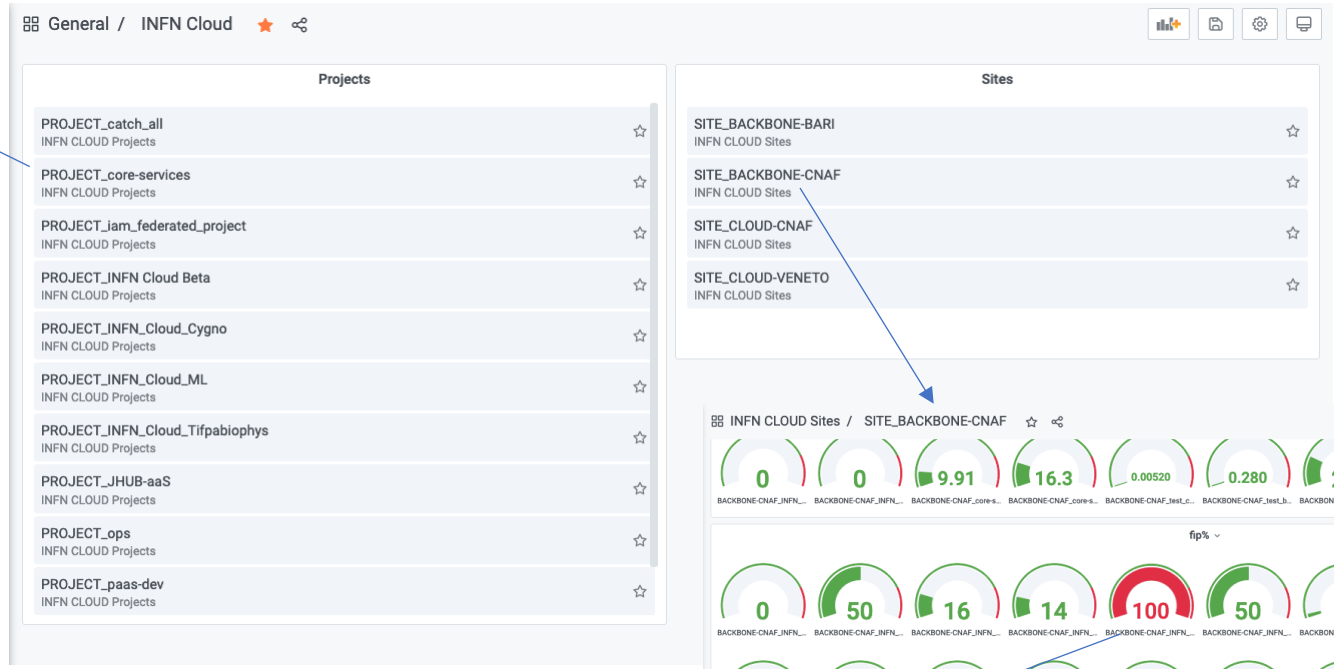
INFN Cloud monitoring

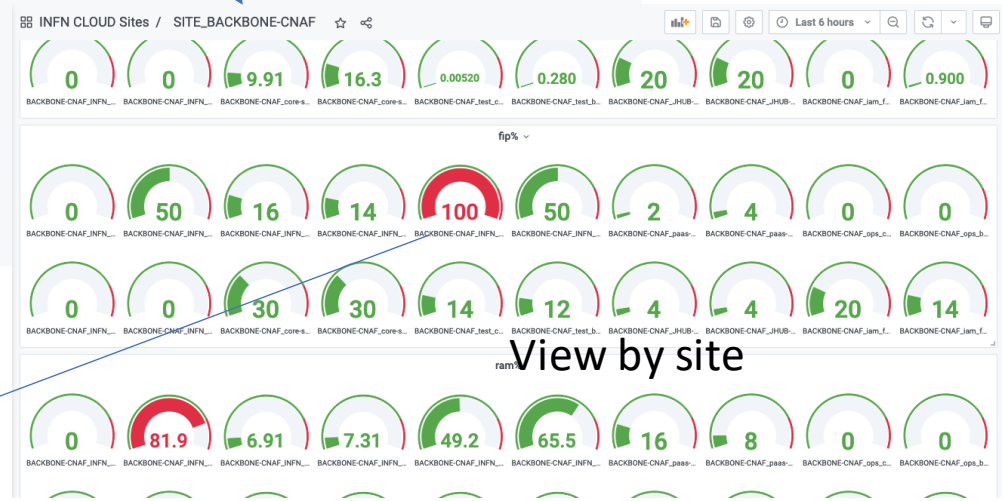
- A separated instance of Zabbix exposed over the Internet
- Provides monitoring for
  - All the INFN Cloud PaaS «building blocks»
  - Federated clouds (VM cycle probe in production) → A/R computation (in progress)
  - Capacity metering ensures project resource quota allocation is properly monitored
    - i.e. VCPUs, instances, RAM, floating IPs, block storage
      - by site
      - by project
- Zabbix collects the metrics, Grafana exposes the aggregated graphs/gauges
  - All automated (using OpenStack/Zabbix/Grafana APIs, code based on python3/requests)
  - All authenticated via Indigo-IAM
- Notifications use Teams channels
  - Subscription model: you add yourself to a channel to get specific notifications



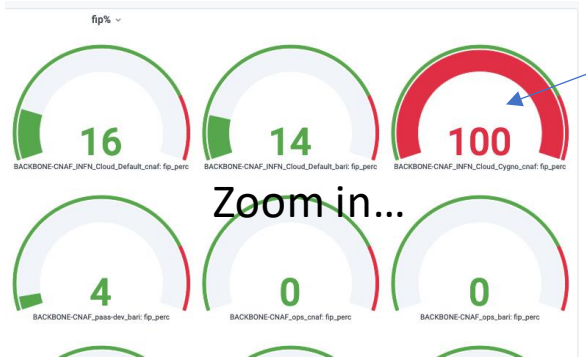


View by project





View by site



Zoom in…

INFN Cloud Status

https://status.cloud.infn.it/

1. INFN Cloud

Object Storage	Operational
Backbone - Cloud Compute (Bari)	Operational
Backbone - Cloud Compute (CNAF)	Operational
Authentication	Operational

Backbone

Federated clouds

2. Federated Cloud - CloudVeneto

CloudVeneto - Cloud Compute	Operational

3. Federated Cloud - ReCaS-Bari

RECAS-BARI – Cloud Compute	Operational

4. Federated Cloud - Cloud@CNAF

Cloud@CNAF - Cloud Compute	Operational

The INFN Cloud status exposes
the overall status of the federation to the users

PaaS

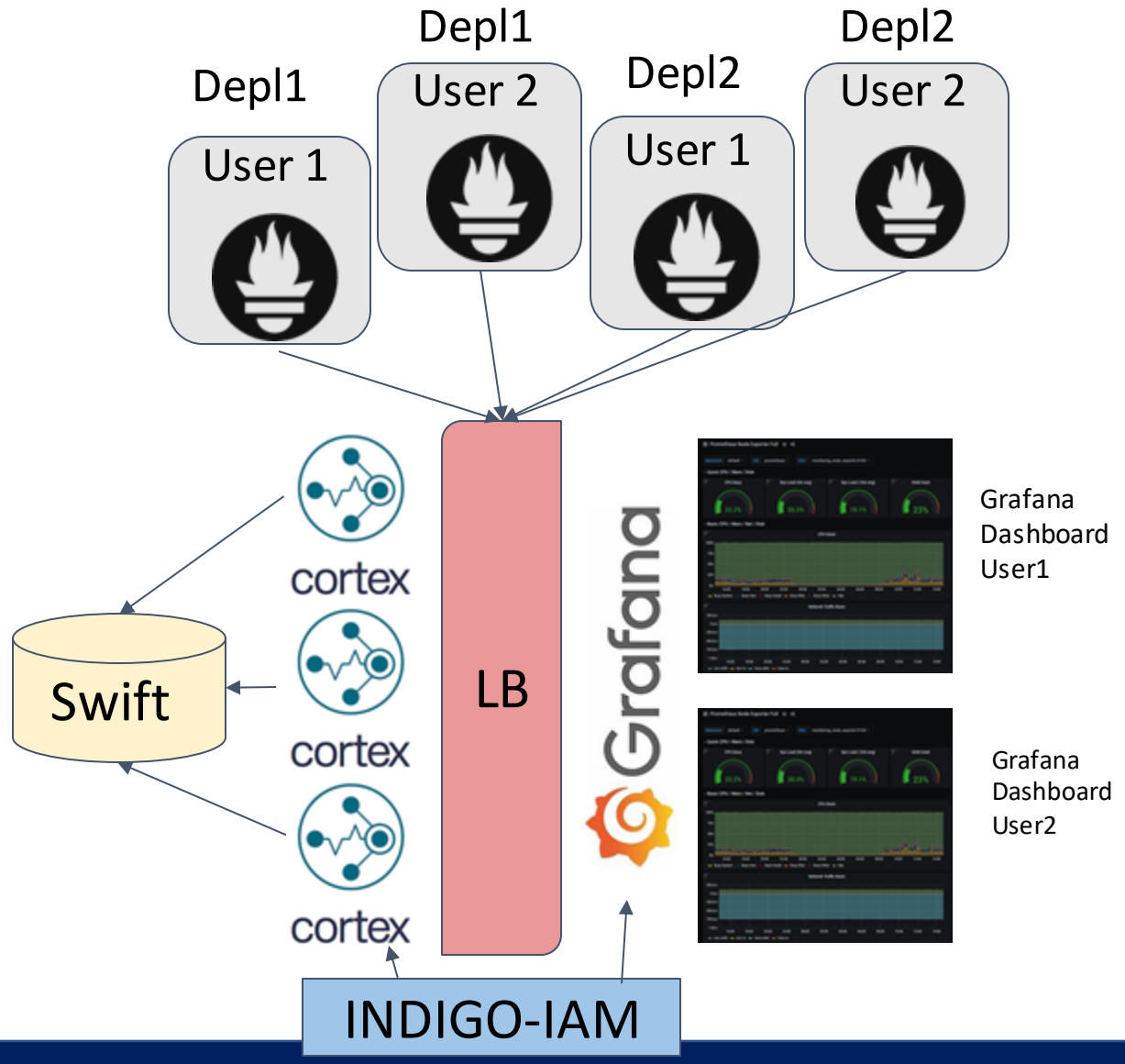
5. PaaS services

Infrastructure Manager	Operational
Orchestrator	Operational
CPR	Operational
CMDB	Operational
Dashboard	Operational

[*] based on https://staytus.co/

User level monitoring

- Objective
  - Find a solution enabling users to monitor their deployments
  - Integrated with INDIGO-IAM
  - Integrated with the solutions proposed by the PaaS and the backbone
- How it works
  - Prometheus is collecting the metrics from the deployments
  - Cortex is receiving the data, stored in Swift, and implements multitenancy
  - Grafana then visualize the data from cortex, showing only those related to the authenticated user



INFN Cloud Accounting

Objectives

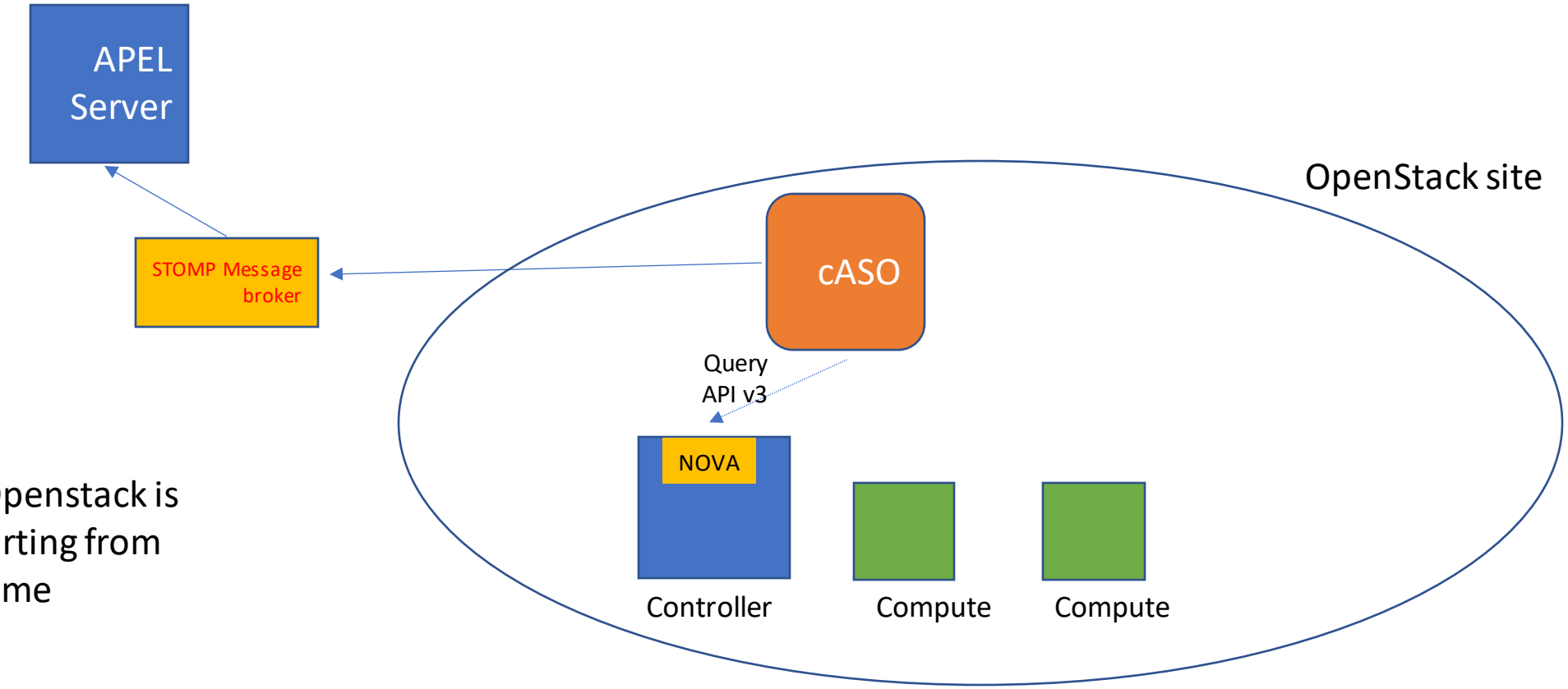
- Collect the usage of resources from users and groups

Architecture

- Based on EGI accounting service
- Client-Server
  - Server: based on APEL accounting tool, RabbitMQ as message broker and Grafana for data visulization
  - Client: based on Collectd/cASO for data collection and APEL-SSM (based on STOMP) to send messages
- Status
  - Implemented in the backbone and in federated clouds
    - CLOUD-VENETO
    - CNAF and RECAS BARI ongoing
- Collected metrics
  - wallclocktime, cputime, disk usage, instances, etc.

INFN Cloud Accounting

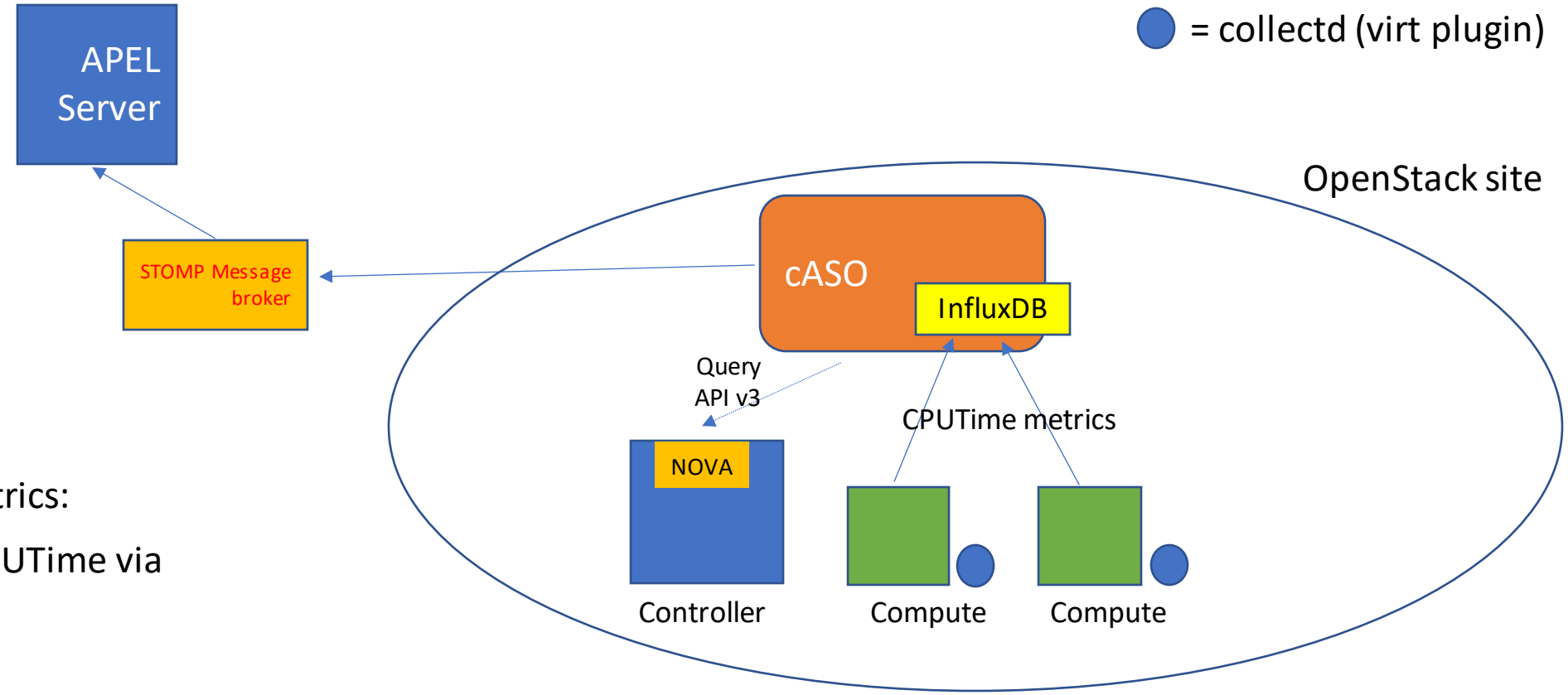
Original Architecture (Openstack site)



CPUTime in Openstack is calculated starting from the VM walltime

INFN Cloud Accounting

Original Architecture (Openstack site)

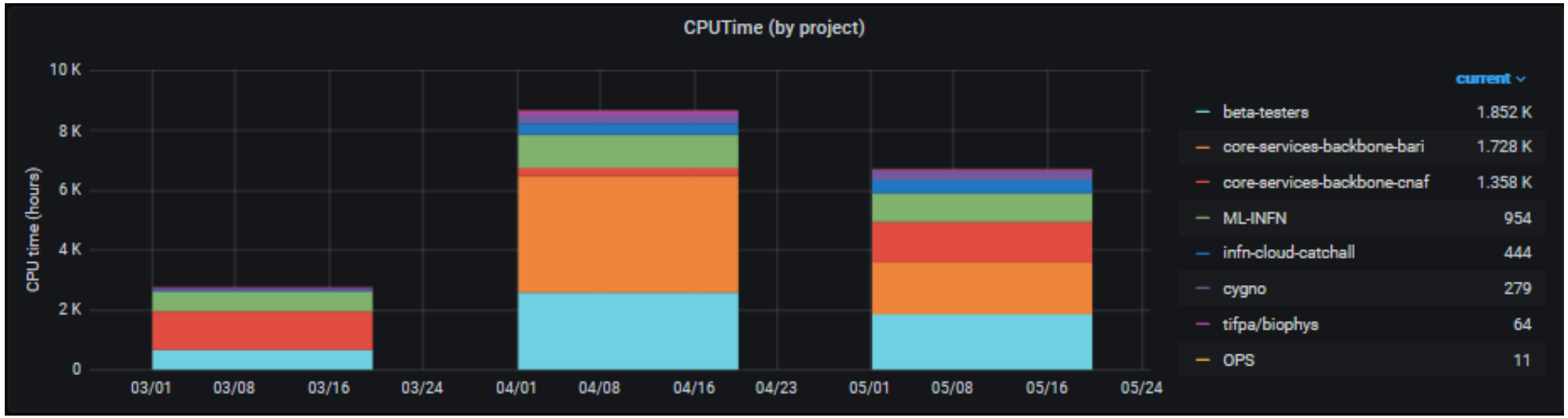


Improving metrics:

Getting the CPUTime via collectd

INFN Cloud Accounting

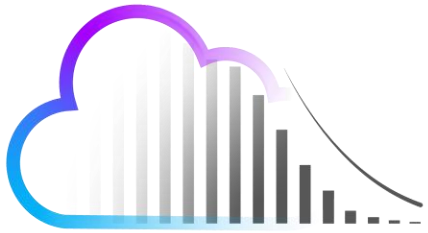
- Data visualization
  - Garafana used for data aggregation and visualization
    - Access via INDIGO-IAM
    - Dashboard showing data aggregated for groups and providers (open to the public)
    - Dashboard showing all collected values (used by the support team)



Future work

- Improve the **robustness** of the services
  - Zabbix HA implementation relying on frontend/server/DB replication
- Improve visualization (user/project views, support view…)
- Evaluate the Operational Level Agreement fulfilment with the federated clouds
  - Add and improve probes to allow testing of the federated clouds, implement A/R evaluation
- Implement user level services monitoring
- Define and implement a user level service accounting

Computer Service



Analytics

Scientific Community Customizations

Machine Learning

Data service