Federating and orchestrating resources across the EOSC infrastructures

Marica Antonacci (INFN)
Giacinto Donvito (INFN), Michele Perniola (UNIBA, INFN)

EGI Conference 2021
19-22 October - Virtual

Outline

- Objectives
- INDIGO PaaS Overview
- Main new functionalities
- Future work and conclusions

Objectives

- **Gather together multiple and heterogeneous service providers** providing a unified view on a heterogeneous pool of resources while using a single access point to control user deployments
- **Deliver a context-aware and customized federation** of compute and data resources, which can dynamically match the user requirements with the available providers
- **Provide high-level composable services**, deployable on demand through simple interfaces, limiting the direct intervention of the user to the definition of policies and requirements that should guide the deployment

The INDIGO PaaS

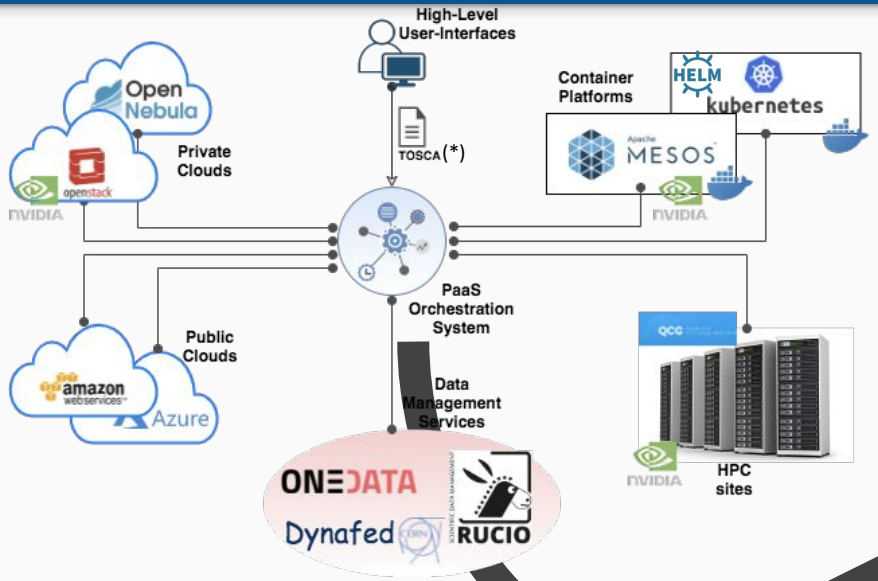
- It allows to coordinate the provisioning of virtualized compute and storage resources on different **Cloud Management Frameworks** (like OpenStack, OpenNebula, AWS, etc.) and the deployment of dockerized services and jobs on **Mesos clusters** and **Kubernetes clusters**.
- The development started during the European H2020 project "**INDIGO-DataCloud**" and continued during the following projects DEEP-Hybrid DataCloud, eXtreme-DataCloud and EOSC-Hub
  - Evolving the functionalities to **TRL8**
  - Ensuring the **scalability** and **performance** of the developed solutions
  - Providing relevant contributions to the **EOSC**
- **Further improvements are being designed and implemented in the framework of the INFN Cloud project, EGI-ACE, C-SCALE, IoTwins, etc.**

Foundations and key enablers

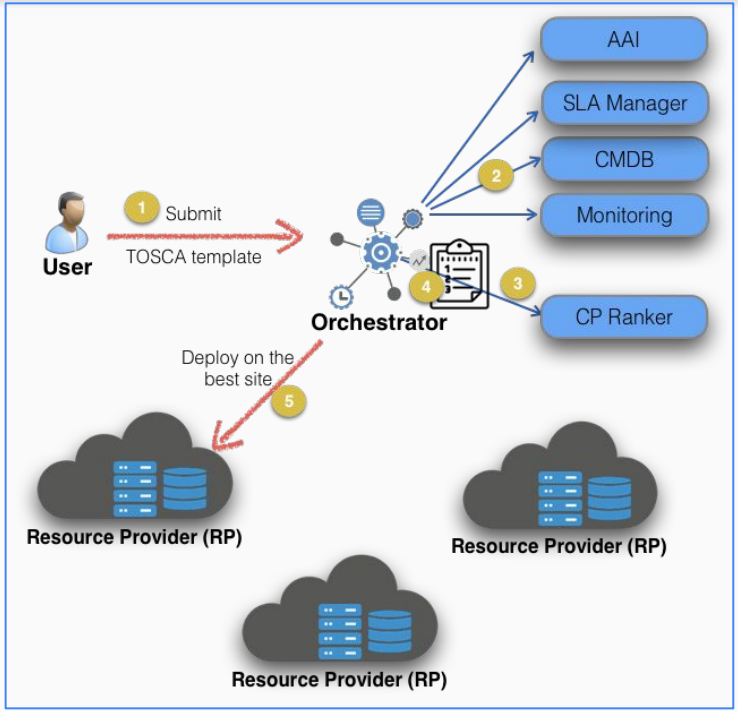
- Develop and integrate **open-source** and **open standard-based** components to ensure portability and interoperability
- Adopt a **modular micro-service** architecture
- Use **Infrastructure as Code** (IaC), DevOps and **containers** to reduce manual processes and increase flexibility and portability across environments
- Leverage **federated Authentication and Authorization** technologies based on **OpenID-Connect** (supporting also legacy AAI solutions)
- Enable flexible **service composition** and **re-use**

PaaS Orchestration System

High-level architecture



(*) Topology and Orchestration Specification for Cloud Applications Ref: TOSCA Simple Profile in YAML Version 1.1



The Orchestrator interacts with the provider services through:

- the **Infrastructure Manager** for deploying complex and customized virtual infrastructures on multiple IaaS Cloud backends (Openstack, AWS, etc.)
- direct APIs for deploying dockerized workloads on container platforms

Deployment retry strategy

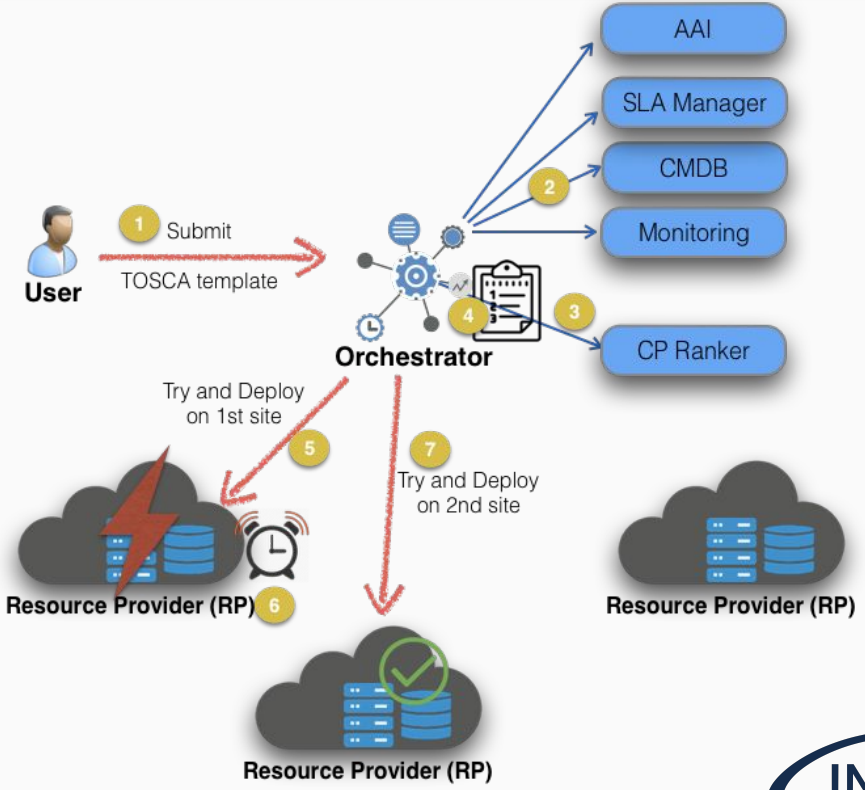
The Orchestrator implements a **trial-and-error** mechanism that allows to **reschedule** the deployment on the next available cloud provider from the list of candidate sites.

Example: the deployment fails due to a runtime error on the chosen site

The mechanism is able to address also the **timeout** in the deployment creation.

The user can specify

- the maximum time for the single trial at each provider;
- the overall maximum time for the deployment creation (including the possible retries).



Support for specialized hardware requirements

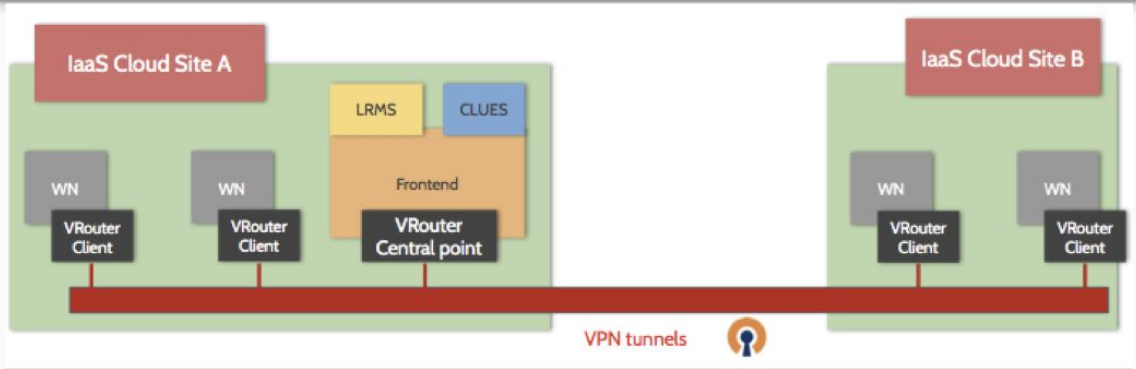
The PaaS layer allows to federate and access specialized hardware resources, mainly **GPUs** and **infiniband**, using high level interfaces.

Both complex clusters of virtual machines and containers can be allocated to sites providing specialized computing hardware devices needed for running time consuming workloads, like the deep learning applications.

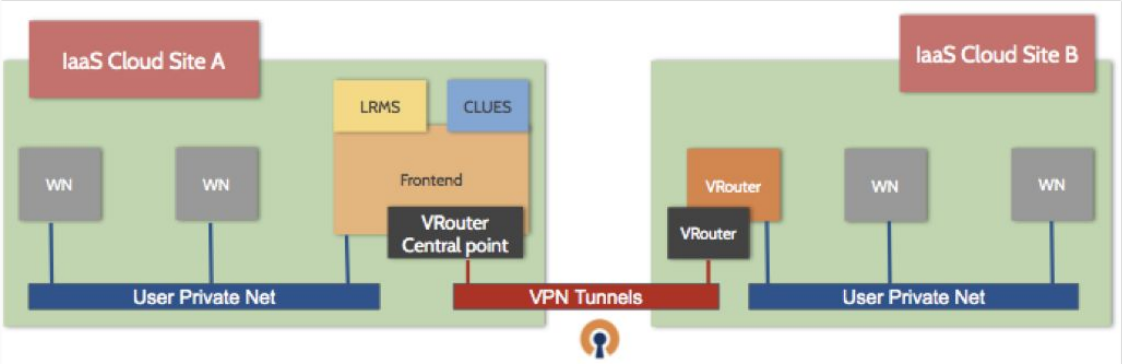
- The information system has been extended in order to collect information about the availability of GPUs and infiniband support at the sites
  - This information is read and consumed by the Orchestrator to select the best site where the resources will be allocated
- The TOSCA model for compute and container nodes has been extended in order to include these requirements

Virtual Networking Orchestration

Scenario I:
exploits private networks already existing at the sites



Scenario II:
a dedicated private network is created for the deployment in both sites



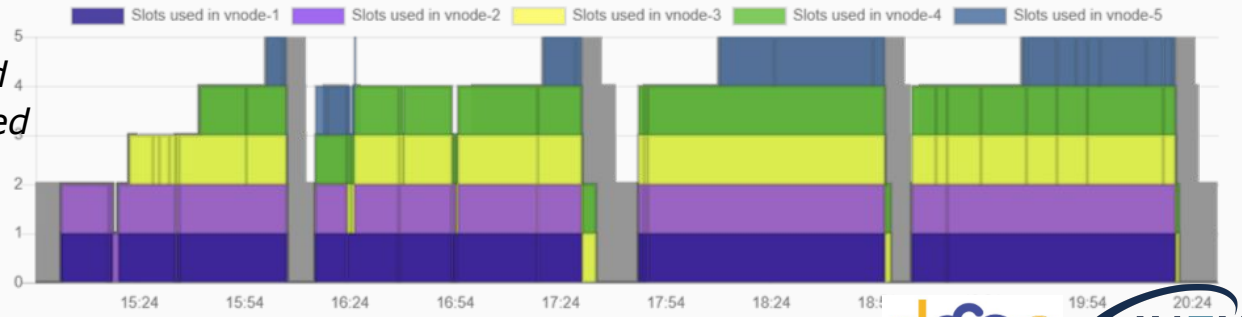
Autoscaling clusters with L2 isolated networks

Slurm cluster deployed on CESNET cloud and AWS

- L2 networks created at both sites and interconnected through vRouter components
- CLUES (Elasticity Manager) used to add/remove nodes in the cluster depending on the workload

Reference Test: 3676 audio files from the Urban Sound Dataset classified with the Audio Classifier model from the DEEP Open Catalog using udocker

"Deployment of Elastic Virtual Hybrid Clusters Across Cloud Sites" published on Journal of Grid Computing

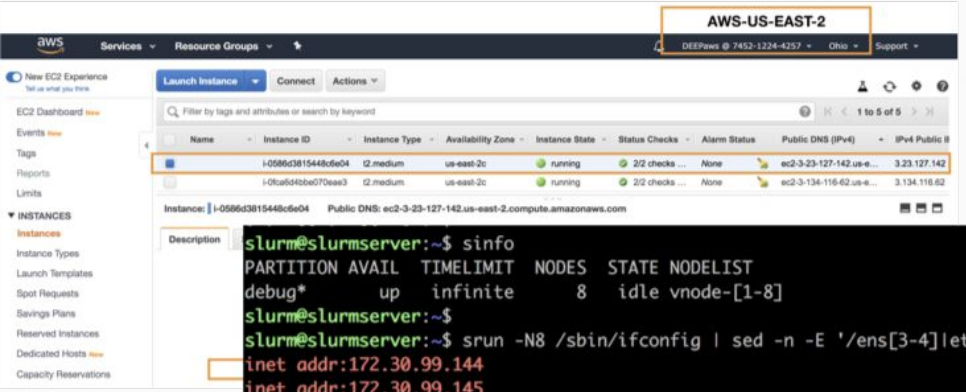


Cluster usage evolution

Scaling across multiple cloud providers

Reference test: Slurm cluster with nodes on 4 sites including AWS

- INFN-BARI
- INFN-CNAF
- IFCA-LCG2
- AWS-US-EAST-2 (Ohio)




```
slurm@slurmserver:~$ sinfo
PARTITION AVAIL  TIMELIMIT  NODES  STATE NODELIST
debug*       up   infinite      8   idle vnode-[1-8]
slurm@slurmserver:~$
slurm@slurmserver:~$ srun -N8 /sbin/ifconfig | sed -n -E '/ens[3-4]|eth[0-1]/{n;p;}' | grep -Eo 'inet addr:[0-9]{1,3}\.[0-9]{1,3}\.[0-9]{1,3}\.[0-9]{1,3}'
inet addr:172.30.99.144
inet addr:172.30.99.145
inet addr:10.10.40.145
inet addr:10.10.40.146
inet addr:172.16.99.11
inet addr:172.16.99.7
inet addr:172.31.47.7
inet addr:172.31.40.223
slurm@slurmserver:~$
```

Support for multiple OIDC identity providers

The PaaS layer has been initially integrated only with the **INDIGO IAM** solution (developed during the same INDIGO-DataCloud project)

This integration remains the reference implementation for managing the users and the authentication/authorization flows throughout the whole stack (from PaaS to IaaS)

- Some details can be found in this presentation from the IAM Users Workshop (27-28 Jan 2021)

Anyway the PaaS has been adapted in order to support multiple and different OIDC Identity Providers, e.g. EGI Checkin

Multi-tenancy support

Initially all the authenticated users were considered part of the same group (organization) → flat structure →same roles, users mapped on the same IaaS project, etc.

A **finer-grained control** has been implemented later - based on the group membership attribute provided in the user oauth token:

- The Orchestrator uses this information to authorize the user to perform admin operations
- At IaaS level the user is mapped onto a dedicated project (Openstack tenant, Kubernetes namespace, etc.)
- Multiple groups membership is managed at the Orchestrator level (the user can specify the current “active group”)

Kubernetes provider integration

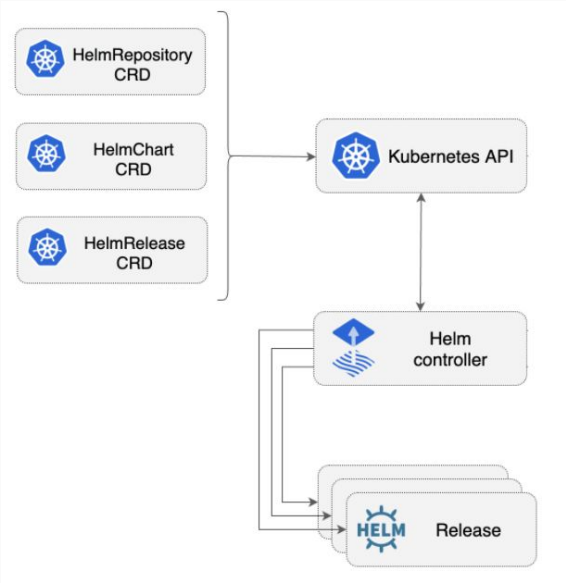
Integration requirements:

- The k8s cluster must support one of the OIDC IdP trusted by the PaaS
  - Fully tested with INDIGO IAM
- RBAC enabled and proper namespaces created
- FluxCD Helm operator installed

Deployment Workflow:

The Orchestrator interacts with the k8s APIs to create and monitor the helm release (through the helm operator)

The deployment is created in the proper namespace (depending on the user group) and is tagged with a unique deployment id generated by the Orchestrator.



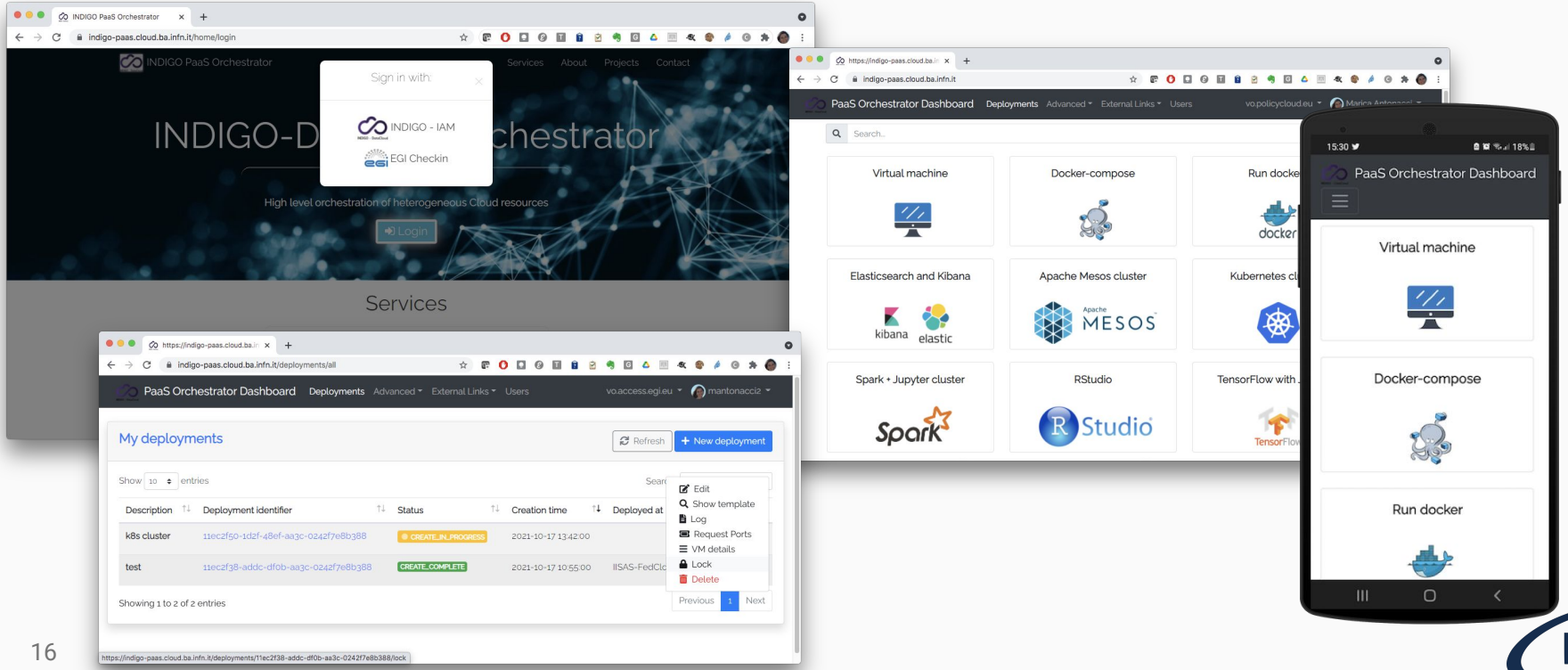
Data Placement and Orchestration

- **Data-aware scheduling**: the INDIGO Orchestrator is able to select automatically the best computing site based on the user data location
  - Data placement plugins available for Onedata and Dynafed

- **Data movement and orchestration**: Rucio, the data management system developed by Cern (initially for the ATLAS experiment), has been integrated in the INDIGO Orchestrator as a plugin to be used to steer the data movement
  - support workflows for data pre-processing at ingestion

High-level user interfaces

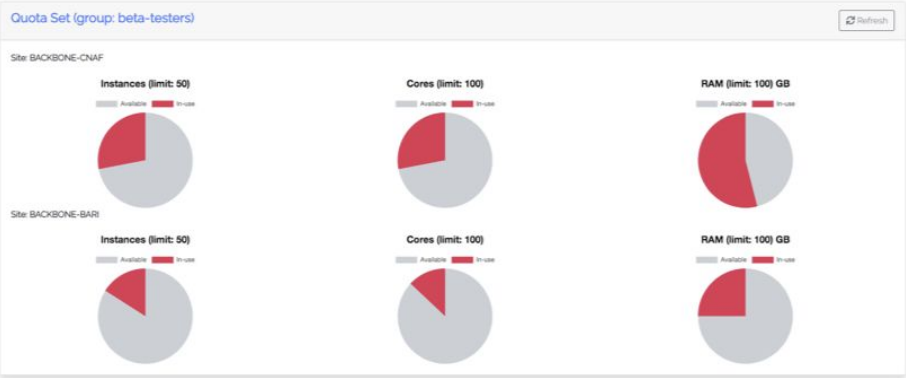
Empower users lowering the access barriers

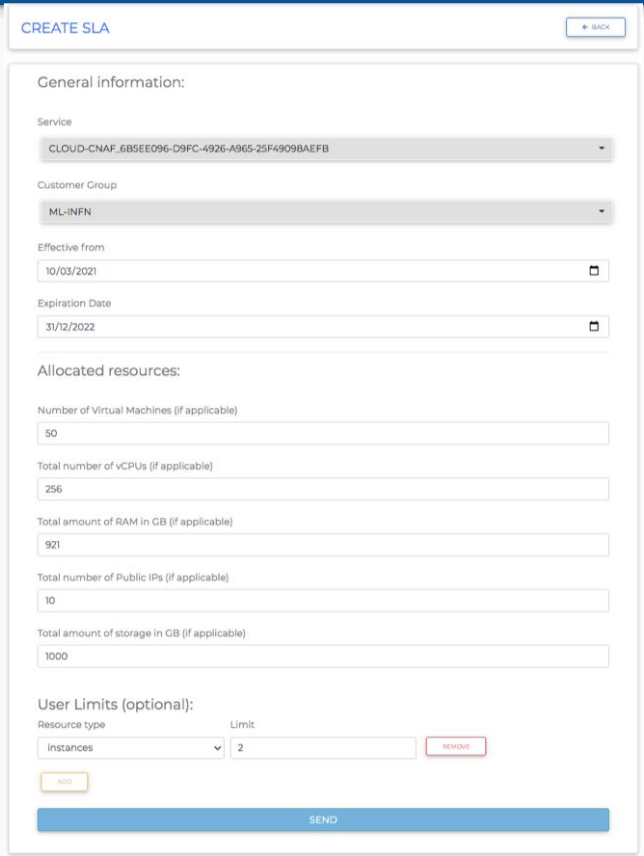


A new tool for SLAs configuration - WIP

SLAT (Service Level Agreement Tool)

- The development started during the EOSC-Hub project and is continuing in INFN Cloud project
- It allows to describe the SLAs for each user group
- New features will be available soon, e.g. user limits and dynamic quota management.





Conclusions

- The INDIGO PaaS Orchestrator service is available through the **EOSC Marketplace**
  - Authentication/Authorization via INDIGO IAM and EGI Checkin
    - ensuring also the integration with **EGI FedCloud** sites
  - New use-cases are being implemented in the framework of the **EGI-ACE** project
- The INDIGO PaaS Orchestrator has been already adopted and used in diverse **production** services:
  - INFN is currently exploiting the INDIGO PaaS capabilities for the **INFN Cloud Infrastructure** federating the resources provided by different INFN sites distributed across Italy
  - Laniakea@ReCaS (Elixir Italy), a service to provide Galaxy “on-demand” instances over heterogeneous cloud infrastructures
- New developments and integrations will be carried out during the on-going projects, like **C-SCALE** to support EO use-cases

Thank you for your attention!

indigo-paas-support@lists.infn.it

https://indico.egi.eu/event/5720/