

# WP1 – Infrastructure and Resource Provisioning

Stefano dal Pra, CNAF  
[stefano.dalpra@cnafe.infn.it](mailto:stefano.dalpra@cnafe.infn.it)



# Introduction

WP1 aims at developing and support an infrastructure, composed of **hardware and software** components, to empower the development of *machine-learning techniques* with **cloud-native technologies**.

Special attention is devoted to the **provisioning of hardware accelerators in INFN Cloud**, and in particular of **GPUs**.

AI\_INFN inherits from ML\_INFN a cluster in *Cloud@CNAF* with multiple GPU models (T4, RTX5000, A100) perfect for **development** and mid-scale tests. **Scalability via offloading will be studied**.

# Where we stand now?

ML\_INFN has been operating since June 2023 a cluster based on Kubernetes provisioning  $7 \times 10$ GB “Multi Instance GPU” (MIG) partitions of an NVIDIA A100 GPU for machine learning research across the INFN units and experiments (“**Production cluster**”).

A second Kubernetes cluster with  $2 \times$  non-partitioned A100 is currently used for machine-learning studies awaiting for resources from ICSC to be available. It is used to experiment with scaling (CPU, RAM and GPU resources can be added or removed transparently) and multi-node filesystems (“**A100 cluster**”).

The same configuration was used on a dedicated cluster, now dismantled, provisioning  $3 \times$  A100 for the [5th advanced hackathon](#) in Pisa (“**Hackathon cluster**”). The setup was transparently replicated at ReCaS.

Several other k8s clusters are created and destroyed frequently for development purposes.

# Milestones

## **CLUSTER 12/24**

31/12/2024 – a kubernetes cluster scalable on multiple GPUs is available for test

## **OBSERVABILITY 6/25**

30/06/2025 – monitoring and accounting tools implemented and documented

## **BATCH 12/25**

31/12/2025 – opportunistic batch system available for test

## **DataCloud 12/26**

31/12/2026 – the overlay prototype can be integrated in DataCloud

While the platform is evolving quickly, some **key requirements are emerging** and our bread&butter solutions should be cleaned and **integrated now** in DataCloud (*i.e.* Ansible, Dashboard, Helm) to avoid diverging development.

Activity in progress for combining multiple flavors of GPUs in one cluster.

Next step Jupyter with multi-node distributed filesystem (*e.g.* NFS).

# Official conda environment management

**CLUSTER 12/24**

Users are encouraged to create their custom conda environments for their applications, but creating a GPU-enabled conda environment might be non-trivial.

AI\_INFNUM will provide and maintain a minimal set of working environment (*e.g. Tensorflow, Pytorch, Jax*) that can then be cloned and extended.

Successful experience at the Advanced Hackathon, providing 5 read-only different environments, possibly extended in user-space.

Extending the cluster to multiple nodes with heterogeneous resources requires a **filesystem distributed across the nodes**.

Example: Two user with runtimes with different GPUs might want to collaborate working on the same notebook.

**Current test-bed is based on NFS and is clearly suboptimal in terms of performance.**

We need to investigate technologies to distribute the user's home and the conda environments across nodes of the same cluster and elaborate realistic **benchmarks** for comparison of the solutions.

**Alternative solutions** (e.g. OpenEBS, longhorn, GlusterFS) may outperform NFS on those benchmarks.

Interesting activity by Diego C. on OpenEBS, but the overhead at spawning time was excessive for interactive usage.

# Opportunistic usage of resources

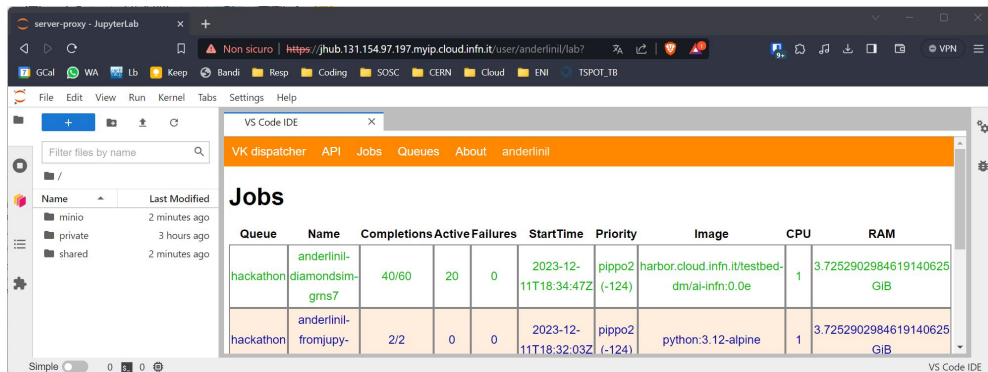
Michele Veltri (UniUrbino) Lorenzo Viliani (Firenze) *et al.*  
Sinergie con DataCloud WP6

BATCH 12/25

We aim at **using GPUs not used interactively for batch processing.**

A first prototype to **submit jobs** from Jupyter to native Kubernetes queues has been deployed and is being tested.

Much more development is needed to tune and document the **job submission**, the **queue management** and **fair-share** policies.



Queue	Name	Completions	Active	Failures	StartTime	Priority	Image	CPU	RAM
hackathon	anderlini-diamondsimgrms7	40/60	20	0	2023-12-11T18:34:47Z	pippo2 (-124)	harbor.cloud.infn.it/testbed-ai-infn:0.0e	1	3.7252902984619140625 GiB
hackathon	anderlini-fromjupy	2/2	0	0	2023-12-11T18:32:03Z	pippo2 (-124)	python:3.12-alpine	1	3.7252902984619140625 GiB

We acknowledge the support of “Cassa di Risparmio di Firenze” to the project “CLOUD\_ML” to explore batch system solution enabling an opportunistic usage of resources.



## Offloading (CVMFS-unpacked)

Marco Verlatto (PD), Diego Ciangottini (PG) *et al.*  
*Sinergie con DataCloud WP6 (in corso!)*

BATCH 12/25

Once the batch system is in place, we might extend it to submit jobs on remote resources (**offloading**), provided we **manage the data flow** properly.

Marco V. developed a mechanism based on **cvmfs-unpacked**, docker and harbor to automatically deploy on cvmfs images loaded to a set of repositories in harbor.cloud.infn.it.

A prototype of **automatic generation and upload of docker images** based on user's job submission is also in preparation (using kaniko).

## Frontend (JupyterLab and user web services)

**CLUSTER 12/24**

The default docker image is the base for users to customize their computing environment with custom web applications (*e.g. MLFlow, OpenRefine*).

It should be sufficiently light, coherent with JupyterHub and integrating some important services (*e.g. Minio via sts-wire*).

We should explore Collaborative Jupyter.

**OBSERVABILITY 6/25**

Profiling applications with GPUs provisioned through the cloud is more difficult.

Simone C. started a precious study for using TensorBoard for profiling GPU applications (even beyond TensorFlow) that should be completed and documented.

Thanks to the integrations available for monitoring Kubernetes services and report the status of the system (e.g. in Grafana), Prometheus has emerged as the hub for the system information.

Rosa P. started a survey of the available exporter and of the technologies to define custom Prometheus exporters to enhance the observability of the cluster status and resource allocation.

A discussion with INFN Cloud for integrating our metrics in a centralized Grafana service was also started.

Part of the effort will be to ensure that all the relevant metrics are available and well organized, to be easy to interpret.

## Explore Kubernetes support for FGPA

Provisioning FPGAs via Kubernetes should be already feasible (to be tested ) but a dedicated effort is needed to define a recipe like the one available for GPUs.

Many tools to develop with FPGAs require a graphics interface beyond Jupyter, we should decide whether we want to support those tools (licensing?) and how to serve them.

# Security and security scans

Persone ad oggi attive: Nessuno. *Prematuro?*  
*Sinergie con DataCloud WP4*

**DataCloud 12/26**

Before discussing how to integrate the platform in DataCloud we must enhance its security, for example including our servers in frequent security scans.

Dedicated effort for frequent updates of exposed software packages should also be identified.

As for the integration part, sinergies with DataCloud are of primary importance!

# Conclusion

An intense program of R&D ahead of us.

With additional resources becoming available in the context of ICSC (e.g. CINECA *Leonardo*) and Terabit (e.g. HPC bubbles) having a fully functional, production-ready provisioning model is critical.

Milestones are relatively relaxed with respect to the needs of the community, giving us time to **document and disseminate the activities** as part of the project.

Contributions on any of the task listed here (or on additional topics) are more than welcome!