Finanziato dall'Unione europea
NextGenerationEU

Ministero dell'Università e della Ricerca

Italiadomani
PIANO NAZIONALE DI RIPRESA E RESILIENZA

ICSC
Centro Nazionale di Ricerca in HPC, Big Data and Quantum Computing

October 19 - 25, 2024
CHEP 2024

ICSC
Centro Nazionale di Ricerca in HPC, Big Data and Quantum Computing

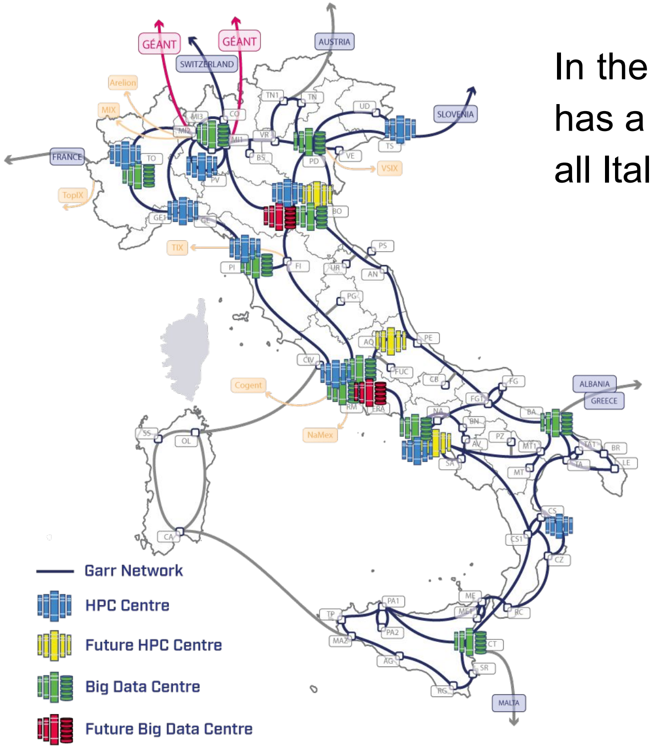
Supporting the development of Machine Learning for fundamental science in a federated Cloud with the AI_INFN platform

L. Anderlini[1], **M. Barbetti**[2], G. Bianchini[3,4], D. Ciangottini[3], C. Pellegrino[2], R. Petrini[1], D. Spiga[3]

[1] Istituto Nazionale di Fisica Nucleare (INFN), Sezione di Firenze, Italy
[2] Istituto Nazionale di Fisica Nucleare (INFN), CNAF, Italy
[3] Istituto Nazionale di Fisica Nucleare (INFN), Sezione di Perugia, Italy
[4] Department of Physics, University of Perugia, Italy

27th International Conference on Computing in High Energy & Nuclear Physics (CHEP 2024) | 19-25 October 2024

The INFN DataCloud project



In the framework of the current NRRP projects (ICSC and TeRABIT), INFN has a leading role in the creation of the ***Italian Cloud Federation*** to access all Italian scientific computing resources through uniform interface

- **Tier-1** (CNAF)
- **Tier-2** (BA, CT, LNF, LNL/PD, NA, MI, PI, RM1, TO)
- Backbone and federated clouds
- HPC4DR (LNGS)
- INFN Cloud
    - a Data Lake-centric Cloud infrastructure
    - heterogeneous federated resources on multiple sites across Italy
    - provider of an extensible portfolio of solutions for multidisciplinary scientific communities
    - more details in Claudio's talk

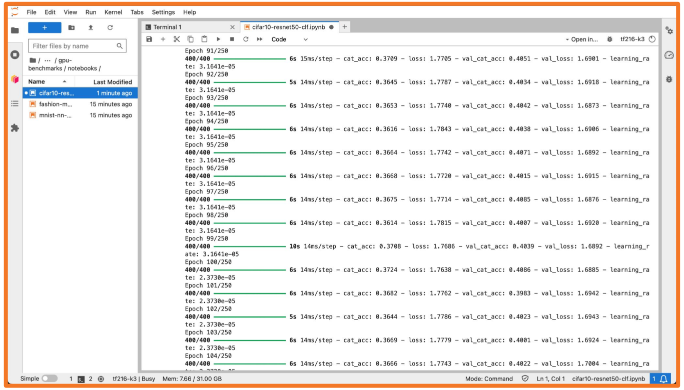
INFN CLOUD

Fostering the use of AI with the AI_INFN platform

Sitting on top of INFN Cloud, the ***AI_INFN*** initiative [preliminary docs] promotes the adoption of **machine learning** and **artificial intelligence** techniques for fundamental scientific research

The core of AI_INFN is its **platform** [https://hub.ai.cloud.infn.it] that provides multi-user access to GPU via Notebooks, powered by JupyterHub

AI_INFN
Artificial Intelligence technologies for INFN research



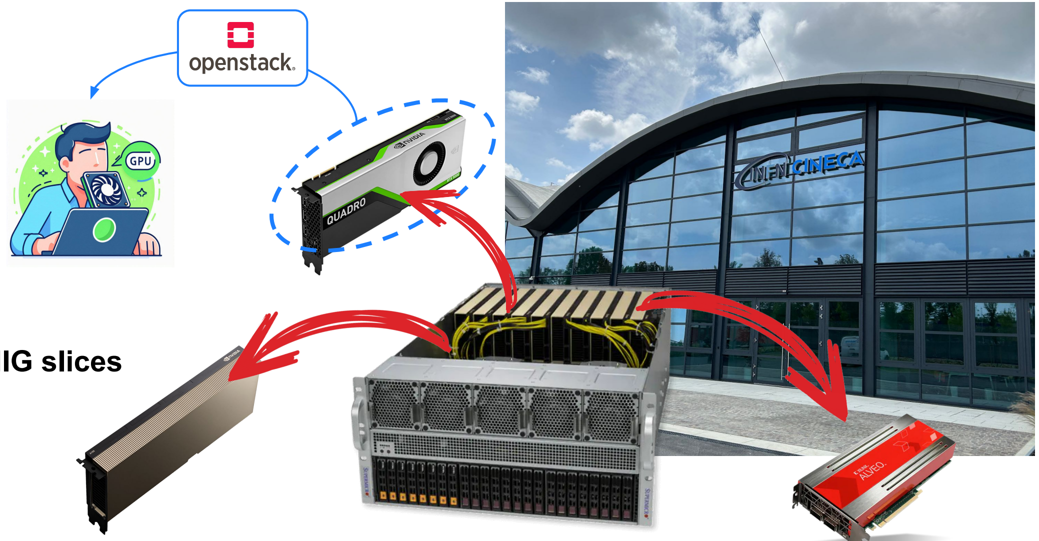
AI_INFN's primary activities include:

- facilitating access to **HPC and GPU resources**
- organizing **educational and training events**
- fostering the **AI community** within INFN
- conducting **R&D** to integrate new technologies (*e.g.*, FPGA, quantum computing) into the platform

Federated bare-metal resources

Computing resources available to AI_INFN are located at Bologna Technopole within the new CNAF Data Center facility [more details in Daniele's talk], and managed through a **virtualization layer** (OpenStack of Cloud@CNAF) in **INFN Cloud**:

- 4× servers:
  - 1× 64 CPU cores with 750 GB RAM
  - 3× 128 CPU cores with 1024 GB RAM
- Total local storage: 60 TB of **NVMe disk**
- GPU cards:
  - 8× NVIDIA **Tesla T4**
  - 5× NVIDIA **RTX 5000**
  - 1× NVIDIA **A30**
  - 4× NVIDIA **A100**, potentially served as **4×7 MIG slices**
- FPGA boards:
  - 2× AMD Xilinx **Alveo V70**
- 10 GbE connection to CNAF resources

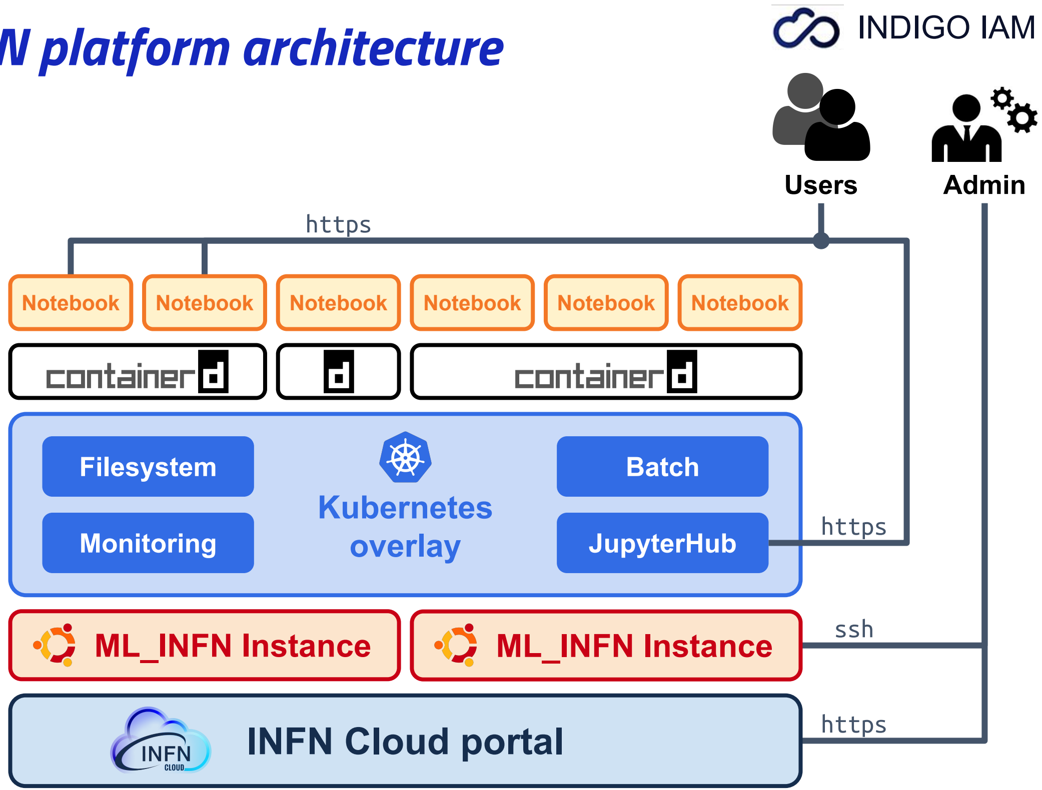


The AI_INFN platform architecture

The feasibility and effectivity of **sharing GPU resources** via the Cloud have been successfully demonstrated through a *proof-of-concept* project [EPJ Web Conf. **295** (2024) 08013]

AI_INFN improves the sharing capabilities:

- addition of an abstract and elastic overlay powered by **Kubernetes**
  - login via AAI → **INDIGO IAM**
  - distributed filesystem
  - managed environments for ML
  - monitoring & accounting
- **data decoupled from computing resources** with a filesystem shared across the VMs
- adding and removing VMs enables manual **horizontal scaling**



Filesystems and data persistency

Local filesystem

- ***ephemeral* filesystem**
- used to install packages in its own container
- provisioned via OverlayFS
  - allows to mimic write ops on top of an immutable fs (Docker image)
  - introduces additional logic to read and write ops

`/tmp` is **directly mapped** to a logical volume in the NVMe storage, avoiding the **OverlayFS overhead**

Distributed filesystem

- **platform filesystem**
- used to make softwares and tiny datasets persistent, and accessible from different nodes
- provisioned via NFS

NFS is relatively **slow** and **not suitable** for large datasets

NFS **cannot be mounted from remote sites** and there is no tools to upload files beyond JupyterLab

Cloud storage

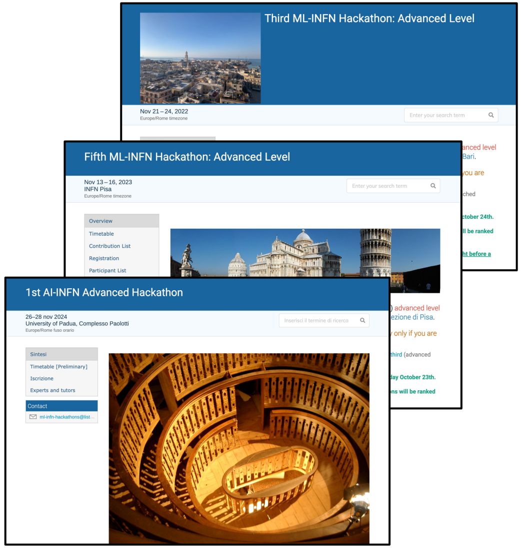
- **cloud-based object storage**
- used to store large datasets
- provisioned via RadosGW and mounted POSIX using sts-wire
- service **centrally managed** by INFN Cloud
  - data access through Web interface or using S3 clients

A **Ceph volume** is used to store the **encrypted backups** of the platform filesystem (based on BorgBackup)

Stress tests for the platform: hands-on during hackathons

In-person training events ("***hackathons***") serve both to **onboard users** to the platform and to provide newcomers with valuable theoretical materials and **ready-to-use notebooks**:

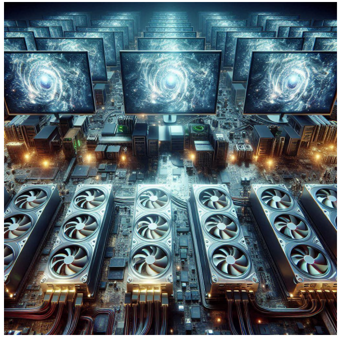
- 3rd ML-INFN Hackathon: Advanced Level (Bari, November 2022)
- 5th ML-INFN Hackathon: Advanced Level (Pisa, November 2023)
- NEW 1st AI-INFN Hackathon: Advanced Level (Padua, November 2024)



Since the first edition, hackathons have served as a **stress test** for the platform, as it had to provide GPU access to several concurrent users (participants + tutors) combining resources from **Cloud@CNAF** and **ReCaS-Bari**:

- independent networks and filesystems
- shared IAM authentication
- synchronized software environments
- intense use of the GPUs during hands-on

From interactive mode to batch system



Once model development reaches sufficient maturity, analysts may want scale it on more resources, **moving beyond the interactive mode**:

- freeing up interactive resources for other developments
- extending training time for model refinement and/or scaling up model size
- enabling parallel execution for intensive *hyperparameter optimization*

Providing the AI_INFN platform with an opportunistic **batch system** is then mandatory!

#1

The primary goal is to enable **opportunistic use of GPUs** dedicated to interactive tasks but left idle (e.g., during the night)

#2

The secondary goal is to enable workflows that combine developments on the platform with heavy computation on remote HPCs (e.g., Leonardo) through ***offloading***

vk-dispatcher

The “interface” between the interactive world and the batch system is enabled by a microservice called `vk-dispatcher` (vkd) currently under development

Development is our priority!

Batch workloads must not affect the interactive use of the platform

Kueue: k8s-native batch system

Kueue offers a set of APIs and dedicated controllers to simplify and enhance **job queue management** in Kubernetes clusters for batch processing, HPC, AI/ML, and similar applications:

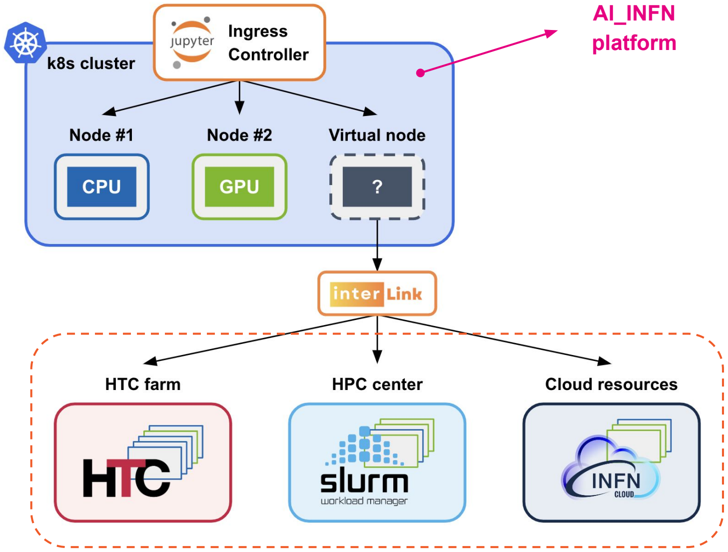
- **Queue management.** Provides a robust infrastructure for job queue management, ensuring reliable and scalable job execution within the Kubernetes cluster
- **Integration with Kubernetes resources.** Kueue integrates natively with Kubernetes resources and functionalities, leveraging the cluster's orchestration and management capabilities
- **Monitoring and Scalability.** With dedicated controllers, Kueue simplifies job state monitoring and enables automatic resource scaling based on workload demands

vk-dispatcher

vkd provides an **authenticated delegation layer** between JupyterHub and Kubernetes, enabling the translation of a user's interactive session into a Kubernetes job

The **Kubernetes job** are submitted to queues managed by **Kueue** that may be enabled for specific projects through the JupyterHub groups

Enabling offloading using interLink as Virtual Kubelet provider



extension of the AI_INFN platform through the VK mechanism

Once AI models are developed, researchers often seek to scale them **beyond development-dedicated resources**

The AI_INFN platform is exploring a solution to transparently extend the resource pool accessible to Kueue using the ***Virtual Kubelet*** (VK) mechanism:

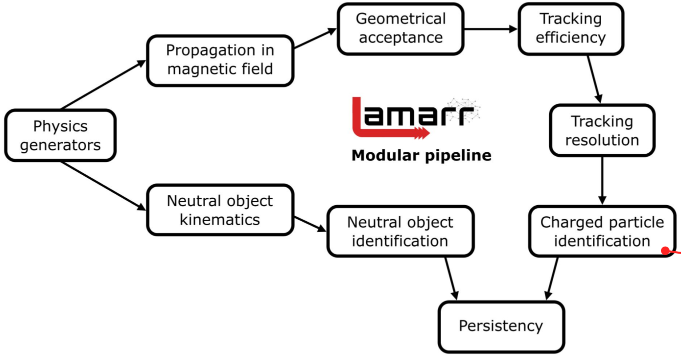
- VKs provide k8s cluster with "***Virtual Computing Nodes***" that have no networking towards the API server or other services
- VKs are **ideal for batch processing**, where the connection between the cluster and the working node is only needed at job submission and retrieval

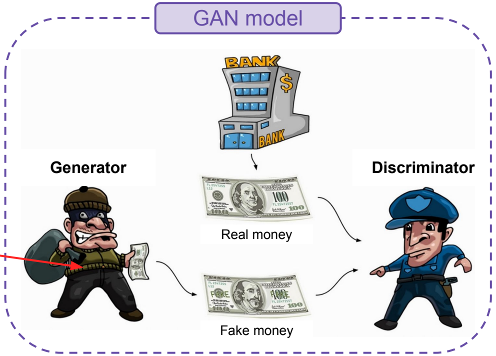
The **interLink** protocol offers a batch-system native backend for Virtual Kubelets (e.g., SLURM, HTCondor, or other Kueue instances)

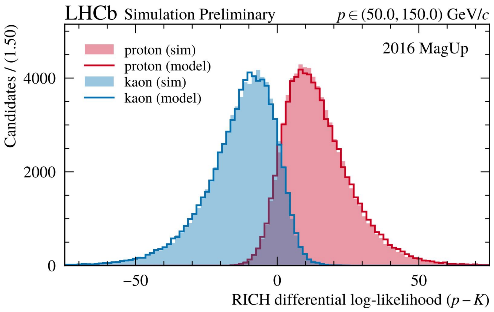
- more details in Diego's talk

Generative models for flash simulation at LHCb

- Simulation consumes the majority of CPU time in HEP experiments, making it necessary to develop **faster simulation options** for *next-generation* detectors [some examples in talks from Michał@LHCb, Andrea@CMS, Federico@ATLAS]
- ***Lamarr*** [see these slides from ICHEP 2024] offers the fastest option (*flash*) for simulation at LHCb relying on a modular framework powered by **AI-based parameterizations**
- ***Generative Adversarial Nets*** (GAN) are used to reproduce the errors introduced during detection and reconstruction mimicking the *high-level* response of the detector [EPJ Web Conf. **295** (2024) 03040]







Generative models for flash simulation at LHCb

PU time in HEP experiments, making

some examples in talks from Michał@

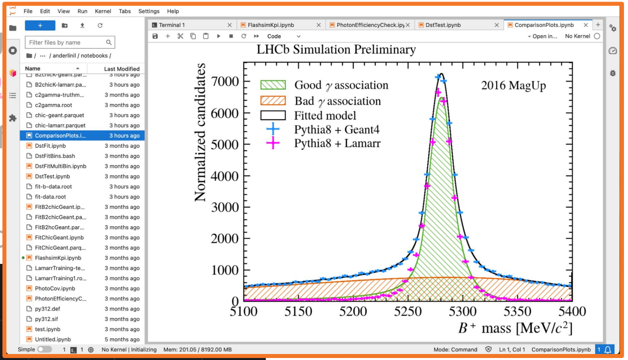
2024] offers the fastest option (*flash*)

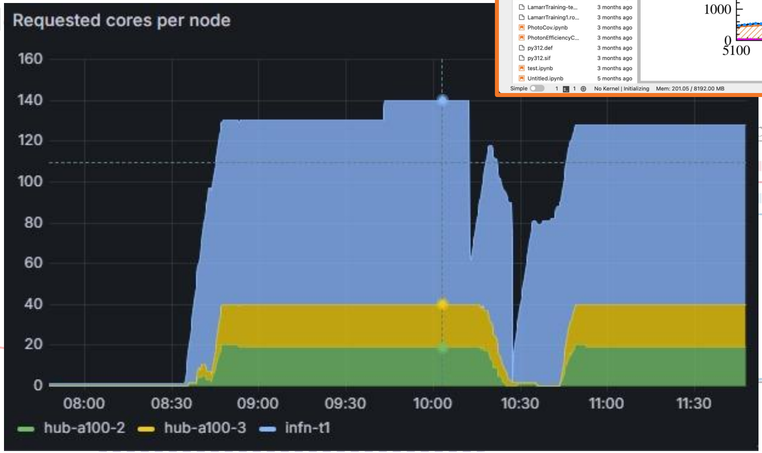
heterizations

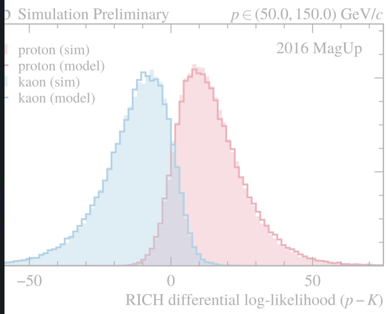
are used to reproduce the errors intro

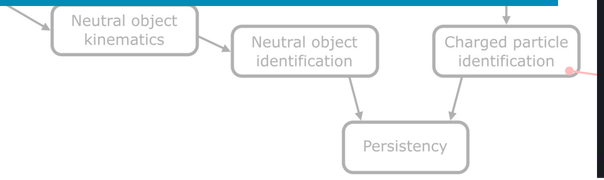
The Lamarr framework represents the perfect laboratory to prototype **AI_INFN offloading capabilities** (based on the interLink protocol)

Lamarr validation campaigns have been distributed among **3 Cloud sites** (Cloud@CNAF, CloudVeneto, and Cloud@ReCaS) and, more recently, also involving the **CNAF Tier-1 resources**









SUMMARY AND CONCLUSIONS

- Sitting on top of INFN Cloud, the **AI_INFN initiative** aims to simplify access to hardware accelerators (e.g., GPU, FPGA, QC) and promote the adoption of AI technologies for INFN use-cases
- While the **interactive mode** is highly beneficial during the development phase, it can become a **limitation** when researchers seek to scale up model performance (e.g., extended training time, larger model size):
  - AI_INFN is exploring the possibility of translating interactive sessions into **Kubernetes jobs**, allowing them to be submitted to a *local* **batch system** using **`vk-dispatcher`** and **Kueue**
  - Ongoing developments focus on extending platform capabilities beyond the local cluster through **offloading**, namely enabling job submission to computing nodes provided via the **Virtual Kubelet mechanism** and the **interLink provider**
- The offloading capabilities of the AI_INFN platform are currently being explored
  - the validation campaigns of the LHCb *flash-simulation*, **Lamarr**, represents the perfect laboratory for **heterogeneous workloads**
  - results presented at ICHEP 2024 obtained CPU resources coming from 3 different Italian **Cloud sites**
  - more recently, validation jobs have been run using **HTC nodes** like the one provided by the CNAF Tier-1
  - ongoing works aim to integrate also **HPC resources** (*e.g.*, Leonardo supercomputer) both for training and validation

Thanks!

Any questions or comments?

Matteo Barbetti (INFN CNAF)
email: matteo.barbetti@cnaf.infn.it

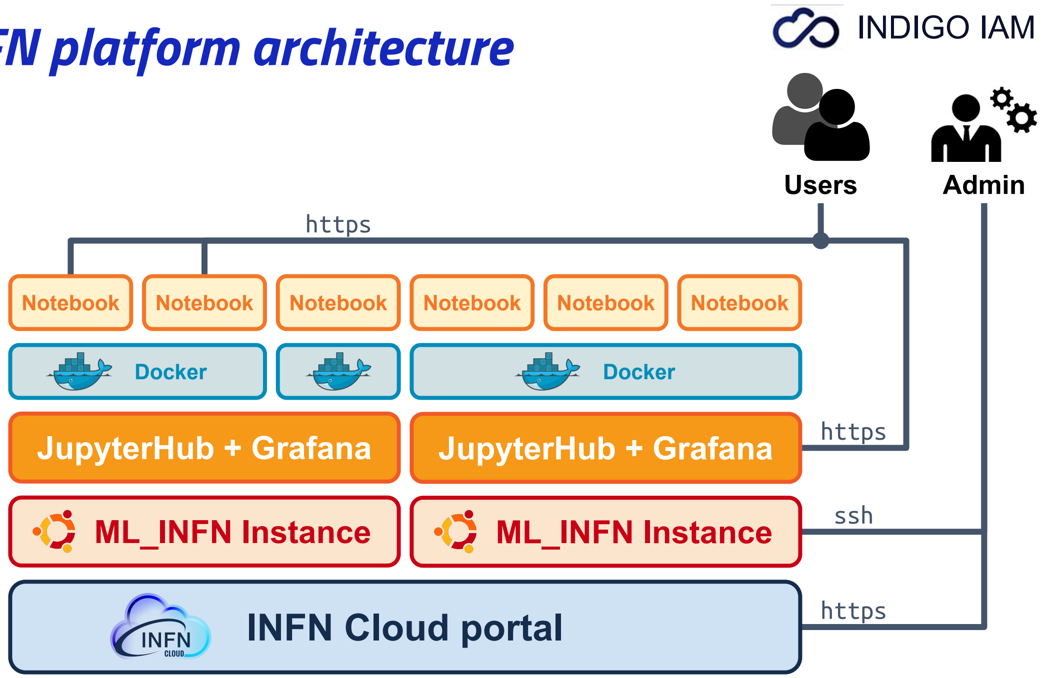
October 19 - 25, 2024
CHEP 2024

BACKUP

The ML_INFN platform architecture

The **ML_INFN** outcome:

> *Sharing precious GPUs through the Cloud is feasible and effective!*



Managed software environments

One of the most common support requests during the ML_INFN experience was setting up of a **GPU-accelerated Python software stack**, due to the complex configuration of NVIDIA drivers, CUDA/cuDNN versions, and the specific ML framework version required for the application

The AI_INFN platform offers different strategies to customize the **development software environment**:

docker

The most radical option is to **extend the default OCI image** by adding system libraries or software packages

This is often done when teams or single users want to use web-based dashboards or single-user web applications, which can be served via Jupyter Server Proxy

CONDA

Conda is a cross-platform and language agnostic environment manager that ensures **portability** between collaborators and is adopted particularly when **Python external tools** are used

Users are encouraged to clone and customize the centrally **managed conda environments** to suit their needs

APPTAINER

The main issue with Conda is that it creates environments with **10000+ files**, stressing any filesystem

Apptainer is a containerization platform offering an **isolated** and **reproducible** environment for application execution by packing all the needed dependencies in a **single file** (container image)

Monitoring and accounting

Balance and distribution of the AI_INFN resources among the participating projects is ensured through a **monitoring and accounting service** that operates at the Kubernetes overlay-level to collect information on the computing resources and expose it to a Prometheus instance running within the platform. All the metrics collected are then **made accessible** through a Grafana dashboard running in a VM independent of the platform cluster.

`kube-eagle` exports information on the use of **cluster CPUs** and **memory resources** by the various Kubernetes components

NVIDIA GPU DCGM-exporter allows admins to collect **GPU metrics** and monitor the **workloads** running on the cluster GPUs

custom exporters have been developed to monitor also the usage of **storage resources**

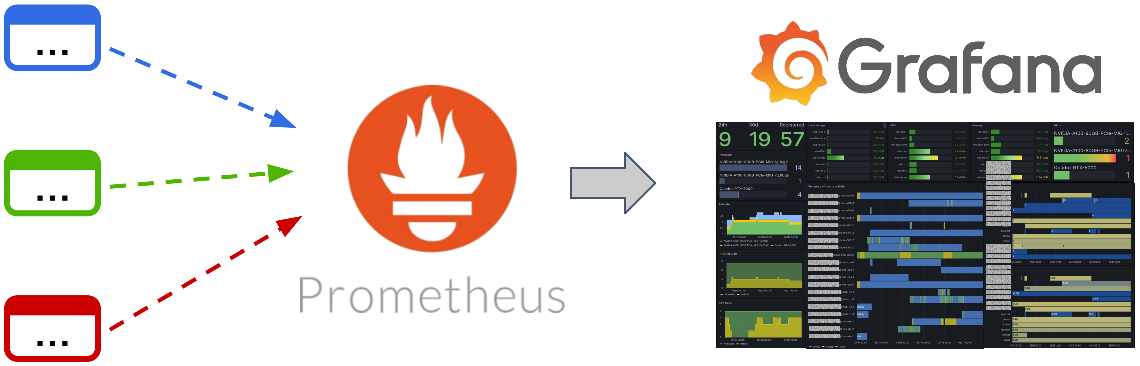
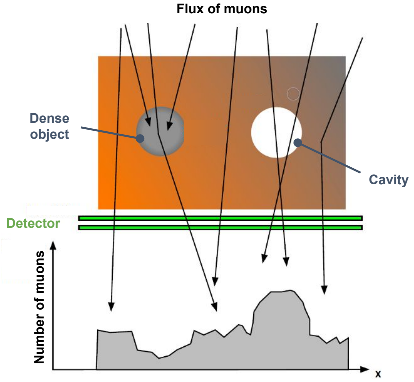
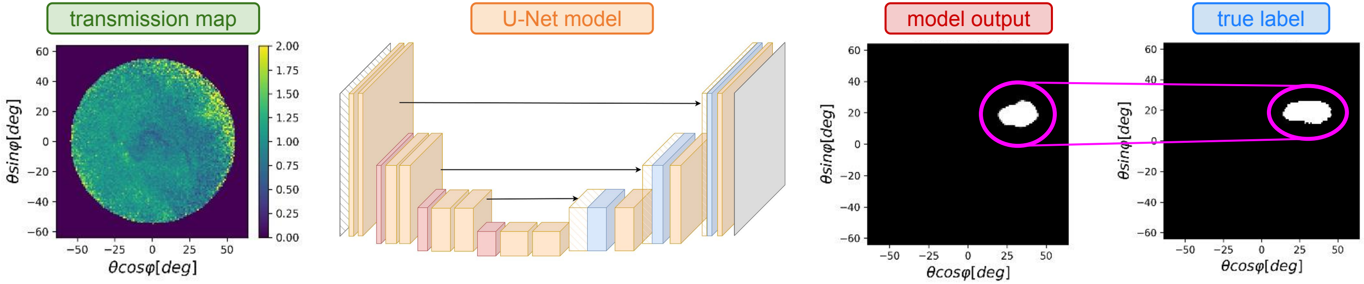


Image segmentation for muon radiography



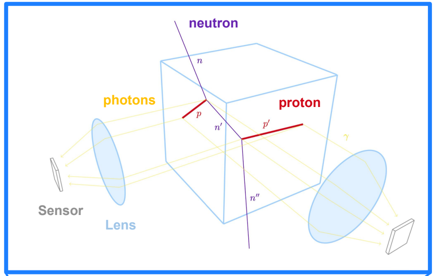
- ***Muon radiography*** is an innovative technique that allows to inspect very large objects (e.g., pyramids, mines, factories) exploiting the penetration capacity of muons
- The goal is to detect **cavities** or **fractures** comparing the muon transmission between the target and the free-sky configuration as measured by a specialized detector
- The AI_INFN platform has been used to develop a CNN-based model for **detecting and mapping cavities** inside the Temperino mine [work recently presented at APSAC 2024]

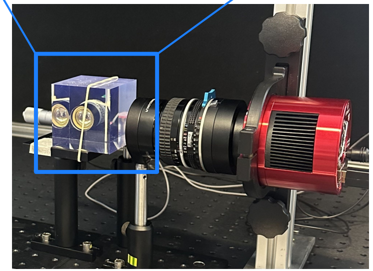


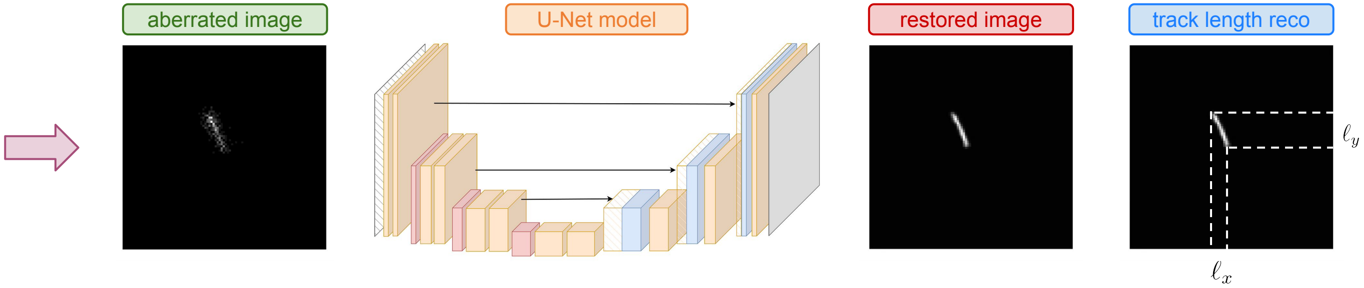
Analysts: A. Paccagnella, V. Ciulli, C. Frosin (UniFi and INFN Firenze)

Image restoration for proton tracking

- **Neutron tracking** plays a key role in fundamental science studies and dosimetry, despite being challenging due to the absence of charge
- ***Recoil Proton Track Imaging*** (RPTI) allows to measure neutron momentum exploiting the scintillating light produced by protons after an elastic scattering
- The **RIPTIDE detector** [JINST **19** (2024) C02074] relies on RPTI techniques for neutron tracking combining a plastic 3D scintillator with an advanced optical system
- A prototypal CNN-based model for **removing optical aberrations** from the collected images has been developed on the AI_INFN platform and trained on simulated data

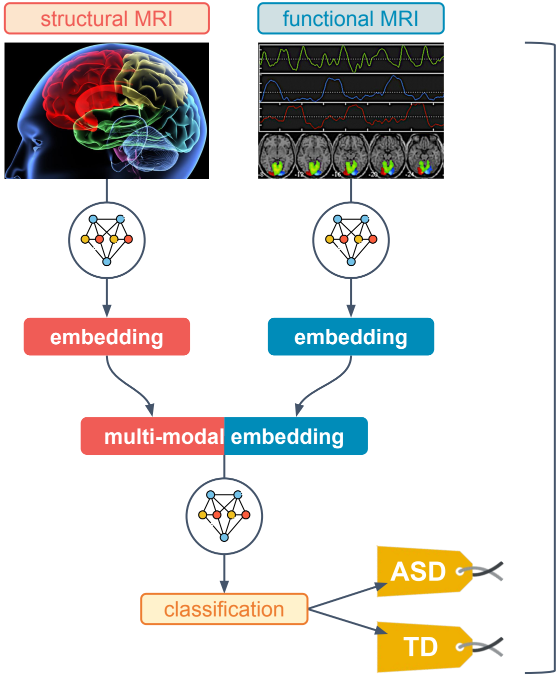




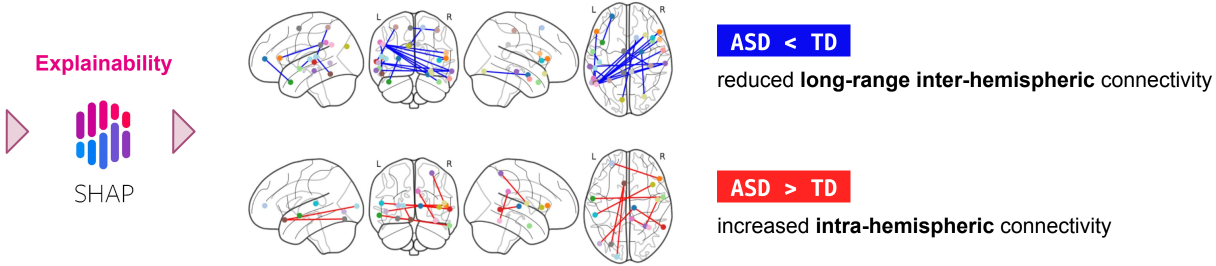


Analysts: S. Lanzi, C. Massimi, F. Giacomini (UniBo and INFN CNAF)

Explained AI for autism diagnosis



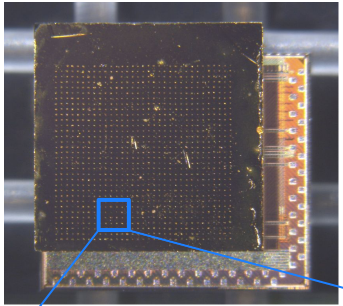
- **next_AIM** is one of the most enthusiastic users of the AI_INFN platform to fulfill its wide scientific program (see I. Postuma and P. Oliva contributions)
- Among the various works, we discuss here the use of deep learning for the **diagnosis** of ***Autism Spectrum Disorder*** (ASD)
- A next_AIM team shows that employing a **multi-modal architecture** allows to obtain state-of-the-art diagnosis accuracy for ASD [Brain Inf. **11** (2024) 2]
- Processing the trained model with **explainability techniques** allows to select relevant brain features for distinguishing ASD from TD



Analysts: F. Lizzi, S. Saponaro, G. Serra, F. Mainas, P. Oliva, A. Giuliano, S. Calderoni, A. Retico (UniPi, UniCa, UniSS, INFN-Pisa, and INFN-Cagliari)

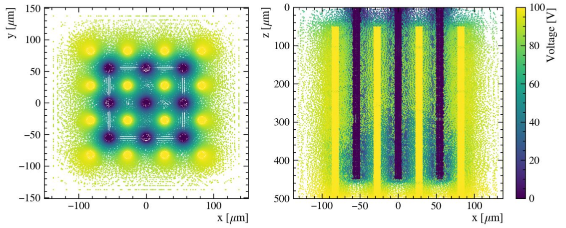
Physics Informed Neural Net for diamond detector fabrication

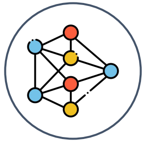
- The simulation of **3D diamond pixel sensors** [Nucl. Instrum. Meth. A **1046** (2023) 167692] is based on *finite element methods* relying on the ROOT-based Garfield++ software package
- Optimizing 3D diamond detectors would benefit from **faster simulation techniques** that can ideally infer detector performance directly from construction parameters
- ***Physics Informed Neural Networks*** (PINNs) are under investigation as a method to solve the set of PDEs used to compute **time-dependent potential maps** (ICSC Spoke 2 in partnership with ENI)



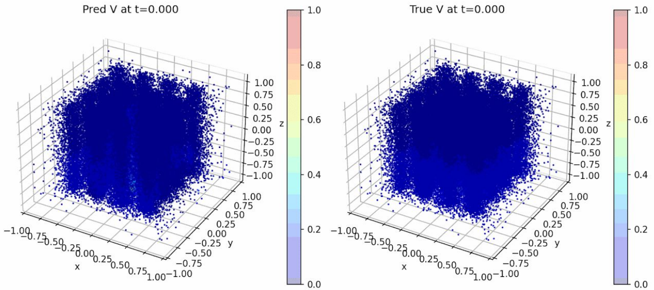
Laboratory to test **AI_INFN batch features** (based on vkd and Kueue)
Used up to **50 CPU cores**, **100 GB of RAM** and **6 GPUs**, opportunistically

Potential of the simulated sensor

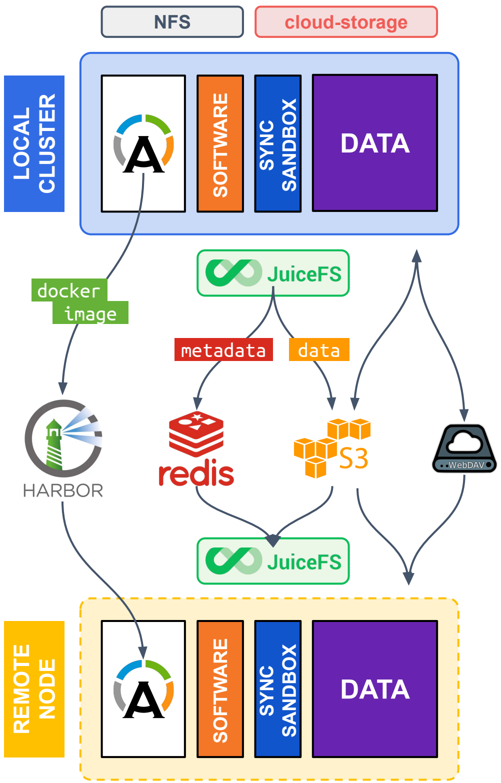




PINN model



Analysts: C. Buti, A. Bombini (UniFi and INFN Firenze)



Software and data crossing the platform borders

The combination of vkd, Kueue, and interLink enables the translation of an interactive session into a batch job, which can be then scheduled on a remote computing node

Remote execution of workloads also requires **replicating** the development software environment provided by the platform, as well as **accessing** data, configuration files and scripts/notebooks:

- In the current implementation, the software environments provided by AI_INFN are packaged as Apptainer images and **distributed to remote resources** by uploading and downloading them via the Harbor registry
- Configuration data and scripts transfer **crossing the platform borders** is enabled by JuiceFS, a Cloud-based, high-performance, POSIX-compliant distributed filesystem designed for multi-cloud and serverless computing
- Data can be directly accessed through **S3** or **WebDAV** protocols