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

Flash Simulation and bleeding-edge Machine Learning applications

Report July 2025

Lucio Anderlini
Istituto Nazionale di Fisica Nucleare, Sezione di Firenze

External Partner

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

Who we are

Staff members:

- Alessandro Bombini [j], INFN
- Giuseppe Piparo[l], INFN
- Maurizio Martinelli [a], Università Milano Bicocca
- Simone Capelli [a], Università Milano Bicocca
- Federica Maria Simone [i], Politecnico di Bari
- Nicola De Filippis [i], Politecnico di Bari
- Vieri Candelise [h], Università di Trieste
- Giuseppe Della Ricca [h], Università di Trieste
- Valentina Zaccolo [k], Università di Trieste
- Mattia Faggin [k], Università di Trieste
- Lorenzo Rinaldi [e], Università di Bologna
- Piergiulio Lenzi [g], Università di Firenze
- Vitaliano Ciulli [g], Università di Firenze
- Sharam Rahatlou[h], Università Roma 1
- Daniele del Re [h], Università Roma 1
- Lorenzo Capriotti [f], Università di Ferrara
- Francesco Conventi [e], Università di Napoli
- Francesco Cirotto [e], Università di Napoli

PhD students:

- Francesco Vaselli [c], Scuola Normale Superiore di Pisa
- Matteo Barbetti [b], Università di Firenze
- Muhammad Numan Anwar [j], Politecnico di Bari
- Benedetta Camaiani [g], Università di Firenze
- Alkis Papanastassiou [g], Università di Firenze
- Antonio D'Avanzo [e], Università di Napoli

External collaborators:

- Andrea Rizzi [c], Università di Pisa

KPIs

KPI ID	Description	Acceptance threshold	2024-09-24
KPI2.2.1.1	N_{MC} billion events obtained from ML-based simulation, as demonstrated by official links in experiments' simulation databases	N_{MC} >= 1	**778 M** events (completed: **78%**)
KPI2.2.1.2	N_{EXP} experiments have tested a machine-learning based simulation	N_{EXP} >= 2	3 experiment (completed: **150%**)
KPI2.2.1.3	Machine-learning use-cases tested in the context of the CN were presented at N_{CONF} international and national events	N_{CONF} >= 3	17 use-cases (since Sept. '23) (completed: **567%**)
KPI2.2.1.4	N_{UC} different machine-learning use-cases were tested in the context of the CN and made available in git repositories	N_{UC} >= 5	5 use-cases (completed: **100%**)

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

Expected targets for M10

Final report:

- validation of the model quality **[proposed for M11]**
- comparison with full simulation **[on track]**
- integration with the computing model of the experiment **[on track]**
- release of the software packages **[on track]**

Propose to delay KPI2.2.1.1 and validation of the model quality to M11.

Motivation: *the delays on the availability of the computing resources had an impact on the overall timeline of the project from which we have almost completely recovered, but better usage of the resources can be done with further refinement of the models before full validation.*

Risk Analysis

Identifier	Description	Update
R1	The CN is unable to provide the needed resources	We have access to Leonardo resources, and the provisioning model enabling offloading via InterLink has been validated in Integration PoC. Offloading from the AI_INFN Platform is being commissioned: ● Access to CINECA Leonardo: *granted* ● Access to CNAF-Tier-1: *granted* ● Access to ReCaS-Bari (condor): ***granted****, see *INFN-CLOUD#1704* ● Access to HPC Bubble Padova (*for ENI-PIML IG*): ***granted***
R2	The provisioning model is not ready for production	Status: ● CINECA Leonardo: ***in production*** ● CNAF-Tier-1: *in production* ● ReCaS-Bari: ***in production, with limitations*** ● HPC Bubble Padova (ENI-PIML IG): ***in production*** ● ***VEGA HPC: Preliminary tests performed***

R3	The recruitment process has limited or delayed success due to the large number of ML positions opening	Two new post-docs are starting (ENI-PIML IG funds):
		● **Alessandro Rosa**, INFN Firenze
		  ○ *Foundational aspects of Physics Informed Neural Networks*
		● **Rosa Petrini**, INFN Firenze
		  ○ *Computing infrastructure for training Physics Informed NN*

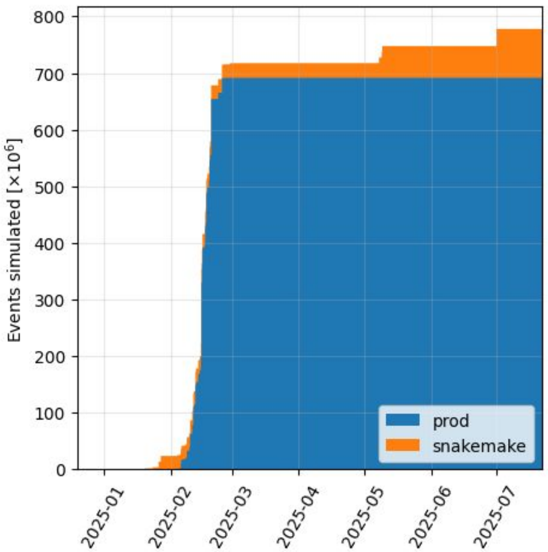
Validation productions

Two methods to run Lamarr productions were developed:

- **snakemake**: *for development and managing DAGs on heterogeneous resources (HPC, HTC, GPU)*
- **prod**: *pushed on scalability as number of grid-like jobs*

Combining the two approaches, we produced 778 M events, most of which in "**prod**" while debugging scalability.

The Snakemake workflow is more general and powerful, but requires more experience to orchestrate.



In July 2025, we lost almost 100M events because of a misconfiguration of the workflow.

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

Leonardo is now in production

Since June 2025, we can run LHCb jobs on Leonardo, thanks to fixes in the cvmfs configuration (thanks DataCloud, thanks CINECA support).

Because of the non-caching policy adopted by CINECA, though, the access to container images through cvmfs is less efficient than in other backends (*e.g. HPC bubbles, CNAF Tier-1...*).

The absence of local storage in the GPU partition also introduces some limitations (only network file systems are available).

Overall (and as expected), Leonardo is better for training than for validation.

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

Update on storage

We are now using **cvmfs unpacked** to distribute the software containers.

Thanks to the work by DataCloud and in particular Marco Verlato:

- Singularity containers can be created as part of the workflow and uploaded to INFN Cloud harbor (harbor.cloud.infn.it);
- They are automatically **unsquashed and distributed through cvmfs** as sandbox images to the computing nodes connected via InterLink, where they are cached;
- They can be used by **interlink** (emulating docker) and snakemake-on-interlink jobs (apptainer-in-apptainer).

Work has been done in the integration of **Snakemake with interlink** to **avoid relying on a file system** (we still do for this flagship, migration to the new setup is pending).

Moving to **cvmfs** for custom software containers has been a major turnaround of this flagship.

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

Release and documentation of the software

The submission of workflows "in-cluster" or via interlink is being documented: https://ai-infn.baltig-pages.infn.it/wp-1/docs/elearning/batch/

The GAN training has been documented in the GitHub package mbarbetti/pidgan

The integration with the experiment software is documented in the LamarrSim GitHub organization and in particular in the packages SQLamarr and PyLamarr

We have also contributed to Snakemake package and ecosystem. In particular, on the integration with Kubernetes, with WebDav and with JuiceFS.

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

Integration with the experiment computing model

The deployment model for Lamarr is based on **transpilation** of the Machine Learning pipelines in C files, compiled and distributed to the WLCG nodes as shared objects.

There are several advantages in this approach, but **shared objects have no guarantee of being forward-compatible**, especially when relying on specialied CPU instructions.

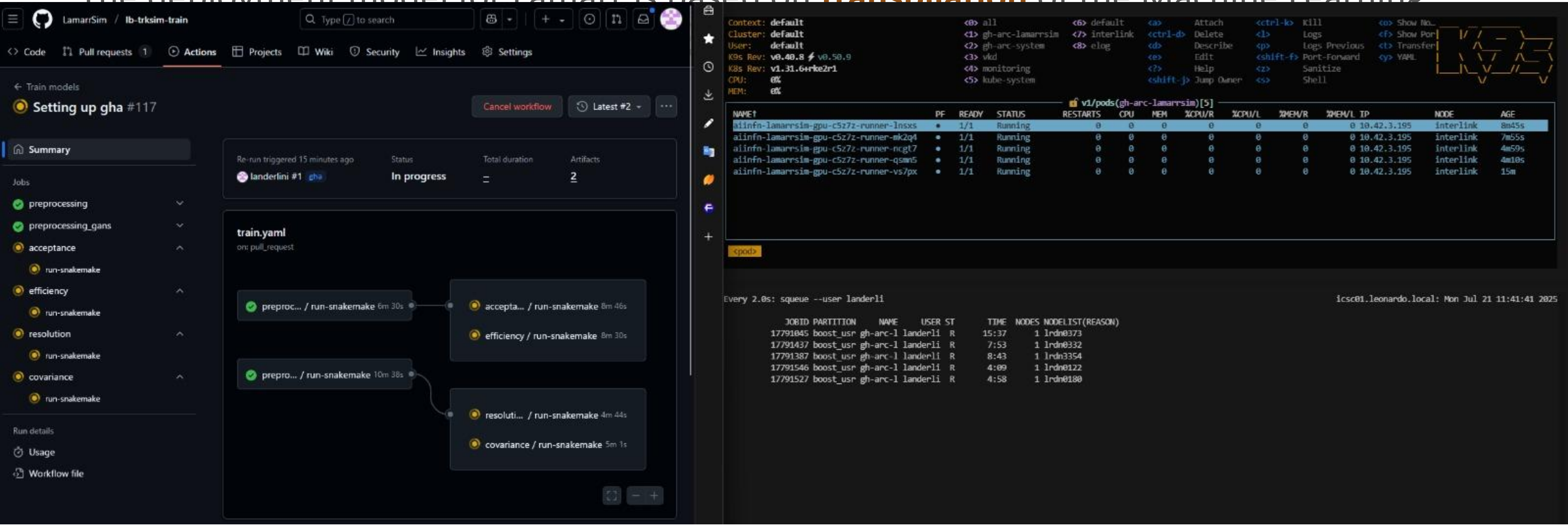
We are integrating model training and compilation as GitHub workflows, backed by aprivate runner, executed via InterLink on Leonardo (for the training part).

A specialized action (landerlini/apptainer-gha-runner/container@main) is being developed to execute workflows using apptainer instead of Docker, consistently with Interlink.

GitHub CI/CD supports mounting cvmfs volumes which enables compiling and releasing the trained models for the exact BINARY_TAG used by the experiment.

Integration with the experiment computing model

The deployment model for Lamarr is based on **transpilation** of the Machine Learning



GitHub CI/CD supports mounting cvmfs volumes which enables compiling and releasing the trained models for the exact BINARY_TAG used by the experiment.

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

Conclusion

Setting up the distributed infrastructure for this flagship required significant effort, but is now in place, for both the training and the validation parts.

Managing the storage has been one of the most critical points, with which many trial&error iterations happened, the current setup involves:

- Dedicated S3 storage for data
- INFN Cloud harbors with cvmfs unpacked for software
- Distributed file system (not needed any longer) for configuration files

The focus is currently on the training part and the utilization of the GPUs via Interlink.
In order to validate the new models trained via interlink, we propose to spend the simulation of the remaining 200M events in fall 2025, moving the milestone to M11.

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

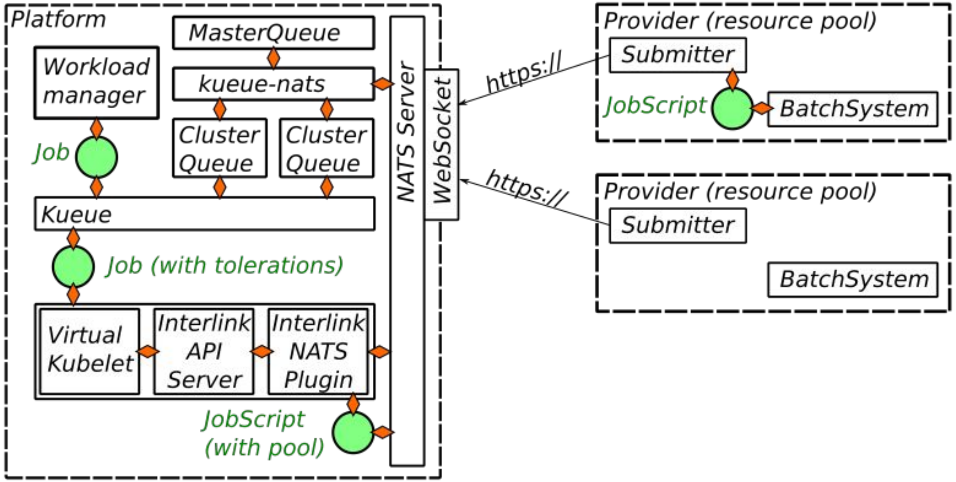
Backup

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

Interlink through Internet or NATS

Most of the jobs run through InterLink from the AI_INFN Platform rely on a custom path, passing through a NATS connection.

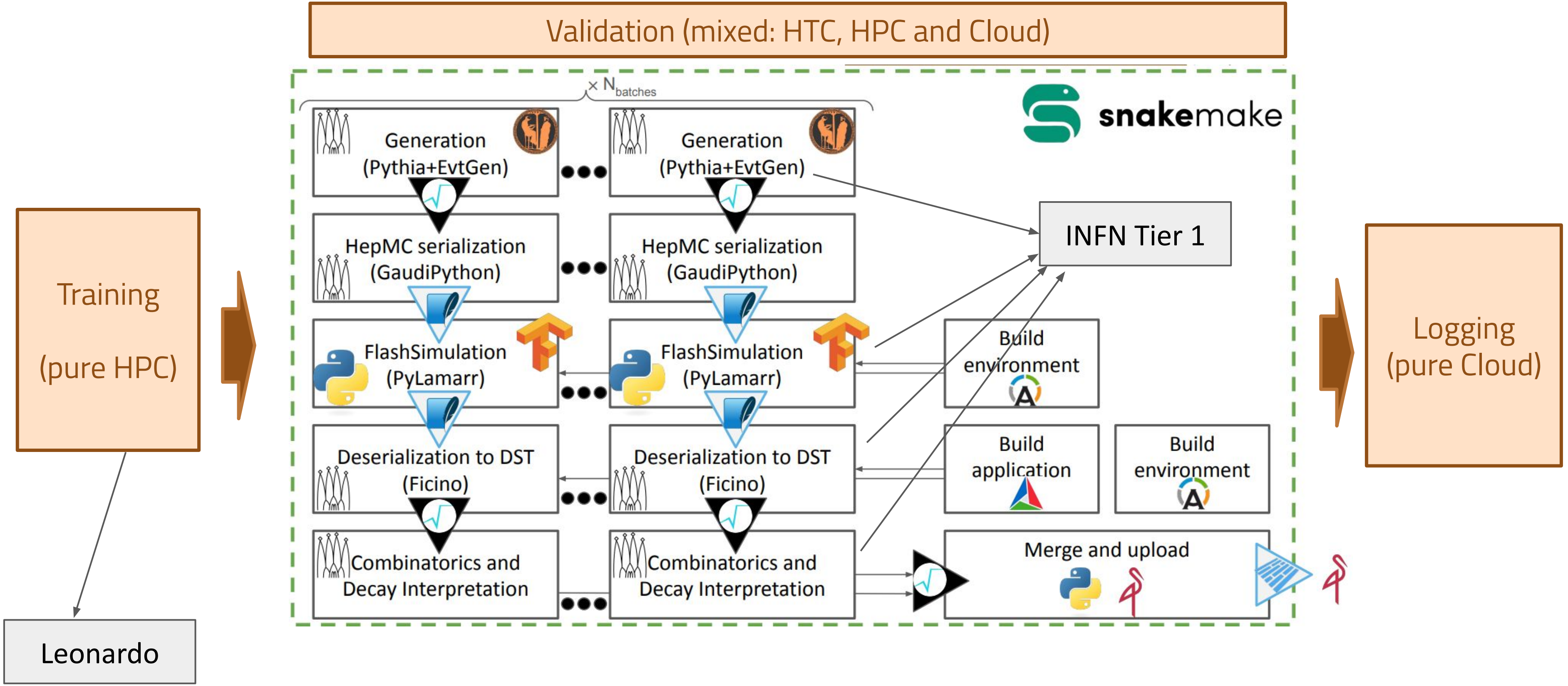
The official InterLink support provides instead a site-managed entry-point which is more sustainable in the long term.



The specific requirements for managing jobs from the AI_INFN platform to the official InterLink are being implemented in the code base.

A first successful test of executing snakemake-managed simulation jobs with the official InterLink were carried on using VEGA HPC.

Flash Simulation Workflow – offloaded



Scalability tests

- Most payloads are single-threaded WLCG-compatible applications
  - *Ideal for scalability tests of job-management infrastructure (many small jobs)*
- Every job mounts the distributed file system via fuse
- Container images are retrieved via the distributed file system (*should be improved*)
- Data from one step to the other is exchanged via S3 (small development instance)

Preliminary results.

Can sustain up to 100 concurrent jobs, independently of the provider.
Beyond, the metadata engine of the file system (tested both Postgres and Redis) cannot cope with the high number of concurrent connections.

The S3 instance as well suffers for the many concurrent operations.

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

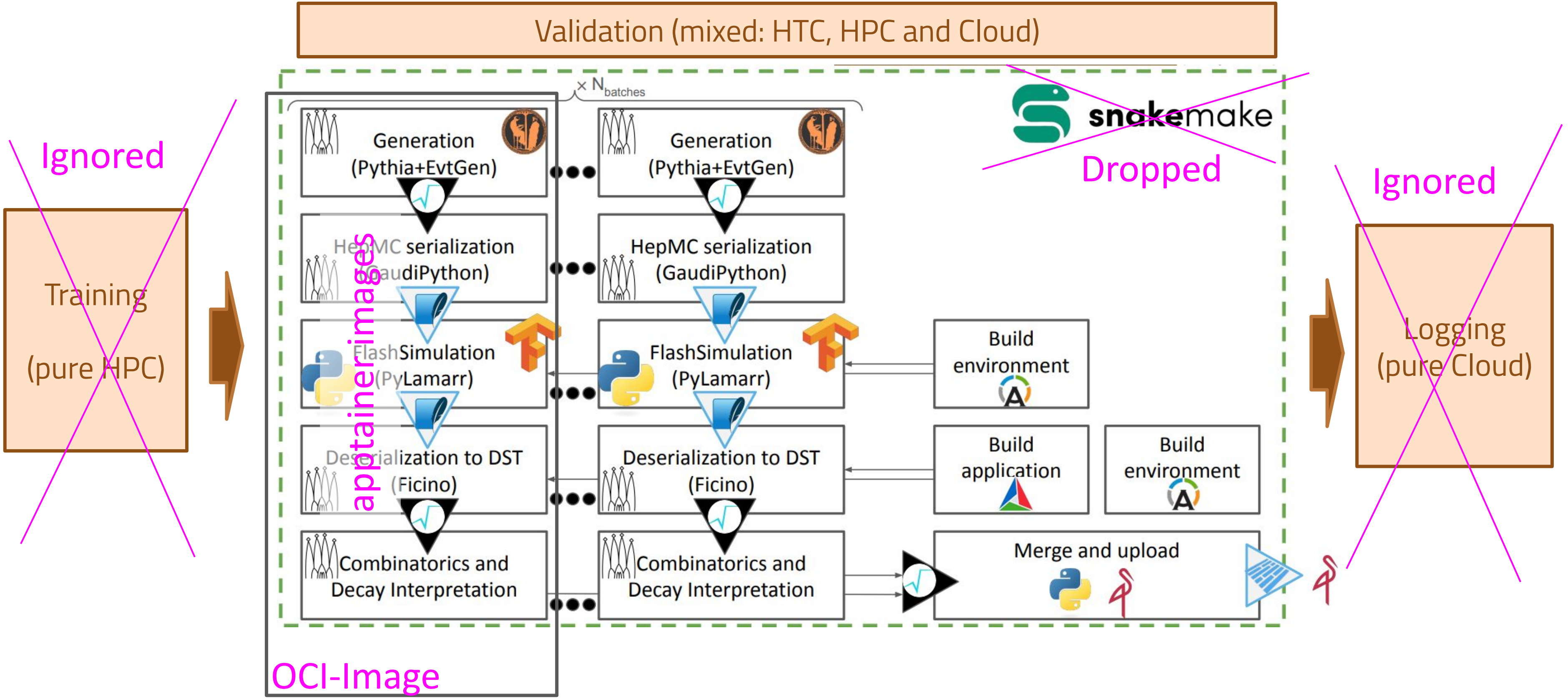
Comments and next steps

1. 100 job is a ridiculously tiny number if compared to the WLCG frameworks, but
   a. this infrastructure is intended for AI. Controlling 100 GPU-powered nodes is less ridiculous;
   b. the bottleneck are *the storage* and *database* infrastructure, where we put no effort and have extremely limited experience → it is very likely this threshold could be improved with a better setup (*e.g.* using a Redis cluster, MinIO on bare metal with more-abundant RAM)
   c. improvements at application level are possible, in particular software distribution *(for example, simply shipping snakemake in the docker image made a factor 3 on the number of jobs)*
   ❤by Gioacchino Vino, DataCloud

2. The distributed file system is fragile (with no surprise)
   → it hides other offloading-specific instabilities.

   To clear-up the offloading-specific aspects, we rewrote the workflow to be as gentle as possible on storage and network, removing dependencies on a distribute fs.

Flash Simulation Workflow – offloaded – prod



Comments on the "prod" workflow

Pro

Scale better, *enabling tests of the offloading infrastructure itself*

Run everywhere, the resulting payload is a single-core, 2-GB memory job, ***consistent with WLCG standards***.

Drastically reduce data exchange, ***ideal for network-bound steps***.

Cons

All steps run in the same node, ***extremely inefficient*** *for some task (overall CPU efficiency @CNAF: 4%)*

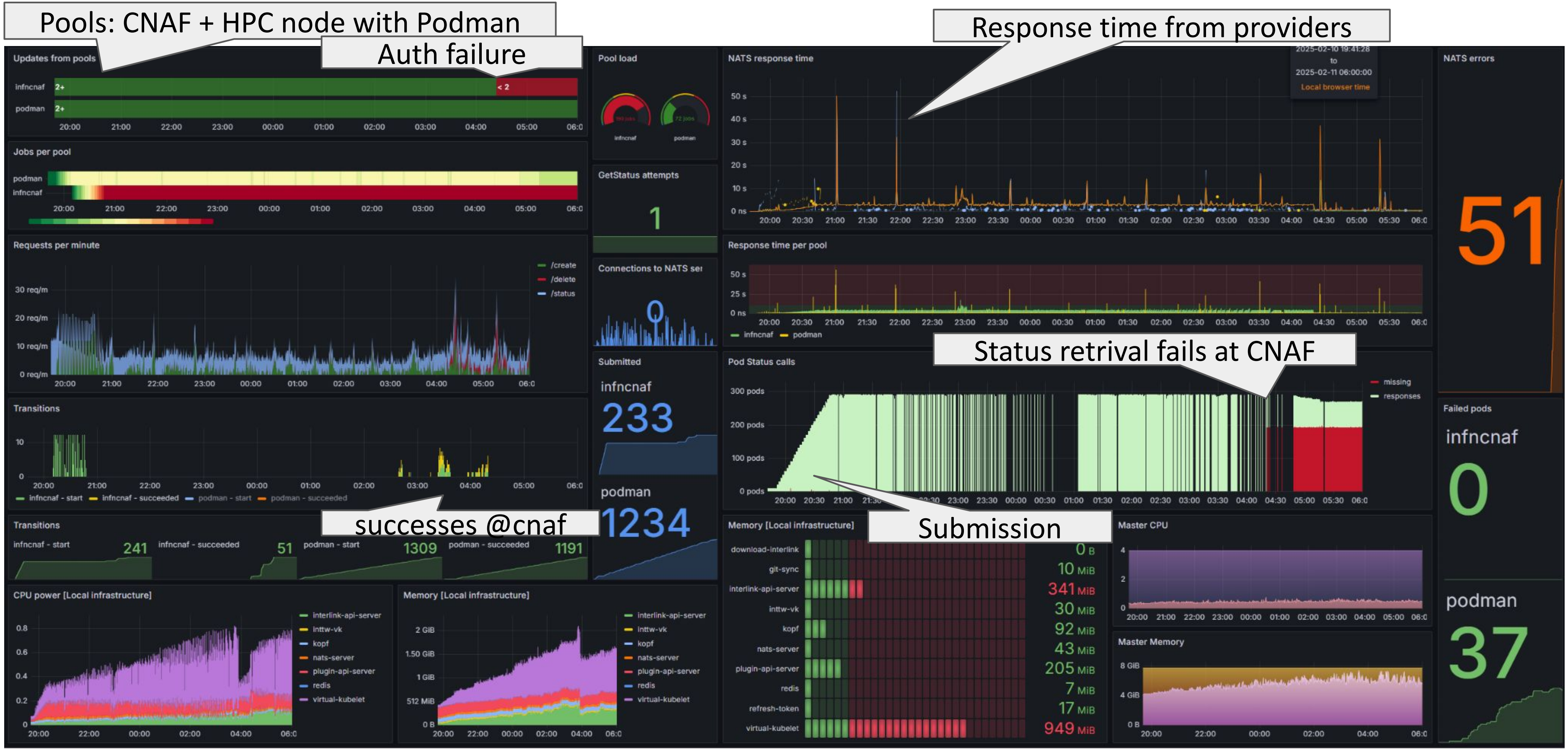
Tedious for development, any modification requires ***rebuilding*** *the OCI-image and* ***invalidating*** *the cache on the nodes.*

Significantly ***more difficult*** *to setup and submit.*

A required step to validate and demonstrate the infrastructure, but not the goal of this flagship.

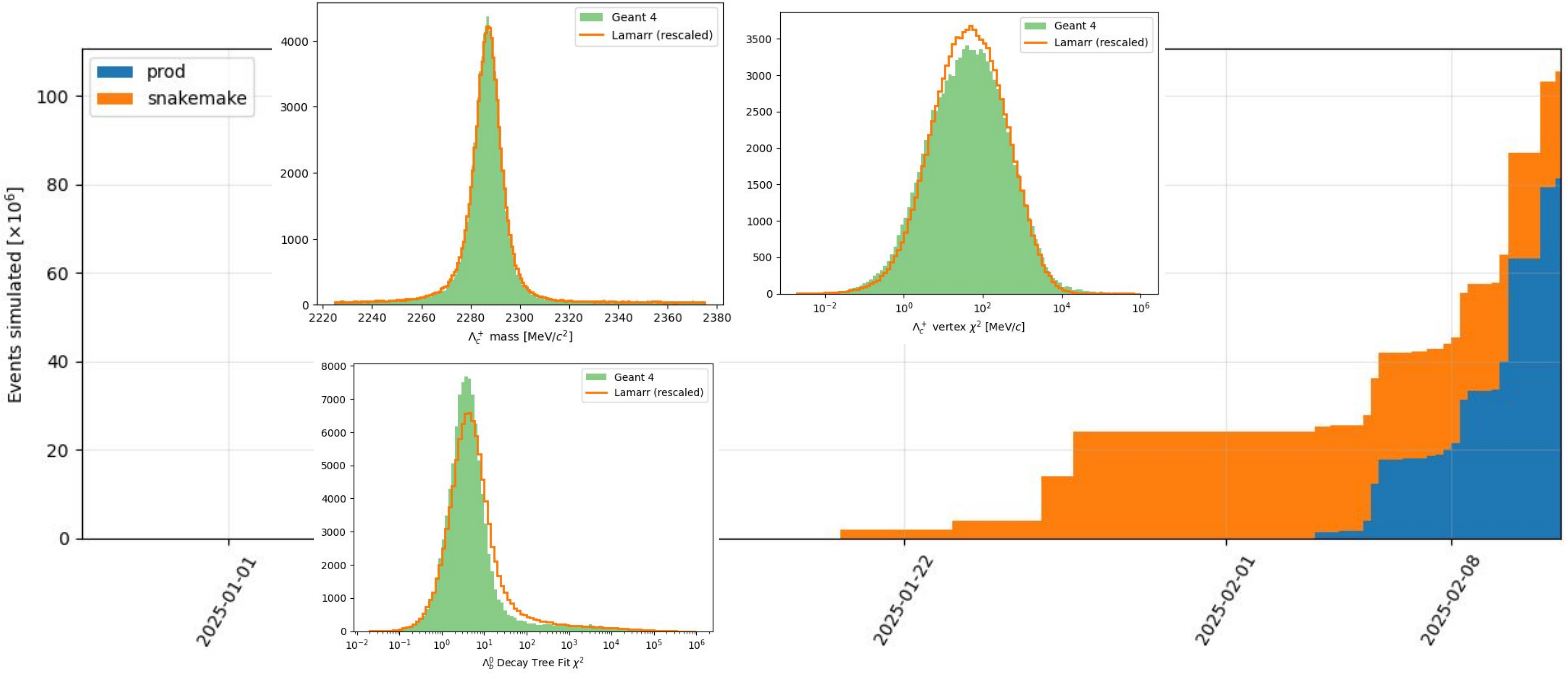
Finanziato dall'Unione europea NextGenerationEU
Ministero dell'Università e della Ricerca
Italiadomani
ICSC

Results (tonight)



Events produced in the exercise

As of today, we have produced 20M with Snakemake and 80M events with the “prod” workflow. Most of these events are Lambda_b decays generated with a Particle Gun.



Next steps

1. Continue with the production to **keep the system under pressure** and solve problems while they arise (also good for the KPI)
2. Integrate **Leonardo** in the new offloading setup
3. Integrate the **HPC Bubble** from Padova
4. Validate the infrastructure with ***"prod"*** workflow
5. Reintroduce snakemake and juicefs (and ***hetherogeneity***, ***GPUs***, …)

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

A recap on Physics Informed Machine Learning

Lead: *Alessandro Bombini*
[dedicated talk at the annual meeting]

In Physics Informed Neural Network, the network is the "*Ansatz*" of a problem defined by a Partial-Difference Equation (PDE).

The solution is reached by combining information on the PDE, experimental or simulated data, and boundary conditions.

The solution can be parametric and meshless.

We are studying two problems relative to the response of solid-state sensors with resistive electrodes, and two-fluid hydrodinamycs.

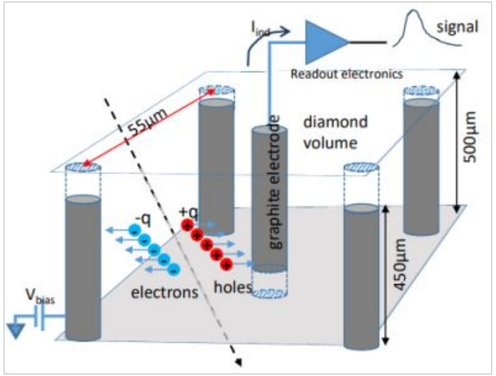
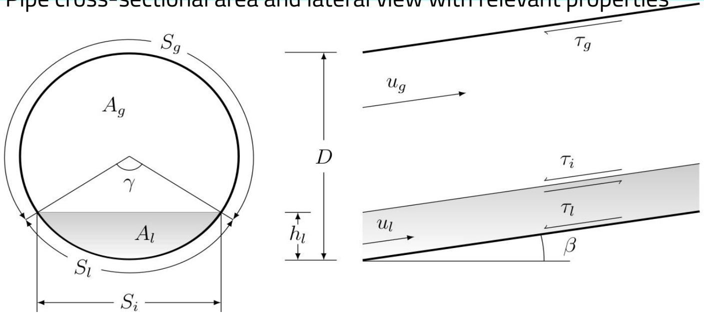


Figure 1.1 Geometry and working principle of 3D diamond detectors fabricated by electrode graphitization.
The dashed line represent a traversing particle depositing energy by ionization.

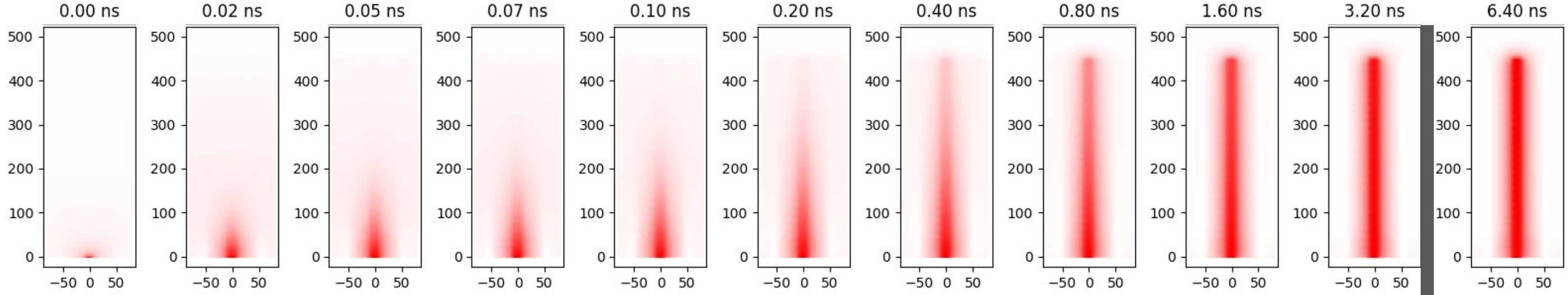
Fig. 2.1. Pipe cross-sectional area and lateral view with relevant properties



Physics Informed Neural Netorks as interpolator

Meshed time-dependent simulation with spectral methods (cheapest option, but still very expensive)

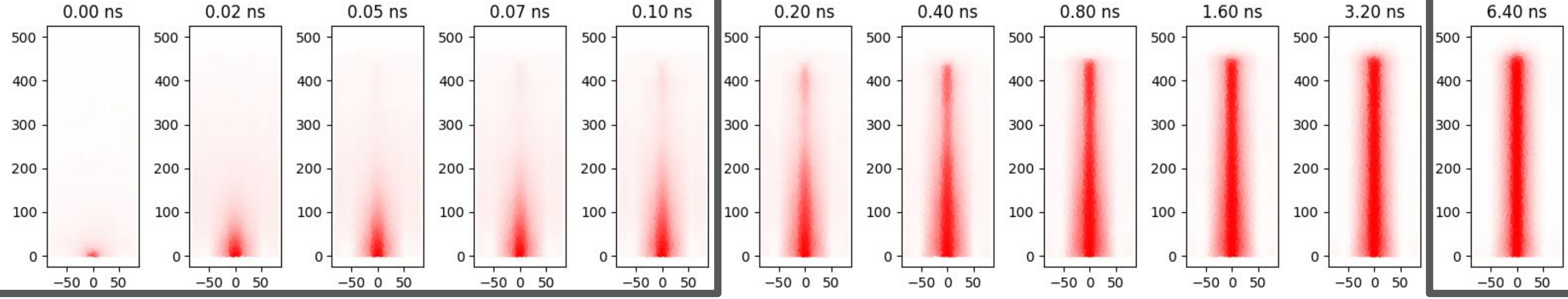
Static simulation (cheap)



Constrained

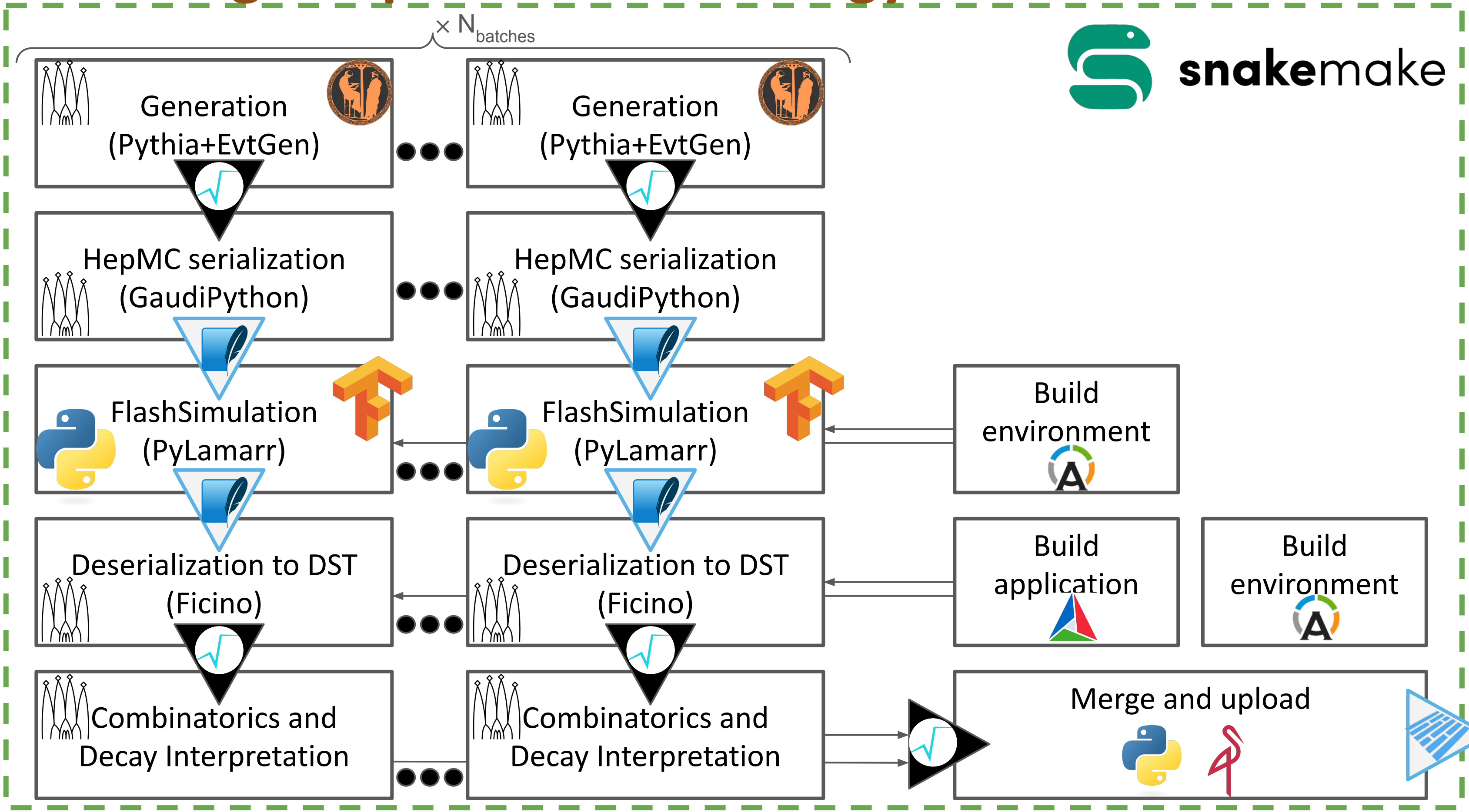
Constrained

Neural-Network interpolation



Meshless time-dependent simulation with a Neural Network

Recalling the production strategy



Resources

Pythia8 (full event)

Generates the whole proton-proton collision event, with pileup and spill-over.
Then processes all particles with Lamarr and Bender to produce nTuples.

1M events (on 50 parallel jobs) require:

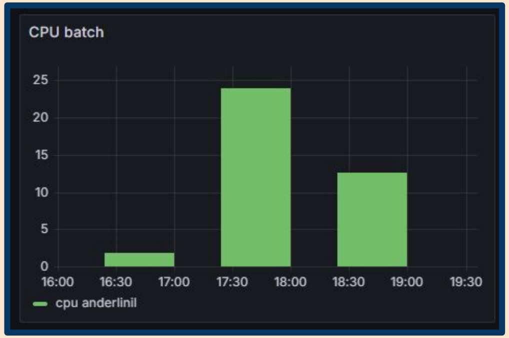
- O(48h) × 50 CPUs
- 0.8 TB of buffer in S3.

Particle Gun (signal-only)

Generates only the heavy hadron decay.
Then processes particles with Lamarr and Bender to produce nTuples.
Less tested than Pythia8 productions

1M events (on **up to** 50 parallel jobs) require:

- O(1h), *limited by submission latency*
- 4 GB of buffer in S3



Requests for the validation part

Resource	Full Request	Strictly required for KPI 1 (Full-Pythia option)
CPU on INFN Cloud	2 M CPU hours	2.4 M CPU hours*
GPU on INFN Cloud	4 H200 for 18 months	0
GPU on Leonardo Booster via InterLink	10000 hours	0
Storage	25 TB	10 TB

Preliminary

- 0.5 M hours from opportunistic borrowing from AI_INFN Platform

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

Status of the integration of INFN-T1 resources

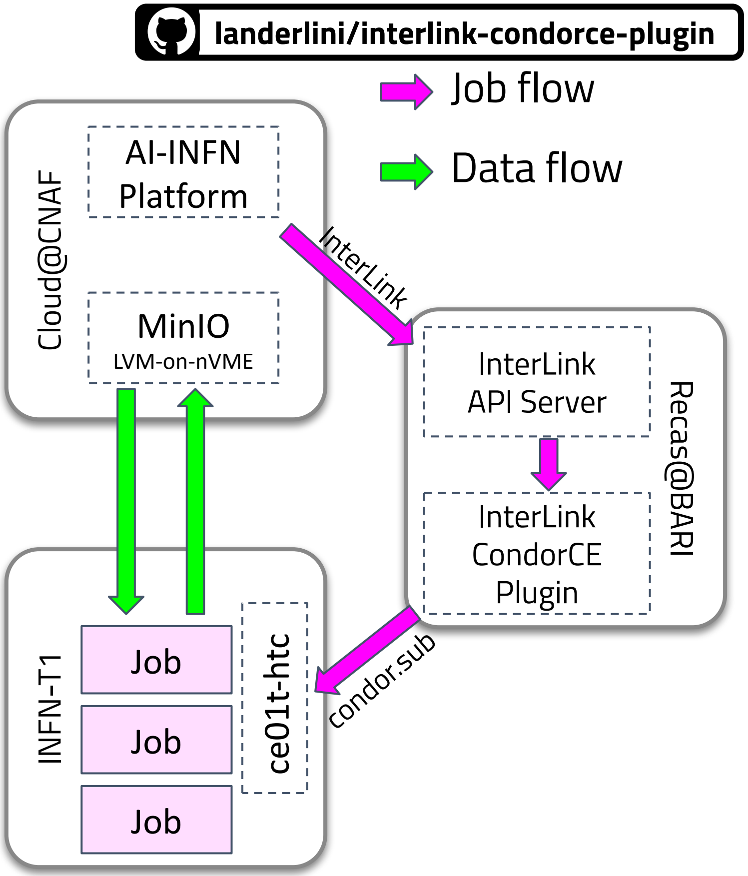
Developing the **HERD Computing Model**,
CNAF defined a CondorCE submitting jobs from remote locations through authentication.

Unfortunately, the CondorCE is not reachable from Cloud@CNAF for network policies, but it is, for example, from ReCaS@BARI.

We developed an **InterLink plugin** sitting in a VM in Bari, accepting InterLink submissions from Cloud@CNAF and forwarding them to CNAF Tier-1 test CE.

The plugin converts the **Kubernetes Pod** specifications into a (possibly rather long) shell script running **Apptainer** containers in multiple subprocesses.

Input and output data is managed through a self-managed **MinIO instance on LVM-on-nVME** hosted in **Cloud@CNAF**.



(re)Defining cvmfs and fuse volumes

Converting Pod's requests to access cvmfs or fuse data should be responsibility of the plugin, as different compute backend may be subject to different rules.

In CondorCE plugin I use generic annotations to define volumes.

For Leonardo, this require hacking the singularity submission command (very verbose).

```yaml
apiVersion: v1
kind: Pod
metadata:
  name: cern-vm-fs
  annotations:
    cvmfs.vk.io/my-volume: sft.cern.ch
spec:
  containers:
    - name: main
      image: ubuntu:latest
      command:
        - /bin/bash
        - -c
        - ls /
      volumeMounts:
        - name: my-volume
          mountPath: /cvmfs
          readOnly: True

  volumes:
    - name: my-volume
      persistentVolumeClaim:
        claimName: intentionally-not-existing
```

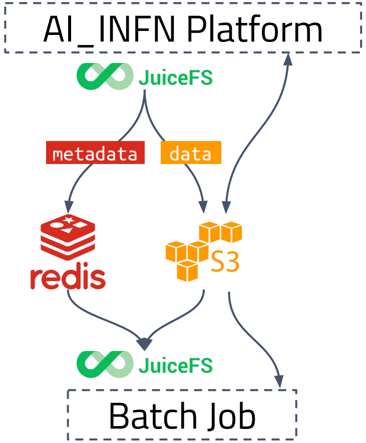
```yaml
apiVersion: v1
kind: Pod
metadata:
  name: fuse-vol
  annotations:
    fuse.vk.io/my-fuse-vol: |
      cat << EOS > /tmp/rclone.conf
      [example]
      type = local
      EOS

      # Mimic a remote
      mkdir -p /tmp
      echo "hello world" > /tmp/file.txt

      # Mount the remote
      rclone mount2 \
        --config /tmp/rclone.conf \
        --allow-non-empty example:/tmp \
        $MOUNT_POINT

spec:
  containers:
    - name: main
      image: rclone/rclone:latest
      command:
        - cat
      args:
        - /mnt/fuse-vol/file.txt
      volumeMounts:
        - name: my-fuse-vol
          mountPath: /mnt/fuse-vol
  volumes:
    - name: my-fuse-vol
      persistentVolumeClaim:  # deliberately fake pvc
        claimName: csi.example.com
```

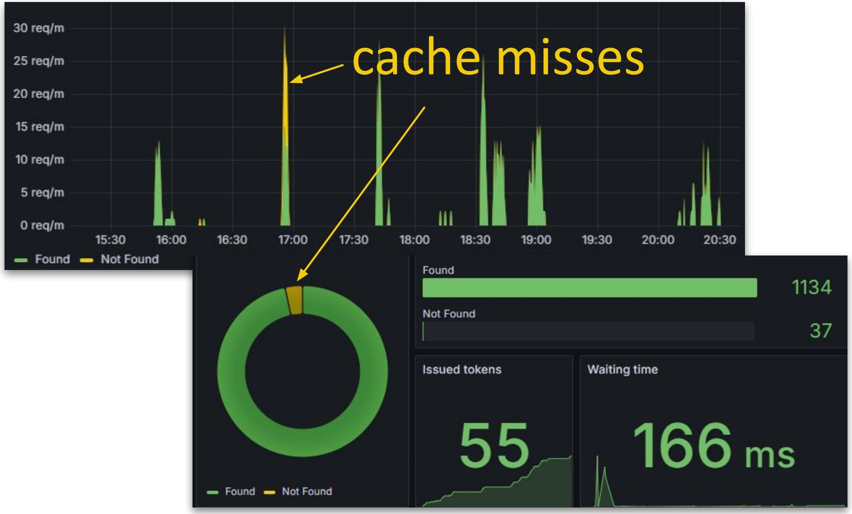
Solving the distributed cache problem

Focus on data flow

A **shared virtual file system** is mounted by the condor nodes with fuse using JuiceFS.

JuiceFS falls back on **MinIO** for the data and **Redis** (part of the AI-INFN platform) for the metadata





[ShubProxy]

Downloading and building docker images into SIF for each jobs

- ➢ would cause a periodic bans of CNAF by DockerHub;
- ➢ cause large inefficiency in short jobs.

We deployed a simple web application defining a shared cache.

If the image is not available in S3, the web app schedule its build, otherwise it return the built artifact from cache.

Status of the integration with Leonardo

The slurm plugin in production in Leonardo, does not accept Pod requests from the Flash Simulation workflow.

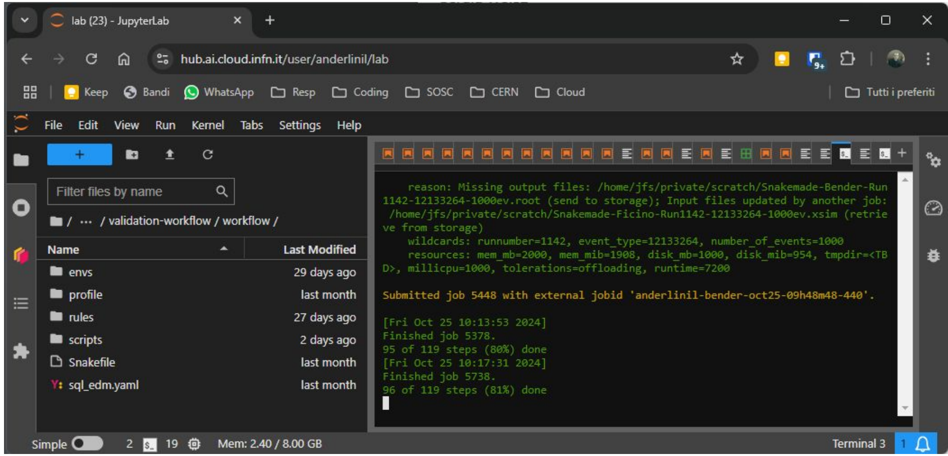
All the building blocks were tested separately and we expect no fundamental reason for the plugin not to work.

Still, some polishing would be needed, probably in a joint debugging session.

Alternatively, we may try to use the CondorCE plugin submitting to slurm.

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

Combining CNAF Tier-1 and CINECA Leonardo resources



- Kubernetes cluster
  - AI_INFN Platform (RKE2 with Kubernetes 1.27)
  - Virtual Nodes installed manually (no helm chart)
  - Snakemake as workload manager
- Tier1 setup
  - CondorCE (originally developed for HERD) mapped to ce01t
  - InterLink server and dedicated plugin running in a VM in ReCaS
- Leonardo setup
  - Slurm submission from edge node icsc01
  - Official interlink slurm plugin

```
(miniconda3)[lucio@pclhcb06 v3]$ k get nodes
NAME                  STATUS   ROLES
hub-a100-2            Ready    <none>
hub-a100-3            Ready    <none>
hub-a102-b            Ready    <none>
hub-cpu-2             Ready    <none>
hub-master            Ready    control-plane,etcd,master
hub-rtx-2             Ready    <none>
hub-rtx-3             Ready    <none>
hub-storage           Ready    <none>
infn-t1               Ready    agent
leonardo-virtual-node Ready    agent
```

vk → infn-t1
vk → leonardo-virtual-node