# The HERD computing model: status and exploration of CLOUD solutions for cosmic ray data analysis

D. Ciangottini, S. Dal Pra, M. Duranti, V. Formato, N. Mori, D. Spiga

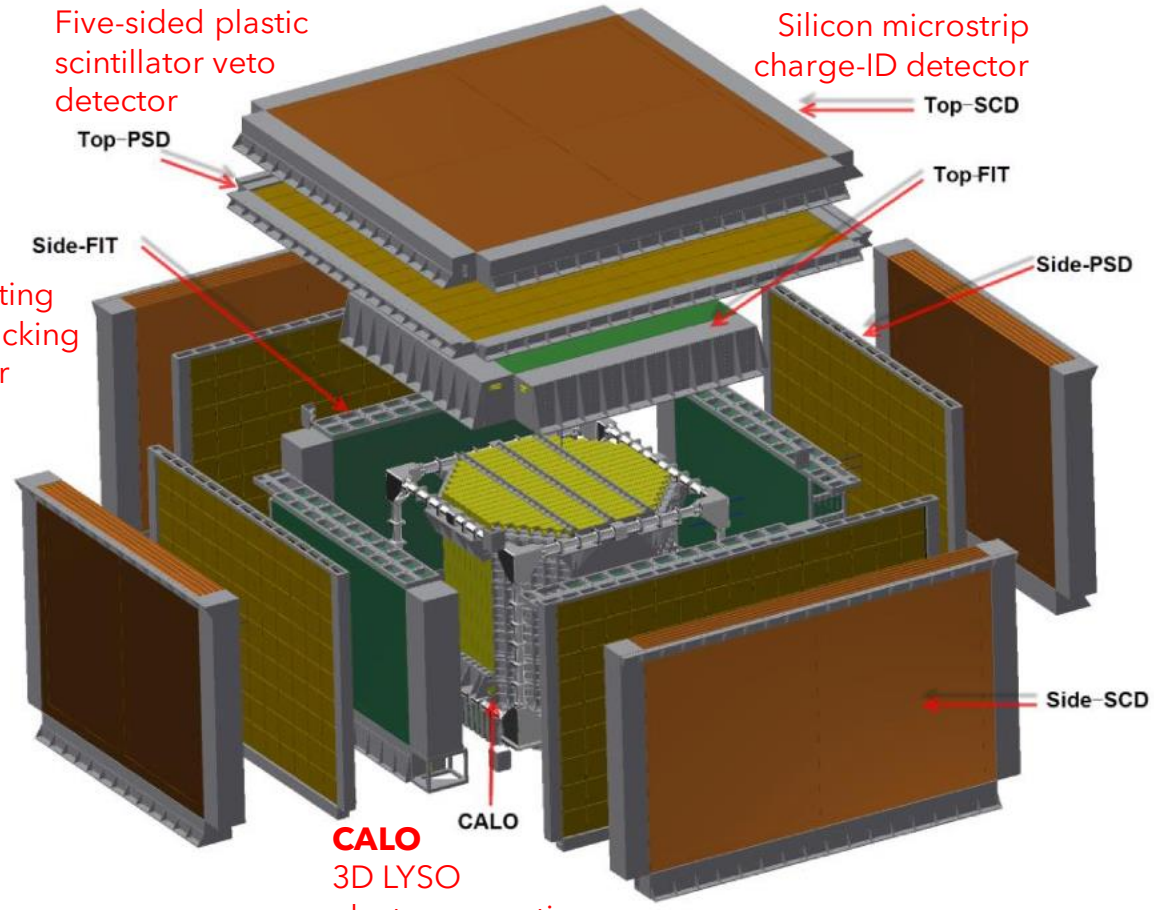
Workshop CCR – Loano - 22/05/2023

INFN ROMA TOR VERGATA

HERD

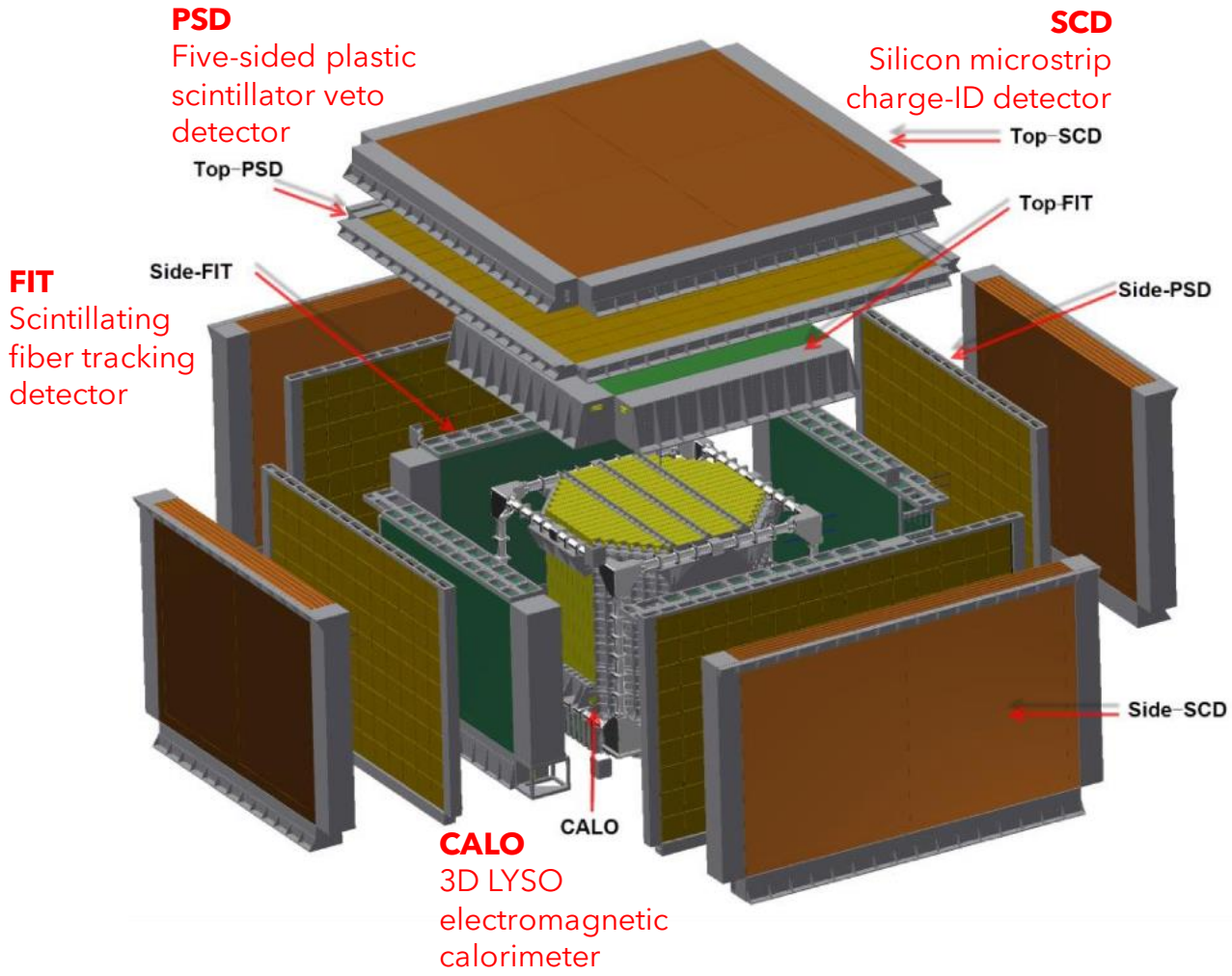
# HERD

- Flagship and landmark scientific experiment, China-led large international collaboration
  - Main Scientific Objectives:
    - Dark matter: dark matter search with unprecedented sensitivity
    - Cosmic-ray: Precise cosmic ray spectrum and composition measurements up to the knee energy
    - Gamma-ray: Gamma-ray monitoring and full sky survey
- Foreseen to operate in space, on board the Chinese Space Station
- Charged CR physics but also γ-ray physics
- ~ O(1M) read-out channels
- The detector is designed to be "isotropic" and accept CR from all (5) the sides



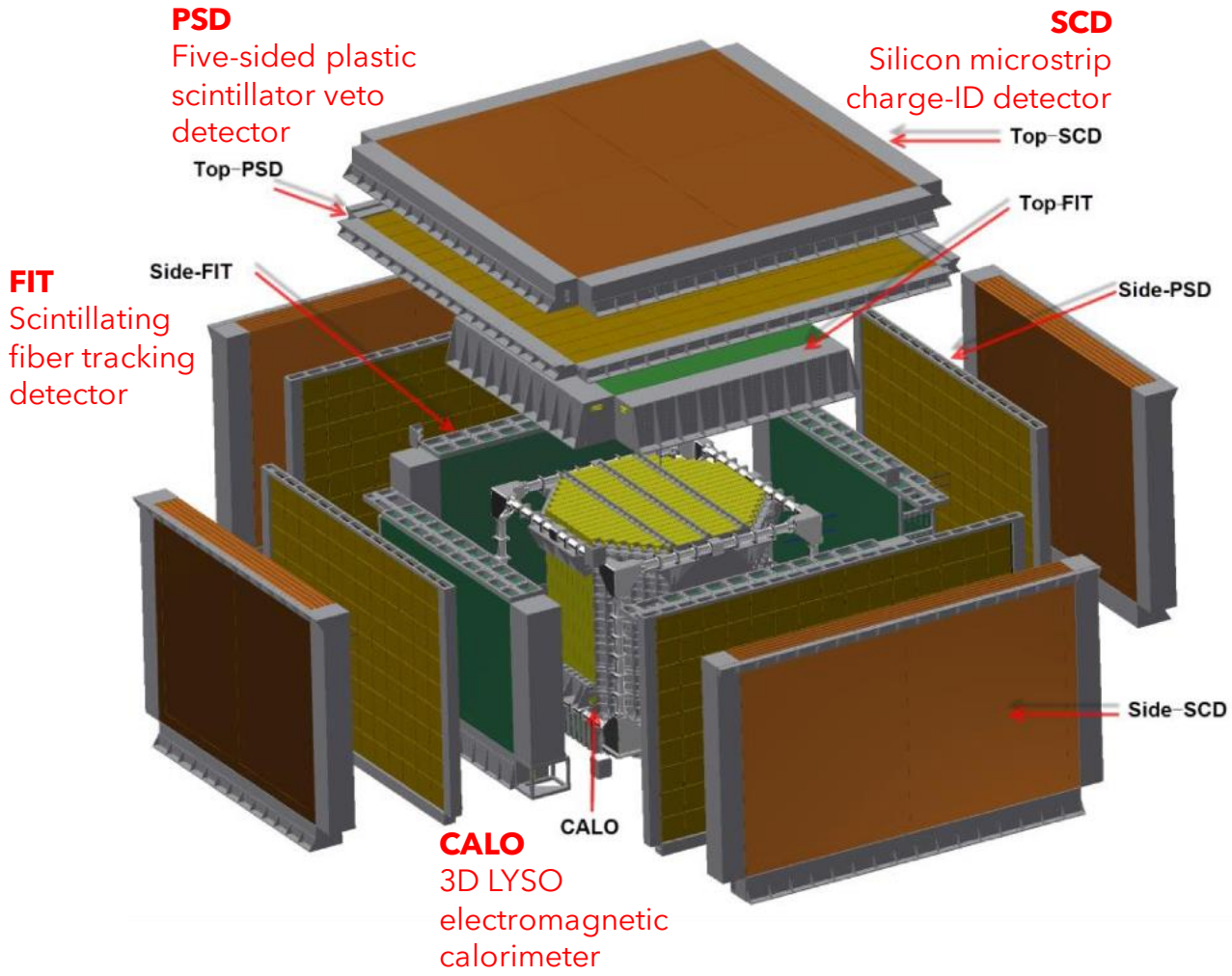
# HERD

- *Currently in advanced design and prototyping stage*
  - Possibility to start developing code and computing model from scratch
- *Facts about computing models for cosmic-ray experiments:*
  - Exp. "size" << LHC
    - "Easy", but…
  - High demand of raw computing power in some scenarios
    - E.g. upper energy limit for MC simulations in the PeV region
  - Difficulties in exploiting some computing optimization techniques
    - E.g. optimized workloads like fast MC simulations may not be suitable for assessing the e/p rejection power with sufficient accuracy
    - Small community with relatively little manpower to devote to this topic
  - Highly dynamic use of computing resources (short-period bursts)
    - Due to e.g. small number of involved people leading to significant usage fluctuations

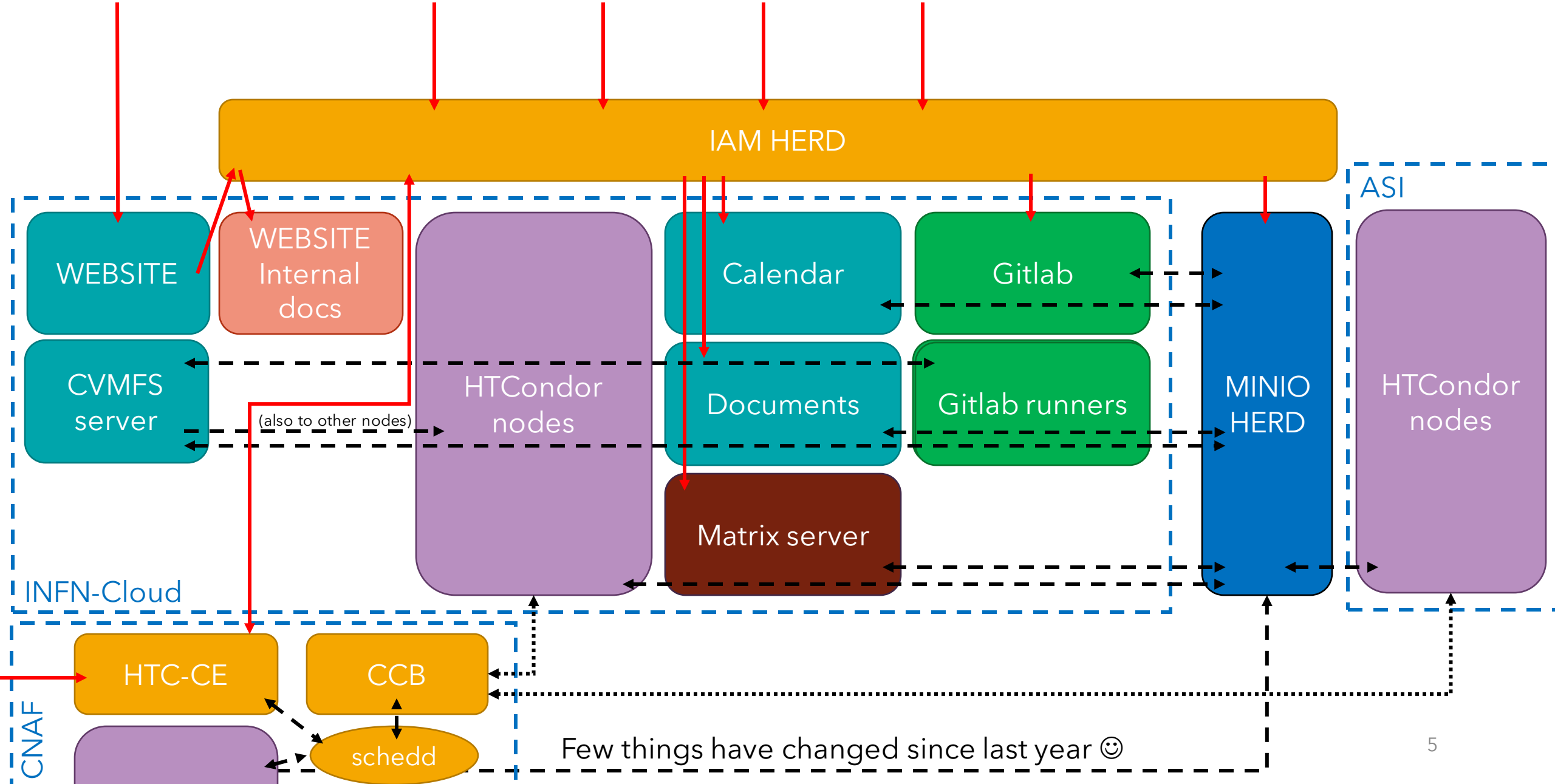


# HERD

- *The cloud can provide some solution:*
  - Add/remove resources with minimal overhead to dynamically cope with high/low demand periods
  - Efficiently exploit opportunistic resources with cloud-native solutions
  - Deploy self-hosted, self-managed services
- *The computing model must be designed to fully profit of the cloud*
  - E.g. to maximize the usage of opportunistic resources by minimizing the set-up period after the resources are made available in the cloud
- *Towards a fully-cloud based model for HERD:*
  - R&D and prototyping work
  - Based on INFN Cloud infrastructure and resources



# HERD – infrastructure model

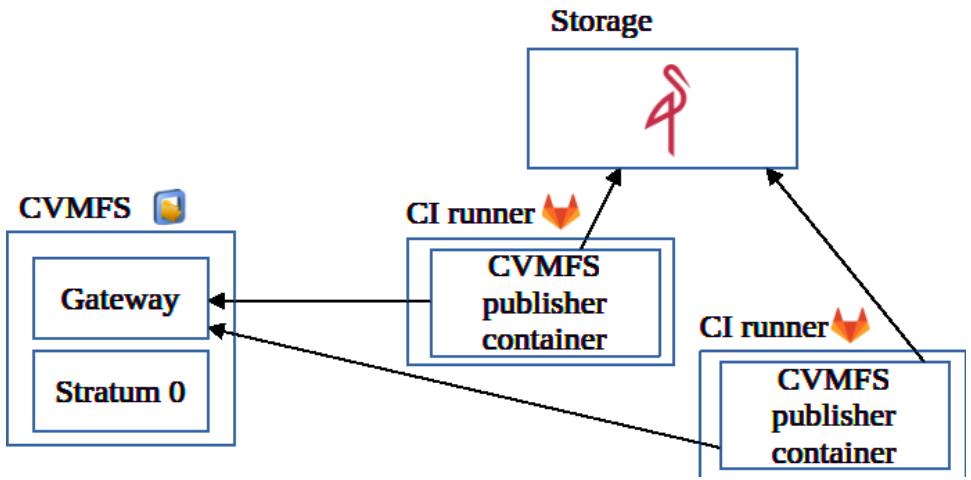


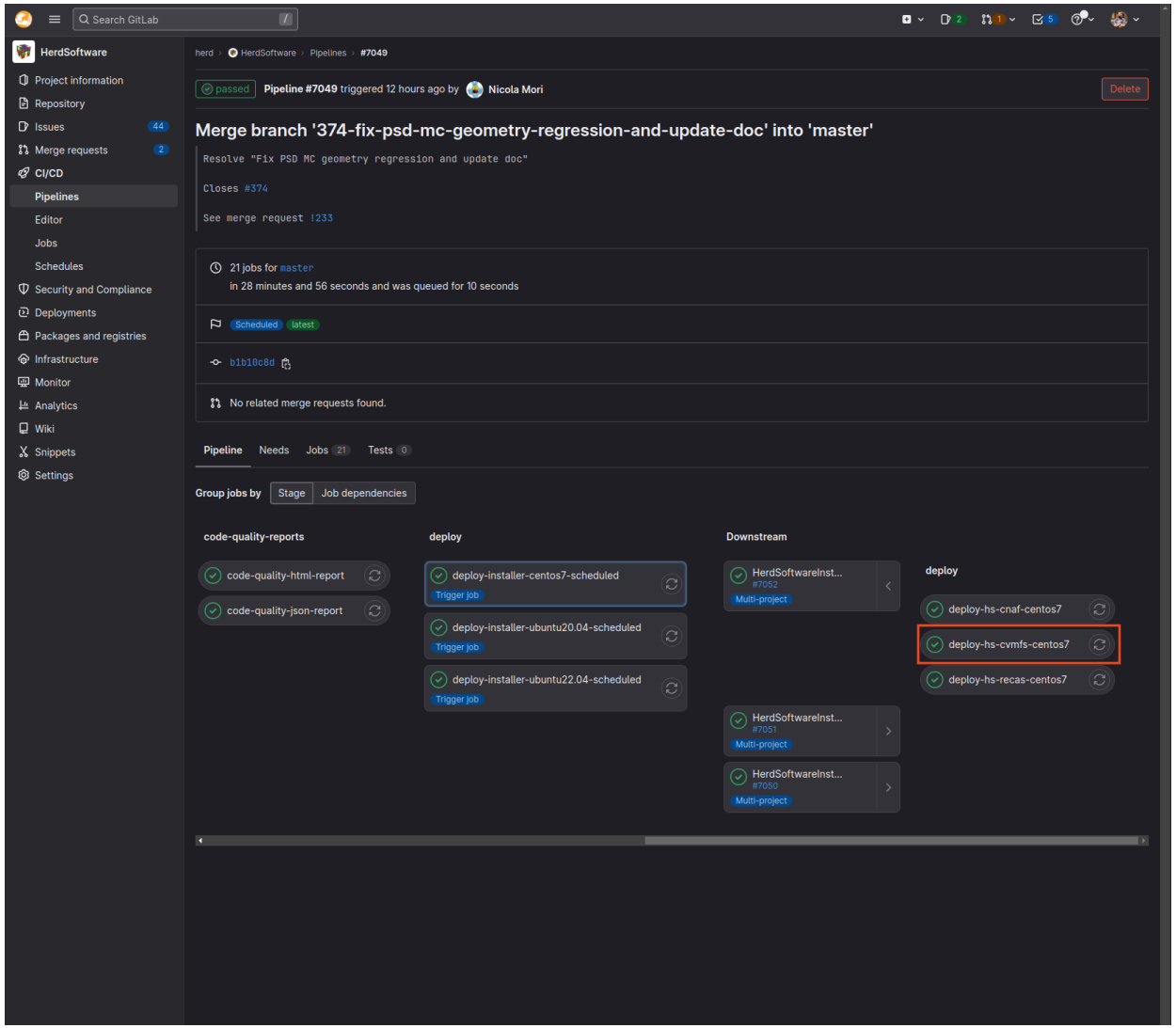
# 2022: Feedback and next steps

- Needed:
  - CVMFS as-a-service. ✅
    To continue testing the HTCondor service (and eventually use it in production) we need a CVMFS server for software distribution.
- Desirable:
  - Self-managed DNS on select subdomains
    To ease the workflow of deploying new services or maintaining existing ones
- Feedback:
  - Support team with high availability for critical, non-self-managed parts of the infrastructure ✅
    We had several issues with our Minio backend which we couldn't debug/fix on our own. We relied on a single contact person during the test phase but this solution clearly doesn't scale to production.
  - The whole INFN-Cloud infrastructure feels a bit "user-centric" rather than "team-centric"
    For example: within the HERD tenant each user can see and manage only his own deployments
- Next:
  - Continue testing and improve the HTCondor workflow for analysis
    Now testing S3 storage solutions ✅, need CVMFS integration ✅, flocking, user-mapping, …
  - Backport hand-made deployments in INFN-Cloud dashboard (*e.g.* Gitlab as-a-service)

# CVMFS

- Self-deployed CVMFS server on INFN Cloud resources
- Activities synergic with DataCloud WP6
- Using S3 storage as a backend
- Fully integrated with Gitlab CI/CD for deployment of new SW releases / development branch
  - Triggered by commits on release tags (x.y.z) or on master branch

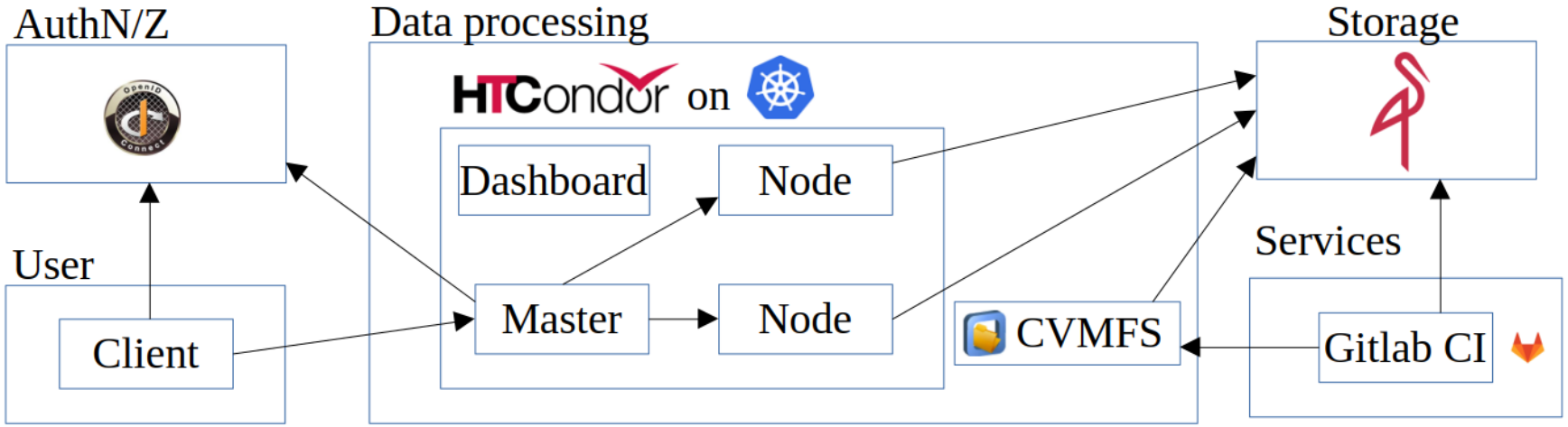


# Batch system

We continued the tests with the on-demand HTCondor cluster on K8S

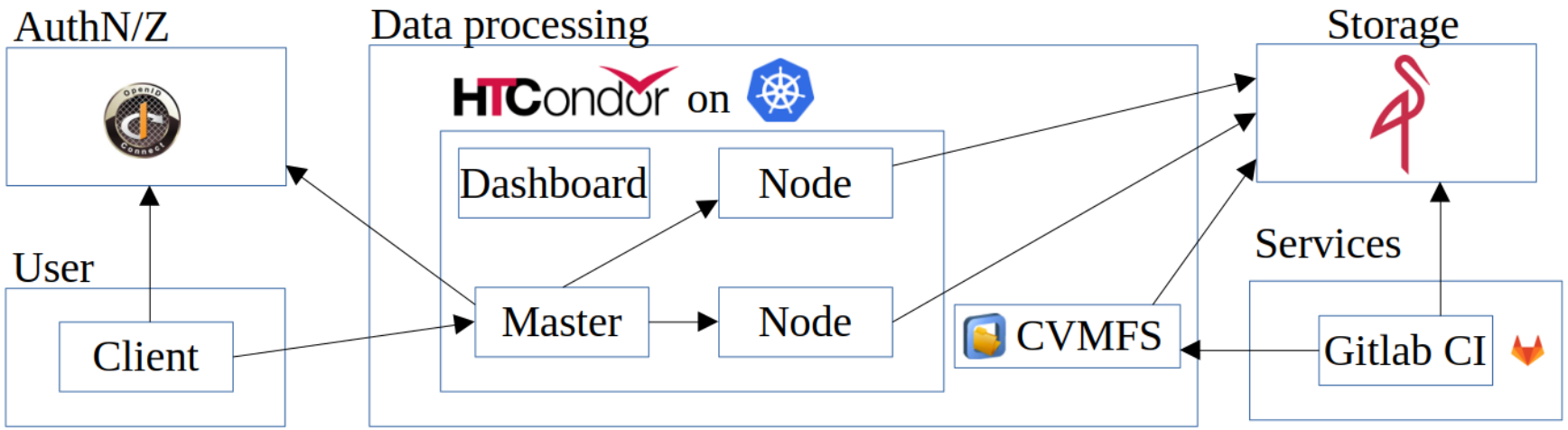
- Realized a docker image for a HERD HTCondor client
  - Based on the htcondor/submit image
  - Additional dependencies added:
    - Rclone
    - Oidc-agent
    - Boto3-sts
    - Cvmfs client
    - Devtools (gcc, cmake, boost, etc…)
  - The user should be able to build and submit its SW using this image



# Batch system

We continued the tests with the on-demand HTCondor cluster on K8S

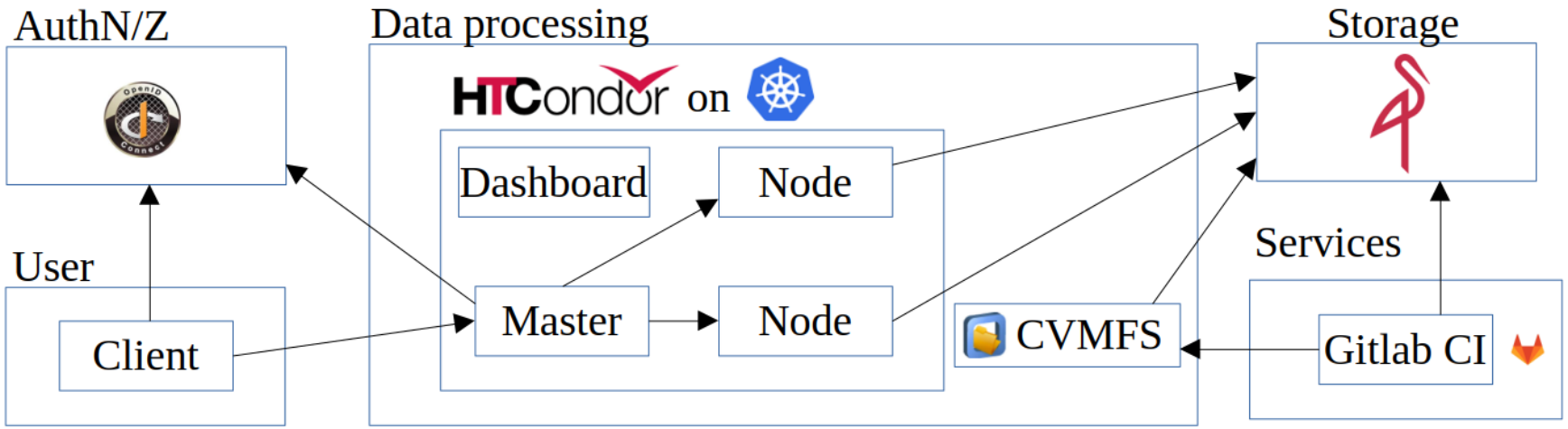
- Jobs perform I/O from/to the S3 storage using pre-signed URLs
  - psURLs are retrieved at job submission using the user's token and are stored on the local storage of the WN
  - Currently the psURLs cannot last more than 7 days, so a solution is needed for long-lasting jobs (e.g. ~PeV simulations)
- WNs mount the HERD CVMFS server automatically
  - On each K8S node of the underlying cluster a daemonset is added exposing the CVMFS mount as a volume to all the pods running on that node



# Batch system

Tested real life pipelines:

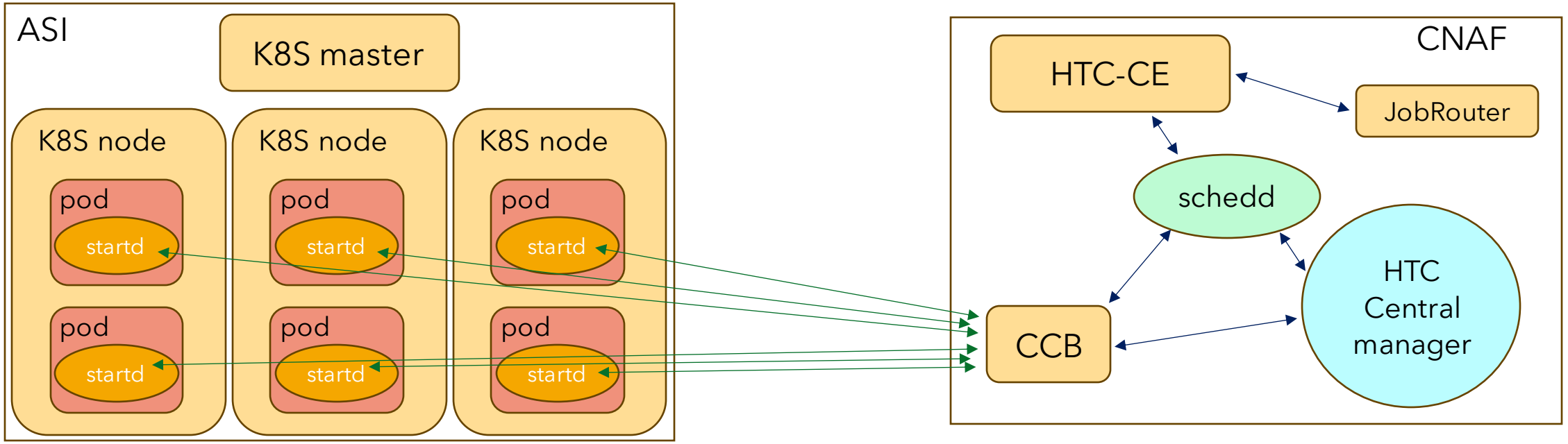
- A launch script retrieves the psURLs for all input and output files, prepares the Condor submit file and the job script, and submits the job cluster
  - Simulation jobs: standard Geant4 simulation, output file is copied on S3 with curl using a psURL
  - Digitization jobs: the job script reads the simulation input file from S3 using curl and a psURL, executes the digitization, and stores the digitized output file on S3 using a psURL
  - Reconstruction: same workflow as digitization, with a digitized file as input
  - Analysis: same workflow as reconstruction, with a reconstructed file as input



# Batch system: WIP

Currently working on:

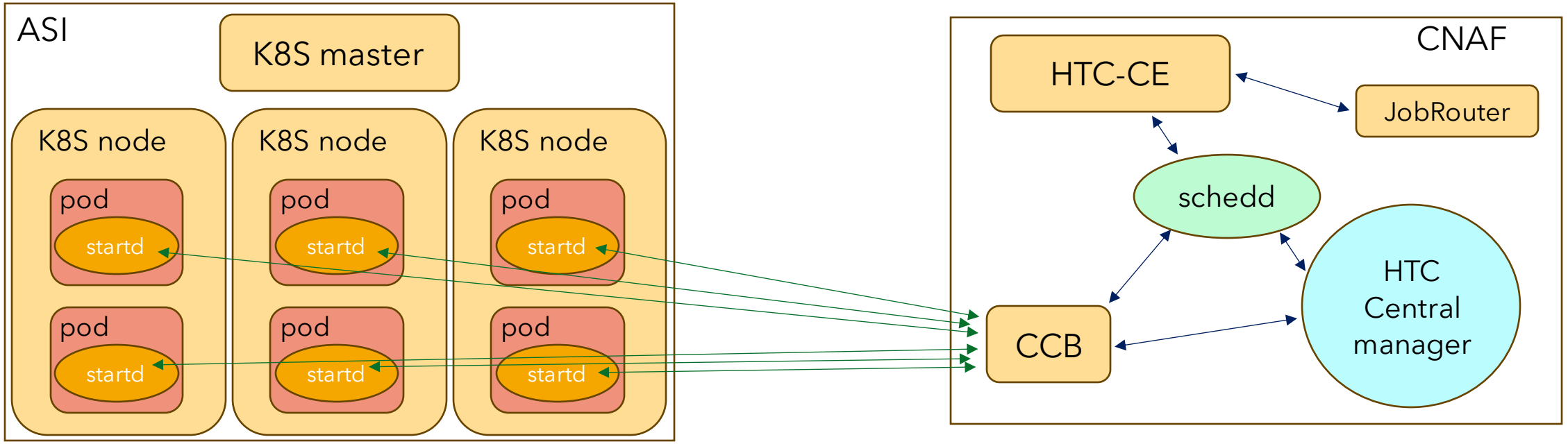
- Include opportunistic resources and merge them with resources @ T1
  - Same workflow as in CNAF-Reloaded (see talk by S. Dal Pra @ CHEP 2023): HTCondor-CE element managing additional external nodes
    - Submission to HTC-CE with custom attribute.
    - Attribute is translated to relevant ClassAd by JobRouter
    - Remote WN spawn startd processes that join the CNAF POOL through the CCB. All traffic to remote nodes goes through CCB.
  - Tested with VM @ Recas running a dockerized WN
  - Setting up a dedicated K8S cluster in ASI (activity under ASI-INFN Agreement No. 2021-43-HH.0) where working nodes will be spawned



# Batch system: WIP

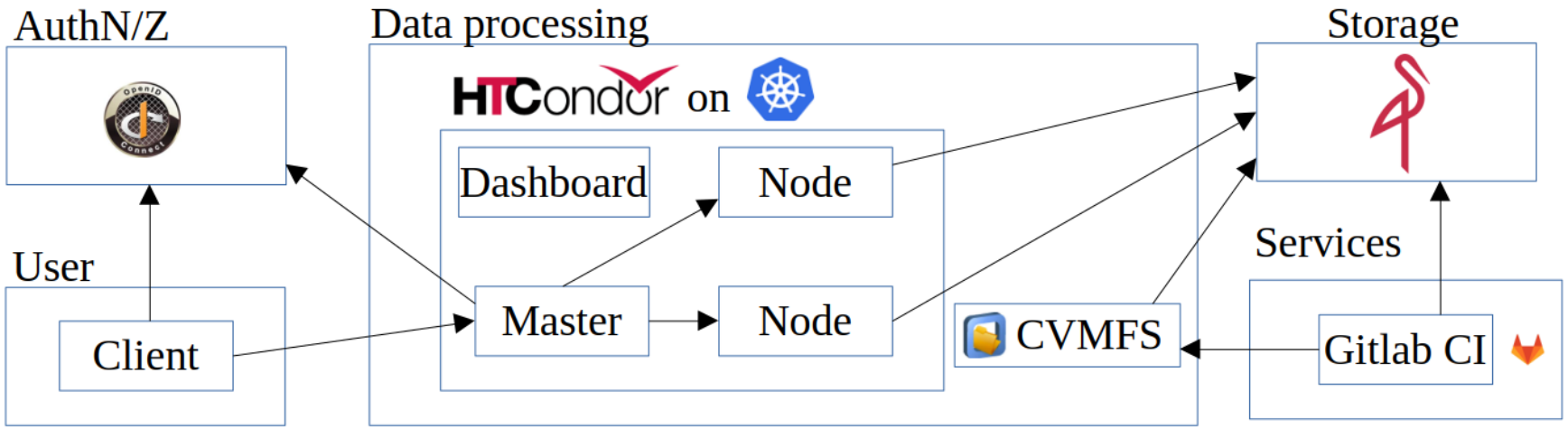
Currently working on:

- Include opportunistic resources and merge them with resources @ T1
  - Since all traffic goes through CCB it's recommended to minimize the usage of HTCondor file transfer mechanism
    - Most of the SW stack is on cvmfs, but a small part of the job payload (user code) is currently transferred in this way
    - Exploring automated solutions to deploy also user code on cvmfs or on a remote host (and maybe copy via gfal)
  - In the next future we could adapt the auto-scale machinery used for M100 to handle multi-tenant setup (where each experiment requires a custom WN image)



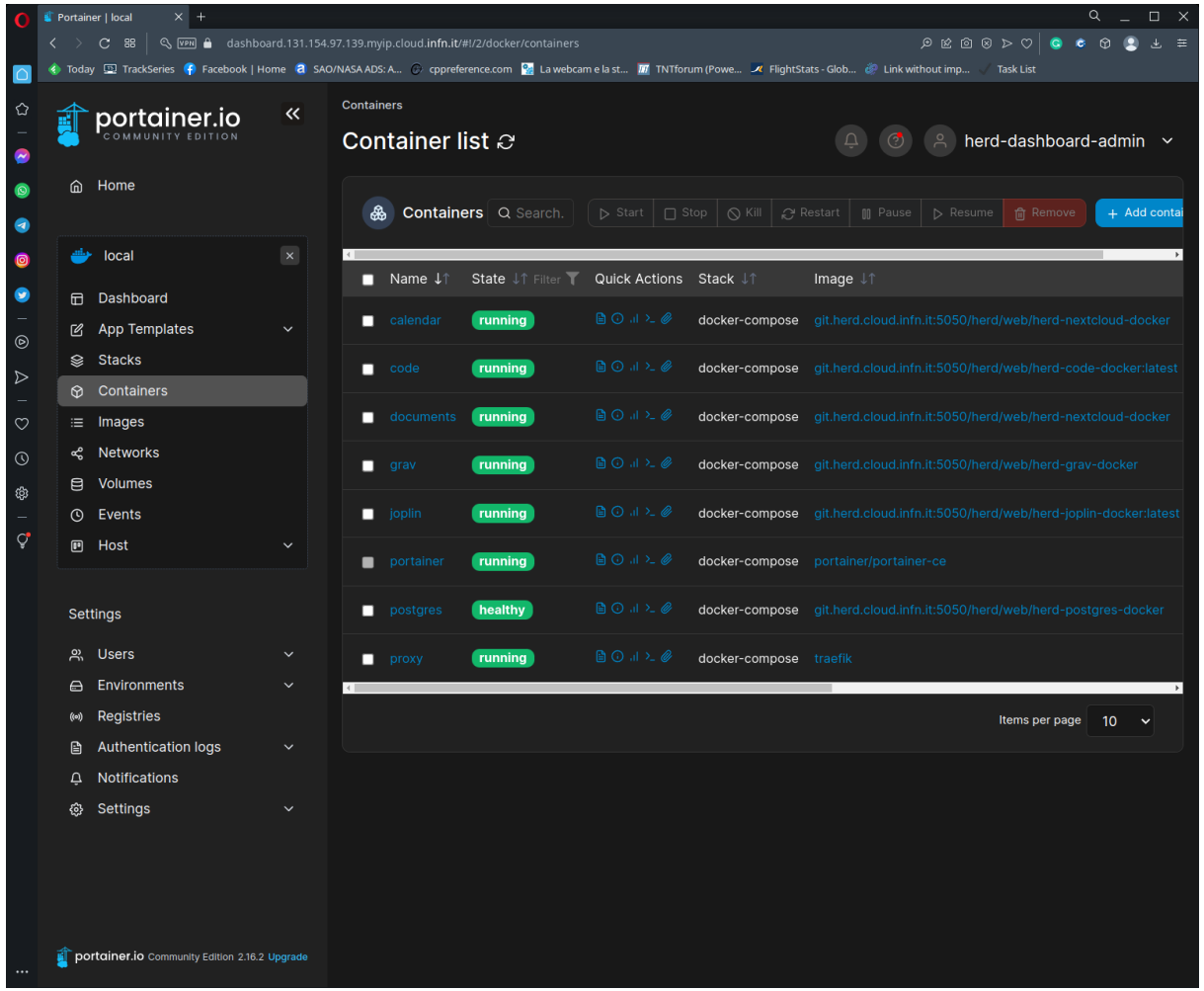
# Batch system: Open issues

- Performance still to be investigated
  - Sub-optimal storage testbed (MinIO on VMs with Ceph backend storage)
  - Single-site setup, no multi-site optimizations yet
- S3 access control for long-queueing/long-running jobs
  - psURLs validity currently limited to 1 week
- Distribution of user's code (CVMFS?)
  - Using HTCondor file transfer at the moment
- Integration with "conventional" pledged resources: ongoing



# Web infrastructure

- Over the time we have set up an extensive set of web applications in support of the activities of the collaboration.
- Peculiarities of the HERD collaboration:
  - Cosmic-ray experiment → no umbrella organization like CERN → no institutional services available to every collaboration member
  - Chinese institutions → no access to free tools from e.g. Google
- Solution: self-host the needed services at collaboration level
  - The cloud approach offers a valid approach for this purpose
- Fully-containerized web applications running on cloud VMs
  - Currently planning a migration to a K8S cluster to allow automatic re-deploy and load balancing



# Authentication

Goal: everyone in the collaboration should have access to the experiment resources.

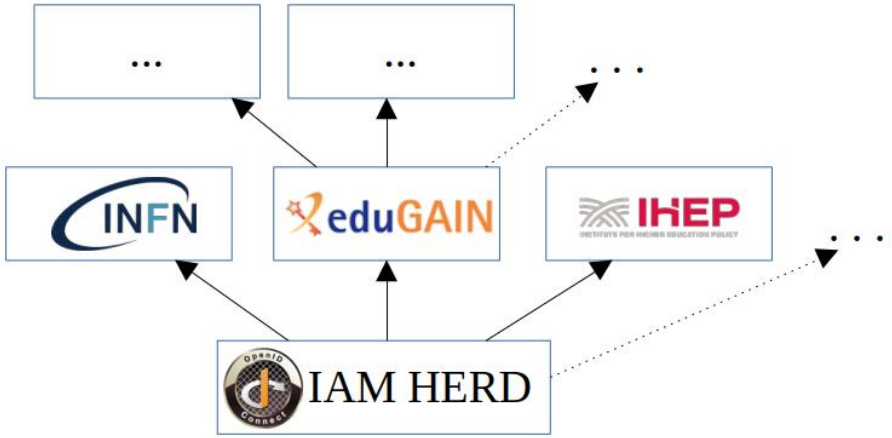
Issues:

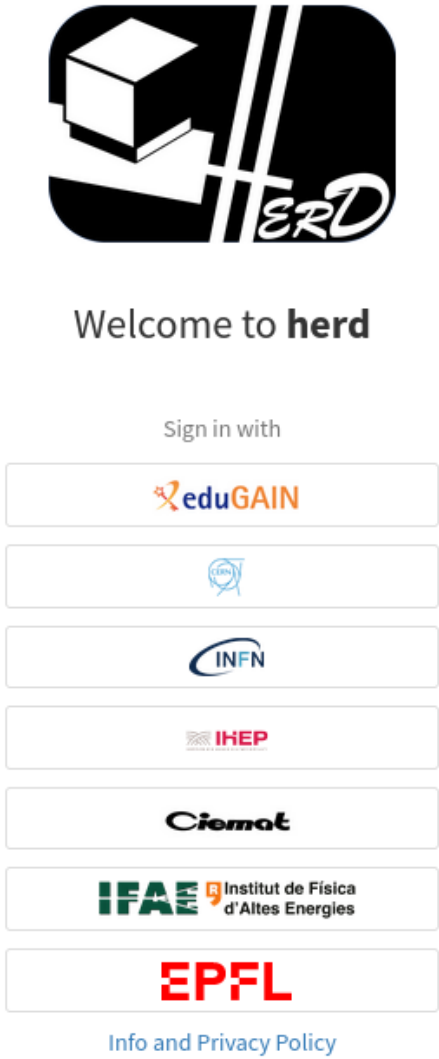
- Not everyone has a CERN account
- Not everyone has a INFN-AAI account
- Not everyone has access to Google

Solution: Single Sign On based on a dedicated INDIGO-IAM instance (IAM HERD)

Federated with institutional SSO services directly and through EduGAIN

OpenID Connect protocol



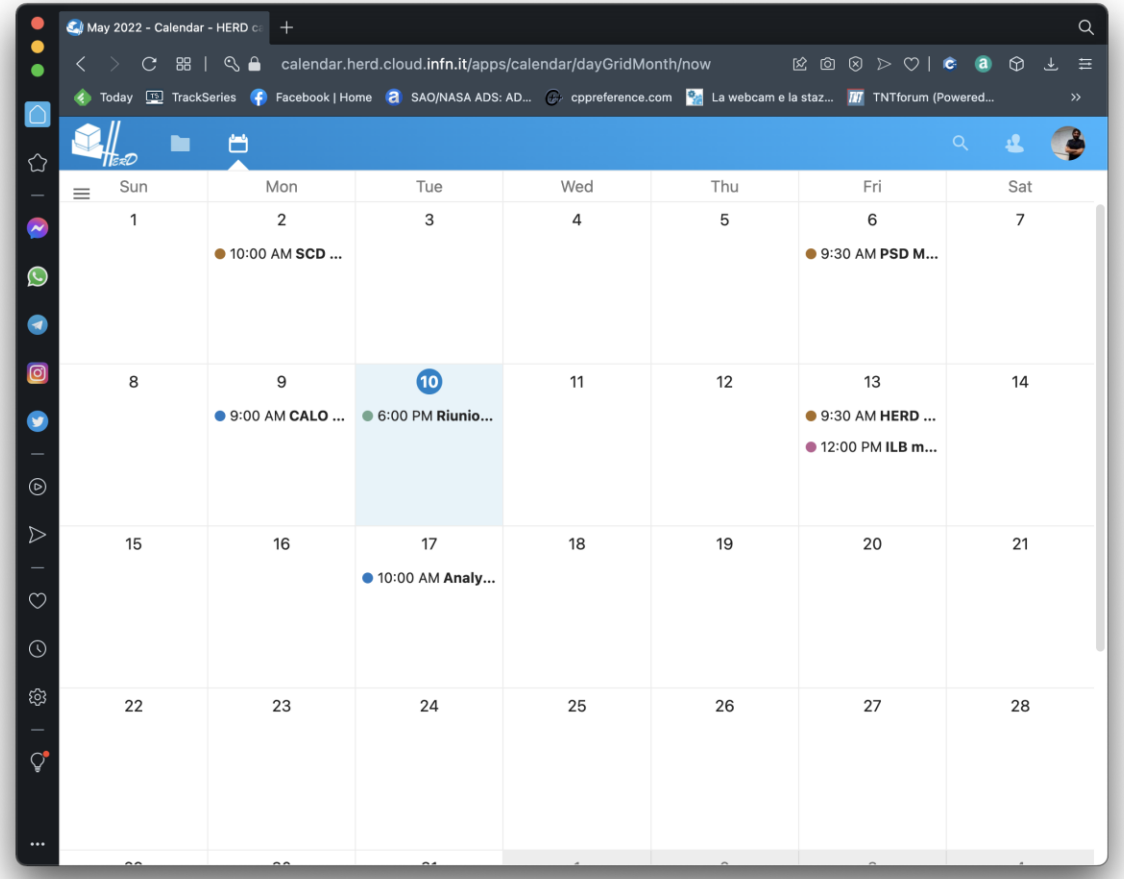
# HERD – services

Requirement: we need several services to ease scheduling of day-to-day operations and/or meetings, activities, as well as document sharing, and more.

Issue: Almost every mainstream tool is unavailable to colleagues in China.

Solution: self-hosting of all services:

- Calendar (based on Nextcloud)



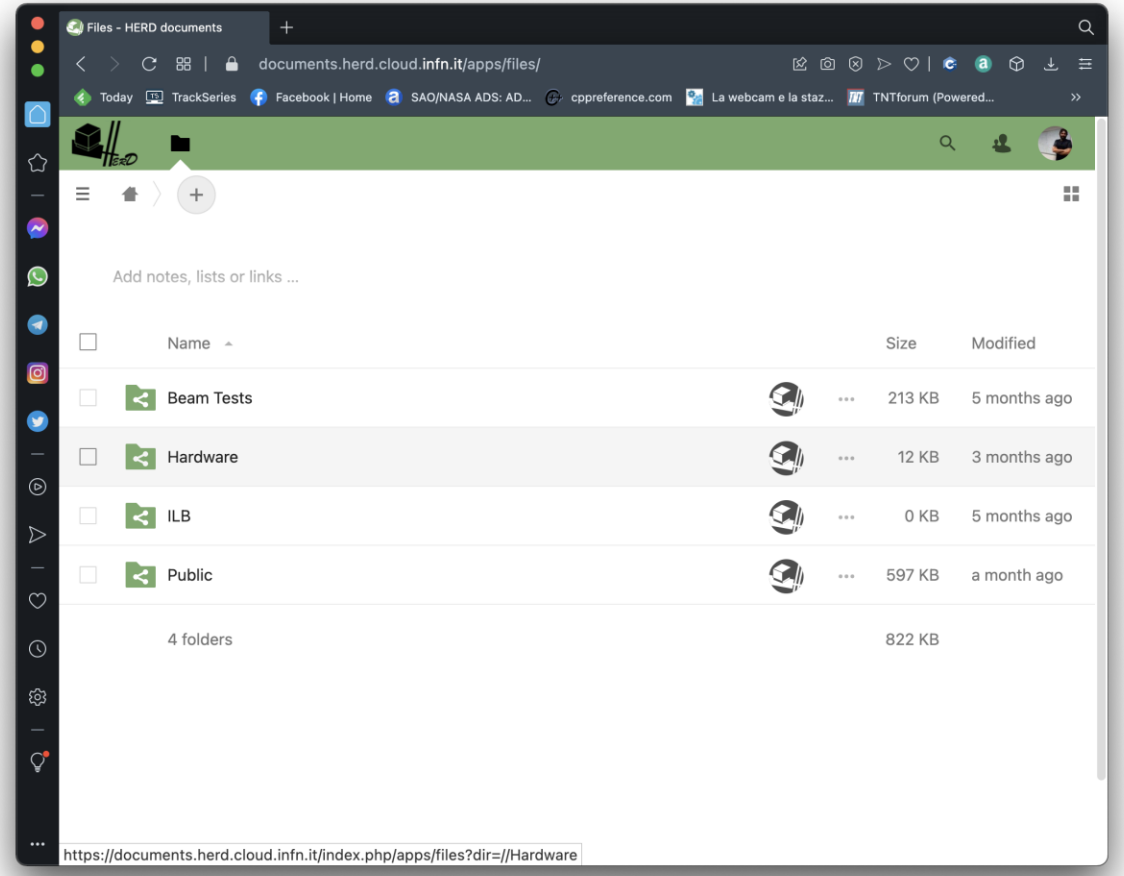
# HERD – services

Requirement: we need several services to ease scheduling of day-to-day operations and/or meetings, activities, as well as document sharing, and more.

Issue: Almost every mainstream tool is unavailable to colleagues in China.

Solution: self-hosting of all services:

- Calendar (based on Nextcloud)
- Document server (based on Nextcloud + Collabora)



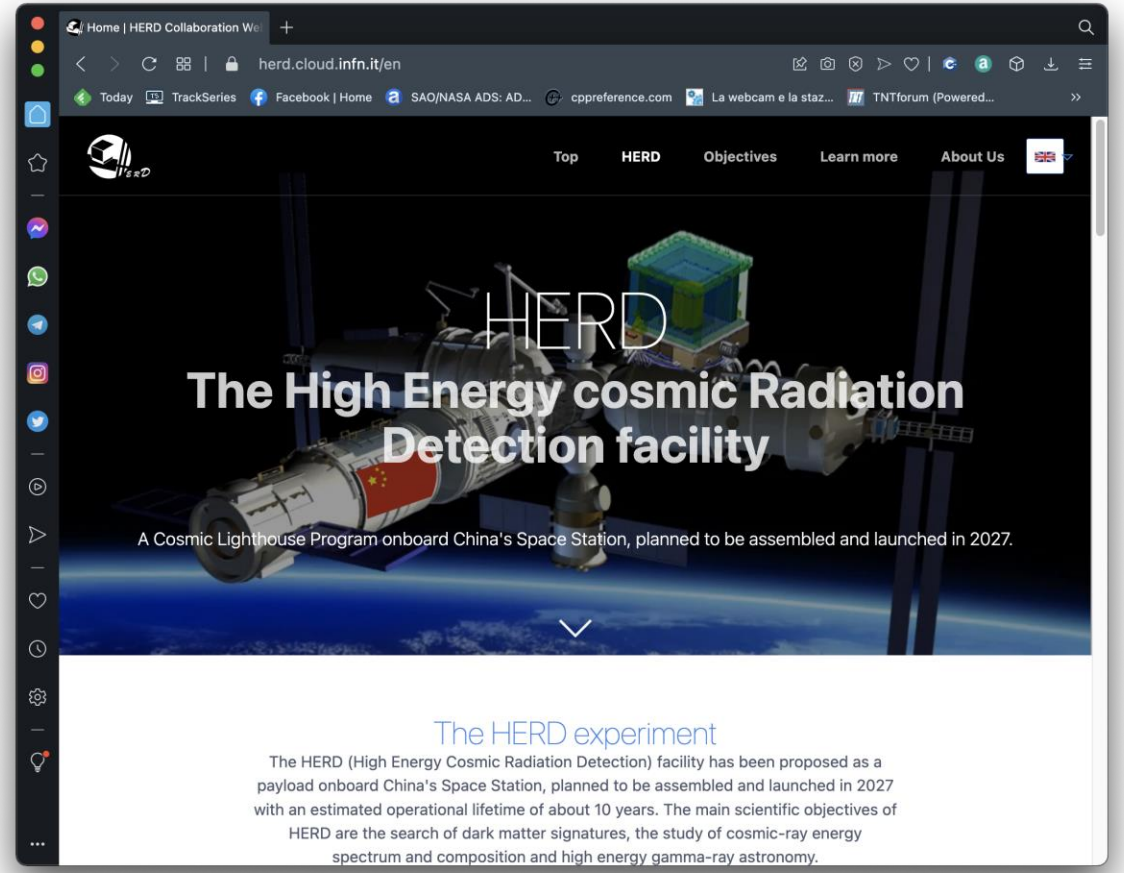
# HERD – services

Requirement: we need several services to ease scheduling of day-to-day operations and/or meetings, activities, as well as document sharing, and more.

Issue: Almost every mainstream tool is unavailable to colleagues in China.

Solution: self-hosting of all services:

- Calendar (based on Nextcloud)
- Document server (based on Nextcloud + Collabora)
- Experiment website (with dedicated restricted area for internal documentation, behind IAM login)



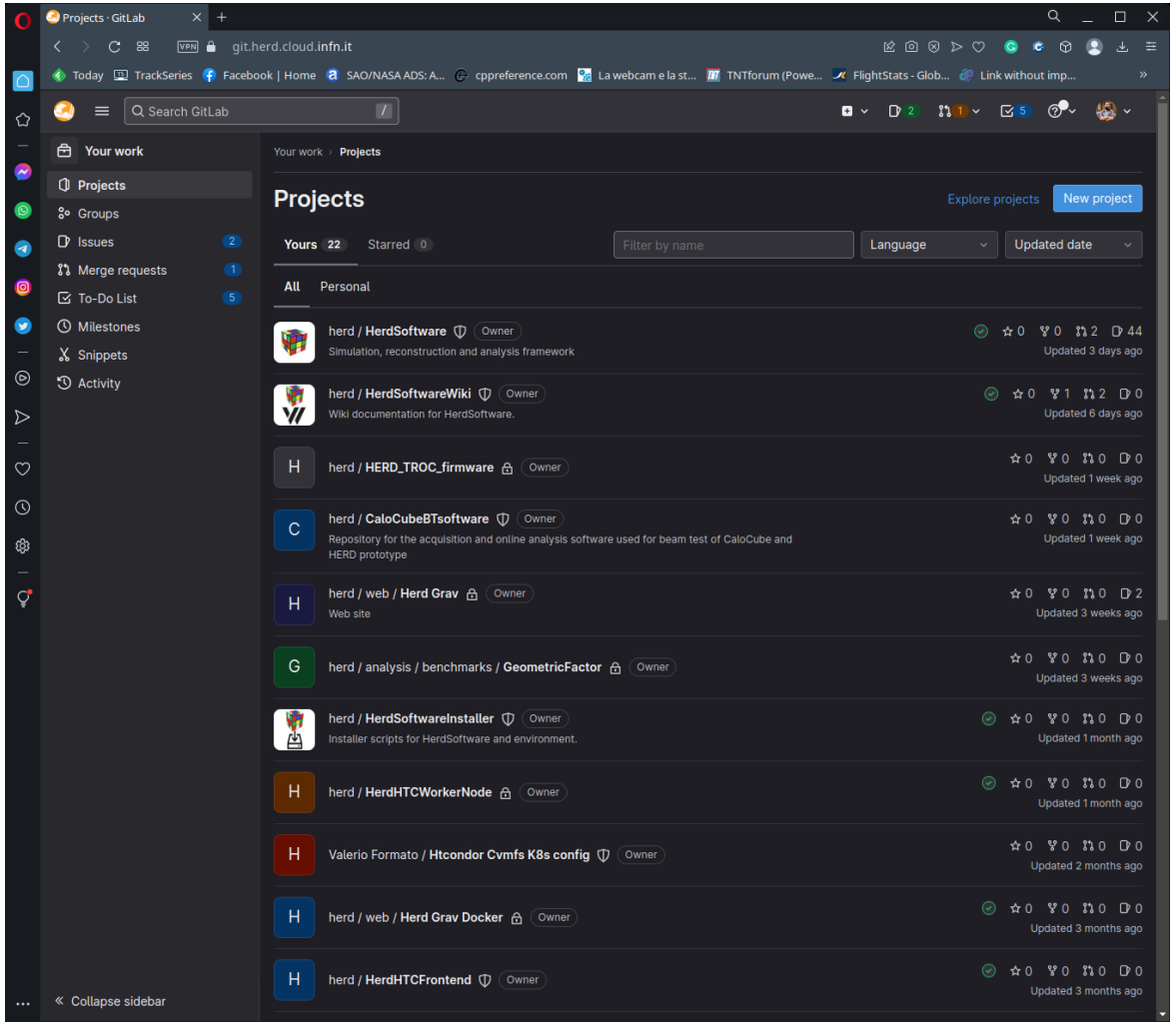
# HERD – services

Requirement: we need several services to ease scheduling of day-to-day operations and/or meetings, activities, as well as document sharing, and more.

Issue: Almost every mainstream tool is unavailable to colleagues in China.

Solution: self-hosting of all services:

- Calendar (based on Nextcloud)
- Document server (based on Nextcloud)
- Experiment website (with dedicated restricted area for internal documentation, behind IAM login)
- Gitlab instance (with runners)



# HERD – services

Requirement: we need several services to ease scheduling of day-to-day operations and/or meetings, activities, as well as document sharing, and more.

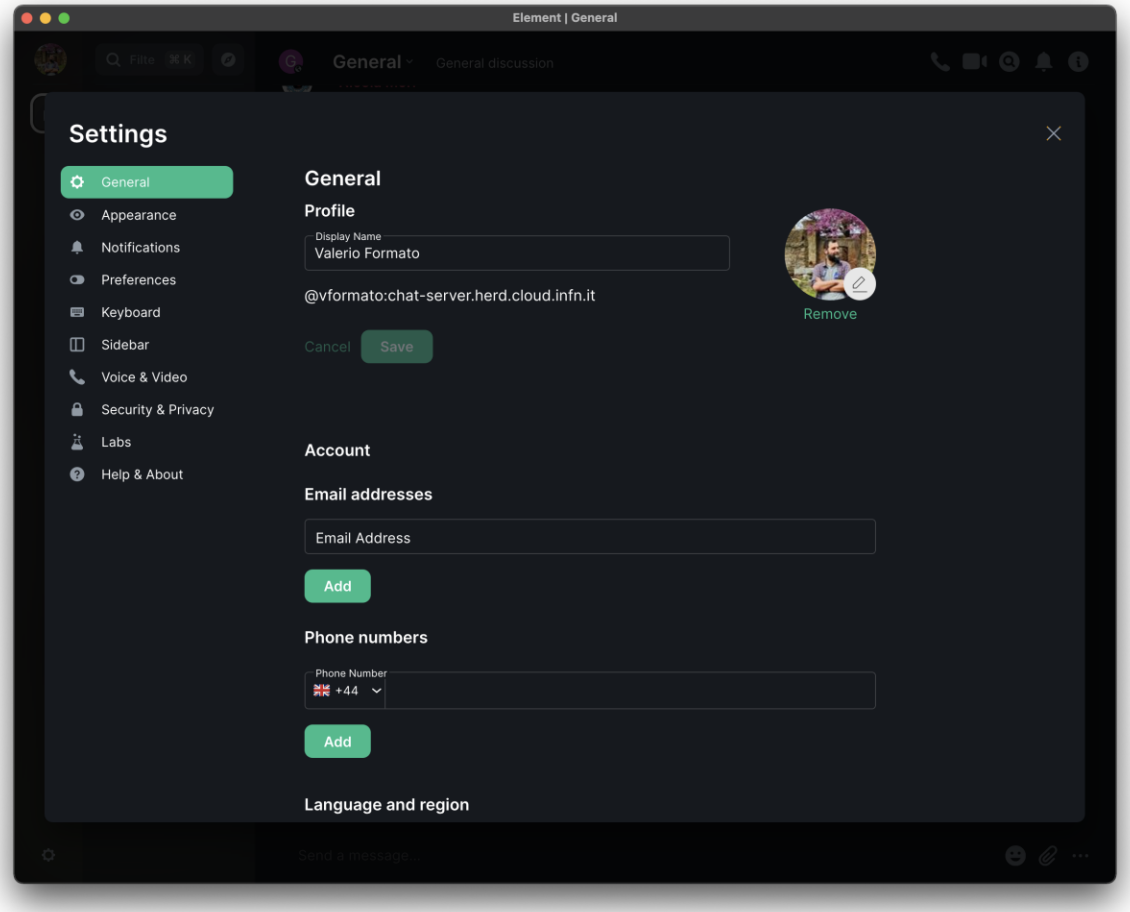
Issue: Almost every mainstream tool is unavailable to colleagues in China.

Solution: self-hosting of all services:

- Calendar (based on Nextcloud)
- Document server (based on Nextcloud)
- Experiment website (with dedicated restricted area for internal documentation, behind IAM login)
- Gitlab instance (with runners)

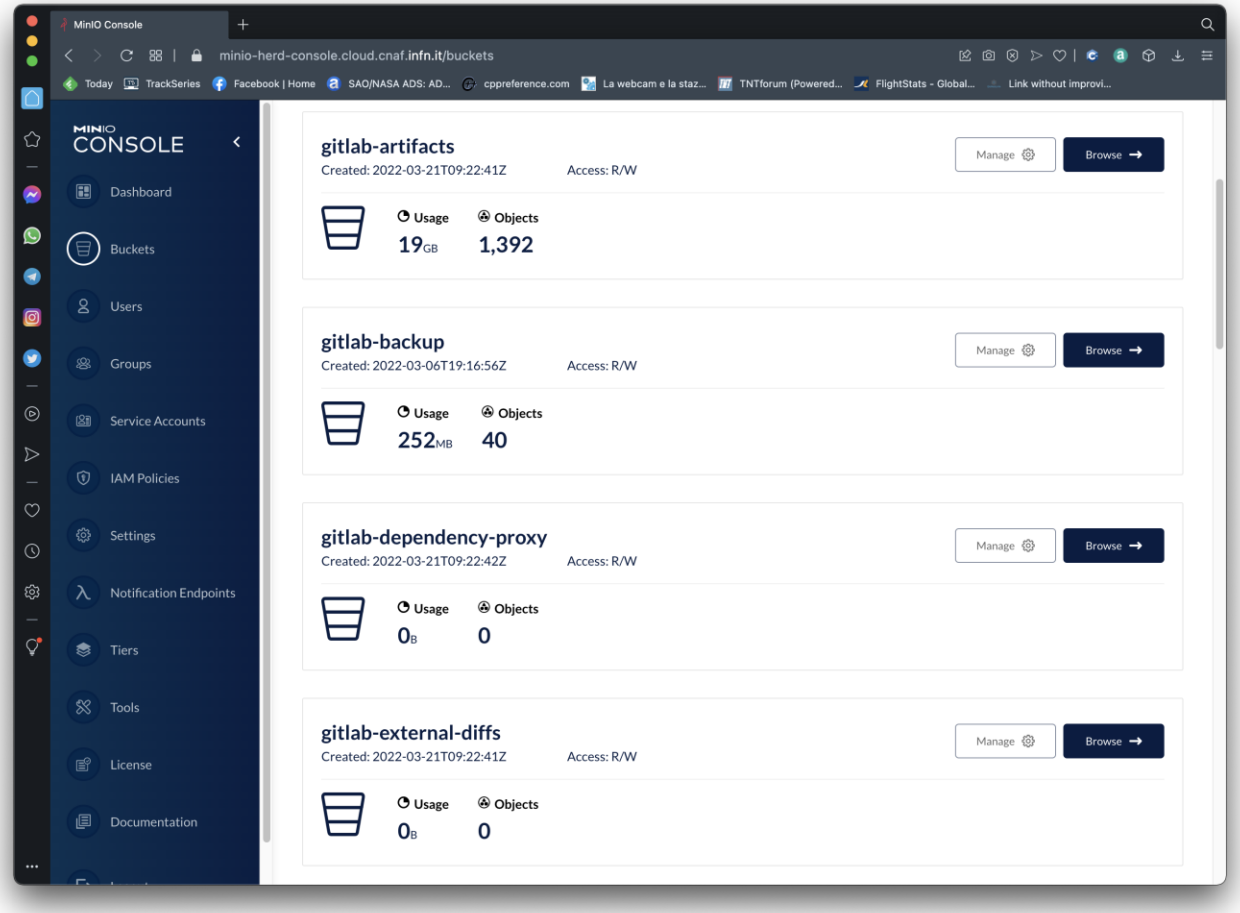
Currently testing also:

- Experiment-wide chat service (based on Matrix)
- INDICO



# HERD – storage

- S3-compatible storage based on MinIO
  - Well-established cloud storage technology
  - Available storage backends for many components (e.g. CVMFS)
- Used for data, software and services
- Testbed based on 4 MinIO instances running on Kubernetes
  - 100 TB raw capacity provided via Ceph (DICE project, grant agreement ID: 101017207)
    - 3+1 erasure coded storage
  - 100 TB on INFN-Cloud resources, to be integrated with existing resources
- Access control:
  - OIDC tokens managed via oidc-agent (users)
  - Access keys (services)



# 2023: Feedback and next steps

- Needed:
  - Cannot reach HTCondor-CE element from VMs hosted by Cloud@CNAF
    Since our resources are primarily hosted on Cloud@CNAF
- Desirable:
  - Self-managed DNS on select subdomains
    To ease the workflow of deploying new services or maintaining existing ones
- Feedback:
  - The whole INFN-Cloud infrastructure feels a bit "user-centric" rather than "team-centric"
    For example: within the HERD tenant each user can see and manage only his own deployments
- Next:
  - Continue testing and improve the HTCondor workflow for analysis
    Finish work on resource merging, user-mapping, …
  - Backport hand-made deployments in INFN-Cloud dashboard (*e.g.* Gitlab as-a-service)
  - Assess the fate of the DICE storage once the project ends

# Conclusions

- The "astro-particle in space" community is eager of resources and poor in terms of man-power for computing: we will always be willing to test any solution to increase our pool of resources and to keep up with the software infrastructure developments, given the limited man-power available
- Given the nature of the partners for the various projects (ASI, Chinese collaborators, …) we can have small and/or temporary resources at our disposal: merging them in a single batch system is a big added value. *We are getting really close now*.
- As a new experiment in a design phase, HERD is proving to be the perfect opportunity to migrate towards a deeper integration with INFN-Cloud provided services and we are continuously improving in both creating new services and maintaining them.