Continuous Integration (CI/CD) at CINECA

Continuous Integration, continuous delivery and deployment also known as CI/CD is the practice of a **constant monitoring of the code development** through the activation of an automatic pipeline. Every time a developer applies a change to the code, the **automatic pipeline validates it** by building the code and running some simple tests (typically unit tests).

CINECA has now activated a **new service** where to run your Continuous Integration (CI/CD) pipelines on CINECA clusters, based on CINECA GitLab service. On GitLab website there is a detailed documentation about CI/CD.
This service is active on both **Galileo100** and **Marconi100**.
This service is at the moment in **experimental phase**.

How to use it

You can **activate our CI/CD service** in projects defined into our GitLab instance. If you are already a CINECA HPC user, you can access the CINECA GitLab using the same credentials. If you are interested and you are still not an HPC User you can find here the instructions how to get access.

Once logged to CINECA GitLab, you can activate the CI/CD service by **enabling shared runners** that pick up and execute your CI/CD pipeline on our cluster.
They can be enabled as in the following:

1. From your project's web page, select "*Settings"* and then on "*CI/CD*", from the menu on the left;
2. Once in the web page, expand the section *"Runners";*
3. Activate the switch under "*Enable shared runners for this project*" on the right, in the *"Shared Runners"* right column. The *shared runners* are listed in that section along with blue labels specifying the *tags* associated to them.

Now shared runners are available to your CI/CD pipeline.

The CI/CD pipeline has to be specified inside the `.gitlab-ci.yml` file through tags (see Gitlab documentation for how to create and manage pipelines).

IMPORTANT: There are **two different kind of runners.** You have to identify **which runner** you would like to run your pipeline by **specifying one or more tags** summarized in the table at the bottom of the page.
IMPORTANT: If you do not select any tag, the pipeline **will never start**.

We set a **time limit** for the execution of each single job of a given pipeline that cannot lasts for more than **10 minutes**.

Runners description

Galileo100

All shared runners are based on docker images, so in your CI/CD pipeline you can optionally choose in which **container image** your pipeline job will run.

You will find **4 distinct shared runners**, consisting of:

- **2 CPU-only** runners, with access up to 24 CPUs each. Jobs are executed in concurrent execution. (specific tags: **x86_64**, **cpu**, **docker**)
- **2 CPU+GPU** runners, limited to run 1 CI job each at the time. Each runner has access to a dedicated GPU. **No concurrent execution** is allowed on these runners. (specific tags: **x86_64**, **docker**, **nvidia-sm70**, **nvidia-volta, nvidia-cuda**)

All shared runners run on a dedicated node of Galileo100 with **Intel x86_64 architectures** (2 x CPU Intel CascadeLake 8260 processors with 24 cores each, 2.4 GHz, 384GB RAM).
GPU runners make use of **Nvidia V100** GPU.

Marconi100 (pre-production)

Since RedHat 8 does not support docker, on M100 you can find Podman to deal with containers. Basically there are no important differences in the usage of Podman with respect to docker from the point of view of the Continuous Integration.

On Marconi100 you will find **6 distinct shared runners**, consisting of:

- **2 CPU-only** runners, with access up to 16 CPUs each. Jobs are executed in concurrent execution. (specific tags: **ppc64le**, **cpu**, **podman**)
- **4 CPU+GPU** runners, limited to run 1 CI job each at the time. Each runner has access to a dedicated GPU. **No concurrent execution** is allowed on these runners. (specific tags: **ppc64le**, **podman**, **nvidia-volta**)

Summary

Below we summarize the runners and the tags needed to select the correct one.

Cluster	Runners	Tags	Notes
Galileo100	2 CPU-only	x86_64 cpu docker	Up to 24 cpus each. Concurrent execution

	2 CPU+GPU	x86_64 docker nvidia-sm70 nvidia-volta nvidia-cuda	Each runner has a dedicated GPU. No concurrent execution
Marconi100	2 CPU-only	ppc64le cpu podman	Up to 16 cpus each. Concurrent execution
	4 CPU+GPU	ppc64le podman nvidia-volta	Each runner has a dedicated GPU. No concurrent execution

IMPORTANT: If you do not select any tag, the pipeline **will never start**.