UG2.1 Getting started

In the following, you can find a simple and quick start guide for users new to HPC systems and expert users who would like to use our systems. It describes all the steps to be followed to access our systems up to the first job submission.

This is a schematic guide with a few examples, and it is not intended to be complete. We always strongly recommend reading the full documentation that can be reached using the links you can find along with the text.

1. Registration

The first step is to get a username on our database and a password to enter our HPC clusters.

   1. Register to our userDB database at userdb.hpc.cineca.it by clicking on the "Create a New User" button and filling in the required fields.
   2. Once you get access, complete the info on your user page by uploading an Identity Card in the Documents for HPC tab, complete the information about your Institution and check your Personal Data.

This step alone does not grant you access to our clusters. You also need to be associated with an account that has budget resources with "cpu-hours" to be used in the clusters.

2. Account association

There are multiple ways to get an account budget (see also UG2.2 Become a User):

- A Principal Investigator (PI) of an already existing account can add you to it in the UserDB portal;
- You can apply for your own project by submitting your proposal for an ISCRA, PRACE projects or HPC Europe Transnational Access Programme;
- If you are a member of an Italian research Institution with already existing agreements with CINECA (in this case, send an email to superc@cineca.it);
- General users and Industrial Applications (send a request to superc@cineca.it)

Once your username has been associated with an active account, you can request access to our HPC clusters by clicking on the button "Submit" on your HPC Access page. (The button appears only after you have been associated with an account).

![HPC Access](image)

After we have granted you the access, you will receive two emails with the username and the password to be used to login. Once logged, you will be asked to change your password (you can always change your password later on with "passwd" command). Our policy about password can be found on the "UG2.3 Access to the Systems" page.

**Remember:** Login credentials are to be considered strictly personal, meaning that NO SHARING between members of the same working group is expected to happen. Every single user entitled with login credentials is to be considered personally responsible for any misuse that should take place.

3. Connecting to the cluster

Once you have received your credentials, you can login to the cluster in which the budget account you are associated with is active.

The simplest way is to open a terminal and type

...
> ssh <username>@login.<cluster>.cineca.it

(Windows users can find here the instructions)

Alternative ways to connect to CINECA HPC clusters are described on the "UG2.3 Access to the Systems" page.

Available clusters are Galileo100, Marconi and Marconi100 (m100) (UG3.0 System specific guides).

4. The "saldo" command

On the cluster, you can keep an eye on your budget of hours using the command:

> saldo -b

This lists all the accounts associated with your username on the current cluster, together with the "budget" and the consumed resources. Additional info on this very useful command and our billing policy can be found on the "UG2.4 Accounting" page.

One username can use multiple accounts, and one single account can be used by multiple usernames (and even on multiple platforms), all competing for the same budget.

5. Modules

Some software, environments and compilers are already installed on the clusters and are available as modules (UG2.6 Production Environment). They are divided into different profiles depending on the category of research. The command

> modmap

shows all the profiles, categories and modules that you can load.

To load a module type:

> module load <module_name>

If the module is inside a specific profile, you have to load the profile before the module:

> module load profile/<profile_name>

There are many useful options for the functions "modmap" and "module" that are described here.

You can also install your libraries and programs on your local folder by yourself or using python environment or spack manager. Please write to superc@cineca.it for any questions or requests about modules and software installations.

6. Storage

Our HPC systems offer several options for data storage:

- $HOME: to store programs and small light results
- $CINECA_SCRATCH: where you can execute your programs
- $WORK: An area visible to all the users associated with the same budget account
- $DRES: An additional area to store your results if they are heavy. This space is not automatic. You need to request for it writing to superc@cineca.it

Important details and suggestions on how to use each space can be found on the "UG2.5 Data Storage and Filesystem" page.

To monitor the occupancy of your space, you can use the "cindata" command

7. How to submit a job

Once you have your compiled program and prepared its input data, the last step is to run it on the cluster’s compute nodes.

Important: the node you are logged in is a login node and cannot be used to execute parallel programs. Login nodes are not used for production therefore the execution of any command in the login nodes is limited up to 10 minutes.

For longer runs, you need to use the "batch" mode. On CINECA machines we use the SLURM scheduler.

Batch job

A simple batch script to submit a job is the following
```bash
#!/bin/bash

#SBATCH --nodes=<nodes_no>   # number of nodes
#SBATCH --ntasks-per-node=<tasks_no>   # number of tasks per node
#SBATCH --time=01:00:00   # time limits: here 1 hour
#SBATCH --mem=<memory>GB   # total memory per node requested in GB (optional)
#SBATCH --error=myJob.err   # standard error file
#SBATCH --output=myJob.out   # standard output file
#SBATCH --account=<account_no>   # account name
#SBATCH --partition=<partition_name>   # partition name
#SBATCH --qos=<qos_name>   # quality of service (optional)

srun ./my_application
```

In the script we tell the scheduler the amount of resources needed (--nodes, --ntasks-per-node and --mem) on which partition (--partition and --qos) and which budget of hours to be used (--account). The session has a walltime (--time) and the outputs of the code are collected in myJob.out and myJob.err (--output and --error respectively).

The partition and the resources depend on the machine you are considering. All you need to know to properly fill your batch script can be found in the "U3.0 System specific guide" page.

To submit the job to the scheduler type

```
> sbatch [opts] job_script
```

You can find a complete list of examples of job scripts here.

---

**Interactive job**

As an alternative you can submit an interactive job opening a terminal session directly in a compute node:

```
> salloc --nodes=<nodes_no> --ntasks-per-node=<tasks_no> --account=<account_no> --partition=<partition_name>
```

in which you can execute your program with the same inputs as the batch job.

Remember to close the interactive job with the command "exit" when you have finished, in order not to waste your account budget.

Congratulations! You have successfully executed your first job on our clusters.

Remind: Going through our detailed documentation may help you optimize your programs on our clusters and save hours of your budget.

For any problem or question, please refer to our Help Desk writing to superc@cineca.it