EUROfusion users: Marconi and Leonardo environments

EUROfusion community has dedicated partitions on CINECA clusters:

- on Marconi-A3 (SkyLake) SKL, consisting in 2912 dedicated nodes;
- on Leonardo (currently) with 72 nodes starting from August 2023.

Previous dedicated partitions:

- From Jul 2016 up to Aug 2017 EUROfusion community had 806 dedicated nodes of the Marconi-A1 partition (from Jul 2016 up to Aug 2017). Starting from October 15th, 2017 the EUROfusion activity on Marconi-A1 has been transferred to Marconi-A3.
- From beginning of 2017 up to the end of 2019 the community could use 449 dedicated nodes of Marconi-A2 (Knights landing) KNL partition. From January 2020 up to May 2020 the number of dedicated nodes has been 288. Starting from May 2020 the activity on Marconi-A2 has been stopped.
- From January 2019 up to February 2020 the community had a dedicated partition on D.A.V.I.D.E. consisting in 40 nodes. On February 2020 the activity on D.A.V.I.D.E. has been stopped.
- From April 2020 the community had 80 dedicated nodes on Marconi100, increased to 99 from April 2021. The Marconi100 production has been stopped on 24 of July of 2023.

The general environment defined on our clusters for the EUROfusion cmmunity is the same as the one defined for all the users of the cluster. The general environment refers to:

- Access
- Accounting
- Disks and Filesystems
- Module environment
- Programming environment depends on the peculiarities of Marconi-A3, Marconi100 and Leonardo.

In the following you find an extraction of the basic knowledge needed to properly take advantage of our clusters

Data storage and Filesystems

The storage organization conforms to the CINECA infrastructure (see Section Data Storage and Filesystems). In addition to the home directory $HOME, for each user is defined a scratch area $CINECA_SCRATCH, a large disk for the storage of run time data and files. A $WORK area is defined for each active project on the system, reserved for all the collaborators of the project. This is a safe storage area to keep run time data for the whole life of the project.

Storage Production Areas: $WORK and $CINECA_SCRATCH

These two areas share the same physical device, have the same block size and they also have the same performance in terms of data throughput. $WORK and $CINECA_SCRATCH are conceived as working directories for large files used and produced by batch jobs. Also, the blocking features make these areas more suitable for large binary files.

* $WORK: permanent, project specific, local *

There is one $WORK area for each active project on the machine that all users belonging to can use for production runs and storage of their output data. The owner of the main directory is the PI (Principal Investigator), but all collaborators are allowed to read/write in there. Collaborators are advised to create a personal directory in $WORK for storing their personal files. By default the personal directory will be protected (only the owner can read/write), but protection can be easily modified, for example by allowing write permission to project collaborators through "chmod" command. *The default quota for a project $WORK area is 1TB, but it is possible to consider a quota extension if needed* (please mailto: superc@cineca.it). File retention in the $WORK area is related to the life of the project. Files in this area will be conserved up to 6 months after the project expiring date, and then they will be cancelled. Please note that **there is no back-up on this area**.

To manage different WORK areas for different projects please use the "chprj" command. For a brief description of the command, just type "chprj --help" to print the help page.

To check for the occupancy of this area please use the "cindata" command, that will list all filesets containing any file owned by your username.

* $CINECA_SCRATCH: temporary, user specific, local *

The main difference of this area with respect to $WORK is that it is user specific (not project specific) and that it can be used for sharing data with people outside your project. There is one $CINECA_SCRATCH area for each username on the machine. By default, file access is open to everyone, in case you need more restrictive protections, you can set them with "chmod" command. **On this area a periodic cleaning procedure is applied, with a normal retention time of 40 days:** files are cancelled on a daily basis by an automatic procedure if not accessed for more than 40 days. Please take in mind that this time interval of 40 days may be reduced in case of critical usage ratio of the area. In this case, users will be notified via HPC-News. When files are deleted, a file listing al deleted files for a given day will be created: CLEAN_<yyyymmdd>.log, where <yyyymmdd> = date when files were cancelled.

$CINECA_SCRATCH **does not have any disk quota.** However, it is strongly recommended to maintain a low occupancy of this area in order to prevent very dangerous filling condition. Please, be aware that on Galileo and Marconi clusters, in order to prevent a very dangerous filling condition, a *20TB disk quota will be temporarily imposed* to all users when the global quota area will reach the *88% of occupancy*; this disk quota will be removed when the global occupancy lowers back to normal. To check for the occupancy of this area please use the "cindata" command, that will list all filesets containing any file owned by your username.

* $TMPDIR: temporary, user specific, local *

Each compute node is equipped with a **local storage** which dimension differs depending on the cluster (please look at the specific page of the cluster for more details).
When a job starts, a **temporary area** is defined on the storage **local to each compute node**:

TMPDIR=/scratch_local/slurm_job.$SLURM_JOB_ID

which can be used **exclusively** by the job's owner. During your jobs, you can access the area with the (local) variable $TMPDIR. In your sbatch script, for example, you can move the input data of your simulations to the $TMPDIR before the beginning of your run and also write on $TMPDIR your results. This would further improve the I/O speed of your code.

However, the directory is **removed at the end of the job**, hence always remember to save the data stored in such area to a permanent directory in your sbatch script at the end of the run. Please note that the area is located on local disks, so it can be accessed only by the processes running on the specific node. For multinode jobs, if you need all the processes to access some data, please use the shared filesystems $HOME, $WORK, $CINECA_SCRATCH.

On Marconi100 the $TMPDIR area has 1 TB of available space, while on Marconi the available space is about 49 GB.

Repository area for collaborative work among different projects and across platforms: $DRES

This is a data archive area available only on-request (please mailto: superc@cineca.it), shared with all CINECA HPC systems and among different projects. $DRES is not mounted on the compute nodes. This means that you cannot access it within a batch job: all data needed during the batch execution has to be moved to $WORK or $CINECA_SCRATCH before the run starts.

MARCONI-A3

This partition, made of 2410 nodes (SkyLake, 48 cores, 192000MB) is in production since August 2017 (initially with 1512 nodes, enlarged to 2410 nodes since November 2018) and is reserved to the EUROfusion community.

As on the rest of the cluster, all production jobs must be submitted using a queuing system. Batch jobs are managed by the SLURM batch scheduler, that is described in section Batch Scheduler SLURM.

The **maximum number of cores per job** and the **maximum walltime** depend on the chosen partition.

In the following table you can find all the main features and limits imposed on the SLURM partitions. For up-to-date information, use the "sinfo" and "scontrol show partition <partition_name>" commands on the system itself.

partition	QOS (quality of service)	min/max # nodes per job min/max # nodes per user	max walltime	max memory (MB)	priority	notes
skl_fua_dbg	*no QOS*	4 max per job 4 max per user	1:00	182000	40	16 dedicated nodes Job submission to the partition is limited to max 10 jobs. Only 1 job running per user.
skl_fua_prod	*no QOS*	65 max per job	24:00	182000	40	
skl_fua_prod	skl_qos_fuabprod	66 min per job 512 max per job	24:00	182000	85	Max 1024 nodes total
skl_fua_prod	skl_qos_fualprod	11 max per job	72:00	182000	85	Max 22 nodes/user Max 66 nodes total
skl_fua_prod	qos_special	> 512 nodes	> 24:00:00	182000	40	**on request** *write to:* superc@cineca.it

skl_fua_prod	skl_qos_fualowprio	65 max per job	24:00	182000	0	Max 2 jobs/user Max 950 nodes total **for exhausted active projects**

For information on how to submit and manage jobs for SKL partition, please refer to the Marconi UserGuide.

As usual on systems with the SLURM scheduler, you submit a batch job script with the command:

```
> sbatch <options> script
```

You can get a list of defined partitions (on the entire cluster - you will be allowed to submit jobs only to the "***_fua_***" partitions) with the command

```
> sinfo
```

For more information and **examples of job scripts**, see section Batch Scheduler SLURM.

Euterpe

It has been noticed a problem with EUTERPE. The job may go in hang using the command *mpiexec*. The *mpiexec* command launches only the execution of the application on the selected cores without initialising them, therefore in some cases, the run could crash without exiting.

In order to avoid this issue, we strongly suggest users to use *SRUN* when launching jobs of EUTERPE.

LEONARDO

A **presentation** of Leonardo dedicated to the EUROfusion community was held on June 6th, 2023, and the slides and recording are available here (you should log in through the button Log in as a guest).

The mandatory access to Leonardo is the two-factor authentication (**2FA**). Please refer to this link of the User Guide to activate and connect via 2FA and to the slides of the 2FA presentation (07/06/2023) dedicated to the EUROfusion community.

All the login nodes have an identical environment and can be reached with **SSH (Secure Shell)** protocol using the "collective" hostname:

```
> login.leonardo.cineca.it
```

On Leonardo, EUROfusion community has (currently) access to 72 nodes, each one containing 4 NVIDIA A100 GPUs.

Similarly to other systems jobs must be submitted using a queuing system. Batch jobs are managed by the SLURM batch scheduler, described in section Batch Scheduler SLURM.

In the following table you can find all the main features and limits (e.g. **maximum number of cores per job** or **maximum walltime**) imposed on the SLURM partitions available to EUROfusion community.
For up-to-date information, use the "sinfo" and "scontrol show partition <partition_name>" commands on the system itself.

It is **not possible to occupy more than 32 nodes for a single user** on the partition boost_fua_prod.

SLURM partition	**Job QOS**	**# cores/# GPU per job**	**max walltime**	**max running jobs per user/ max n. of cores/nodes/GPUs per Grp**	**priority**	**notes**
boost_fua_prod	*normal*	max = 16 nodes	24:00:00		40	
	boost_qos_fuabprod	min = 17 nodes max =32 nodes	24:00:00	49 nodes / 1568 cores / 196 GPUs	60	runs on 49 nodes, min is 17 FULL nodes
boost_fua_dbg	*normal*	max = 2 nodes	00:10:00	2 nodes / 64 cores / 8 GPUs	40	runs on 2 nodes
	qos_fualowprio	max = 16 nodes	08:00:00		0	▪ automatically added to the active accounts with exhausted budget ▪ to be used with the FUAL7_LOWPRIO account

Low-priority jobs

1) If you consume all the budget assigned to your projects, you can keep running on Leonardo boost_fua_prod partition at low priority by requesting in your submission script the qos_fualowprio QOS:

#SBATCH --qos=qos_fualowprio

The QOS is automatically added to your account upon budget exhaustion.

2) You can also request to run low priority jobs, without consuming your active budget, by association to the FUAL7_LOWPRIO account (write a mail to superc@cineca.it). You always need to specify the qos_fualowprio QOS in your submission script as above together with this LOWPRIO account.

2FA Access

The mandatory access to Leonardo and Marconi (starting from June 13 2023) is the two-factor authentication (**2FA**). Please refer to this link of the User Guide to activate and connect via 2FA and to the slides of the 2FA presentation (07/06/2023) dedicated to the EUROfusion community.

Other remarks

PLEASE NOTE THE FOLLOWING IMPORTANT REMARKS:

1) on Marconi-A3, in order to have a one-node granularity on your jobs we imposed **a scattered arrangement and exclusive placement of resources** for your jobs. This is not valid for Marconi100.

Hence, each resource will be placed on a different node and each node will be job-exclusive. We also defined a default value of cpus per chunk equal to the full number of cores/node (48 on Marconi-A3). Hence, if you need for example 3 nodes you can simply write:

```
#SBATCH -N3
```

You can always specify the other resource, e.g. mpiprocs, for example for a hybrid job using 24 OMP threads and 2 MPI processes per node (on A3):

```
#SBATCH -N3 --ntasks-per-node=2 --cpus-per-task=24
```

Please note that you can overwrite the default value if you ask explicitly for a specific number of ncpus, nevertheless you will pay always for the full node.

2) following the granularity request on the resources, on Marconi-A3 we also set the **default memory** for each chunk to be equal to the total memory available on a compute node . Again, the parameter only applies if the user do not ask explicitly for a specific amount of memory.

3) You always have to specify the partition (and when needed the qos) accordingly to the needed resources. For example, if you need to run on 64 nodes the "skl_fua_prod" queue is what you need. As an example consider a pure-MPI job requiring 64 nodes and 1 hour of walltime on **Marconi-A3**:

```
#!/bin/bash

#SBATCH -N64 --ntasks-per-node=36
#SBATCH --mem=182000
#SBATCH --time=01:00:00
#SBATCH --account=FUA32_XXXXX
#SBATCH --partition=skl_fua_prod

<load some modules or set some env variables>
<execute your code>
```

on **Marconi100**:

```
#!/bin/bash
#SBATCH -N 64                # 1 node
#SBATCH --ntasks-per-node=8 # 8 tasks out of 128
#SBATCH --time=01:00:00
#SBATCH --gres=gpu:1        # 1 gpus per node out of 4
#SBATCH --mem=7100          # memory per node out of 246000MB
#SBATCH -A FUAC4_XXX
#SBATCH -p m100_usr_prod

<load some modules or set some env variables>
<execute your code>
```

4) Please note that **on Marconi-A3 the recommended way to launch parallel tasks in slurm jobs is with srun. On Marconi100 it is mpirun.** By using the correct command you will get full support for process tracking, accounting, task affinity, suspend/resume and other features.

5) Controlling the processes and threads affinity is crucial to ensure the optimal performances on Marconi-A3 and Marconi100. Do not rely on slurm autoaffinity and use the proper SLURM --cpu-bind option.

6) All users with active (i.e., not expired) projects but exhausted budget are automatically enabled to use the QOS **skl_qos_fualowprio** (on Marconi-A3) or **qos_fualowprio** (on Marconi100), which allow to keep submitting jobs with no priority.

Marconi-A3:

```
#SBATCH --account=FUA35_XXXXX
#SBATCH --partition=skl_fua_prod
#SBATCH --qos=skl_qos_fualowprio
```

Marconi100:

```
#SBATCH --account=FUAC5_XXXXX
#SBATCH --partition=m100_fua_prod
#SBATCH --qos=qos_fualowprio
```

For more information and **examples of job scripts**, see section **Batch Scheduler SLURM**.

7) All users with active and non exhausted budgets can request to be enabled to use the lowprio QOS by association to FUA36_LOWPRIO (on Marconi-A3) or FUAC6_LOWPRIO (on Marconi100) projects by writing to superc@cineca.it. You can then submit job (at low priority) using these accounts as above:

Marconi-A3:

```
#SBATCH --account=FUA36_LOWPRIO
#SBATCH --partition=skl_fua_prod
#SBATCH --qos=skl_qos_fualowprio
```

Marconi100:

```
#SBATCH --account=FUAC6_LOWPRIO
#SBATCH --partition=m100_fua_prod
#SBATCH --qos=qos_fualowprio
```

8) On Marconi-A3, EUROfusion community users are suggested to use some particular settings for the Intel I_MPI_ADJUST family environment variables. This choice should guarantee optimal performances controlling the corresponding collective algorithm selection. In particular:

I_MPI_ADJUST_GATHERV=3

has been found optimal for a given workload containing a huge number of collective calls and for a large number of nodes.

9) INTEL compiler options: processors specific optimization flags

To optimize the performance of your code on SKL architectures follow the guidelines reported here.

10) (OLD) CPU binding on Marconi.

Please refer to this document *(Courtesy of T. Ribeiro)*, that provides practical guidelines on how to use the process pinning options within the IntelMPI library together. The document refers to the PBS scheduler system, that was available on Marconi before the transition to SLURM.