UG3.2: LEONARDO UserGuide

Sections Production environment, Programming environment are specific for the two partitions, Booster and Data Centric General Purpose (DCGP):

- LEONARDO Booster UserGuide
- LEONARDO DCGP UserGuide

---

hostname: login.leonardo.cineca.it

login01-ext.leonardo.cineca.it

login02-ext.leonardo.cineca.it

login05-ext.leonardo.cineca.it

login07-ext.leonardo.cineca.it

early availability: March, 2023 (Booster)

start of pre-production: June, 2023 (Booster)

January 2024 (DCGP)

start of production: August 2023 (Booster)

February 2024 (DCGP)

This HPC system is the new pre-exascale Tier-0 EuroHPC Joint Undertaking supercomputer hosted by CINECA and currently built in the Bologna Technopole, Italy. It is supplied by EVIDEN ATOS, and it is based on two new specifically-designed compute blades, which are available through two distinct SLURM partitions on the cluster:

- X2135 **GPU** blade based on NVIDIA Ampere A100-64 accelerators - **LEONARDO Booster partition**
- X2140 **CPU-only** blade based on Intel Sapphire Rapids processors - **LEONARDO Data Centric General Purpose (DCGP) partition**

The overall system architecture also uses NVIDIA Mellanox InfiniBand High Data Rate (HDR) connectivity, with smart in-network computing acceleration engines that enable extremely low latency and high data throughput to provide the highest AI and HPC application performance and scalability.

The system also includes a Capacity Tier and a Fast Tier storage, based on DDN Exascaler.

The Operating System is **RedHat Enterprise Linux 8.6**.

System Architecture

Login nodes: 4 nodes, icelake no-gpu

	Booster	DCGP
Model	Atos BullSequana X2135 "Da Vinci" single-node GPU blade	Atos BullSequana X2140 three-node CPU blade
Racks	116	22
Nodes	3456	1536
Processors	single socket 32 cores Intel Ice Lake CPU 1 x Intel Xeon Platinum 8358, 2.60 GHz TDP 250W	dual socket 56 cores Intel Sapphire Rapids CPU 2 x Intel Xeon Platinum 8480p, 2.00 GHz TDP 350W
Accelerators	4 x NVIDIA Ampere GPUs /node, 64GB HBM2e NVLink 3.0 (200GB/s)	-
Cores	32 cores /node	112 cores /node
RAM	512 (8x64) GB DDR4 3200 MHz	512 (16 x 32) GB DDR5 4800 MHz
Peak Performance	about 309 Pflop/s	9 Pflops /s
Internal Network	DragonFly+ 200 Gbps (NVIDIA Mellanox Infiniband HDR)	
	2 x dual port HDR100 per node	single port HDR100 per node
Storage (raw capacity)	137.6 PB based on DDN ES7990X and Hard Drive Disks (Capacity Tier) 5.7 PB based on DDN ES400NVX2 and Solid State Drives (Fast Tier)	

Peak performance details

Node Performance		
Theoretical Peak Performance	CPU (nominal/peak freq.)	1680 Gflops
	GPU	75000 Gflops
	Total	76680 GFlops
Memory Bandwidth (nominal/peak freq.)		24.4 GB/s

Access

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
```

which establishes a connection to one of the available login nodes. To connect to LEONARDO you can also indicate explicitly the login nodes:

```
$ login01-ext.leonardo.cineca.it
$ login02-ext.leonardo.cineca.it
$ login05-ext.leonardo.cineca.it
$ login07-ext.leonardo.cineca.it
```

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. For information about data transfer from other computers please follow the instructions and caveats on the dedicated section Data storage or the document Data Management.

Accounting

The accounting (consumed budget) is active from the start of the production phase. For accounting information please consult our dedicated section.

The account_name (or project) is important for batch executions. You need to indicate an account_name to be accounted for in the scheduler, using the flag "-A"

```
#SBATCH -A <account_name>
```

With the "saldo -b" command you can list all the account_name associated with your username.

```
$ saldo -b          (reports projects defined on LEONARDO Booster)
$ saldo --dcgp -b   (reports projects defined on LEONARDO DCGP)
```

Please note that **the accounting is in terms of consumed core hours, but it strongly depends also on the requested memory and local storage, and number of GPUs**, please refer to the dedicated section.

Budget Linearization policy

On LEONARDO, as on the other HPC clusters in CINECA, a linearization policy for the usage of project budgets has been defined and implemented. The goal is to improve the response time, giving users the opportunity of using the cpu hours assigned to their project in relation to their actual size (total amount of core-hours).

Disks and Filesystems

The storage organization conforms to the CINECA infrastructure (see Section Data Storage and Filesystems).

In addition to the home directory **$HOME**, for each user is defined a scratch area **$SCRATCH** (or **$CINECA_SCRATCH**), a large disk for the storage of run time data and files.
An new user specific area **$PUBLIC** is defined on LEONARDO, useful for example to share installations with other users (it is indeed the default directory for SPACK sub-directories, see more details in the dedicated page).
A **$WORK** area is defined for each active project on the system, reserved to all the collaborators of the project. A corresponding **$FAST** area is defined for each active project on the scratch filesystem, on its subset of "fast" NVMe SSD flash drives. As for $WORK, the $FAST area is reserved to all the collaborators of the project. An extension of the default $WORK quota (1 TB) can be granted if justified and essential for the course of the project's activity, while the use of the $FAST is limited to 1 TB of space per project.

	Total Dimension (TB)	Quota (GB)	Notes
$HOME	0.46 PiB	50GB per user	• permanent • backed up • user specific
$CINECA_SCRATCH	40 PiB	no quota	• HDD storage • temporary • user specific • no backup • automatic cleaning procedure of data older than 40 days (time interval can be reduced in case of critical usage ratio of the area. In this case, users will be notified via HPC-News).
$PUBLIC	0.46 PiB	50GB per user	• permanent • user specific • no backup
$WORK	30 PB	1TB per project	• permanent • project specific • no backup • extensions can be considered if needed (mailto: superc@cineca.it)
$FAST	3.5PB	1TB per project	• permanent • project specific • no backup

- The automatic cleaning of the scratch area is NOT active yet, but it will soon be enforced.

It is also available a temporary area **local to nodes** on login and compute nodes (on the latter it is generated when the job starts and removed when it ends) and accessible via environment variable **$TMPDIR**. This area is:

- on the local SSD disks on login nodes (14 TB of capacity), mounted as /scratch_local (TMPDIR=/scratch_local). This is a shared area with no quota, remove all the files once they are not requested anymore. A cleaning procedure will be enforced in case of improper use of the area.
- on the local SSD disks on the *serial* node (lrd_all_serial, 14TB of capacity), managed via the slurm job_container/tmpfs plugin. This plugin provides a job-specific, private temporary file system space, with private instances of /tmp and /dev/shm in the job's user space (TMPDIR=/tmp, visible via the command "df -h"), removed at the end of the serial job. You can request the resource via sbatch directive or srun option "--gres=tmpfs:XX" (for instance: --gres=tmpfs:200GB), with a maximum of 1 TB for the serial jobs. If not explicitly requested, the /tmp has the default dimension of 10 GB.
- on the local SSD disks on DCGP nodes (3 TB of capacity). As for the serial node, the local /tmp and /dev/shm areas are managed via plugin, which at the start of the jobs mounts private instances of /tmp and /dev/shm in the job's user space (TMPDIR=/tmp, visible via the command "df -h /tmp"), and unmounts them at the end of the job (all data will be lost). You can request the resource via sbatch directive or srun option "--gres=tmpfs:XX", with a maximum of all the available 3 TB for DCGP nodes. As for the serial node, if not explicitly requested, the /tmp has the default dimension of 10 GB. Please note: for the DCGP jobs the requested amount of gres/tmpfs resource **contributes to the consumed budget**, changing the number of accounted equivalent core hours, see the dedicated section on the Accounting
- on RAM on the diskless booster nodes (with a fixed size of 10 GB, no increase is allowed, and the gres/tmpfs resource is disabled).

For a general discussion on the TMPDIR area, please see the dedicated section of Data storage and FileSystems.

Since all the filesystems are based on Lustre, the usual unix command "quota" is not working. Use the local command **cindata** to query for disk usage and quota ("cindata -h" for help):

```
$ cindata
```

or the tool "cinQuota" available in the module cintools

```
$ cinQuota
```

For more details about both these commands, please consult the section dedicated to how to monitor the occupancy.

Software environment

Module environment

The software modules are collected in different profiles and organized by functional categories (compilers, libraries, tools, applications, ...). The profiles are of two types: “programming” type (base and advanced) for compilation, debugging and profiling activities, and “domain” type (chem-phys, lifesc, ...) for the production activity. They can be loaded together.

"Base" profile is the default. It is automatically loaded after login and it contains basic modules for the programming activities (ibm, gnu, pgi, cuda compilers, math libraries, profiling and debugging tools, ...).

If you want to use a module placed under other profiles, for example an application module, you will have to previously load the corresponding profile:

```
$ module load profile/<profile name>
$ module load <module name>
```

Almost all the softwares on LEONARDO were installed with Spack manager, which loads automatically the possible dependencies, so "autoload" command is unnecessary**.**

For listing all profiles you have loaded you can use the following command:

```
$ module list
```

In order to detect all profiles, categories and modules available on LEONARDO, the command "**modmap" is available** as for the other clusters. With modmap you can see if the desired module is available and which profile you have to load to use it.

```
$ modmap -m <module_name>
```

Note: on LEONARDO you can find **modules compiled to support GPUs and modules suitable only for CPUs**. You can check the compiler in the full name of the module, where the version is specified (e.g. gromacs/2022.3--intel-oneapi-mpi--2021.10.0--oneapi–2023.2.0). Remind that modules compiled with gcc, nvhpc, cuda should be used only on the Booster partition, while modules compiled with intel oneapi are suitable for running on the DGCP partition. Please refer to the specific sections of the two partitons for more details on the available compilers: Booster Programming environment and DCGP Programming environment.

Spack environment

In case you don't find a software you are interested in, you can install it by yourself.
In this case, on LEONARDO we offer the possibility to use the "spack" environment by loading the corresponding module. Please refer to the dedicated section in UG2.6: Production Environment

Please note that we are still optimizing LEONARDO software stack, and more installations may be added/replaced. Always check with "module av" (the hash in the module name can change).

Remind that, on LEONARDO (at variance with other CINECA clusters), the default area where Spack directories are created (/cache, /install, /modules, /user_cache) is the $PUBLIC one (described in section Disks and Filesystems).

Graphic session

It will be available soon.

You can proceed with the sections related to Production environment and Programming environment in the specific pages for the two partitions:

- LEONARDO Booster UserGuide
- LEONARDO DCGP UserGuide