

Rsync

- [Rsync via batch job or interactive session](#)
 - [Interactive session](#)
 - [Single job](#)
 - [Chaining multiple jobs](#)
- [Rsync via command line](#)
- [Dedicated node for Marconi100 and Galileo100](#)

This page describes how to use the rsync function to **transfer files** from, to and between CINECA HPC machines. Rsync parameters shown here are **tuned for CINECA** internal network. Please refer to the rsync manual (man rsync) if you look for different optimizations.

The most important advantage of choosing rsync instead of scp is the possibility to **restart the file transfer** from the point it was interrupted (in case of connection problems or time limits of the jobs) by relaunching the command without starting again from zero.

There are two different ways to use rsync that will be described below in detail:

- via **batch job** or **interactive session**: it is the recommended choice when transferring files between **two machines with public IP** (CINECA HPC but also to machines different from CINECA). You will use a dedicated queue with a **time limit of 4 hours**.
- via **command line** on login nodes or on your pc: the only possibility when transferring files to/from machines **without public IP** (e.g. your personal PC). Regardless you run the command on our clusters or on your personal PC the command will create a rsync process on a login node of our cluster with a **time limit of 10 minutes**. As a consequence after about 10 minutes the transfer will be killed. This choice is very useful and quick for small files but difficult for files larger than 10 GB. This is valid for all our clusters except **Marconi100** for which we have set up a [different solution](#).

For **very large data set** (>~ 500GB), we strongly suggests using **Globus online** via [GridFTP protocol](#).

Rsync via batch job or interactive session

This is the recommended solution in case you would like to move or retrieve data from other CINECA HPC machines or other places with a public IP. By using the serial queue you have up to 4 hours to complete your data transfer for each job. In addition by using **the serial queue you will not consume your budget**.

Interactive session

When running rsync command you need to **insert the password** if you don't have exchanged a public key between the two clusters (not recommended for security reasons).

You can open an interactive session:

```
srunk -N1 -n1 --cpus-per-task=1 -A <account_name> -p <serial_queue_name> --time=04:00:00 --pty /bin/bash
```

then run your rsync command like in the dedicated section and insert the password when requested.

Single job

If you have exchanged a public key between the two clusters (see [How to connect by a public key](#)) you don't have to type the password.

WARNING: it is not recommended to leave private keys for a long time on public clusters because in case of security breach they can be stolen and used to move towards other clusters. Therefore we recommend to use rsync between clusters via interactive session and use this possibility only when really needed. We strongly recommend to **remove the private key** after the data transfer is completed.

In this case you can submit a job script like in the example below where we move data between two CINECA HPC machines:

```
##### serial queue 1 cpu #####

#!/bin/bash
#SBATCH --out=job.out
#SBATCH --time=04:00:00
#SBATCH --nodes=1 --tasks-per-node=1 --cpus-per-task=1 --mem=4096
#SBATCH --account=<account name>                                ##you can find the name of the account by using
"saldo -b" command
#SBATCH --partition=<serial queue name>                          ##you can find the name of the dedicated queue
by using "sinfo|grep serial" command
#

cd <data_path_to>

rsync -PrvzHS </data_path_from/dir> <username>@login.<hostname>.cineca.it:<data_path_to>

#####
```

You can ask for more than a single cpu (--cpu-per-tasks) and execute on each cpu an rsync command for different data.
Example:

```
##### serial queue 2 cpu #####
```

```
#!/bin/bash
#SBATCH --out=%j.job.out
#SBATCH --time=04:00:00
#SBATCH --nodes=1 --tasks-per-node=1 --cpus-per-task=2 --mem=4096
#SBATCH --account=<account name>                                ##you can find the name of the account by using
"saldo -b" command
#SBATCH --partition=<serial queue name>                          ##you can find the name of the dedicated queue
by using "sinfo|grep serial" command
#

cd <data_path_to>

rsync -PravzHS <username>@login.<hostname>.cineca.it:</data_path_from/dir1> <data_path_to1> &

rsync -PravzHS <username>@login.<hostname>.cineca.it:</data_path_from/dir2> <data_path_to2> &

wait
#####
```

Chaining multiple jobs

If your data copy requires more than 4 hours you can run concatenated jobs taking advantage of the fact that you can interrupt and restart the file transfer with rsync.

Each concatenated job has up to 4 hours of time limit in order to copy the data starting from the file where the previous job was interrupted.

```
$sbatch job1.cmd
  submitted batch job 100
$sbatch --dependency=afternotok:100 job2.cmd
  submitted batch job 102
```

where job1.cmd and job2.cmd are job scripts like the ones shown above.

The available options for -d or --dependency are:

afterany:job_id[:jobid...], afternotok:job_id[:jobid...], afterok:job_id[:jobid...], ... etc..

See the sbatch man page for additional details.

Rsync via command line

You can launch rsync via command line on login nodes in the following way:

```
----- CINECA /space1/ <-> CINECA /space2/ -----

rsync -PravzHS </data_path_from/file> <data_path_to>

----- CINECA /HPC machine 1/ <-> CINECA /HPC machine 2/ -----

rsync -PravzHS </data_path_from/file> username@login.<hostname>.cineca.it:<data_path_to>

----- CINECA -> LOCAL/HPC machine -----
rsync -PravzHS username@login.<hostname>.cineca.it:</data_path_from/dir> <data_path_to>

----- LOCAL/HPC machine -> CINECA -----
rsync -PravzHS <data_path_from/dir> username@login.<hostname>.cineca.it:<data_path_to>
```

We remind again that, on CINECA's cluster, the maximum cpu time available via the command line is 10 min. If your rsync connection will be killed after this time (i.e for big file >10 GB) and your transfer has not been completed execute again the same rsync command until the data transfer will be completed.

Dedicated node for Marconi100 and Galileo100

For Marconi100 and Galileo100 in order to avoid the 10 minutes limit we set up a dedicated VM accessible with a dedicated alias.

Login via ssh to this VM is not allowed. As a consequence you always have to explicitate the complete path to the files you need to copy. Environment variables as \$HOME or \$WORK are not recognized.

In this case you can use the command:

```
rsync -PravzHS </data_path_from/file> <your_username>@data.<cluster>.cineca.it:<complete_data_path_to>
```

where <cluster> can be m100 as Marconi100 or g100 as Galileo100.

In similar ways you can use also scp and sftp commands if you prefer them.

You can also use the data VM from login nodes to move data from Marconi100 or Galileo100 to another location with public IP.

```
ssh -xt data.<cluster>.cineca.it rsync -PravzHS <complete_data_path_from/file> </data_path_to>
```

this command will open a session on the VM that will not be closed until the rsync command is completed.