

Istruzioni tecniche d'uso del cluster GPU DISI (aggiornamento luglio 2022)

Per l'utilizzo del cluster il primo passo è l'**abilitazione dell'account istituzionale** unibo.it all'accesso ai sistemi dipartimentali e al cluster stesso. In caso non si sia già abilitati si riceverà una e-mail di avvenuta abilitazione. Con le credenziali istituzionali si avrà accesso, anche in remoto, a tutte le macchine dei laboratori Ercolani e Ranzani. Nell'e-mail si trova il link dedicato ai servizi informatici dipartimentali (<https://disi.unibo.it/it/dipartimento/servizi-tecnici-e-amministrativi/servizi-informatici>) e nella sezione Accesso remoto si trovano dettagli su tali macchine. Oltre a ciò, è possibile accedere allo stesso modo alla macchina **slurm.cs.unibo.it** su cui si trova lo schedatore del cluster e su cui è necessario predisporre l'ambiente di esecuzione dei job in quanto sono presenti le versioni aggiornate di Python ed eventuali ulteriori librerie richieste.

La **quota utente** massima è ad oggi impostata a 400 MB. In caso di necessità di maggiore spazio è possibile ricorrere alla creazione di una propria directory in **/public.hpc/**, che non è sottoposta ad alcuna politica di cancellazione forzata (la directory similare **/public/** viene invece di norma cancellata ogni prima domenica del mese). La home utente è uno spazio di archiviazione condiviso tra le macchine; quindi l'ambiente di esecuzione e i file necessari all'elaborazione presenti sulla macchina **slurm.cs.unibo.it** da cui poi avviare il job che verrà eseguito sulle macchine dotate di GPU saranno visibili anche su tutte le altre macchine dei laboratori. La directory **/public.hpc/** è invece visibile esclusivamente dalla macchina **slurm.cs.unibo.it**.

Una possibile **impostazione del lavoro** è quella di creare sulla macchina **slurm** un virtual environment Python inserendo all'interno tutto ciò di cui si ha bisogno e utilizzando **pip** per l'installazione dei moduli necessari. Si segnala che per utilizzare Python 3 è necessario invocarlo esplicitamente in quanto sulle macchine il default è Python 2. Nel cluster sono presenti GPU Turing pilotate con driver Nvidia v. 470 e librerie di computazione CUDA 11.4, quindi in caso di installazione di **pytorch** bisognerà utilizzare il comando **pip3 install torch --no-cache-dir --extra-index-url <https://download.pytorch.org/whl/cu113>** (rif. <https://pytorch.org/>).

N.B.: Il gestore di pacchetti pip utilizza per default una cache nello spazio utente, e la relativa quota potrebbe esaurirsi velocemente. Si consiglia quindi di includere sempre il parametro **--no-cache-dir** nel comando di installazione di moduli, e nel caso si debba cancellare una cache esistente di utilizzare il comando **pip3 cache purge**.

Il cluster utilizza uno schedatore SLURM (<https://slurm.schedmd.com/overview.html>) per la distribuzione dei job. Per effettuare il submit di un job bisogna predisporre nella propria area di lavoro un file di script SLURM (ad es. *script.sbatch*) in cui inserire le direttive per la configurazione del job stesso. Dopo le direttive è possibile inserire comandi di script (ad es. BASH). Un esempio di script è il seguente:

```
#!/bin/bash
#SBATCH --job-name=nomejob
#SBATCH --mail-type=ALL
#SBATCH --mail-user=nome.cognome@unibo.it
#SBATCH --time=01:00:00
#SBATCH --nodes=1
#SBATCH --ntasks-per-node=1
#SBATCH --output=nomeoutput
#SBATCH --gres=gpu:1
```

```
. bin/activate # per attivare il virtual environment python
```

```
python test.py
```

Nell'esempio precedente la direttiva da riportare immutata è `--gres=gpu:1` (ogni nodo di computazione ha un'unica GPU a disposizione e deve essere attivata per poterla utilizzare). Le altre possono essere personalizzate. Per la definizione di queste e altre direttive si rimanda alla documentazione di SLURM (<https://slurm.schedmd.com/sbatch.html>). Nell'esempio, dopo le direttive è stato invocato il programma. Il processo deve essere accodato dalla macchina `slurm.cs.unibo.it` (accessibile via `ssh`) e lanciato il comando **`sbatch nomecript`** (ad es. **`sbatch script.sbatch`**). Con le direttive specificate nell'esempio saranno inviate e-mail all'indirizzo specificato alla partenza del job, al termine e nel caso di errori. I risultati dell'elaborazione saranno presenti nel file `nomeoutput` come indicato nella direttiva.

L'esecuzione sulle macchine avviene all'interno dello stesso path relativo che, essendo condiviso, viene visto dalle macchine dei laboratori, dalla macchina `slurm` e dai relativi nodi di elaborazione (ad eccezione della directory `/public.hpc/` che non è visibile dalle macchine dei laboratori).

Si segnala inoltre questo [repository Github](#) con alcune risorse per l'utilizzo del cluster.