

Exécution de code sur une infrastructure HPC

Les clusters des hautes écoles genevoises

Pierre Künzli

hepia

Présentation générale

- **Cluster de calcul :**

- Ensemble d'ordinateurs (les noeuds),
- fonctionnant sous Linux,
- souvent équipés de coprocesseurs (en général des GPUs),
- interconnectés par un réseau rapide (souvent en topologie *fat tree*)
- disposants d'un stockage partagé.

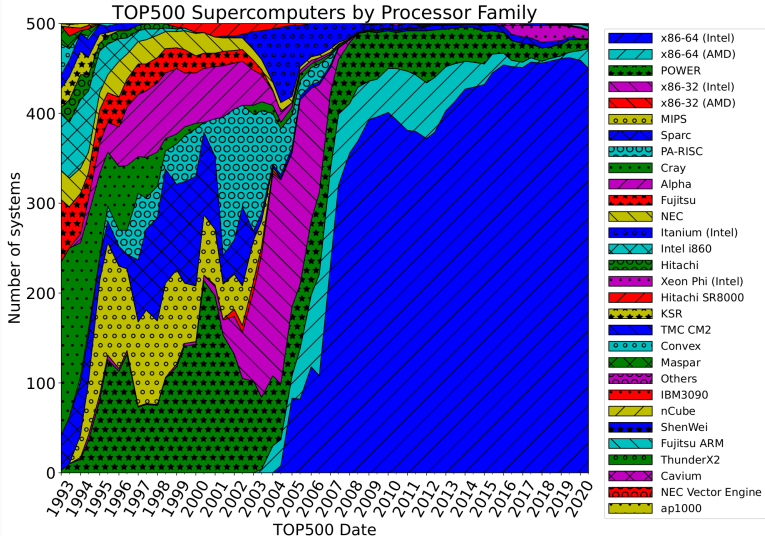
- **Accès :**

- On accède au cluster via un noeud *frontal*,
- depuis ce frontal, on utilise un système de queuing qui gère l'accès aux ressources.

- **Machine de calcul parallèle fortement couplée à mémoire distribuée.**



https://en.wikipedia.org/wiki/Beowulf_cluster



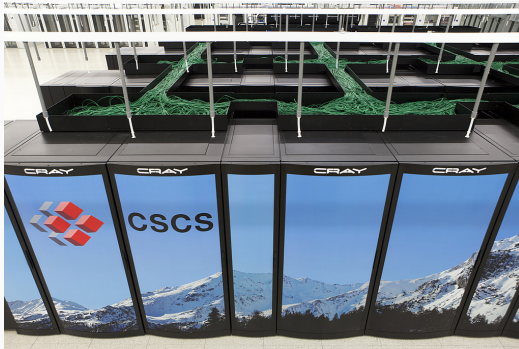
Un cluster de Playstation 3 (EPFL)

h e p i a

Haute école du paysage, d'ingénierie
et d'architecture de Genève



<https://www.win.tue.nl/~bdeweger/PS3Lab/>



Piz Daint 7517 noeuds, > 130'000 coeurs, > 480 TB de RAM, 5700
NVIDIA Tesla P100



Arolla et Tsa (Meteosuisse), deux machines jumelles (production + R&D et failover), Cray, 18 noeuds, 144 NVIDIA V100 GPUs, 4.6TB VRAM, 288 coeurs CPU Intel Xeon, 6.9TB RAM.

20 pre/post processing nodes, 800 coeurs Intel Xeon, 7.7TB RAM.

Clusters des hautes écoles genevoises.

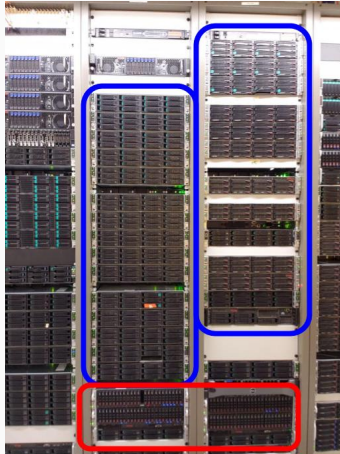
- **Baobab**, mis en service en 2013 à Uni Dufour.
 - 900 coeurs publics, 4200 coeurs au total,
 - 273 GPUs,
 - réseau 100GB IB.
- **Yggdrasil**, mis en service en 2020 à l'observatoire de Genève, Sauverny.
 - 3000 coeurs publics, 4300 coeurs au total,
 - 52 GPUs,
 - réseau 100GB IB.
- **Bamboo**, mis en service en 2024 au campus Biotech.
 - 5700 coeurs,
 - 20 GPUs,
 - réseau 100GB IB.

Machines hétérogènes, détails sur [https:](https://doc.ereseach.unige.ch/hpc/hpc_clusters#compute_nodes)

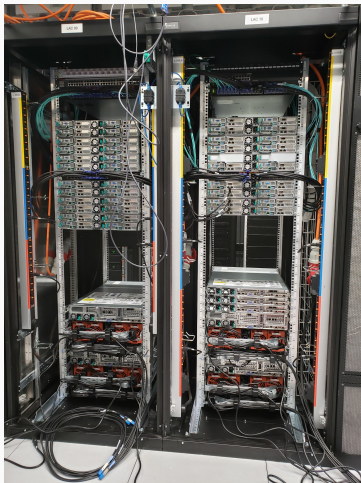
[//doc.ereseach.unige.ch/hpc/hpc_clusters#compute_nodes](https://doc.ereseach.unige.ch/hpc/hpc_clusters#compute_nodes)



Baobab vu de face et un châssis.



Baobab vu de face, en bleu noeuds de calcul, en rouge noeuds de stockage.

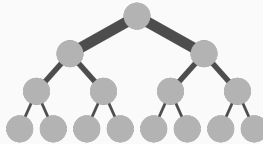


Deux armoires de Yggdrasil pendant le montage (six en tout).



Yggdrasil pendant le montage. Un câble infiniband par noeud, deux alimentations par châssis.

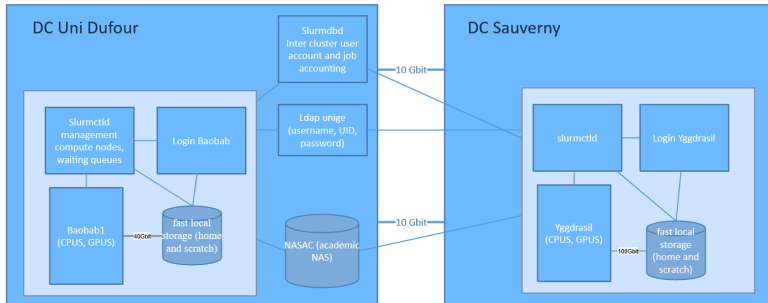
La topologie réseau *fat tree* a l'avantage d'être performante et scalable (facile d'ajouter des noeuds). Elle est souvent utilisée pour les clusters.



https://en.wikipedia.org/wiki/Fat_tree



Cluster choice: Baobab (legacy) or Yggdrasil (launched in 2020)
You can use either cluster, but remember that data aren't
shared between clusters, at least for the moment.
If you have private nodes on one of the cluster, use this one.



https://doc.eresearch.unige.ch/hpc/hpc_clusters#how_do_our_clusters_work

Utilisation

Depuis une machine sur laquelle se trouve le couple clé privé/publique dont vous avez fourni la clé publique :

```
~$ ssh username@login1.baobab.hpc.unige.ch
```

ou

```
~$ ssh username@login1.yggdrasil.hpc.unige.ch
```

ou

```
~$ ssh username@login1.bamboo.hpc.unige.ch
```

Vérifiez les adresses des login nodes :

https://doc.eresearch.unige.ch/hpc/access_the_hpc_clusters

Utilisez une clé ssh avec une passphrase !

Sur votre machine, vous pouvez aussi créer un fichier `.ssh/config` pour donner des noms courts aux hosts ssh.

Par exemple :

```
Host baobab
    Hostname login1.baobab.hpc.unige.ch
    User USERNAME
```

Il vous suffira de faire `ssh baobab` ou `ssh yggdrasil` pour vous connecter.

Exemples donnés avec Baobab, fonctionne aussi avec Yggdrasil.

Pour copier un fichier vers baobab:

```
~$ scp fichier  
    username@login1.baobab.hpc.unige.ch:/destination
```

Et dans l'autre sens:

```
~$ scp  
    username@login1.baobab.hpc.unige.ch:/source  
    fichier
```

Pour copier tout un repertoire:

```
~$ scp -r  
    username@login1.baobab.hpc.unige.ch:/destination  
    dossier
```

- Commandes à exécuter depuis votre machine locale, pas depuis un cluster.
- Attention à ne pas supprimer le dossier local avec le montage activé ! (suppression récursive du contenu du dossier).

```
# creer un dossier qui sert de point de montage
~$ mkdir baobab
```

```
# monter le home de baobab sur le point de
montage
~$ sshfs username@login1.baobab.hpc.unige.ch:.
baobab
```


<https://rclone.org/>

Une fois connecté, vous devez charger les modules à utiliser, par exemple

```
~$ module load NVHPC/24.1-CUDA-12.3.0
```

Vous pouvez ajouter cette commande dans votre fichier `.bashrc` pour qu'elle soit exécutée automatiquement à chaque connexion.

Vous pouvez aussi essayer

```
~$ module list
```

```
~$ module spider "appToLoad"
```

```
~$ module purge "appToRemove"
```

Ordonnanceur Slurm

- Les clusters genevois utilisent le système de queuing *slurm* (standard dans le domaine),
- vous devez **TOUJOURS** passer par Slurm (commande `srun` ou `sbatch`) pour lancer un calcul (un *job*) sur un cluster,
- il ne faut **JAMAIS** exécuter un calcul sans passer par Slurm.

Si vous n'utilisez pas Slurm, votre programme s'exécutera sur le noeud de login au lieu des noeuds de calcul. Cela vous empêche d'utiliser des GPUs et impacte l'expérience de tous les utilisateurs du cluster.

```
#!/bin/sh
#SBATCH --job-name=NomDuJob
#SBATCH --output=NomOutput.o%j
#SBATCH --ntasks=1
#SBATCH --cpus-per-task=1
#SBATCH --mem-per-cpu=1000
#SBATCH --partition=shared-cpu
#SBATCH --time=00:10:00
```

```
echo $SLURM_NODELIST
```

```
srun monprogramme
```

Si vous enregistrez ce script dans un fichier nommé `script.sh`, vous pouvez le soumettre sur le cluster avec la commande :

```
~$ sbatch script.sh
```

Afficher tous les jobs du cluster

```
~$ squeue
```

Limiter l'affichage à vos jobs

```
~$ squeue -u username
```

Affiche des informations a propos du job

```
~$ scontrol show jobid <jobid>
```

```
~$ sacct -j <jobid>
```

Pour supprimer un job, vous pouvez utiliser la commande `scancel`. Soit avec:

```
~$ scancel jobid
```

ou

```
~$ scancel jobname
```

ou encore

```
~$ scancel -u username
```

Partitions : ensembles de noeuds de calcul avec certaines politiques d'exécution (par exemple temps d'exécution maximum).

Pour les voir :

~\$ sinfo

Public partitions

- Partitions public-cpu et public-gpu (Yggdrasil et Bamboo),
- temps d'exécution limité à 4 jours avec uniquement noeuds publics,
- pour les longs calculs.

Shared partitions

- Partitions shared-cpu et shared-gpu,
- temps d'exécution limité à 12 heures avec tous les noeuds,
- bonne option la plupart du temps.

Debug partitions

- Partitions debug-cpu et debug-gpu (Yggdrasil et Bamboo),
- temps d'exécution limité à 15 minutes,
- permet de tester rapidement votre code.


```
salloc --partition=shared-cpu --time=0-01:00:00
```

Une fois la ressource allouée pour le job, vous y serez connecté automatiquement. Vous serez déconnecté automatiquement à la fin du temps alloué.

- **Documentation** <https://doc.eresearch.unige.ch/hpc/start>
- **Forum communautaire** <https://hpc-community.unige.ch/>
- **Contact de l'équipe HPC** hpc@unige.ch
- **Pour mettre à jour votre clé ssh**
<https://applicant.unige.ch/main/outsider-info/update>
- **Outils de monitoring**
<https://monitor.hpc.unige.ch/dashboards>
<https://openxdmod.hpc.unige.ch/>

Prenez un vieux TP (de C par exemple) et faites le fonctionner sur un cluster.