



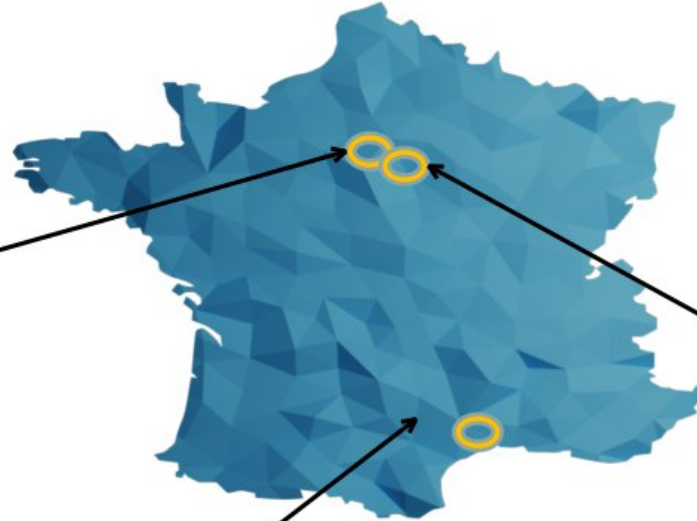
IDRIS / Jean Zay

IDRIS:

Institut du développement et des ressources en informatique scientifique



Jean Zay Supercomputer
HPE SGI 8600 +
Eviden Bull Sequana XH3000
125,9 PFlop/s
2019
IDRIS CNRS



Ressources gratuites

- financement français
- recherche ouverte



Adastra Supercomputer
HPE Cray EX4000,
74.5 PFlop/s
2022
CINES Universities



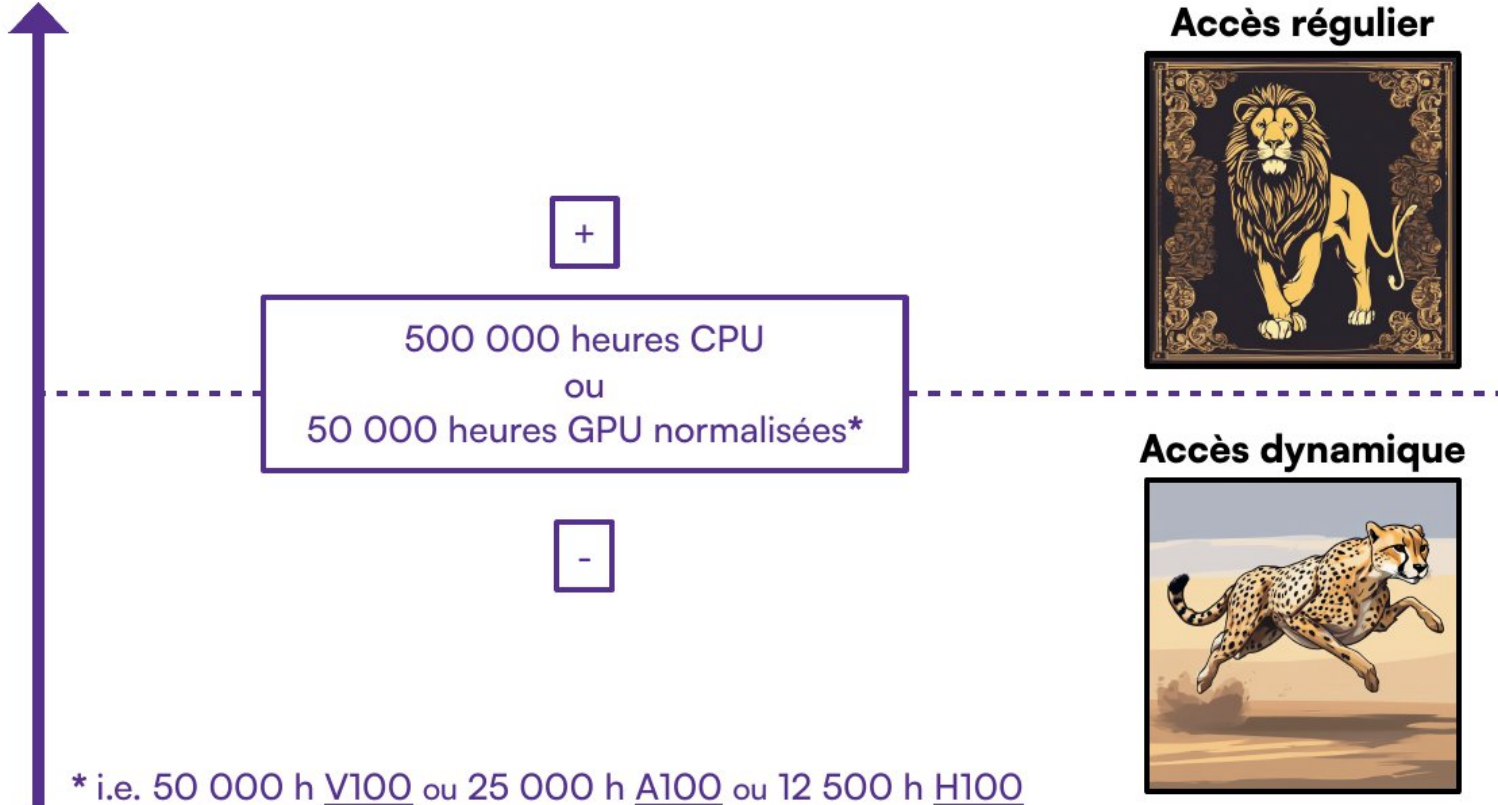
Joliot-Curie Supercomputer
Bull Sequana X1000 and XH2000,
22 PFlop/s
2017
TGCC CEA



IDRIS

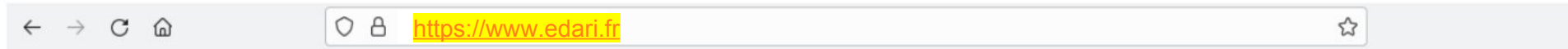
Access to resources

Nombre d'heures demandées



* i.e. 50 000 h V100 ou 25 000 h A100 ou 12 500 h H100

Hours request + account opening



DARI Demande d'Attribution de Ressources Informatiques

Se connecter ou se créer un compte eDARI

Comment obtenir des ressources de calcul et de stockage ?

Demande de ressources ou demande d'accès



Pour qui, quel usage et quelles conditions ?



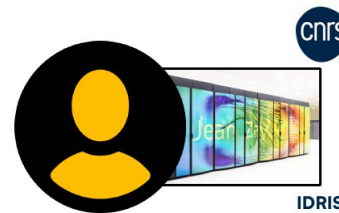
Documents de référence



Actualités

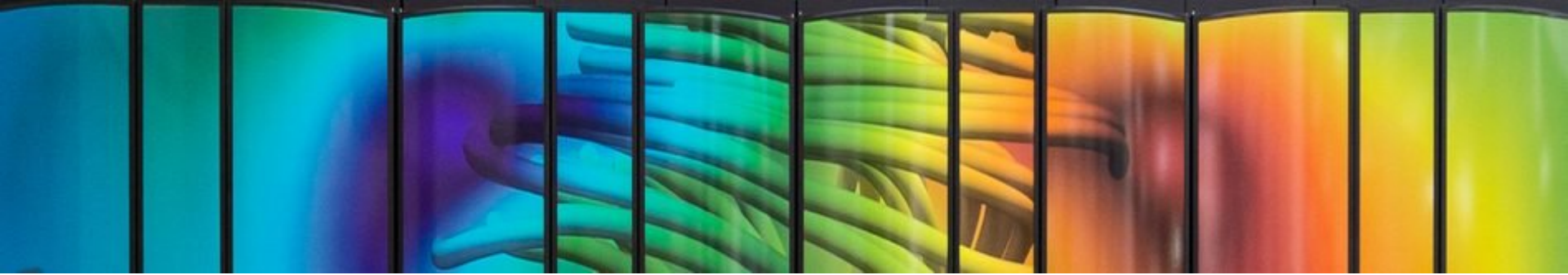


Allocation d'heures



Compte de calcul IDRIS





IDRIS / Jean Zay

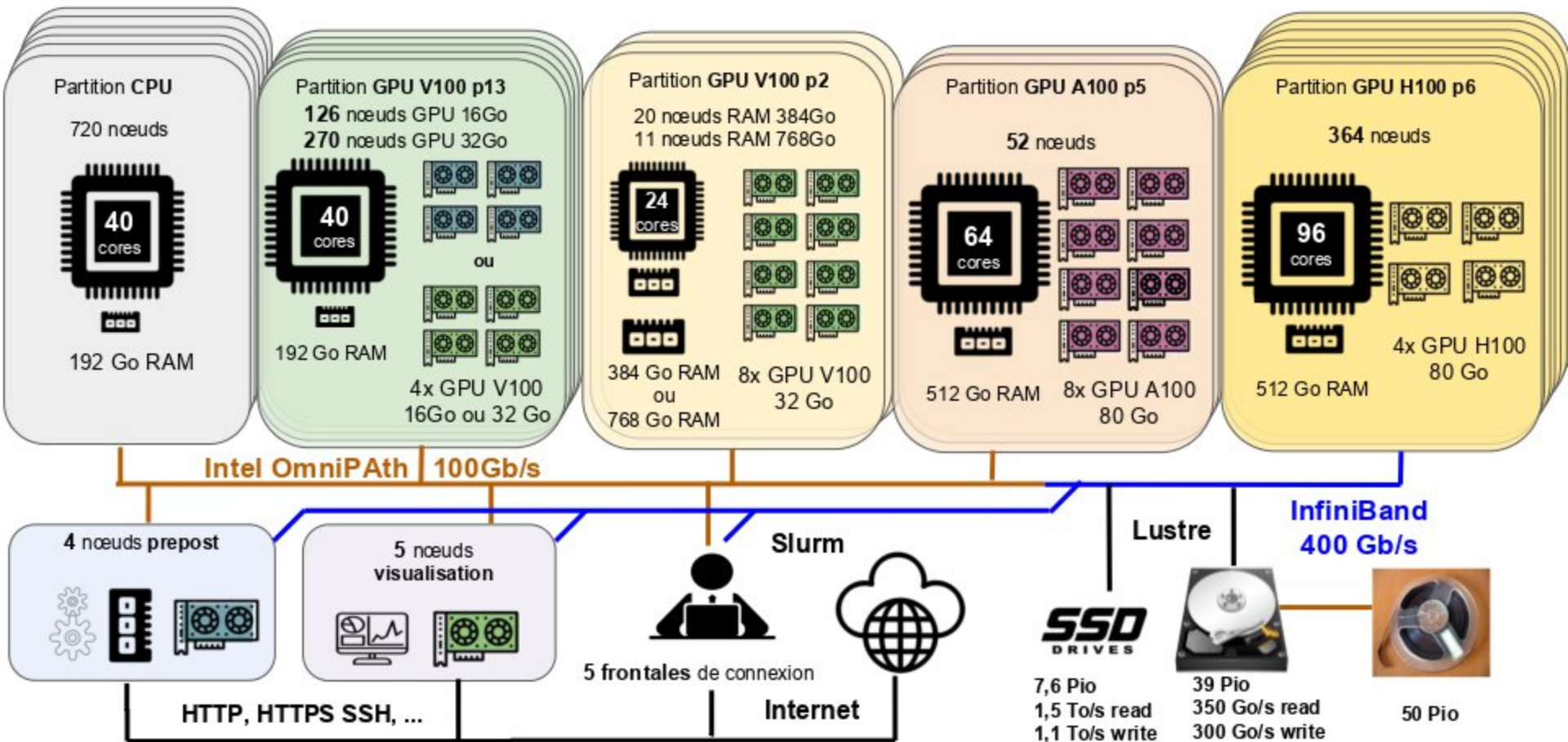
Jean Zay supercomputer

28800 CPU
2,3 PFlop/s

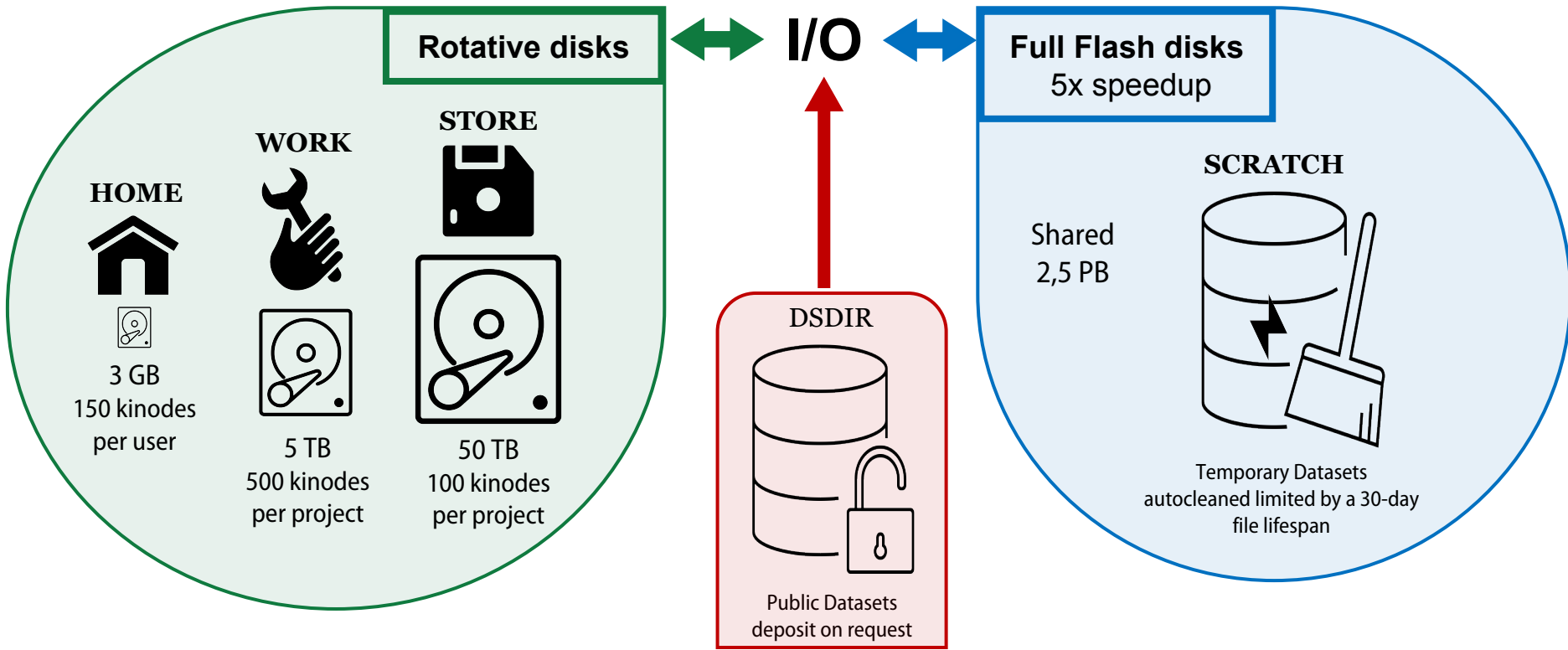
1832 GPU V100
15,5 PFlop/s

416 GPU A100
8,2 PFlop/s

1456 GPU H100
99,9 PFlop/s



Storage spaces





IDRIS / Jean Zay

Work environment on Jean Zay

Connexion

- La connexion à Jean Zay se fait en **SSH** à partir d'une machine dont l'adresse IP est déclarée et rattachée à votre compte de calcul. Sous Windows, la connexion peut se faire via PuTTY, WSL ou MobaXterm.

```
localname@localPC:~$ ssh username@jean-zay.idris.fr
username@jean-zay.idris.fr's password :
Last login : Fri May 20 16 :14 :21 2022 from 255.255.255.255
*****
* Ceci est un serveur de calcul de l'IDRIS. Tout acces au systeme      *
* doit etre specifiquement autorise par l'IDRIS. Si vous tentez de    *
* de continuer a acceder cette machine alors que vous n'y etes pas    *
* autorise, vous vous exposez a des poursuites judiciaires.          *
*                               --                                       *
* This is an IDRIS compute node.  Each access to this system must be  *
* properly authorized by IDRIS. If you go on accessing this machine   *
* without authorization, then you are liable to prosecution.          *
*****
*
* Orsay          CNRS / IDRIS - Frontale - jean-zay.idris.fr  France  *
*
*****
[username@jean-zay3 : ~]$
```

Les frontales

5 nœuds de connexion (frontales sans GPU) :

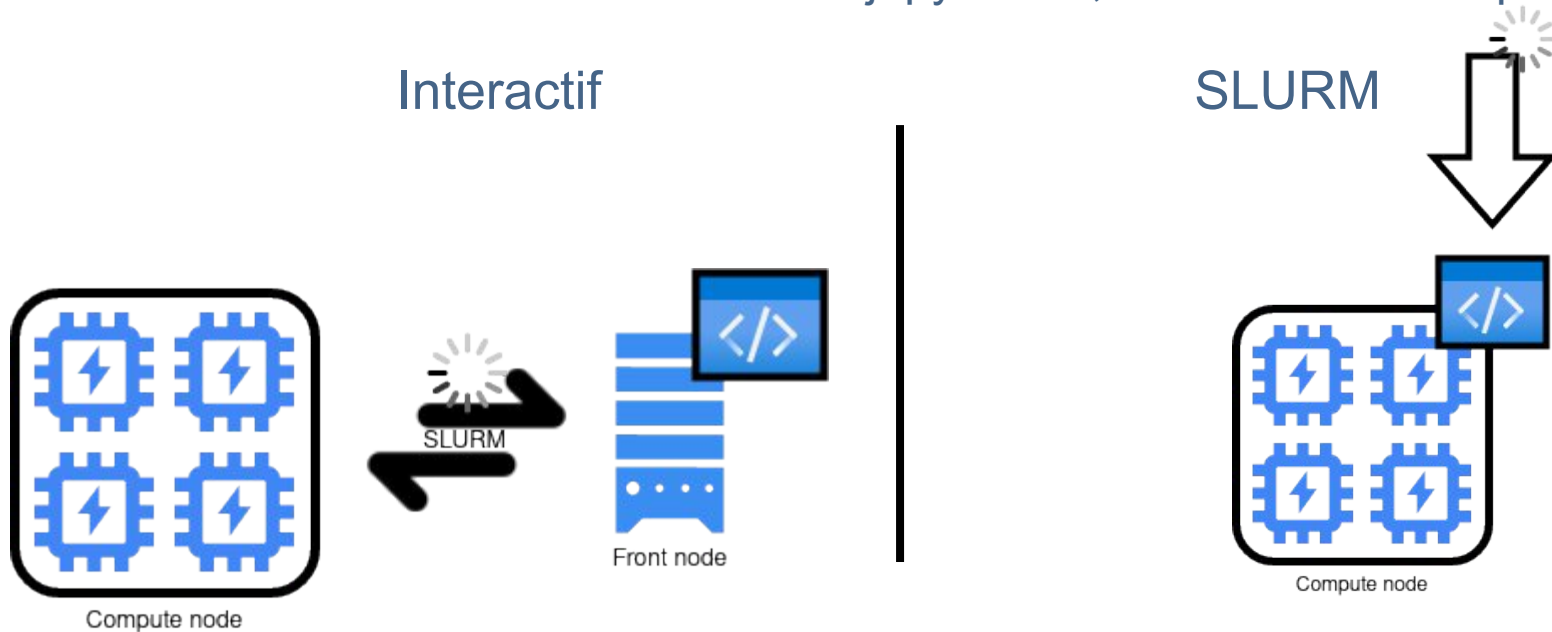
- partagés par l'ensemble des utilisateurs
- dédiés à la mise en place de l'environnement de calcul (compilation, transferts de données,...)
- proxy HTTP permettant de télécharger des données depuis des serveurs distants, avec git ou wget par exemple
- Pas d'exécution en mode production : risque de *crash* du nœud

Limitations:

- 1 cœur par utilisateur
- 5 GB de mémoire RAM par utilisateur
- 30 min temps CPU par processus

Les outils disponibles : Jupyterhub

- Jupyterhub :
 - Pour le développement il est possible de lancer une session jupyterhub en se connectant à <https://jupyterhub.idris.fr/> avec ses identifiants Jean-Zay.
 - Lors de la création d'une instance jupyterhub, deux modes sont proposés :



JupyterHub



1. Authentication on <https://jupyterhub.idris.fr>

Sign in

Username:
user

Password:

Sign in

2. Choose and configure an instance

Server Options

Spawn server on login (frontal) node

Spawn server on SLURM node

Run on a compute node

Run on a login node

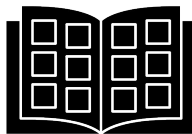
3. Choose a kernel (pytorch-gpu-2.2.0)

Notebook

Python 3 (pykernel)	alphafold2.1.2	atomistic_simulation	bloom-pytorch-1.13.0
mxnet-gpu-1.8.0	parcels	parcels_seq	python-3.10.4
pytorch-1.8.1_py3.8.8-lts	pytorch-cpu-1.7.1	pytorch-gpu-1.1	pytorch-gpu-1.10.0_py3



Catalogue of shared modules (conda environments)



- Installed by IDRIS
- Completed on request

```
login@jean-zay3:~$ module load pytorch-gpu/py3/1.11.0  
Loading requirement: ...  
(pytorch-gpu-1.11.0+py3.9.12) login@jean-zay3:~$
```

- Customizable

```
~$ pip install --user --no-cache-dir <paquet>
```



Conflicts between versions
Storage spaces saturation

Personal conda environments

```
login@jean-zay3:~$ module load anaconda-py3/2023.03  
(base) login@jean-zay3:~$ conda create -n myenv
```



Storage spaces saturation ++

Singularity containers

```
login@jean-zay3:~$ module load singularity
```

Import SIF images on Jean Zay

- From your PC
- From public deposits
- Possible conversion from docker



Les outils disponibles : modules

❖ Un module est :

- Un environnement conda (pour les programmes python)
- Une mise en commun des ressources logicielles
- Maintenu et mis à jour par l'équipe support
- Un accès à des librairies compilées pour l'architecture de Jean Zay

❖ Utilisation des modules :

- `module avail <nom_du_module>` : pour rechercher un module
- `module load <nom_du_module>` : pour charger un module
- `module purge` puis `conda deactivate` pour décharger un module

❖ Informations complémentaires :

- PyTorch, TensorFlow sont disponibles pour l'IA
- Si des paquets sont manquants dans un module :
 - Contacter l'assistance pour qu'ils soient installés (assist@idris.fr)
 - [Installer le paquet en local](#)

Partitions et QOS Slurm

- 2 types de partition : 1 CPU pour utiliser les nœuds CPU et 4 GPU les nœuds convergés (ayant des GPU).
- Plus partitions dédiées à de la compilation (*-p compil*), du pré-post traitement (*-p prepost*), de la visualisation (*-p visu*), aux copies de données (*-p archive*) : heures non comptabilisées
- 2 ou 3 types de QOS (Quality Of Service) suivant la partition CPU ou GPU définissant les temps de calcul maximaux (2h, 20h ou 100h) et le nombre maximal de cœurs ou de GPU pouvant être requis.
 - http://www.idris.fr/jean-zay/cpu/jean-zay-cpu-exec_partition_slurm.html
 - http://www.idris.fr/jean-zay/gpu/jean-zay-gpu-exec_partition_slurm.html

Les espaces disques

Espace	Capacité par défaut	Spécificité	Usage
\$HOME	3 Go et 150 <i>kinodes</i> par utilisateur	- Accueil de connexion	- Stockage de fichiers de configuration et de petits fichiers
\$WORK	5 To (*) et 500 <i>kinodes</i> par projet	- Stockage sur disques rotatifs (350 Go/s en lecture et 300 Go/s en écriture)	- Stockage des sources et des données d'entrée/sortie - Exécution en batch ou interactif
\$SCRATCH	Quotas de sécurité très larges 4,6 Po partagés par tous les utilisateurs	- Espace non sauvegardé - Stockage SSD (1,5 To/s en lecture et 1,1 To/s en écriture) - Durée de vie des fichiers inutilisés : 30 jours (inutilisés = non lus, non modifiés)	- Stockage des données d'entrée/sortie volumineuses - Exécution en batch ou interactif - Performances optimales pour les opérations de lecture/écriture
\$STORE	50 To (*) et 100 <i>kinodes</i> (*) par projet	- Cache disque et bandes magnétiques - Accès longs si fichier uniquement sur bande. - Sécurisé par double copie sur bandes magnétiques des fichiers non modifiés pendant quelques jours.	- Stockage d'archives sur du long terme (durée de vie du projet) - Pas accessible depuis les nœuds de calcul

(*) les quotas par projet peuvent être augmentés sur demande du chef de projet ou de son suppléant via l'interface [Extranet](#) ou sur demande auprès du support utilisateurs.

Les espaces disques

- Partage de données entre les membres d'un même projet : `$ALL_CCFRWORK`, `$ALL_CCFRSCRATCH` et `$ALL_CCFRSTORE`
- `ALL_CCFRWORK` (respectivement `ALL_CCFRSTORE`) est soumis aux quotas du `WORK` (respectivement `STORE`).
- `$DSDIR` : Data-Sets publiques installés par l'assistance.
- Liste des espaces disques : `idrenv`
- Documentation : <http://www.idris.fr/jean-zay/cpu/jean-zay-cpu-calculateurs-disques.html>

Lancer un calcul

Pour exécuter un code **sur les nœuds de calcul de Jean Zay**, il y a deux façons de procéder :

- une exécution interactive
- une exécution en mode batch via un job Slurm
- Exemple d'une exécution interactive CPU sur un nœud de calcul avec 4 cœurs

```
*$ srun --pty --ntasks=1 --cpus-per-task=4 --hint=nomultithread --account=xxx@cpu bash
```

--pty permet d'obtenir un terminal interactif

--hint=nomultithread assure la réservation des cœurs **physiques** (pas d'*hyperthreading*)

Les réservations ont toutes des ressources définies dans Slurm par une partition et une "Qualité de Service" *QoS* (*Quality of Service*) par défaut

Le terminal est opérationnel après validation de l'allocation :

```
$ srun --pty --ntasks=1 --cpus-per-task=4 --hint=nomultithread bash
srun: job 1365358 queued and waiting for resources
srun: job 1365358 has been allocated resources
bash-4.2$ hostname
r4i3n7
```

Lancer un calcul

- Exemple pour une exécution en mode batch sur la partition accélérée quadri-GPU sur un nœud entier de la partition par défaut gpu_p13, soit 4 GPU

```
# !/bin/bash
#SBATCH --job-name=gpu_mono           # nom du job
##SBATCH --partition=gpu_p2          # de-commente pour la partition gpu_p2
#SBATCH --nodes=1                    # on demande un nœud
#SBATCH --ntasks-per-node=1          # avec une tache par nœud (= nombre de GPU ici)
#SBATCH --gres=gpu:1                 # nombre de GPU (1/4 des GPU)
#SBATCH --cpus-per-task=10           # nombre de cœurs CPU par tache (1/4 du nœud 4-GPU)
##SBATCH --cpus-per-task=3           # nombre de cœurs CPU par tache (pour gpu_p2 : 1/8 du nœud 8-
GPU)
# / !\ Attention, « multithread » fait référence à l'hyperthreading dans la terminologie Slurm
#SBATCH --hint=nomultithread         # hyperthreading désactivé
#SBATCH --time=00:10:00              # temps maximum d'exécution demande (HH :MM :SS)
#SBATCH --output=gpu_mono%j.out      # nom du fichier de sortie
#SBATCH --error=gpu_mono%j.out       # nom du fichier d'erreur (ici commun avec la sortie)
#SBATCH --account=xxx@v100

# nettoyage des modules chargés en interactif et hérités par défaut
module purge
# chargement des modules
module load tensorflow-gpu/py3/2.7.0
# écho des commandes lancées
set -x
# exécution du code
srun python file.py
```

Gérer un calcul

- Vous pouvez gérer vos calculs avec les commandes suivantes :
 - Vérifier vos jobs en queue : `squeue --me`
 - Annuler un job : `scancel <jobid>`

Outil CEEMS de mesure d'énergie

Accessible via un service d'authentification Single Sign-On (SSO) à l'adresse <https://jean-zay-ceems.idris.fr>

Modalités de connexion

On utilise le même nom d'utilisateur et le même mot de passe que pour accéder à la machine

Le filtrage des adresses IP par le pare-feu est aussi d'actualité.

Documentation

<http://www.idris.fr/jean-zay/jean-zay-doc-energie.html>

Useful links and contacts

Compute hours request:

- eDARI website: www.edari.fr
- GENCI website: www.genci.fr

Account opening: <http://www.idris.fr/eng/info/gestion/demandes-heures-eng.html>

IDRIS website: www.idris.fr

Jean Zay documentation:

- Main documentation: <http://www.idris.fr/eng/jean-zay/>
- Getting started: <http://www.idris.fr/eng/su/debutant-eng.html>
- Artificial Intelligence: <http://www.idris.fr/eng/ia/>
- Cheat sheet: <http://www.idris.fr/jean-zay/pense-bete.html>

Contacts:

- Compute jours request: acc@gencl.fr
- Account opening: gestutil@idris.fr
- Support team: assist@idris.fr