INSTITUT DU
DÉVELOPPEMENT ET DES
RESSOURCES EN
INFORMATIQUE
SCIENTIFIQUE

www.idris.fr

Débuter sur Jean Zay

Recherche et application en Intelligence Artificielle

Pourquoi cette documentation ?

- Cette présentation vise à guider les *nouveaux utilisateurs IA* de Jean Zay
- L'ensemble des informations délivrées ici est voulu *synthétique* pour assurer une prise en main rapide du supercalculateur
- Une documentation *complète* est mise à jour régulièrement par l'équipe Support aux Utilisateurs de l'IDRIS sur : www.idris.fr/jean-zay

NB : dans cette présentation, les symboles  représentent les liens hypertexte

Au programme

- Principales caractéristiques du supercalculateur Jean Zay

- Environnement administratif
- Demande d'accès aux ressources de calcul

- Environnement machine : connexion et espaces disques
- Environnement de calcul : modules et environnements virtuels
- Soumission des travaux : Slurm, partitions, QoS, batch et interactif
- Consommation des heures de calcul
- Partition de pré/post-traitement

- Pour aller plus loin
- Contacter le Support Utilisateurs

Principales caractéristiques de Jean Zay

- Jean Zay est un calculateur HPE SGI 8600 composé de deux partitions :
  - une partition *scalaire* ou « CPU » contenant 61120 cœurs de calcul
    ⟶ 1528 nœuds scalaires (192Go de mémoire, 40 cœurs @2,5GHz, Intel CSL 6248)
  - une partition *convergée* ou « GPU » contenant 1292 GPU
    ⟶ 261 nœuds convergés **quadri**-GPU
    (nœud scalaire à 192Go + 4 GPUs Nvidia V100 à 32 Go et 4 liens OPA)
    ⟶ 31 nœuds convergés **octo**-GPU
    (nœud scalaire à 384Go ou 768Go + 8 GPUs Nvidia V100 à 32 Go et 4 liens OPA)

- Puissance crête cumulée : 15,9 Pflops/s
- Réseau d'interconnexion Intel Omni-Path (bande passante 100Gb/s)
- Système de fichiers parallèle IBM Spectrum Scale (ex-GPFS)
- Deux dispositifs de stockage parallèle :
  ⟶ 1,3Po sur disques SSD (300Gio/s) et 300Po sur disques rotatifs (100Gio/s)
- 5 nœuds frontaux

Environnement administratif

GENCI

Société civile coordinatrice des trois grands centres de calcul nationaux

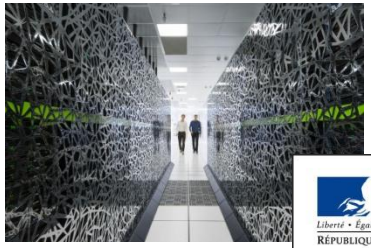
Calculateur hébergé à l'IDRIS

Jean Zay

Calculateur hébergé au TGCC

Joliot-Curie

Calculateur hébergé au CINES



Occigen

Liberté • Égalité • Fraternité
RÉPUBLIQUE FRANÇAISE

MINISTÈRE
DE L'ENSEIGNEMENT
SUPÉRIEUR,
DE LA RECHERCHE
ET DE L'INNOVATION

Demande d'accès aux ressources de calcul
- Procédure -

- L'ensemble de la démarche prend *une dizaine de jours* (un à deux mois supplémentaires si vous sujet à une enquête sécuritaire)

- Les accès dynamiques ont été mis en place pour demander un accès aux ressources de calcul de la partition GPU de Jean Zay
  → ces accès sont à destination des chercheurs et développeurs en *Intelligence Artificielle*
  → les demandes peuvent se faire à n'importe quel moment de l'année

- Une allocation est valable *un an* à partir de l'ouverture du projet

Demande d'accès aux ressources de calcul
- Procédure -

- L'allocation des heures de calcul est gérée par le GENCI via le portail DARI. Vous devez donc tout d'abord vous créer un compte utilisateur sur ce portail

- La demande se fait en deux étapes (onglet « Intelligence Artificielle ») :
  - une déclaration de dossier
    (description du projet scientifique, nombre d'heures demandées,…)
    → les demandes élevées sont étudiées par la direction de l'IDRIS et /ou des experts scientifiques avant approbation
  - une déclaration de compte de calcul
    (déclaration du responsable sécurité informatique, adresses IP,…)
    → pour les personnes ayant déjà un compte de calcul sur Jean Zay, un simple rattachement au nouveau projet suffit, il se fait via le Formulaire de Gestion de Compte (FGC)

- Après une semaine environ, vous recevez un mail de l'IDRIS contenant vos *login* et mot de passe de connexion à Jean Zay

Demande d'accès aux ressources de calcul
- Bon à savoir -

- Des demandes au fil de l'eau exceptionnelles sont déposables tout au long de l'année sur le portail DARI pour demander un complément d'heures
- Vous pouvez apporter des modifications à votre compte de calcul (rattachement à un nouveau projet, ajout d'une adresse IP,…) à n'importe quel moment via le Formulaire de Gestion de Compte (FGC)
- Le contact IDRIS pour l'ensemble de ces procédures est : gestutil@idris.fr

Environnement machine
- Connexion -

- La connexion à Jean Zay se fait en *ssh* à partir d'une machine dont vous avez déclaré l'adresse IP dans votre compte de calcul

```
ssh login@jean-zay.idris.fr
```

- Si vous travaillez sous Windows, la connexion peut se faire via *Putty*

- Vous arrivez sur un des 5 nœuds de connexion, ou *frontales*, de Jean Zay
  - ces nœuds sont partagés par l'ensemble des utilisateurs
  - ils sont dédiés à la mise en place de l'environnement de calcul (compilation, transferts de données, débogage,…)

Environnement machine
- Espaces disques -

- Il existe 4 espaces disques *majeurs* sur Jean Zay
  → leurs usages sont définis en fonction de leurs capacités de stockage (en Go et *inodes*-nombre de fichiers) et leurs spécificités techniques (temporalité, accès mémoire,…)

Espace	Capacité par défaut	Spécificité	Usage
`$HOME`	3Go / 150k *inodes* par utilisateur	Accueil de connexion. Espace sauvegardé.	Stockage de fichiers de configuration.
`$WORK`	5To / 500k *inodes* par projet	Stockage sur disques rotatifs (100Gio/s en lecture/écriture)	Stockage des sources et données d'entrée/sortie. Exécution en batch ou interactif.
`$SCRATCH`	Pas de quota 1,3Po partagés par tous les utilisateurs	Stockage SSD (300Gio/s en lecture/écriture). Suppression des fichiers non utilisés (non lus ou modifiés) au bout de 30 jours.	Stockage des sources et données d'entrée/sortie volumineuses. Exécution en batch ou interactif. Performances optimales pour les opérations de lecture/écriture.
`$STORE`	50To / 100k *inodes* par projet		Stockage d'archives.

Environnement machine
- Espaces disques -

- Pour consulter vos quotas disques : `idrquota -h`
- Les quotas des espaces « projet » `$WORK` et `$STORE` peuvent être augmentés sur demande auprès du Support Utilisateurs (assist@idris.fr)

- Vos espaces disques sont *cloisonnés* par défaut : vous seul avez les droits d'accès sur les fichiers qu'ils contiennent
- Pour partager des fichiers *avec les membres de votre projet*, il existe trois espaces dédiés :
  - dans le `$WORK` : `cd $ALL_CCFRWORK`
  - dans le `$SCRATCH` : `cd $ALL_CCFRSCRATCH`
  - dans le `$STORE` : `cd $ALL_CCFRSTORE`

- Sur demande, l'IDRIS peut importer des bases de données publiques et ouvertes dans l'espace disque `$DSDIR` (accessible à tous les utilisateurs)

Environnement de calcul
- Modules -

- L'IDRIS met à disposition un catalogue d'outils (compilateurs, librairies, environnements virtuels,…) accessibles via la commande *module*
- Ce catalogue peut être enrichi sur demande des utilisateurs

- Pour afficher le catalogue complet : `module avail`
- Pour rechercher un outil précis : `module avail nom_outil`
- Pour charger un module : `module load nom_outil/version`
- Pour décharger un module : `module unload nom_outil`

- Pour afficher la liste des modules chargés : `module list`
- Pour repartir d'un environnement vierge : `module purge`
  → attention, `module purge` ≠ `conda deactivate`

Environnement de calcul
- Modules -

- Certains modules « parents » (logiciels,…) dépendent de modules « fils » (compilateurs, version CUDA,…), appelés *prérequis*. Ces prérequis constituent un *environnement logiciel*.
- Certains produits existent pour différents environnements logiciels. L'inter-dépendance des modules peut donc générer des *conflits*.

- Pour connaître les environnements logiciels disponibles pour un outil donné, les prérequis associés et les conflits de dépendance :

```
module show nom_outil/version
```

- De manière générale, pour construire un environnement propre et sans conflit, nous conseillons de charger l'environnement logiciel **avant** les autres modules.

Environnement de calcul
- Les environnement virtuels -

- Les logiciels pour l'Intelligence Artificiel sont installés dans des environnements virtuels Anaconda, pour Python 2 et Python 3
  → la version Python 2.7 n'est plus maintenue par la communauté depuis le 01/01/2020

- Logiciels installés : *TensorFlow*, *PyTorch, Keras* et *Caffe*
  → les environnements sont activés lors du chargement des modules

- Logiciel de gestion parallèle : *Horovod* installé pour *TensorFlow* et *PyTorch*

- Logiciel de développement : *Jupyter Notebook* (avec *TensorBoard*) et *JupyterLab*

- Plus d'information sur : http://www.idris.fr/jean-zay/gpu/jean-zay-gpu-logiciels-ia.html

Soumission de travaux
- Slurm -

- Les travaux s'exécutent sur les *nœuds de calcul* de Jean Zay, accessibles par script batch ou en interactif.
- La file d'attente pour l'accès aux ressources de calcul est gérée par le gestionnaire *Slurm* pour l'ensemble des utilisateurs.
- Un système de *priorité* est mis en place pour garantir un partage des ressources le plus équitable possible entre les utilisateurs. En particulier :
  → un job peu coûteux en ressources aura une priorité plus forte qu'un job coûteux
  → lorsqu'un job est placé dans la file d'attente, sa priorité augmente au fil du temps
  → votre priorité sera forte si vous avez peu consommé dans un passé proche
  → votre priorité sera faible si vous avez beaucoup consommé dans un passé proche
  → passé proche = 28 derniers jours (la prise en compte décroissant avec une durée de demi-vie de 14 jours)
- Ce système incite les utilisateurs à consommer leurs heures régulièrement.
- Il arrive que la machine soit sous-utilisée (période estivale, fêtes de fin d'année,…). Une priorité faible a alors peu d'impact.

Soumission des travaux
- Partitions et *QoS* -

- Lorsque vous soumettez un job, vous devez spécifier :
  → une *partition* qui définit le type de nœuds de calcul auxquels vous souhaitez accéder
  → une *QoS* (*Quality of Service*) qui calibre vos besoins en ressources (nombre de nœuds, temps d'exécution,…) et entre en jeu dans le calcul de la priorité de vos travaux
- Il existe deux *partitions* de calcul sur Jean Zay :
  → `cpu_p1` pour une exécution sur des nœuds scalaires (CPU)
  → `gpu_p1` pour une exécution sur des nœuds convergés (GPU)
- Chacune de ces partitions propose 3 *QoS*, listées ci-dessous :

Partition	QoS	Limite en temps	Limite en ressources		
			par job	par utilisateur	par QoS
CPU	`qos_cpu-t3` (défaut)	20h	512 nœuds		
	`qos_cpu-t4`	100h	1 nœud	32 nœuds	128 nœuds
	`qos_cpu-dev`	2h	128 nœuds	128 nœuds	1000 nœuds
GPU	`qos_gpu-t3` (défaut)	20h	96 nœuds		
	`qos_gpu-t4`	100h	1 nœud	8 nœuds	32 nœuds
	`qos_gpu-dev`	2h	4 nœuds	4 nœuds	64 nœuds

Soumission des travaux
- En batch -

- Un script *batch* contient :
  → un en-tête de configuration du job (nom du job, ressources demandées,…) sous forme d'une liste d'options Slurm précédées du mot-clef `#SBATCH`
  → l'ensemble des lignes de commande à exécuter (chargement des modules, lancement de l'exécutable,…) sur le nœud de calcul

- Le lancement de l'exécutable se fait via la commande `srun`, qui récupère le paramétrage *batch.*

- Les exemples de script qui suivent sont représentatifs mais non génériques. Un *catalogue complet d'exemples*, couvrant toutes les configurations de calcul possibles sur Jean Zay, est disponible sur la documentation en ligne 

Soumission des travaux
- En batch -

- Exemple de script *batch* pour une exécution sur la partition convergée GPU
  → exécution sur 1 GPU

```
#!/bin/bash
#SBATCH --job-name=TravailGPU              # nom du job
#SBATCH --partition=gpu_p1                 # partition
#SBATCH --qos=qos_gpu-dev                  # QoS
#SBATCH --output=TravailGPU%j.out          # fichier de sortie (%j retourne le job ID)
#SBATCH --error=TravailGPU%j.out           # fichier d'erreur (= fichier de sortie ici)
#SBATCH --time=00:10:00                    # temps maximal d'allocation des ressources
#SBATCH --ntasks=1                         # nombre de tâches (= nombre de GPU ici)
#SBATCH --gres=gpu:1                       # réservation d'un GPU
#SBATCH --cpus-per-task=10                 # réservation de 10 CPU par GPU (et mémoire associée)
#SBATCH --hint=nomultithread               # désactive l'hyperthreading

module purge                               # nettoyage des modules hérités par défaut
conda deactivate                           # désactivation des environnements hérités par défaut

module load pytorch-gpu/py3/1.4.0          # chargement des modules

set -x                                     # écho des commandes lancées
srun python script.py                      # exécution du code avec la commande Slurm srun
```

Soumission des travaux
- En batch -

- Exemple de script *batch* pour une exécution sur la partition convergée GPU
  → exécution sur 4 GPU (1 nœud)

```
#!/bin/bash
#SBATCH --job-name=TravailGPU           # nom du job
#SBATCH --partition=gpu_p1              # partition
#SBATCH --qos=qos_gpu-dev               # QoS
#SBATCH --output=TravailGPU%j.out       # fichier de sortie (%j retourne le job ID)
#SBATCH --error=TravailGPU%j.out        # fichier d'erreur (= fichier de sortie ici)
#SBATCH --time=00:10:00                 # temps maximal d'allocation des ressources
#SBATCH --ntasks=4                      # nombre de tâches (= nombre de GPU ici)
#SBATCH --ntasks-per-node=4             # nombre de tâches par nœud (travail sur 1 nœud ici)
#SBATCH --gres=gpu:4                    # réservation de 4 GPU
#SBATCH --cpus-per-task=10              # réservation de 10 CPU par GPU (et mémoire associée)
#SBATCH --hint=nomultithread            # désactive l'hyperthreading

module purge                            # nettoyage des modules hérités par défaut
conda deactivate                        # désactivation des environnements hérités par défaut

module load pytorch-gpu/py3/1.4.0       # chargement des modules

set -x                                  # écho des commandes lancées
srun python script.py                   # exécution du code avec la commande Slurm srun
```

Soumission des travaux
- En batch -

- Pour soumettre un script *batch* : `sbatch script.slurm`
- Pour suivre l'état de soumission de vos jobs (états possibles : R = running, PD = pending, CG = completing) : `squeue -u $USER`
- Pour afficher l'ensemble des paramètres d'un job soumis : `scontrol show job $JOBID`
- Pour annuler l'exécution d'un job: `scancel $JOBID`
- Pendant l'exécution d'un job, vous pouvez vous connecter au(x) nœud(s) de calcul réquisitionné(s) :

```
srun --jobid $JOBID --ntasks=1 --pty bash
```

Soumission des travaux
- En interactif -

- Vous pouvez aussi ouvrir un *shell bash* directement sur un nœud de calcul :

```
salloc --ntasks=1 --gres=gpu:1 --time=01:00:00 srun --pty bash
```

- La connexion est opérationnelle lorsque l'écho suivant apparaît :

```
salloc: Pending job allocation JOBID
salloc: job JOBID queued and waiting for resources
salloc: job JOBID has been allocated resources
salloc: Granted job allocation JOBID
```

- Cette fonctionnalité vous permet de faire des tests préparatoires en interactif pour configurer vos scripts batch, tout en ayant accès à un ou plusieurs GPU (les frontales de Jean Zay n'ont pas de GPU)

Consommation des heures de calcul

- Le décompte des heures de calcul h se fait de la façon suivante :
  $\longrightarrow h =$ nombre de GPU réservés $\times$ temps *elapsed*

- Un nœud est parfois réservé *en exclusivité* : si plus d'un nœud est demandé, ou si l'option Slurm `--exclusive` est activée. Dans ce cas, le décompte d'heures se fait de la façon suivante :
  $\longrightarrow h =$ nombre de nœuds réservés $\times$ 4 GPU $\times$ temps *elapsed*

- Pour suivre votre consommation : `idracct`

Partition de pré/post-traitement

- Une partition de pré/post-traitement est disponible sur Jean Zay
- Elle contient 4 nœuds convergés à large mémoire :
  → 4 processeurs Intel Skylake 6132 de 12 cœurs à 3,2 GHz,
  → 1 GPU Nvidia V100
  → 3 To de mémoire par nœud

- Pour y accéder, il faut spécifier l'option Slurm `--partition=prepost` dans les scripts batch ou dans les lignes de commande `salloc`

- Les heures utilisées pour le pré/post-traitement ne sont pas décomptées de votre allocation

- Vous pouvez vous connecter en ssh sur les nœuds de pré/post-traitement :

```
ssh login@jean-zay-pp.idris.fr
```

Pour aller plus loin

- Vous trouverez également dans la documentation en ligne comment :
  - lancer une exécution sur plusieurs GPU
  - gérer un compte multi-projets
  - utiliser des outils de pré- et post-traitement

- L'IDRIS dispense également diverses formations à destination des utilisateurs de calcul scientifique

Contacter le Support Utilisateurs

- Pour toute question ou demande sur l'accès à la machine, le déploiement de l'environnement logiciel, le débogage ou l'optimisation de votre code,… le Support Utilisateurs de l'IDRIS est joignable :

  du lundi au jeudi de 9h à 18h
  le vendredi de 9h à 17h30
  par mail à assist@idris.fr
  ou par téléphone au 01 69 35 85 55

- Pour toute demande relative à l'administration de votre compte de calcul (mot de passe, ouverture de compte, autorisation d'accès, enregistrement des adresses IP,…), contactez gestutil@idris.fr