Formation utilisateurs MCIA

A. Darrieutort
P. Gay

Présentation du MCIA

- MCIA pour Mésocentre de Calcul Intensif Aquitain.
- Le MCIA est une unité de services de l'Université de Bordeaux avec une équipe technique, un conseil scientifique, un comité des utilisateurs, ...

Utilisateurs

- Accès gratuit à tout chercheurs et personnels des instituts et établissements de recherche publics de Nouvelle Aquitaine
- 570+ utilisateurs
- Établissements et instituts
  - UB, UPPA, BxINP, Bx Sciences Agro, ULaRochelle,UPoitiers, ULimoges
  - CNRS, INSERM, INRIA, INRAE

Accéder aux services du Mésocentre

- Inscription: https://account.mcia.fr
- Documentation et assistance: https://redmine.mcia.fr/
- Cluster Curta: https://redmine.mcia.fr/projects/cluster-curta/wiki
- Stockage iRODS: https://redmine.mcia.fr/projects/irods-v2/wiki
- Cluster Pyrene (DOREMI U. Pau): https://redmine.mcia.fr/projects/cluster-pyrene/wiki
- Cluster CALI (DOREMI U. Limoges): https://redmine.mcia.fr/projects/cluster-cali3/wiki
- Cluster étudiants Poudlard (DOREMI MCIA): https://redmine.mcia.fr/projects/cluster-doremi-poudlard/wiki
- Site web: https://www.mcia.fr/

Contact

- Pour toutes demandes (d'assistance, d'installation de logiciels, etc), il faut privilégier une demande sur le Redmine dans le projet correspondant (Cluster Curta ou iRODS v2). https://redmine.mcia.fr/projects/irods-v2/issues/new
- https://redmine.mcia.fr/projects/cluster-curta/issues/new
- Pour des besoins en optimisation de code de calcul ou du stockage, n'hésitez pas à nous contacter à l'adresse contact@mcia.fr

Gestion du compte MCIA

- https://account.mcia.fr/documentation
- Demander un compte / réinitialiser son mot de passe : https://account.mcia.fr/login
  - Nécessite une adresse email profssionnelle
- Profil utilisateur : https://account.mcia.fr/profile
  - Ressources
  - Authentification : mots de passe, clefs SSH
  - Groupes
  - Invités
- Durée de vie du compte

Cluster Curta

- Le Cluster Curta est une infrastructure de calcul haute performance (HPC) de 12000 cœurs
  - Destiné à la simulation et à l'analyse de données scientifiques.
  - Hébergé dans la salle SHM1 de l'Université de Bordeaux

nœuds interactifs / frontaux

- Machines sur lesquelles l'utilisateur se connecte pour mettre au point et lancer ses travaux, manipuler ses données et ses résultats.
- curta.mcia.fr
- Architecture et environnement logiciel identique aux nœuds de calcul
- Destiné à la compilation et la préparation des calculs, mais **PAS AU CALCULS EUX-MEMES**!
- la durée des processus sur les nœuds frontaux est limitée pour éviter les surcharges

nœuds de calcul

- **compute (x364)** qui assurent l'exécution des tâches de calcul intensives
  - 2 processeurs hexadécacœurs (32 cœurs par nœud) Intel® Xeon® Gold SKL-6130 @ 2,1 GHz
  - 96 Go RAM
- **bigmem (x4)** qui ont plus de mémoire et plus de cœurs que les compute
  - 4 processeurs hexadécacœurs (64 cœurs par nœud) Intel® Xeon® Gold SKL-6130 @ 2,1 GHz
  - 3 To RAM
- **gpu (x4)** qui contiennent des cartes accélératrices GPU
  - 2 processeurs hexadécacœurs Intel® Xeon® Gold SKL-6130 @ 2,1 GHz
  - 192 Go RAM
  - 2 cartes graphiques NVidia® P100 de 16Go chacune
- **visu (x4)** qui permettent l'analyse et la visualisation de données complexes
  - 2 processeurs hexadécacœurs Intel® Xeon® Gold SKL-6130 @ 2,1 GHz
  - 192 Go RAM
  - 2 cartes graphiques NVidia® Quadro P4000 de 8Go chacune

Serveur d'administration

- Pas accessibles au public

Serveurs de stockage

- La solution de stockage distribué utilisée est le système de fichier performant Spectrum Scale (GPFS) d'une capacité de 512To dont le trafic est supporté par le réseau Omnipath.
- Le système de fichiers est découpé comme suit:
  - un espace /gpfs/home de 83To:
    - pour les données utilisateurs (/gpfs/home).
    - Un quota est défini pour chaque utilisateur au travers de deux limites (soft=128Go et hard=256Go).
    - des sauvegardes sont faites chaque jour avec une durée de retention de 4j.
    - Cet espace n'est pas adapté à un usage intensif. Privilégier le /tmp des nœuds.
  - Un espace /gpfs/softs pour l'installation de logiciels communs
  - Un espace /scratch de 426To pour les données des travaux
    - Les fichiers datant de plus de 90j sont nettoyés automatiquement
    - Pas de quota
    - Pas de sauvegarde

Réseau

- Réseau d'administration
- Réseau Ethernet 10Gb/s
- Intel® Omnipath® 100Gb/s (topologie FatTree avec un facteur de blocage de 2:1)

Accéder au cluster

- https://redmine.mcia.fr/projects/cluster-curta/wiki/Guide_de_l'utilisateur
- Linux / MacOS : Commande SSH
  - ssh user_name@curta.mcia.fr
- Windows : Utiliser PuTTY (voir la documentation)

Environnement Système et Logiciel

- Le cluster Curta fonctionne sous l'environnement Linux RockyLinux 8.6
- Les logiciels sont disponibles soit à partir du système, soit à partir de gestionnaire de paquets (environnement de modules et spack):
  - Editeurs de texte: emacs, vi/vim, nano
  - Debuggers: gdb
  - Profilers: gprof, Vtune
  - Large gamme de compilateurs: gcc, Intel/OneAPI, NVIDIA, Cuda, etc.
  - Librairies: MPI, BLAS, LAPACK, FFTW, NetCDF, etc.
  - Conteneur: Singularity/Apptainer
  - Optimisation énergétique: EnergyScopium
- Les logiciels sont principalement accessibles avec les commandes
  - module av <logiciel>
  - module load <logiciel>/<version>

Le Scheduler

- Afin d'accéder aux nœuds de calcul pour effectuer des simulations ou des analyses, l'utilisateur doit préparer un travail (un *job*), le plus souvent sous forme d'un script (bash, etc.)
- Le job doit être soumis au scheduler Slurm (commande "sbatch") pour se voir attribuer des ressources de calcul (cœurs, mémoire, durée)

Jobs

- Les jobs sont confinés dans un environnement *cgroup* pour empêcher les processus de consommer plus de ressources CPU que ce qui a été alloué pour le job.
- Les ressources nécessaires à chaque job doivent être précisées à la soumission
- valeurs par défaut:
  - 1 cœur
  - durée: 1h
  - mémoire vive: 1Go

Partitions

Partition (option « -p »)	Nœuds	MaxTime	Limites
<groupe>	n[001-315],bigmem[01-04]	5j	
longq	n[001-188],bigmem01	30j	(1)
gpu	gpu[01-04]	5j	
visu	visu[01-04]	12h	1 job / utilisateur
i2m-resources	n[316-336]	5j	Utilisateurs i2m
imb-resources	n[337-364]	5j	Utilisateurs imb
preemptible	n[001-364],gpu[01-04],visu[01-04]	2j	(2)

- (1) : sur nœuds compute: max jobs running=100 et par utilisateur=10; pour les nœuds *bigmem*: max jobs running=1 et par utilisateur=1
- (2) : le job peut-être préempté et remis en file d'attente s'il tourne sur des nœuds hors n[001-315]. Il est préférable d'avoir un mécanisme de protection reprise lorsqu'on soumet sur la partition *preemptible*.

Limites Générales

- Une limite du temps réservé pour tous les processeurs d'un job est définie à 34560 heures (soit 5j sur 288 cœurs ou 30j sur 48 cœurs).
- Une limite sur l'ensemble des jobs alloués en même temps à un utilisateur est définie à 150000 heures.
- La priorité entre les jobs prend en compte plusieurs facteurs:
  - Tous les utilisateurs sont initialement égaux par rapport au fairshare, seule leur consommation va avoir une influence.
  - Les jobs avec beaucoup de cœurs auront une meilleure priorité.
  - Les jobs de faible priorité ou demandant peu de ressources passeront rapidement grâce au backfilling.

Utilisation de Slurm

- Les principales commandes :
  - **sbatch** pour soumettre un job,
  - **squeue** pour afficher la liste des jobs,
  - **scancel** pour supprimer un job,
  - **scontrol** et **sacct** pour avoir des informations sur les jobs.
- D'autres commandes utiles (accessibles avec le module *slurm/wrappers*):
  - **seff** pour avoir une estimation des ressources utilisées par un job
  - **showuserlimits** pour afficher l'usage et les limites de ressource slurm pour un utilisateur
  - **snodes** afficher l'état actuel des nœuds
- Pour soumettre un job, le script batch devra débuter par des directives *#SBATCH* et définir les ressources à allouer.

Slurm : remarques

- Les nœuds *compute* possèdent 96Go de RAM mais seulement 90Go sont accessibles. Par défaut, 1Go de mémoire sera allouée par cœur réservé donc si le job a besoin de plus, il faudra utiliser les options *--mem-per-cpu* ou *--mem*
- Dans certains cas, pour optimiser les performances d'un job, vous pouvez également préciser de prendre en compte la topologie du réseau Omnipath.
  - Exemple: Pour spécifier à slurm d'allouer des nœuds se trouvant sur le même switch

    #SBATCH --switches=1
- La connexion SSH sur un nœud de calcul est interdite, par contre, vous pouvez :
  - soumettre un job interactif (*srun --pty /bin/bash -i*) pour des besoins en développement, compilation ou débugage.
  - vous connectez à l'interieur d'un job en cours d'exécution (*srun --pty --jobid <jobid> /bin/bash -i*)
- Pour lancer un job sur un type de nœuds particulier (valeurs possibles: compute, bigmem, visu, gpu, imb, i2m)

  #SBATCH --constraint=compute
- Utiliser la partition *preemptible* pour avoir accès à des nœuds d'autres partitions qui sont libres:

  #SBATCH -p preemptible

Slurm : messages

- A la soumission d'un job, si les ressources demandées ne sont pas conformes à notre configuration, une erreur est retournée et le job ne sera pas soumis.
- Commentaires visible sur un job avec la commande **squeue**:
  - **AssocMaxCpuMinutesPerJobLimit** → limite sur l'ensemble des jobs d'un utilisateur atteinte (150000 heures)
  - **AssocGrpCPURunMinutesLimit** → limite du *_nombre de coeur * temps d'exécution_* atteinte (34560 heures)
  - **QOSGrpJobsLimit** → limite du nombre maximum de jobs dans une *qos* atteinte (ex: 100 jobs running sur la partition *longq*)
  - **QOSMaxJobsPerUserLimit** → limite du nombre maximum de jobs par utilisateur dans une qos atteinte (ex: 1 job running sur le bigmem dans la partition longq)
  - **Nodes required for job are DOWN, DRAINED or reserved for jobs in higher priority partitions** → message standard qui signifie simplement que le job est en file d'attente
- Messages d'erreurs dans le log:
  - **Job ... on ... cancelled at ... due to time limit** → il faut augmenter le walltime.
  - **oom-kill** ou **out-of-memory** → il faut augmenter la mémoire.

OpenOnDemand

- Service pour accéder au cluster depuis un Navigateur Web https://curta3.mcia.fr
- Consulter et manipuler les fichiers sur le cluster (home et scratch)
- Accéder à un terminal sur la frontale dans le navigateur
- Consulter et lancer des jobs
- Lancer des jobs spéciaux
  - Notebooks Jupyter
  - Sessions graphiques déportées

Stockage iRODS

- Système de gestion des données basé sur le logiciel open source https://www.irods.org
- Destiné au stockage pérenne et long terme de grands volumes de données scientifiques froides ou tièdes
- Permet d'implémenter des politiques de gestion des données adaptées aux besoins des utilisateurs
- Une collaboration MCIA(UB) / UPPA / U. Poitiers

Caractéristiques

- Capacité utile totale : 500To (1+ Po brut)
- Réplication x2 : chaque fichier déposé existe en 2 exemplaires réparti entre les 3 sites partenaires
- Transferts réseau chiffrés, le stockage sur les serveurs ne l’est pas
- Clients :
  - Ligne de commande sur Linux, MacOS et Windows (WSL?)
  - Graphiques : brocoli, autres ?

Configuration

- Mot de passe
  - Différent du mot de passe principal MCIA
  - à créer ici : https://account.mcia.fr/profile#authentication
- Clients :
  - https://redmine.mcia.fr/projects/irods-v2/wiki/Utiliser_la_solution
  - CLI sur Curta : « *module load irods/mcia2* »
  - CLI Linux :
    - Installation : https://redmine.mcia.fr/projects/irods-v2/wiki/Installation_ICommands
    - « *iinit* »
  - Brocoli : « *pip3 install brocoli [--user]* »

Commandes iRODS

- Commande de type ftp :
  - *iput* -
  - *iget* -
- Commandes de type shell
  - *ils* -
  - *ipwd* -
  - *icd* -
  - *imkdir* -

Statistiques d’utilisation MCIA

- Portail https://pandora.mcia.fr/grafana/
- Permet d’accéder à vos statistiques d’utilisation personnelles
  - Curta : nombre d’heures
  - IRODS : volume des données nombre de répliques

Questions