

Calcul COMET-France (tracking)

Réunion du 30 octobre 2024 au CC-IN2P3

Présent(e)s : Sebastien Gadrat, Gino Marchetti, Sybille Voisin, Patrice Lebrun, Wilfrid da Silva, Jean-Claude Angélique.

Excusé Thomas Kachelhoffer.

9h00 Accueil.

9H45 Nouvelles générales : Sébastien Gadrat

Migration sur RHEL9. Sur Jupyter par défaut Notebook server.

CPU sous OS9

GPU sous OS8 (car K80)

Demandes et gestions des ressources

Aides aux utilisateurs (assurer une bonne utilisation des ressources)

Support permanent au CCIN2P3 & Support software ponctuel au CCIN2P3

Authentification AIM nouveau portail.

Géré plus particulièrement par Patricia Warin-Charpentier (LPNHE)

Administration par EDUGAIN: ne doit être utiliser que par les personnes appartenant à la fédération France.

- ARM :

** Patrice est intéressé (Sébastien va lui envoyer les infos)*

** Mise en place d'une partition Slurm basée sur l'architecture ARM, avec deux worker nodes pour le moment (pas de cca, pas de worker interactif).*

** Soumission à partir d'une cca "classique" :*

```
$ sbatch --partition=htc_arm -t 1:00:00 --mem=512 -n 1 job.sh
```

Computing Comet interface utilisateur avec le CC :

Départ de Patrice en 2025. (snif snif)

Sébastien accepte de prendre le relais de Patrice.

Disponibilités et évolution GPU

- Evolution des GPU au CC

---- *Plateforme Jupyter NoteBook*

CPU : deux AMD 7243F - 24 cores - 4.05Ghz

RAM CPU : 768G

GPU : quatre L40S

Mémoire par GPU : 48G

Stockage : 3.2To

---- Plateforme de calcul sur soumission de job

CPU : deux AMD 7243F - 24 cores - 4.05Ghz

RAM CPU : 1536G

GPU : quatre H100 NvLink

Mémoire par GPU : 94G

Stockage : 6.4To

Au final, on devrait avoir au CC:

- 12 GPU H100 (dans Slurm)

- 30 GPU L40S (dans Jupyter)

(Device vs GPU ?)

Le bon de commande a été déposé en octobre. Livraison en fonction des disponibilités et espéré pour le premier trimestre 2025.

Rmq : sur Jupyter : actuellement, en moyenne les GPU sont utilisés à 50%, sauf ponctuellement à 100% pour formation machine learning.

Pour Info : pour COMET : indispensable pour utilisation de Julia pour tracking dans software COMET Via icedust C++).

Julia dans ICEDUST :

11H00 Présentation du travail de Patrice fait avec Julia sur le tracking et l'intégration de Julia dans ICEDUST (C++).

Il a été proposé à Patrice de faire un webinaire au CC pour présenter son travail et l'utilisation de Julia.

11h40 Gitlab CI/CD – Docker et GPU :

Présentation de Sébastien Gadrat.

Rmq : Docker empile les couches dont un des intérêts est de ne pas reprendre des couches qui existent déjà.

Python est installé via Conda (défaut la licence du logiciel deviendra payante à terme!)

Packages pour Julia et R sont les mêmes.

Installation de git

Installation NVidia toolkit

Rmq il existe un docker COMET

L'idée est de construire un docker pool GPU (Julia) qui sera mis dans le docker de COMET

Exple docker avec et sans GPU

Extension de jupyter pour utiliser plotyJP (demande Patrice ? effectuée par Sébastien)

Bilan avoir un point de départ pour commencer

Docker and docker non autorisé sur gitlab

Voir besoin pour les demandes en 2025 faire un docker file sur sa propre machine

Créer un docker and docker

Créer docker file, singularity file (l'image au final doit faire autour de 10Go)

Création d'un container

En conclusion il est décidé que c'est singularity qui sera utilisé avec les GPU du CC.

14h00 reprise de la réunion

JAX for machine learning

Présentation de Sybille Voisin sur la mise en place de JAX sur GPU pour la mesure de luminosité dans le traitement des images télescope ZTF

Exple : Pour le traitement de 100 étoiles on constate un gain de facteur 20 (GPU/ CPU).

Rmq : Il pourrait être intéressant de faire une comparaison de JAX avec flux.jl

15h10 Ressources GPU-Organisation future

Estimation de Patrice : 843 jours pour un GPU en V100 pour analyser les 176 fichiers de données de Yao ZHANG (IHEP). Ceci correspond à 1 milliards de hits environ (bdf inclus).

Soit une durée de traitement de 280 jours environ pour 4 GPU en V100 sur une année.

(durée totale de traitement : ~ 3 ans)

A noter qu' il y a approximativement un facteur 8 à 10 entre V100 et H100 en temps de traitement. (1 an V100 ~ 1,5 mois en H100).

Rmq : pour info le prix d'un tiroir de 4 GPU H100 sans le châssis (100 keuros).

Rmq : Ferme de calcul: sur 200 millions demandé initialement on a consommé 340 millions de HS06.h (n' a pas été problématique).

Question utilisation Gitlab

COMET n'utilise pas Gitlab de façon optimale ce qui peut pourrait poser des problèmes à plus ou moins long terme sur l'espace disque.

Suivis des FORK : 90 Forks recensés. Nombre important (~800Go sur Gitlab)

Mesure à prendre : Suppression des Forks inutilisés depuis 3 ans

16H30 Fin de la réunion.