Inserm

La science pour la santé
From science to health

Présentation générale du HPC Inserm

Inserm - Cloud self-service - Offre HPC

Objectifs de cette présentation

Cette présentation vise à vous présenter la plateforme HPC de l'Inserm. Nous aborderons ses particularités, son architecture et ses ressources.

L'objectif est également de vous présenter les tutoriels mis à votre disposition sur :

- L'exploitation des jeux de données externes hébergés sur un bucket S3
- La gestion et l'inférence de modèles LLM avec le serveur Ollama
- La gestion et l'inférence de modèles LLM avec vLLM
- Le traitement de calculs génomiques

Table des matières

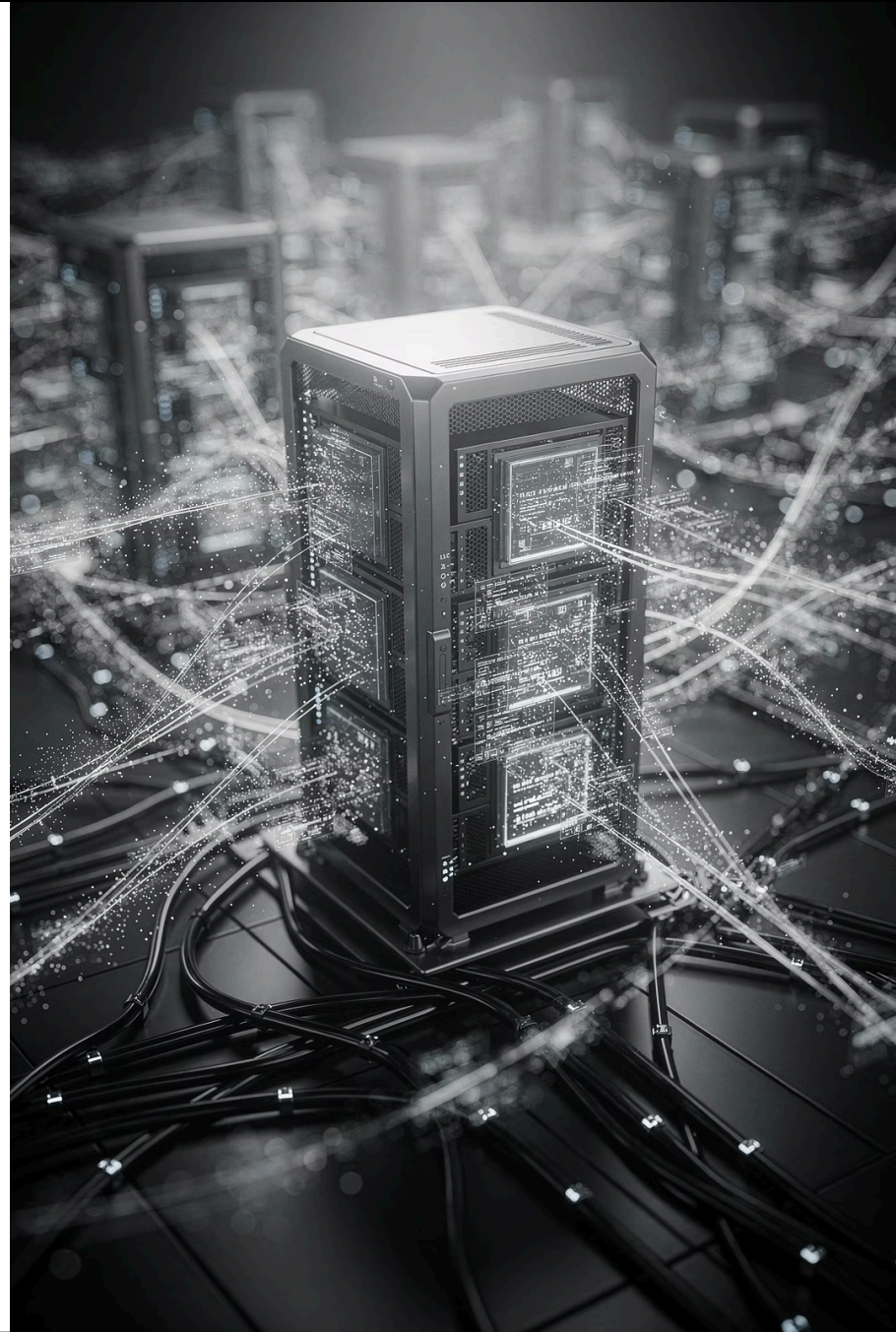
1. HPC

1.1. Qu'est-ce qu'un HPC ?

Un HPC (Calcul Haute Performance) est un **ensemble de plusieurs ordinateurs, appelés nœuds, utilisés pour réaliser des calculs complexes très rapidement** grâce à la parallélisation et la distribution des calculs sur chaque nœud.

Exemples de cas d'usage des HPC :

- Recherches scientifiques : climat, génome, médecine, physique nucléaire...
- Conception de véhicules ou d'avions : aérodynamique, analyse de crash-tests
- Intelligence artificielle et modèles de langage (comme ChatGPT)
- Simulation 3D, météo, volcanologie, sismologie



1.2 Qu'est-ce qu'un job ?

Sur un HPC, le travail ne se fait pas comme sur un PC classique. On exécute des **jobs**, c'est-à-dire des programmes à exécuter.

Le système HPC décide où et quand exécuter un job en fonction des ressources demandées et réservées par l'utilisateur, via un mécanisme de files d'attente.

Un HPC est composé de :

Applications scientifiques

JupyterHub, RStudio, Nvidia NGC containers, BioContainers, etc.

Portail (OpenOnDemand)

Facilite l'accès aux applications scientifiques, déployées via Apptainer sur le nœud choisi par l'ordonnanceur.

Ordonnanceur de jobs (Slurm)

Distribue les calculs sur les nœuds disponibles selon les ressources demandées.

2. Particularités de la plateforme HPC Inserm

2.1 Un environnement HDS / air-gapped

La plateforme HPC de l'Inserm répond à des **exigences de calcul scientifique et à la sensibilité des données traitées**. Dans une **infrastructure HDS** (Hébergement des Données de Santé), il est essentiel de mettre en place des mécanismes d'isolement entre les nœuds pour garantir un usage sécurisé.

La plateforme est **air-gapped**, c'est-à-dire qu'elle n'a pas accès à Internet. Les ressources sont mises à disposition des utilisateurs ou transférées depuis des bulles sécurisées (VDI accessible via **portail.hds.inserm.fr**).

2.2 GPU ou CPU : quelles différences ?

La plateforme HPC de l'Inserm propose des ressources de calcul CPU et GPU. Chacune est optimisée pour des typologies de travail distinctes et des besoins scientifiques spécifiques.

Catégorie	CPU	GPU
Usages typiques Inserm	• Analyses génomiques classiques • Bio-informatique, statistiques, scripts R / Python • Pré-traitement de données, pipelines séquentiels	• Apprentissage automatique (IA, deep learning) • Simulation scientifique (calcul matriciel, modélisation) • Inférence de modèles LLM (Large Language Models) • Traitement d'images ou vidéos
Avantages	• Polyvalence élevée pour les traitements scientifiques standards • Facilité d'usage via un environnement VM de staging disponible • Approche progressive : développement, test, puis passage à l'échelle	• Accélération majeure des temps de calcul • Traitement massif en parallèle, indispensable pour l'IA
Inconvénients	• Temps de calcul plus longs pour les traitements intensifs • Scalabilité limitée pour l'IA et les simulations lourdes	• Potentielle adaptation nécessaire des pipelines de calculs existants • Surdimensionné pour des traitements simples ou exploratoires • Pas de développement en VM de staging, nécessité de réservation des partitions de calcul

2.3 Un environnement adapté à chaque utilisateur

La plateforme HPC s'adresse à des profils variés afin de fournir une expérience utilisateur adaptée à vos besoins et habitudes :

Interface graphique

JupyterHub, VS Code, bureau graphique

Ligne de commande

SSH, Slurm, Apptainer

3. Architecture et ressources

3.1 Architecture de la plateforme HPC

La plateforme utilise un ordonnanceur de jobs, qui facilite l'accès aux ressources via des partitions.

Pour le HPC Inserm, **5 partitions** ont été définies. Elles servent à organiser et contrôler l'accès aux ressources de calcul, y compris les GPU :

Partition A40-48

- 2 nœuds, chacun avec 1 GPU A40-48Gb et 16 vCPU.
- Idéal pour l'inférence de modèles LLM de petite taille (≤ 8b).

Partition 2xH100-96

- 1 nœud avec 2 GPU H100 NVL-96Gb et 32 vCPU.
- Idéal pour l'inférence de modèles LLM de grande taille (> 70b).

Partition A100-10

- 14 nœuds, chacun avec 1 GPU A100-10Gb et 6 vCPU.
- Adapté aux calculs scientifiques nécessitant plusieurs nœuds GPU simultanément.

Partition CPU-16

- 4 nœuds, chacun avec 16 vCPU.
- Adapté aux calculs scientifiques sans GPU.

Partition CPU-4

- 4 nœuds, chacun avec 4 vCPU.
- Adapté au prototypage et à l'élaboration de code sans GPU.

Un nœud réservé ne peut être utilisé que par un seul utilisateur à la fois. D'autres partitions sont en cours d'implémentation.

Chaque réservation est limitée à **30 heures maximum.**

⚠️ **Point crucial : au-delà de cette durée, vos résultats de calcul seront perdus !** Assurez-vous que vos calculs ne dépassent pas 30 heures. Vous pouvez découper vos calculs et réserver des partitions à la suite.

3.2 Accès au stockage

L'utilisateur dispose d'un accès unifié à ses données, quel que soit le nœud de calcul, grâce à un système de fichiers partagé hébergé sur les baies NetApp de l'Inserm.

En plus du répertoire personnel `/home`, l'utilisateur a accès à deux espaces partagés :

`/packages`

Ressources accessibles en lecture seule

`/shared`

Contenus scientifiques partagés (modèles LLM, scripts GPU, calculs génomiques)

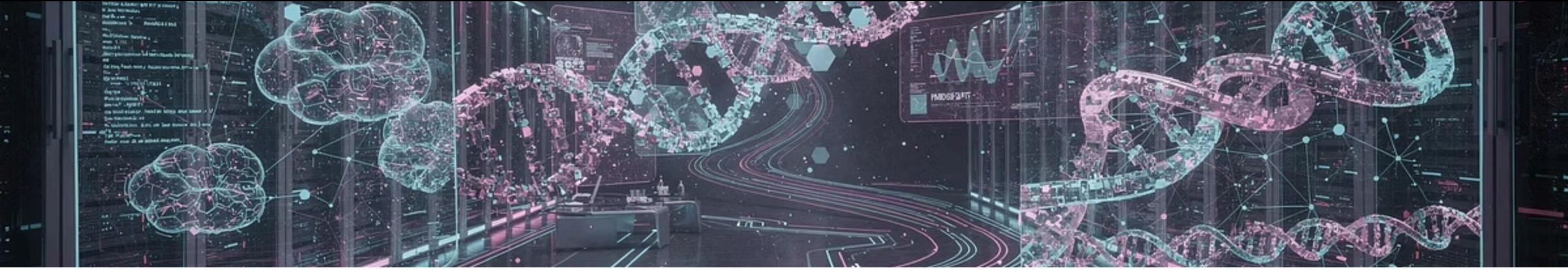
Un bucket S3 est également disponible pour importer/exporter des données, soumis à analyse antivirus.

Pour un guide pas à pas sur l'accès au bucket S3 depuis les nœuds HPC, découvrez notre tutoriel : **"Tuto HPC bucket S3"**.

3.3 Accès aux bibliothèques scientifiques

Chaque utilisateur est autonome pour gérer ses dépendances Python ou R dans son environnement de travail. L'usage de **virtualenv** est recommandé.

> Pour un guide pas à pas sur l'installation d'un **virtuelenv** via un serveur Nexus authentifié : découvrez notre tutoriel : **"Tuto HPC vLLM" (chapitre 2).**



4. Exemples de cas d'usage traités sur la plateforme HPC

Nous avons sélectionné 3 principaux cas d'usage de la plateforme HPC :

La gestion et l'inférence de modèles de langage (LLM) avec **Ollama**

La gestion et l'inférence de modèles de langage (LLM) avec **vLLM**

Le traitement de **calculs génomiques**

4.1 Inférence de modèles LLM

Ollama ou vLLM : quelles différences ?

Les LLM (Large Language Models) sont des **modèles d'intelligence artificielle très volumineux qui nécessitent beaucoup de puissance de calcul et de mémoire**, en particulier lors de l'inférence ou de l'adaptation de modèles. Le HPC permet d'**exploiter des GPU performants et de répartir les calculs sur plusieurs ressources**, ce qui réduit fortement les temps d'exécution.

Parmi les possibilités de mise en œuvre de LLM, 2 solutions sont ici proposées : Ollama et vLLM.

Catégorie	Ollama	vLLM
Modèles	Modèles LLM en conteneurs OCI	Modèles au format Transformers, disponibles sur Hugging Face *(dépôt de plus de 200 000 modèles pré-entraînés : LLMs, modèles de vision, modèles de voix, etc)*
Cible	Simplicité, favorise le prototypage rapide de solution à base de LLM	Haute performance, scalable, adapté à la production et aux clusters HPC, tire pleinement parti des GPU et optimise la vitesse et l'utilisation de la mémoire
Utilisation GPU	Supportée mais limitée	Optimisée pour multi-GPU / HPC
API	REST intégrée	API compatible OpenAI
Cas d'usage	Démo, prototypage, usage personnel	Déploiement à grande échelle, HPC, recherche

Tutoriel Ollama

Pour un guide pas à pas sur Ollama : découvrez notre tutoriel : **"Tuto HPC Ollama"**

Ce tutoriel vous guide dans la **mise en place et l'utilisation d'un serveur Ollama** en s'appuyant sur :

- Apptainer
- un environnement graphique Linux

Tutoriel Python vLLM via HuggingFace

Pour un guide pas à pas sur vLLM : découvrez notre tutoriel : **"Tuto HPC vLLM"**

Ce tutoriel vous guide dans la configuration de **l'accès au serveur nexus et au virtualenv**, dans l'**installation de bibliothèques Python**, puis l'**exécution d'un programme Python** à partir de trois environnements différents :

- la console Linux
- un environnement graphique Linux
- JupyterLab

4.2 Calculs génomiques

Le calcul génomique traite d'énormes quantités de données issues du séquençage d'ADN ou d'ARN. Ces analyses demandent beaucoup de puissance de calcul et de mémoire, par exemple pour :

- Aligner des séquences
- Détecter des variants génomiques
- Assembler des génomes

Le HPC permet de répartir ces calculs sur plusieurs nœuds et CPU/ GPU, ce qui accélère fortement les analyses. De plus, grâce aux bio-conteneurs, les pipelines sont reproductibles et sécurisés, ce qui est essentiel pour les données sensibles.

Tutoriel génomique

Pour un guide pas à pas sur le traitement de calculs génomiques : découvrez notre tutoriel : **"Tuto HPC génomique"**

Ce tutoriel vous guide dans **la mise en place d'un pipeline de calcul** à travers l'**exemple d'extraction de variants génomiques** en s'appuyant sur **:**

- Apptainer
- Biocontainer
- L'orchestrateur Slurm