

# GPGPU

Algorithmes parallèles STL avec C++ 20

---

Michaël El Kharroubi

27.02.2024

HPC 2024 - HEPIA

# Introduction

---

Le terme “*general-purpose computing on graphics processing units*” (GPGPU) désigne l’ensemble des programmes qui ne sont pas directement liés au calcul graphique et qu’on réalise sur GPU.

Par exemple :

- Simulation de phénomènes physiques
- IA et Machine learning
- Minage de crypto-monnaie

# Pourquoi a-t-on besoin de GPU?

Un CPU est conçu pour suivre une série d'instructions séquentiellement (ce qui n'est plus le cas, dans une moindre mesure, avec les instructions vectorielles ex: AVX).

## **Adapté :**

- Ordonnancer des tâches
- Accéder aux périphériques
  - Réseau
  - Disque
  - Clavier/souris

## **Moins adapté :**

- Calcul vectoriel
- Calcul matriciel

Les GPU apportent une réponse à ces problèmes

Un GPU est un type de processeur conçu pour effectuer massivement des calculs en parallèle

# CPU vs GPU

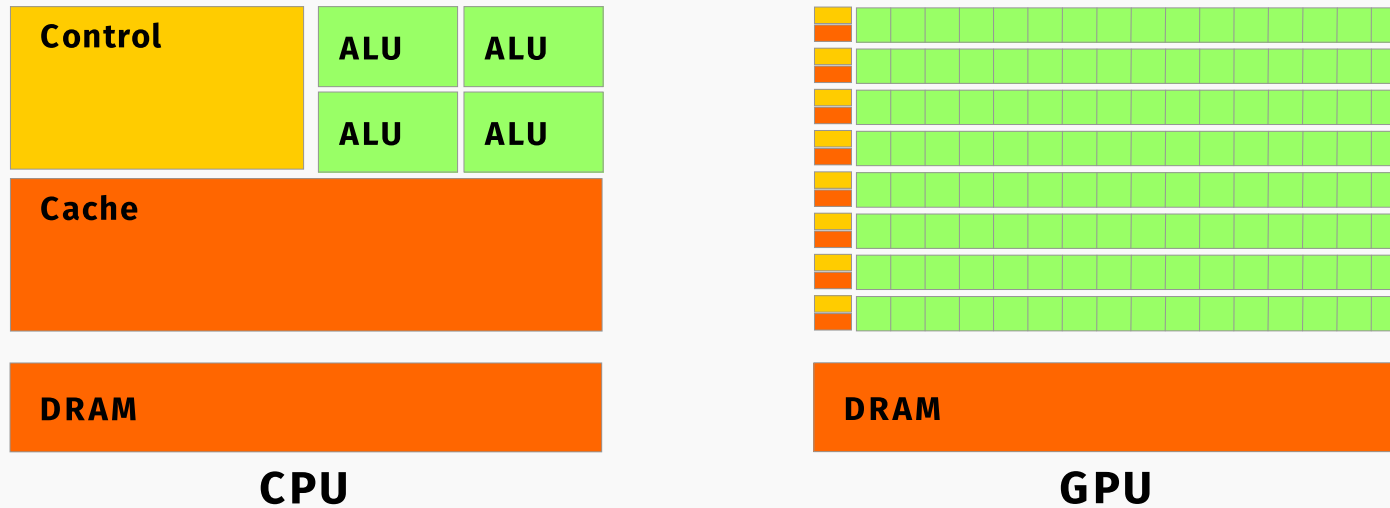


Figure 1: CPU vs GPU Architecture - Source : NVIDIA CUDA Programming Guide version 3.0 - 2010

Voici une petite vidéo du partenariat entre Myth Busters et Nvidia, pour visualiser la différence <https://youtu.be/-P28LKWTzrI>

# La programmation sur GPU

---



CUDA est le kit de développement de NVIDIA pour faire du GPGPU. C'est le kit le plus populaire de développement GPGPU.

CUDA permet d'écrire du code dans un dialecte C/C++ et de le compiler pour qu'il soit exécuté sur des GPU de Nvidia.

Dans la concurrence, on peut également citer OpenCL (multi-plateforme), ROCm pour AMD ou OneAPI pour Intel.

# Déroulement classique d'un programme GPGPU

1. Le CPU envoie les données à traiter au GPU

# Déroulement classique d'un programme GPGPU

1. Le CPU envoie les données à traiter au GPU
2. Le GPU applique un kernel sur les données

# Déroulement classique d'un programme GPGPU

1. Le CPU envoie les données à traiter au GPU
2. Le GPU applique un kernel sur les données
3. Le CPU récupère le résultat

Exemple de code CUDA et notions à  
retenir

---

# Notions essentielles à retenir

- Le CPU et le GPU sont deux périphériques distincts
- Le GPU est prévu pour effectuer massivement des calculs
- La mémoire n'est pas partagée, l'espace d'adressage n'est donc pas commun
- On parle de **Host** pour le CPU et de **Device** pour le GPU

Questions ?

---