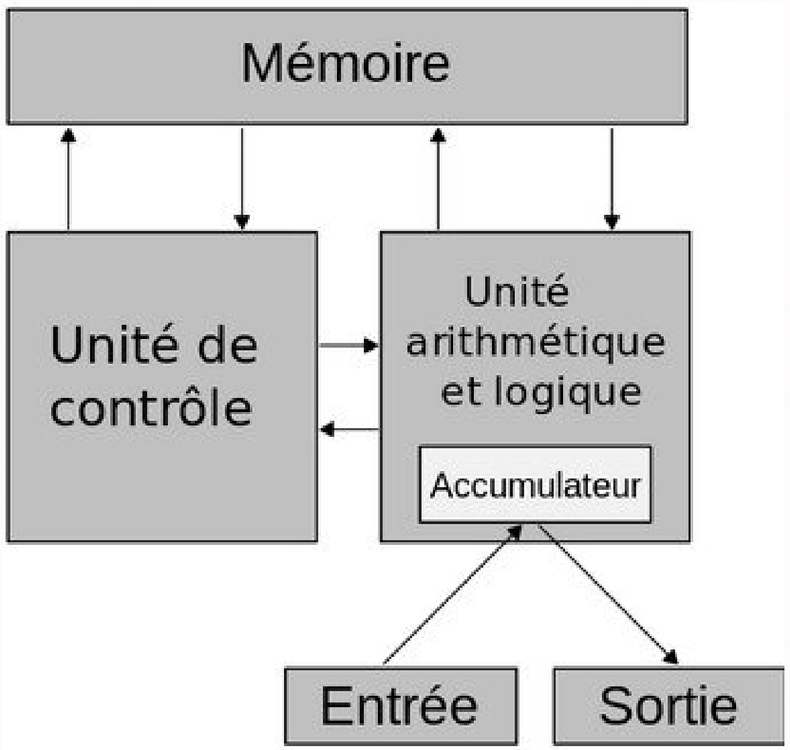
HPDA 2025

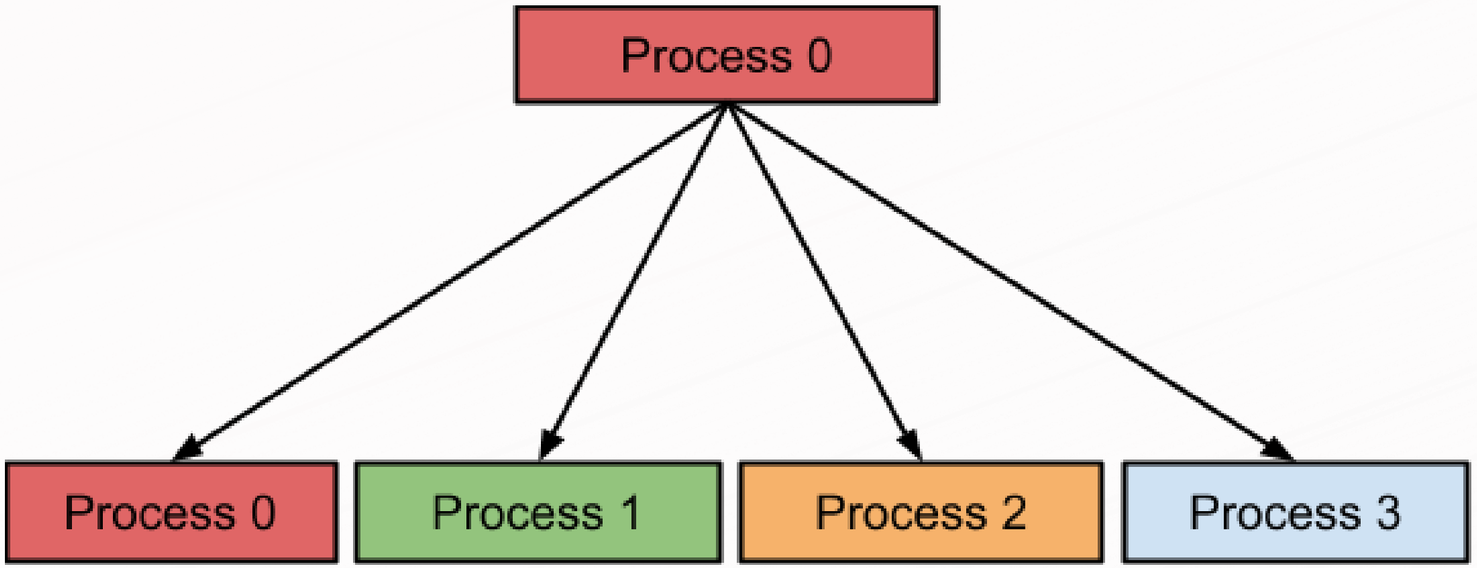
INTRODUCTION PROGRAMMATION HÉTÉROGÈNE

Jordan Rey-Jouanchicot - jordan.reyjouanchicot@orange.com
Thierry Garcia - CYU

Recap'



Architecture des ordinateurs
Von Neumann



Programmation parallèle: MPI

Jordan Rey-Jouanchicot

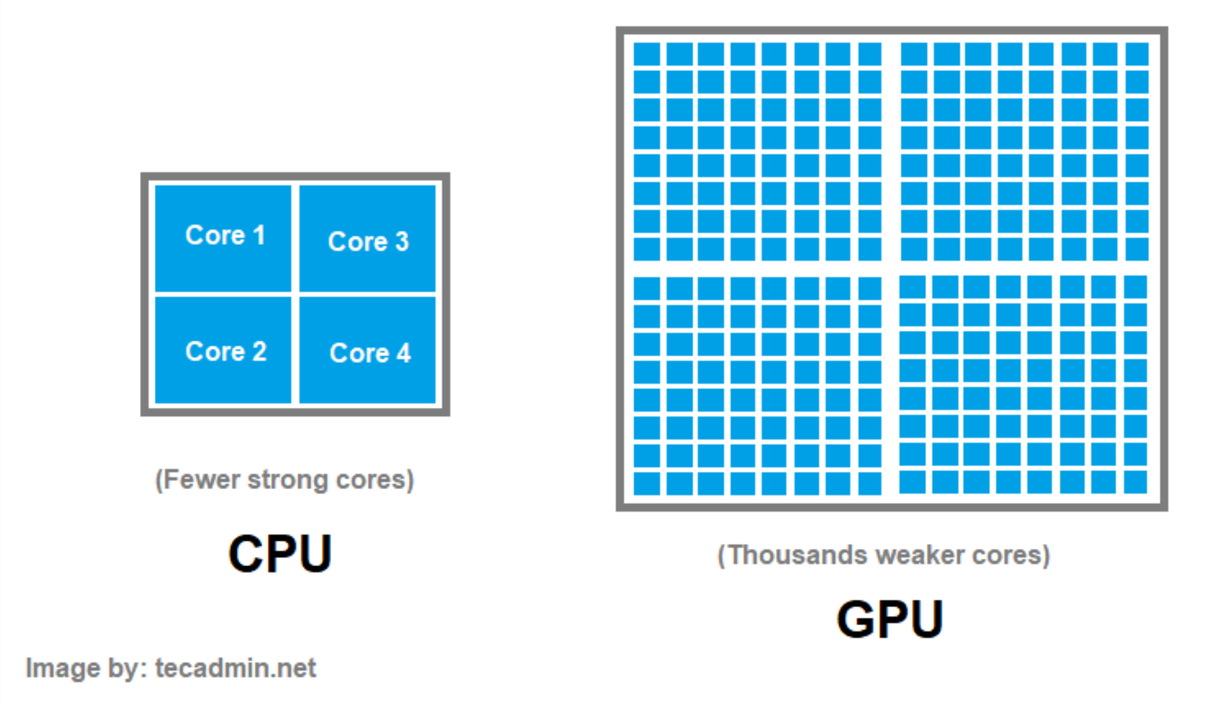
Apparition des GPU

Un Graphics Processing Unit (GPU) est un processeur conçu à l'origine pour le rendu graphique 2D/3D

Utilisé pour des tâches de calcul massivement parallèles, grâce à l'architecture elle même massivement parallèle.

Comparaison avec le CPU :

- **CPU :** Optimisé pour la flexibilité et les calculs séquentiels.
- **GPU :** Optimisé pour le parallélisme massif et les calculs répétitifs.



Jordan Rey-Jouanchicot

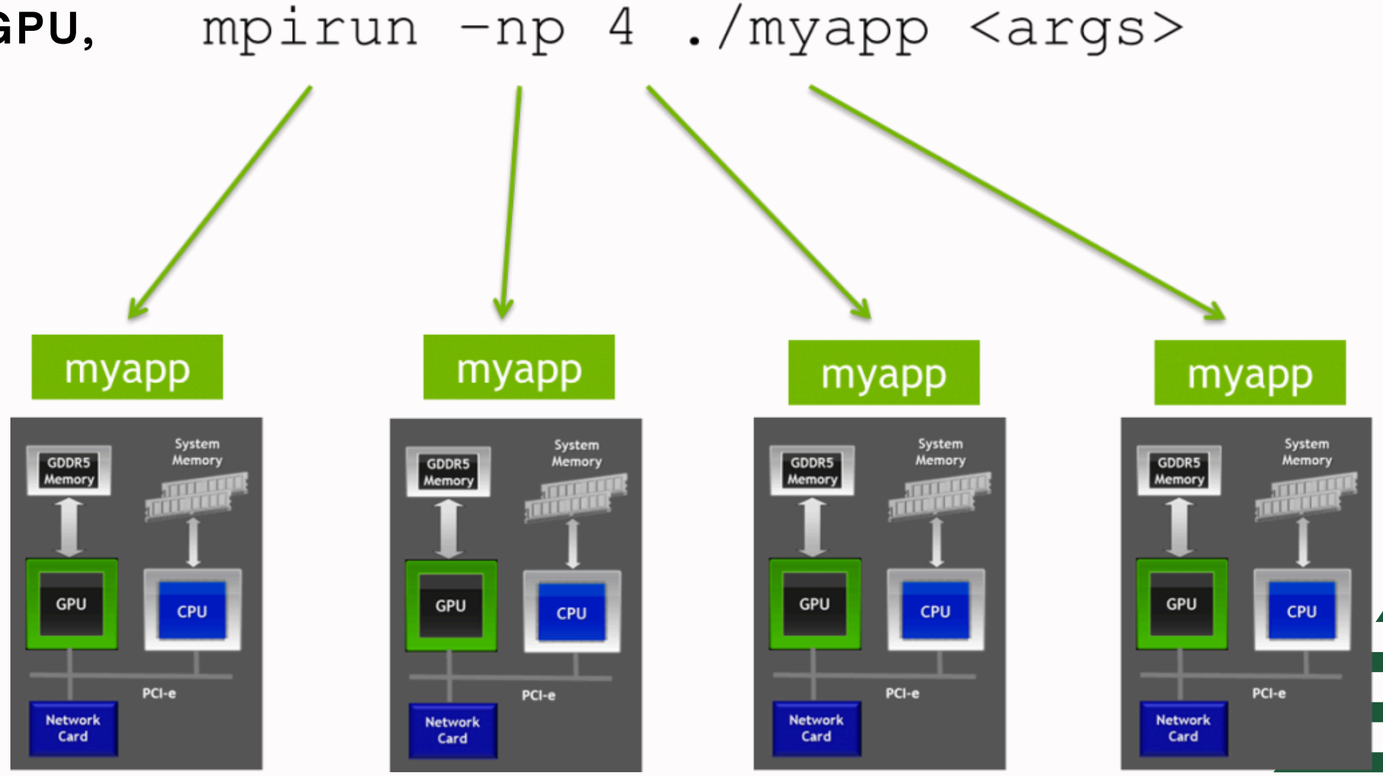
Programmation GPU

Qu’est ce que c’est ?

La programmation hétérogène consiste à écrire des programmes exploitant différents types de matériel ensemble comme les CPU et les GPU, en tirant parti de leurs forces respectives.

Des objectifs différents

- Le CPU excelle dans les tâches séquentielles et le contrôle du système.
- Le GPU est conçu pour exécuter des milliers de tâches en parallèle avec une grande efficacité.
- En répartissant les tâches sur CPU et GPU intelligemment, on optimise les performances d’un programme.



Programmation GPU: Des étapes

- Années 1980-1990 : GPU à fonction fixe (uniquement pour l'affichage).
- Fin des années 1990 : Introduction des shaders programmable/ Dzcouverte de méthodes utilisant OpenGL pour du calcul.
- 2006 : Nvidia introduit CUDA, permettant l'utilisation du GPU pour le calcul général (GPGPU).
- Aujourd'hui : GPUs optimisés pour l'IA, le Big Data et la simulation scientifique.

In 2010, the U.S. Air Force built a supercomputer using 1,760 PlayStation 3 consoles and named it the Condor Cluster.

Architecture CUDA

Une architecture unifiée

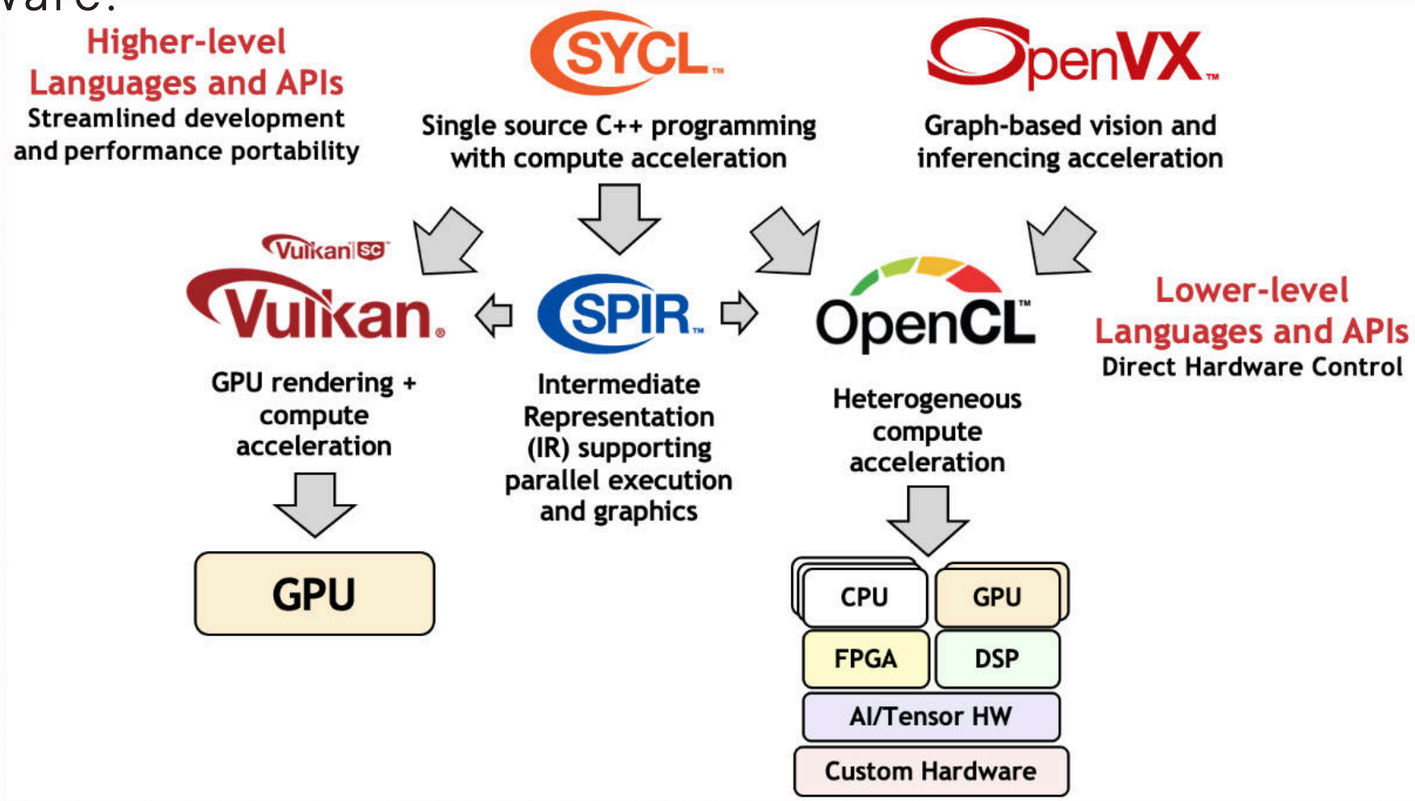
CUDA (Compute Unified Device Architecture), annoncé par Nvidia en 2006.
Permet de programmer un GPU de façon relativement simple.
Exploite la puissance du parallélisme GPU, tout en proposant une abstraction du hardware.

Des alternatives ?

OpenCL
AMD ROCm
SYCL (C++ standard).
OpenACC
CUDNN - CUBLAS

Perspectives:

Vulkan ?



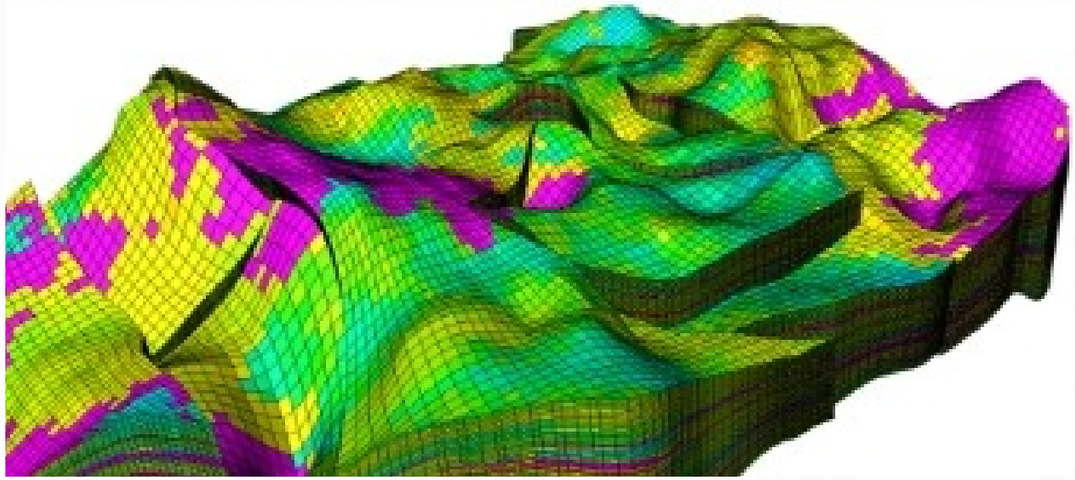
https://github.com/linuxelf001/OpenCL-Vs-Cuda/tree/master

Les cas d’usages actuels

Et l’impact

Jeux

Calcul scientifiques



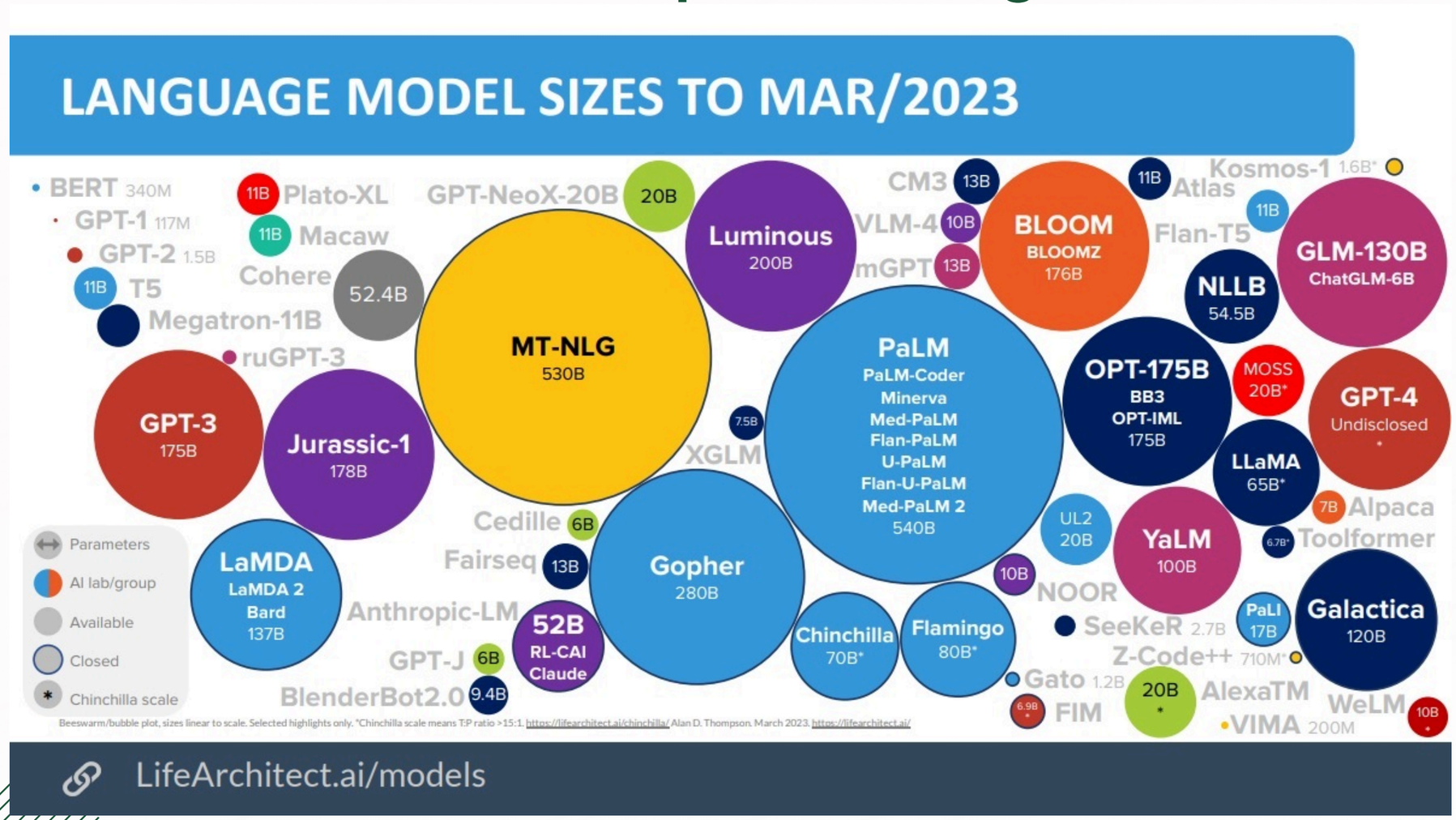
Le minage

Jordan Rey-Jouanchicot

Les cas d’usages actuels

Et l’impact

IA: Deep Learning



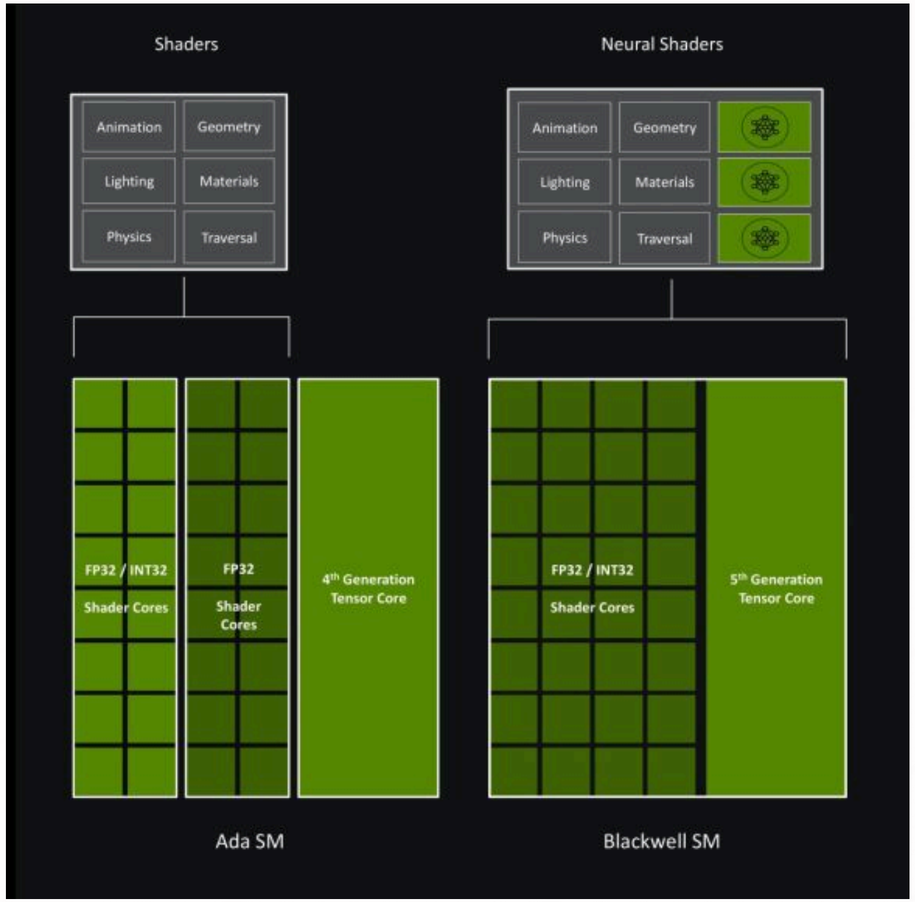
Jordan Rey-Jouanchicot

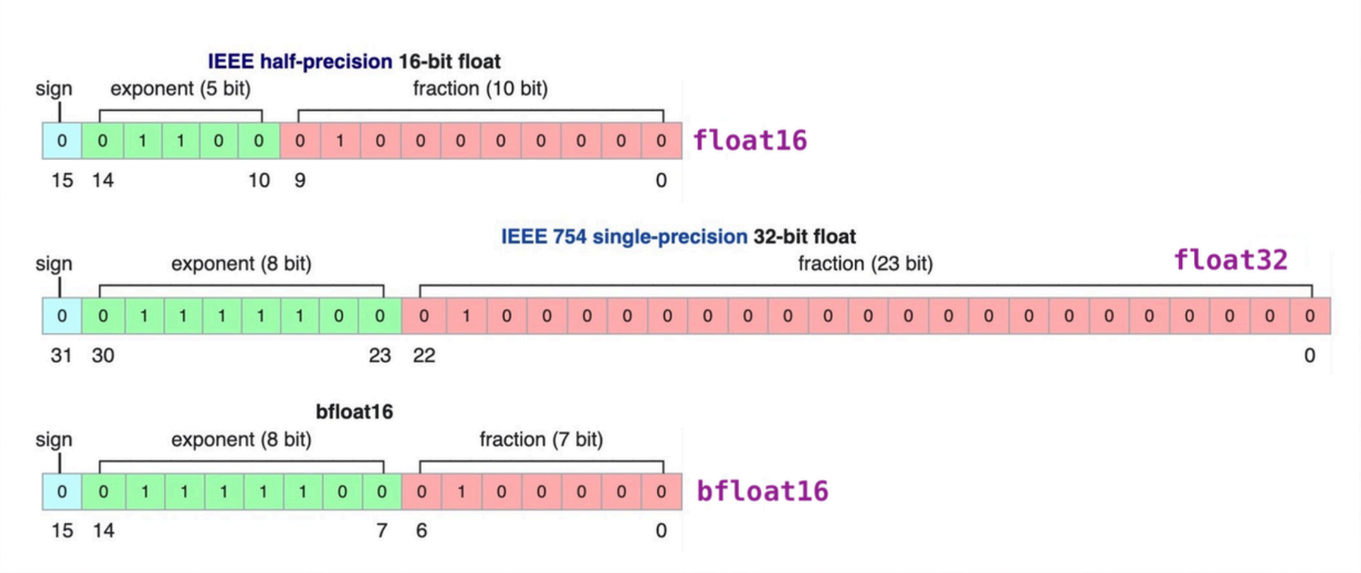
Evolutions architecturelles

Figure 3. GB202 GPU block diagram (full chip).

From RAW power to smarter improvements:

Evolutions architecturelles



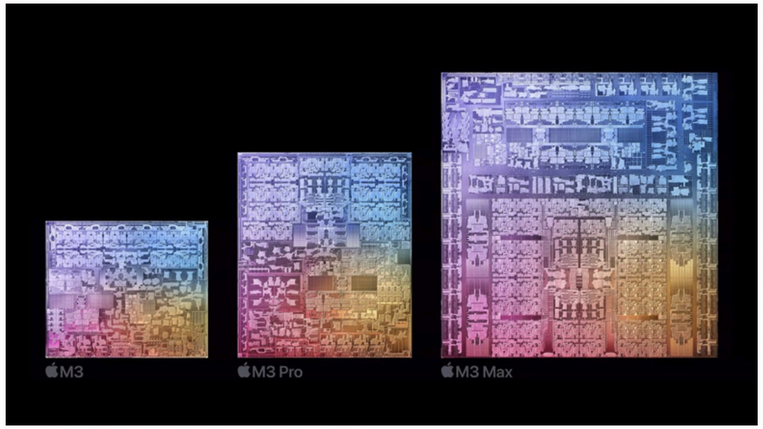


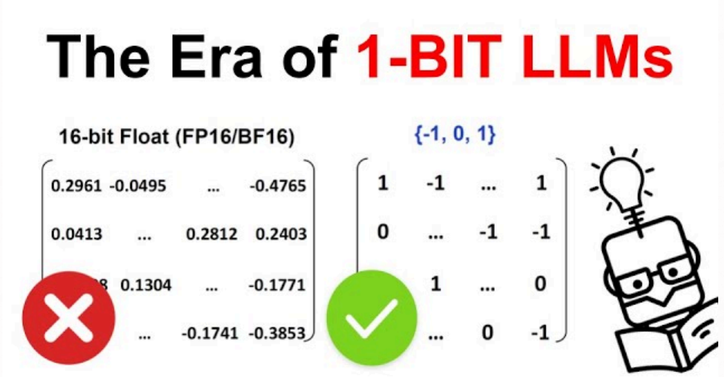
From RAW power to smarter improvements:

Jordan Rey-Jouanchicot

Vers des changements majeurs ?

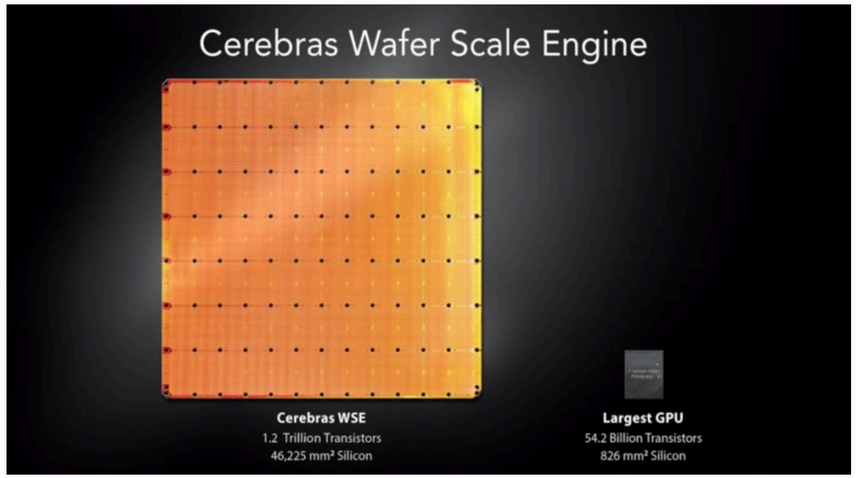
Mémoire unifée et puces unifiées





More specific optimized formats ?

Dedicated hardware



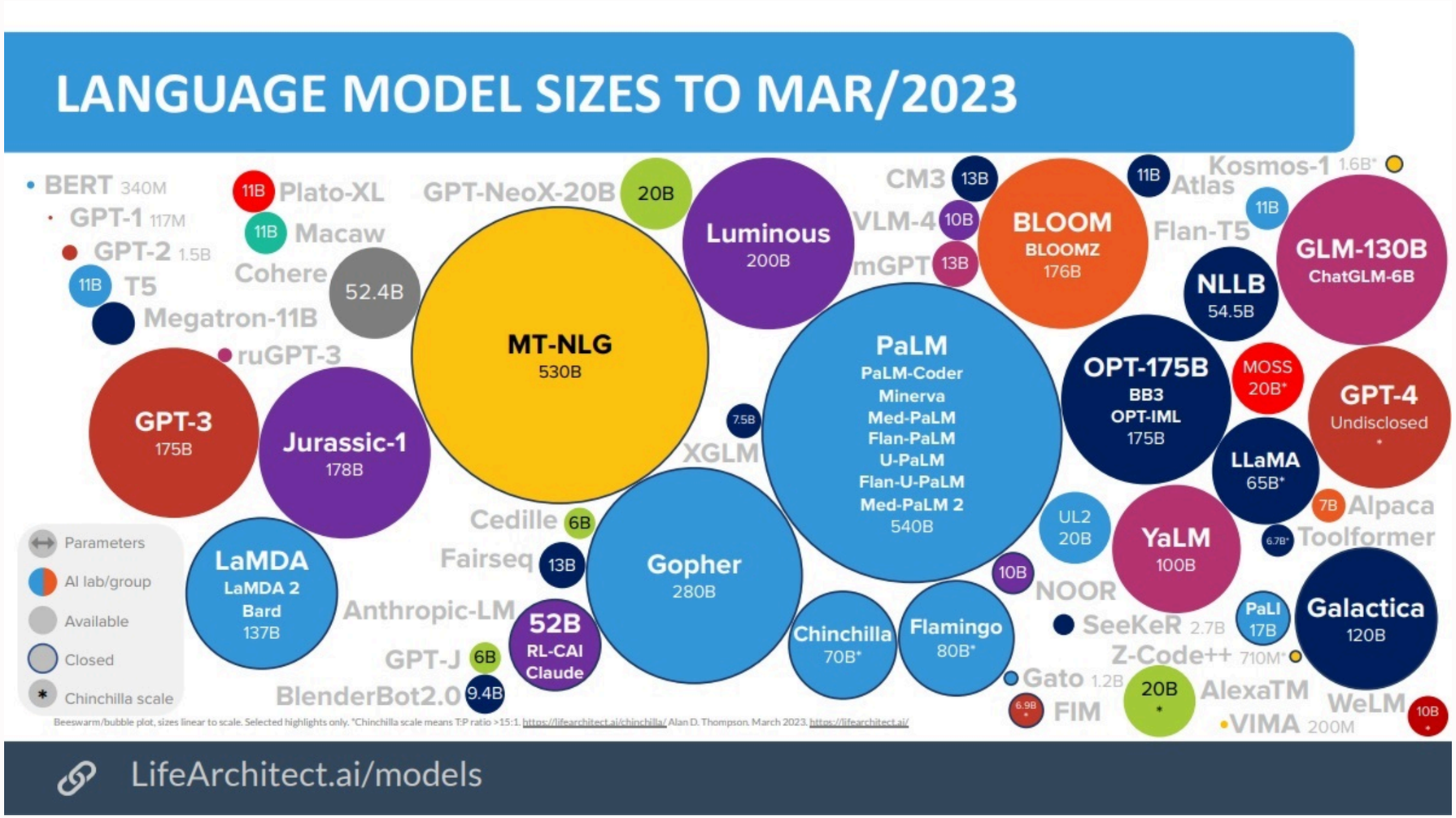
TPU/NPU/DLA, etc.

HPDA 2025

DÉPLOIEMENT IA: HPML

Jordan Rey-Jouanchicot - jordan.reyjouanchicot@orange.com
Thierry Garcia - CYU

Le développement de l'IA



Yolo V8 : 3 à 70 milliards
VGG16 (2014) : 138 millions
LLama 3 (2024): 7 à 70 Billions
DeepSeek R1: 670 Billions*

Qu'est ce qu'un "SLM" en 2025 ?

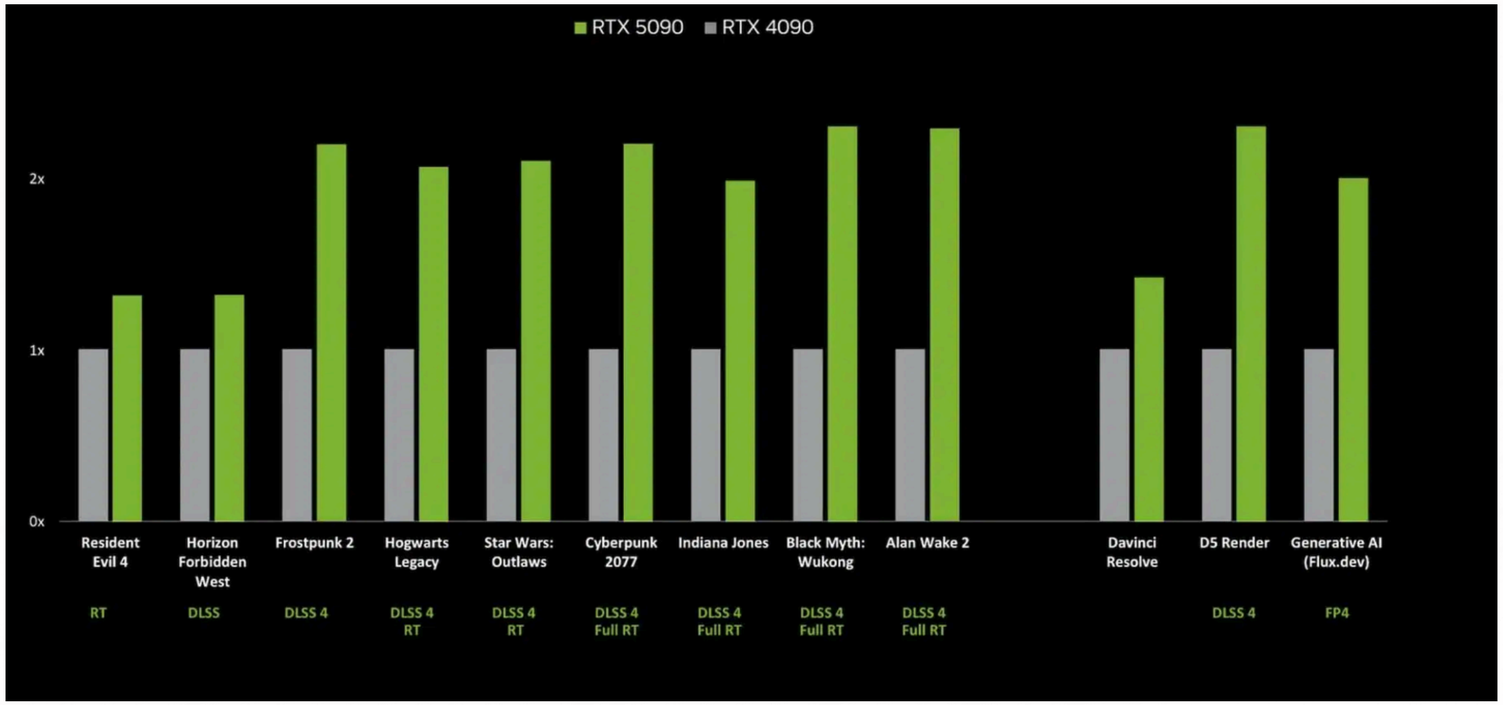
Les modèles

Une architecture unifiée

Augmentation du nombre de paramètres des modèles mais le hardware s’améliore plus lentement

Optimisation du hardware... Mais aussi du software

La plupart des méthodes qu’on va présenter ont été appliqués en premier à des modèles de Computer Vision (CNN) principalement pour du déploiment sur Edge mais la grande démocratisation se fait avec les LLMs.



Jordan Rey-Jouanchicot

Le hardware

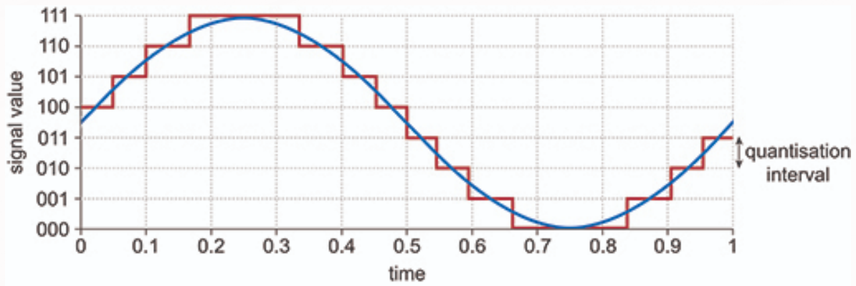
Une architecture unifiée

Du Edge au HPC...
Tout le monde est concerné !

Jordan Rey-Jouanchicot

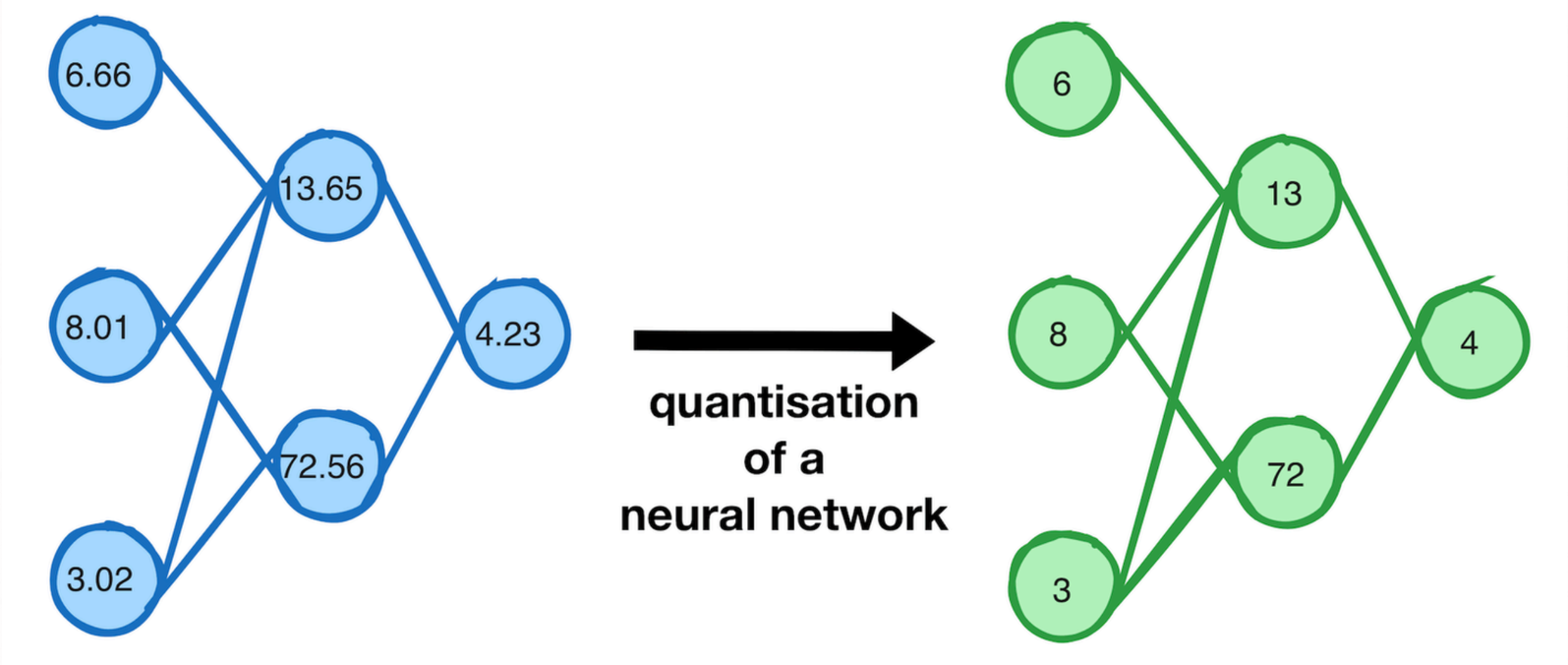
Quantification

On change le format de représentation des poids ?



https://arxiv.org/abs/2402.17764

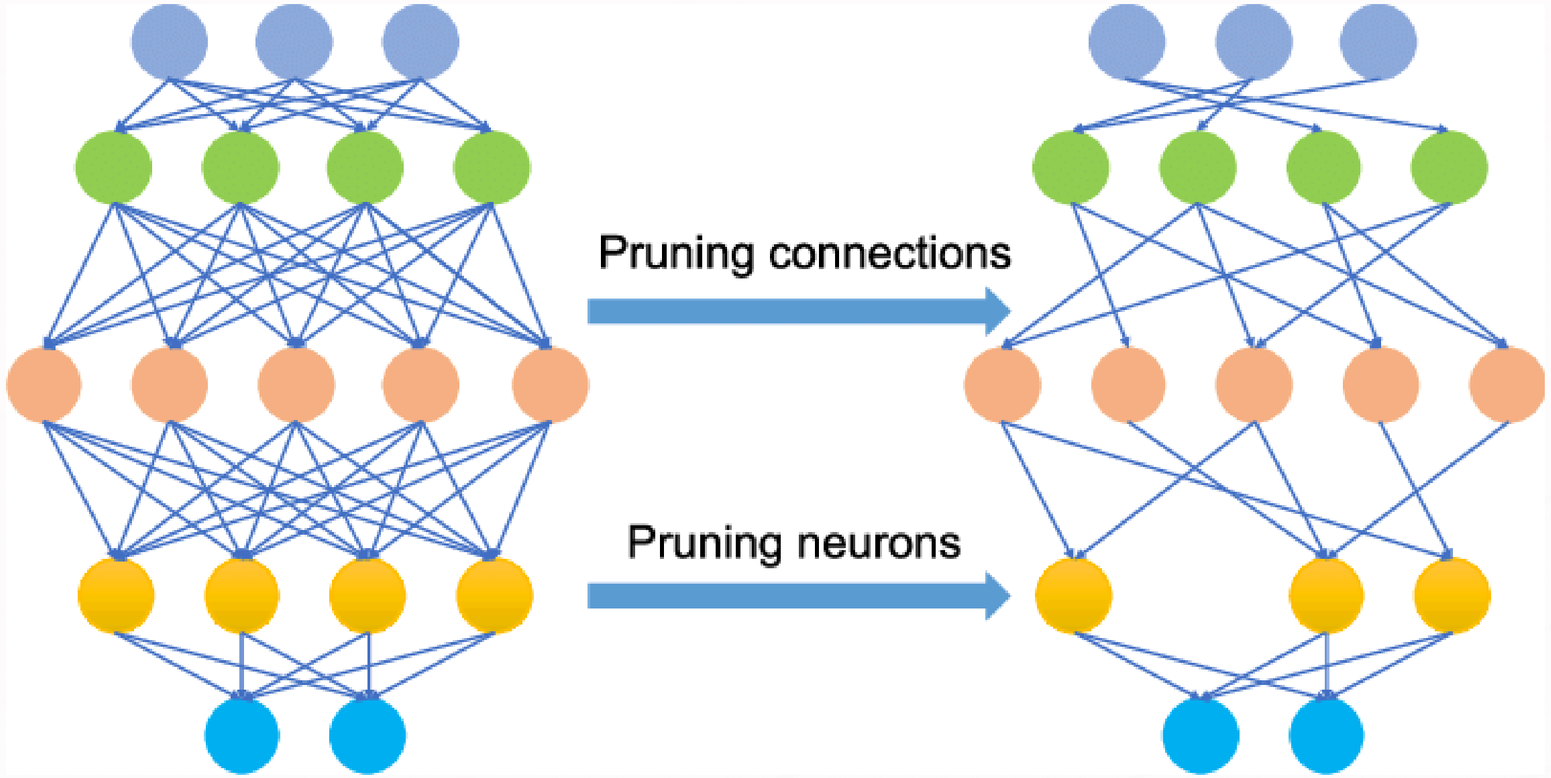
Différentes méthodes existent



Jordan Rey-Jouanchicot

Le pruning

Et si on supprimait des poids dans notre réseaux de neurones ?



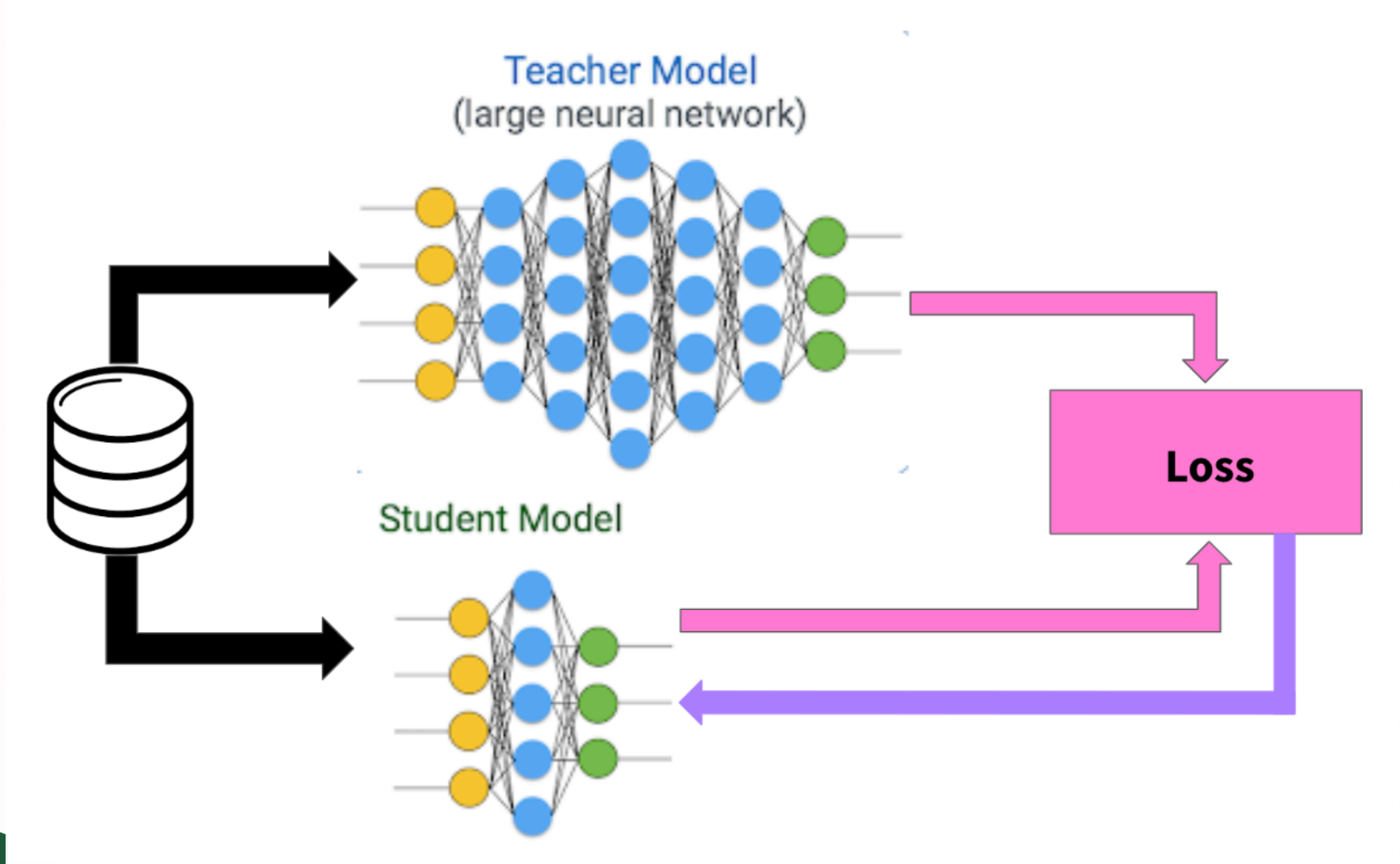
Sparse Tensor Core nécessaire sur GPU pour un gain lors du pruning de connections

https://arxiv.org/abs/2408.11796

Jordan Rey-Jouanchicot

Knowledge distillation

Et si on utilisait un modèle comme professeur pour un plus petit ?



https://arxiv.org/abs/1503.02531

Jordan Rey-Jouanchicot

Speculative decoding

Et si on inférait plus pour aller plus vite ?

Et d’autres !

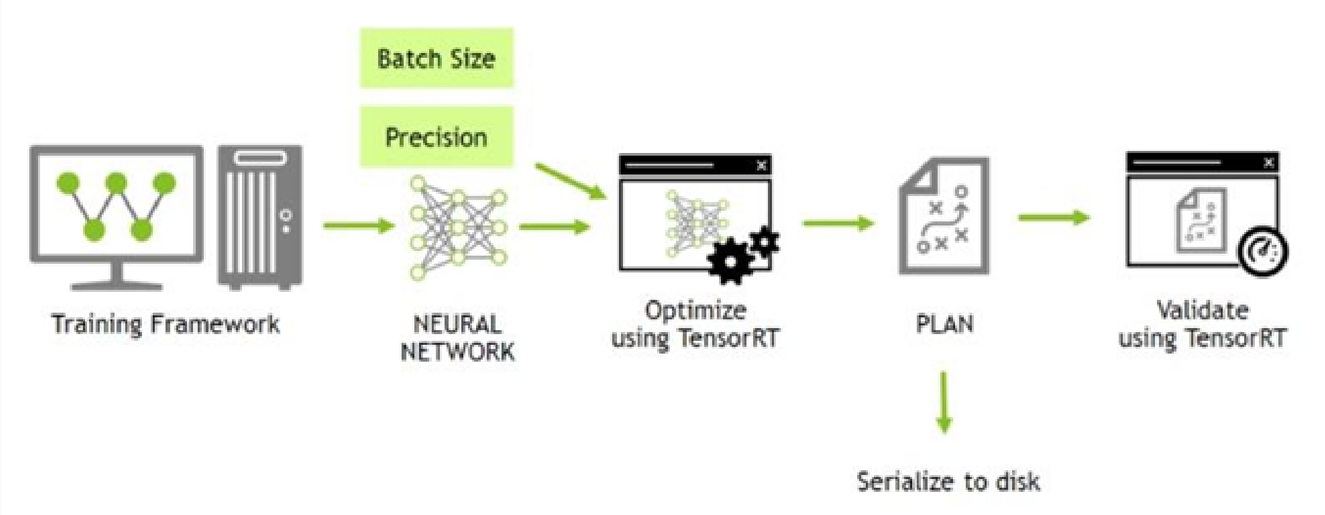
Looking back at speculative decoding

Yaniv Leviathan, Distinguished Engineer, Matan Kalman, Software Engineer, and Yossi Matias, Vice President & Head, Google Research

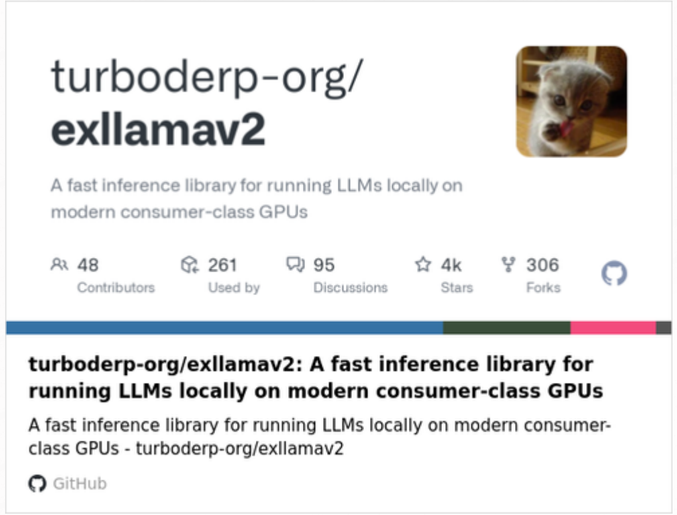
research.google

Des outils

Moteurs d'inférences et/ou de déploiement



TensorRT



La recherche est très active !

CONTACT

Jordan Rey-Jouanchicot

jordan.reyjouanchicot@orange.com

https://www.linkedin.com/in/jordanreyjouanchicot/