Programmation Parallèle

Le calcul haute performance
Architectures mémoire distribuée/mémoire partagée

Sophie Robert

UFR ST Département info

Le déroulement du module

Le contenu

- 15h CM Intro, Mémoire distribuée (MPI), Mémoire partagée (OpenMP), Vectorisation (intervenant Allan Blanchard)
- 14h de TP (ce n'est pas assez, il faut du travail personnel)
- 6h de TD (algorithmique)

La modalité d'évaluation : en contrôle continu

- 3 projets + des évaluations surveillées, individuelles type questionnaires sur Celene

Usage de l'IA

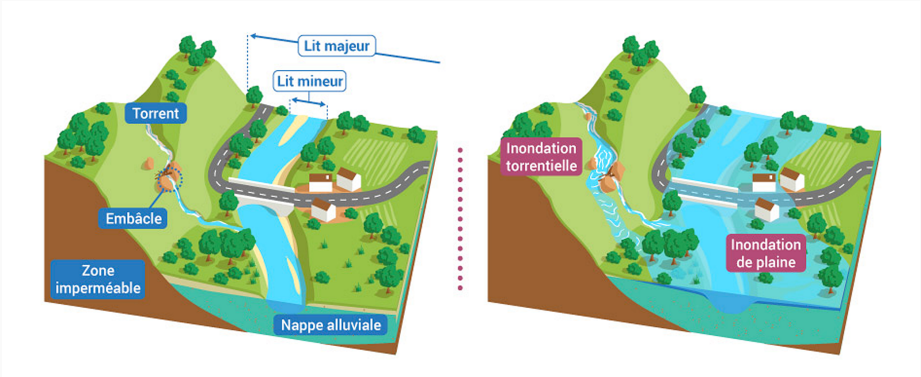
Pour votre apprentissage

- Aucun code généré par l'IA ne sera accepté
- Aide à la compréhension et aide à la recherche d'erreurs dans un code
- Définition de votre prompt pour de l'aide

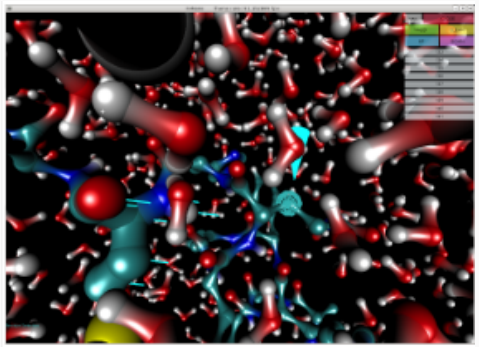
https://celene.univ-orleans.fr/course/view.php?id=13845

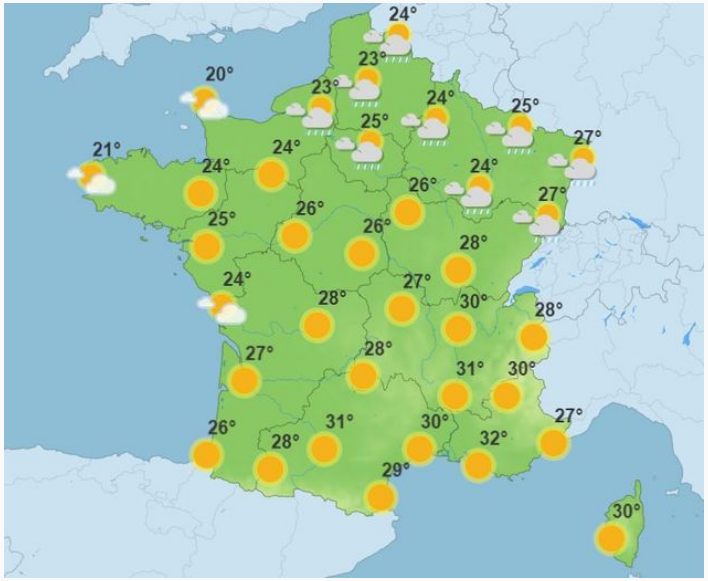
Les caractéristiques

- Premier objectif la performance
- Problèmes visés les problèmes *compute-bound*
  - Les simulations numériques



[1]





[1] eaufrance.fr

Les moyens de calculs

Les plateformes

- Les machines parallèles (à partir des années 1960)
- Les grappes de calculs (début 2000)
    - Démocratisation du calcul parallèle
- Aujourd'hui accès "facile" à des machines très hétérogènes

Paradigme Séparation Données - Calculs

Compute-bound

- Un système de fichiers parallèles POSIX (Lustre, IBM Spectrum Scale/GPFS)
- Un système d'archivage
- Un réseau d'interconnexion efficace inter processus (InfiniBand)

Forte évolution : Architectures hybrides

- Mémoire partagée/Mémoire distribuée
- Accélération avec le calcul sur GPU
- Voir calcul spécialisé sur FPGA
- Nouvelles technologies pour réduire le coût des flux de données mémoire, CPU, GPU (NVlink, DirectStorage).

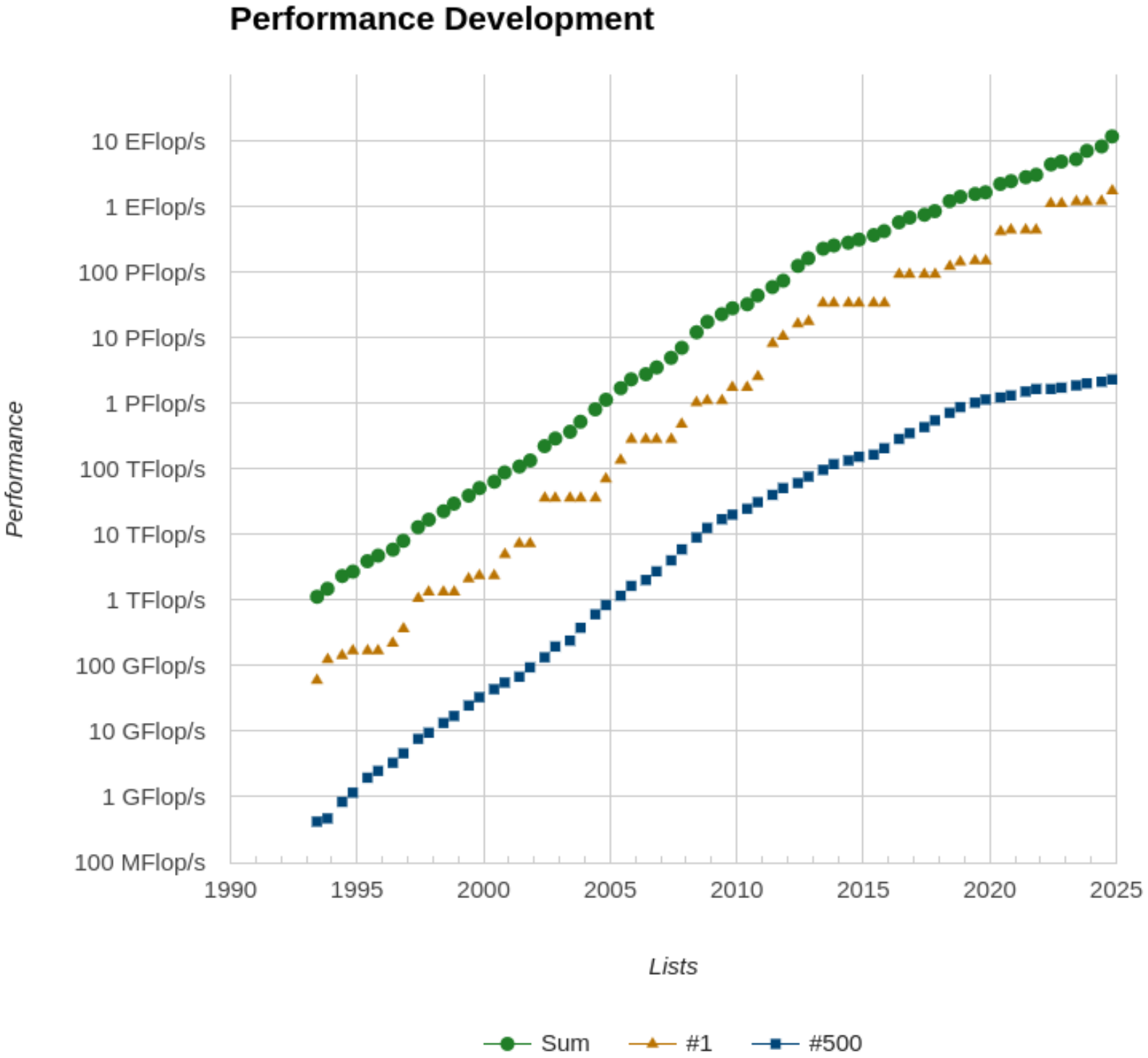
Les moyens de calculs : Le top 500 [2]

Rank	Site	System	Cores	Rmax (TFlop/s)	Rpeak (TFlop/s)	Power (kW)
1	DOE/NNSA/LLNL United States	**El Capitan** - HPE Cray EX255a, AMD 4th Gen EPYC 24C 1.8GHz, AMD Instinct MI300A, Slingshot-11, TOSS HPE	11,039,616	1,742.00	2,746.38	29,581
2	DOE/SC/Oak Ridge National Laboratory United States	**Frontier** - HPE Cray EX235a, AMD Optimized 3rd Generation EPYC 64C 2GHz, AMD Instinct MI250X, Slingshot-11, HPE Cray OS HPE	9,066,176	1,353.00	2,055.72	24,607
3	DOE/SC/Argonne National Laboratory United States	**Aurora** - HPE Cray EX - Intel Exascale Compute Blade, Xeon CPU Max 9470 52C 2.4GHz, Intel Data Center GPU Max, Slingshot-11 Intel	9,264,128	1,012.00	1,980.01	38,698
4	Microsoft Azure United States	**Eagle** - Microsoft NDv5, Xeon Platinum 8480C 48C 2GHz, NVIDIA H100, NVIDIA Infiniband NDR Microsoft Azure	2,073,600	561.20	846.84	
5	Eni S.p.A. Italy	**HPC6** - HPE Cray EX235a, AMD Optimized 3rd Generation EPYC 64C 2GHz, AMD Instinct MI250X, Slingshot-11, RHEL 8.9 HPE	3,143,520	477.90	606.97	8,461

Les moyens de calculs : Le top 500 [2]

- CEA/Recherche 22ième (17ième)
- Jean Zay/Recherche 27ième (premier classement)
- CINES/Académique 30ième (20ième)
- CEA/Recherche 55ième (36ième)
- IBM Industry 60ième
- CEA (encore) 82ième
- ROMEO/Académique 122ième
- Météo France 115ième

[2] https://www.top500.org/


Performance Development
10 EFlop/s
1 EFlop/s
100 PFlop/s
10 PFlop/s
1 PFlop/s
100 TFlop/s
10 TFlop/s
1 TFlop/s
100 GFlop/s
10 GFlop/s
1 GFlop/s
100 MFlop/s
Performance
1990
1995
2000
2005
2010
2015
2020
2025
Lists
Sum
#1
#500

Les moyens de calculs : En France

GENCI : Grand Equipement National de Calcul Intensif

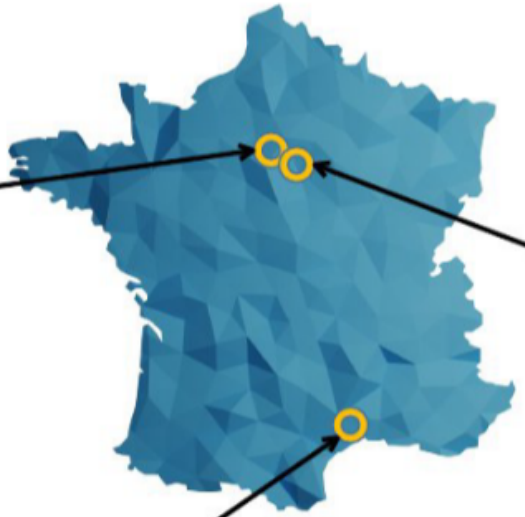
IDRIS: National computing center of the CNRS

GENCI

Ressources gratuites
- financement français
- recherche ouverte



Jean Zay Supercomputer
HPE SGI 8600,
36.8 PFlop/s
2019
IDRIS CNRS





Joliot-Curie Supercomputer
Bull Sequana X1000 and XH2000,
22 PFlop/s
2017
TGCC CEA



Adastra Supercomputer
HPE Cray EX4000,
74.5 PFlop/s
2022
CINES Universities

MINISTÈRE DE L'ENSEIGNEMENT SUPÉRIEUR, DE LA RECHERCHE ET DE L'INNOVATION

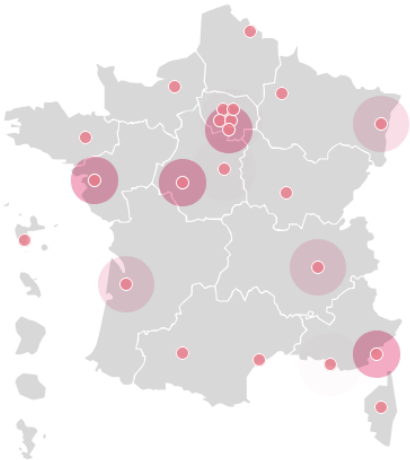
Les moyens de calculs : En France

GENCI : Grand Equipement National de Calcul Intensif

- CINES 30ième
- TGCC(CEA) 132ième
- Jean Zay 27ième mais pour du calcul GPU

Les moyens de calculs : En France

Les méso centres



Les partenaires

GENCI — Le calcul intensif au service de la connaissance
Université des Antilles
UGA Université Grenoble Alpes
eskemm numérique
UBFC Université Bourgogne Franche-Comté
Université d'Orléans
université de TOURS
Università di Corsica Pasquale Paoli
Université de Reims Champagne-Ardenne
Université de Strasbourg
Université de Lille
université PARIS-SACLAY
CentraleSupélec
école normale supérieure paris-saclay
PSL Université Paris
CRIANN
université de BORDEAUX
Université Fédérale Toulouse Midi-Pyrénées
amU Aix Marseille Université
UNIVERSITÉ CÔTE D'AZUR
CENTRALE NANTES
Lyon 1

Les moyens de calculs : En France

En Région Centre Val de Loire



Les moyens de calculs : En France

En Région Centre Val de Loire



LETO : Démarrage rapide

Caractéristiques techniques

	Nœud de calcul	*Nœud gpu intel*	*Nœud gpu amd*
Nom	**node01 à node16**	**gpu01 et gpu02**	**gpu03**
Nombre de nœuds	16	2	1
Nombre de cœurs par nœud	128	40	32
Type de processeur	AMD Epyc 7702 à 2GHz	Intel Xeon Gold 6248 2.5GHz	AMD Epyc 7302 3GHz
Carte Graphique	-	4 cartes NVidia Tesla V100 en Nvlink	3 cartes NVidia Tesla V100 en PCIe
Mémoire disponible par nœud (GB)	512	192	256

Programmation Parallèle ?

Effectuer beaucoup de calculs en utilisant plusieurs cœurs ou processeurs

- Le principe du parallélisme est simple
  - Exécuter en même temps des instructions indépendantes
- La mise en œuvre nécessite de connaître
  - Les architectures des machines parallèles
  - Les techniques de parallélisation
  - Les techniques de programmation

Introduction à l'algorithmique et la programmation parallèles

- Savoir **paralléliser** un problème en fonction de la cible
- Savoir mettre en œuvre en fonction du paradigme de parallélisation

Applications d'aide à la décision

Data-mining

- Gestion d'énormes bases de données
- Exploration efficace de gros volumes de données Exemples : marketing, bio-informatique ...

$\Rightarrow$ **nécessitent de répartir les données**

Simulations

- phénomènes physiques
- phénomènes biochimiques

  Exemple : dynamique moléculaire pour la conception de molécules d'intérêt thérapeutique
- phénomènes environnementaux

  Exemples : météo, inondations, diffusion des pesticides

$\Rightarrow$ **nécessitent une grande puissance de calcul**

Corrections de copies

- des copies : les données
- des exercices constitués de x questions à corriger : les tâches
- des enseignants capables tous de corriger 1 question par minute

Introduction intuitive du parallélisme

Les principes associés

- le processus
- l'équilibrage de charge, l'accélération et l'efficacité
- le parallélisme de tâches ou de données ?
- la mémoire et les coûts d'accès
- la mémoire partagée ou distribuée

Les modèles du parallélisme

Qu'est ce qui est indépendant ?

- Le parallélisme de contrôle
  * Faire plusieurs choses en même temps
- Le parallélisme de données
  * Répéter une action sur des données similaires
- Le parallélisme de flux
  * Travailler à la chaîne

Introduction intuitive du parallélisme

Corrections de copies

- des copies : les données
- des exercices constitués de x questions à corriger : les tâches
- des enseignants capables tous de corriger 1 question par minute

Le stockage des copies

- Initialement les copies sont dans l'amphi et le transfert coûte 30 min.
- x copies peuvent être stockées simultanément dans **un bureau**.
- Les enseignants **partagent ou pas** un même bureau.

La performance ?

Accélération et efficacité

Les 2 valeurs sont importantes et elles vont permettre de juger de la qualité et de la scalabilité de la parallélisation

$\Rightarrow$ Faire des courbes de performance (benchmarks) en faisant varier

- soit les ressources (p le nombre d'enseignants).
- soit la taille du problème (le nombre de copies)

$$A = \frac{T_{seq}}{T_{//}} \quad E = \frac{T_{seq}}{p \times T_{//}}$$

La classification de Flynn

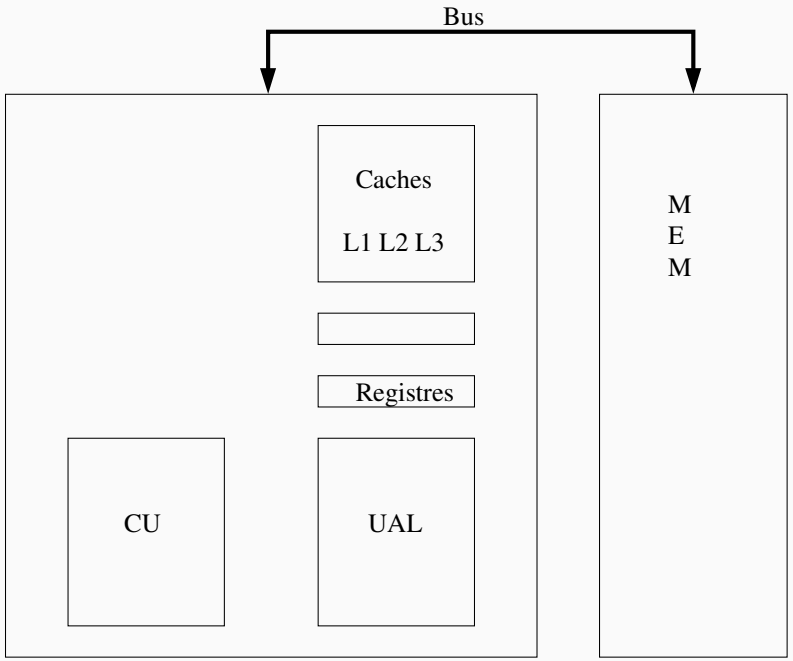
En fonction du flot d'instructions et de données

- Single Instruction Single Data (SISD)
- Single Instruction Multiple Data (SIMD)
- Multiple Instruction Multiple Data (MIMD)

La classification de Flynn

En fonction du flot d'instructions et de données

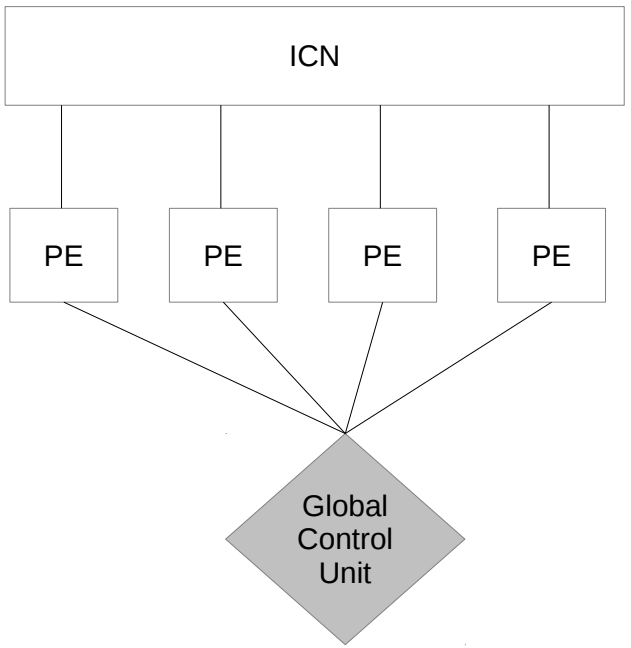
- Single Instruction Single Data (SISD)
- Single Instruction Multiple Data (SIMD)
- Multiple Instruction Multiple Data (MIMD)



La classification de Flynn

En fonction du flot d'instructions et de données

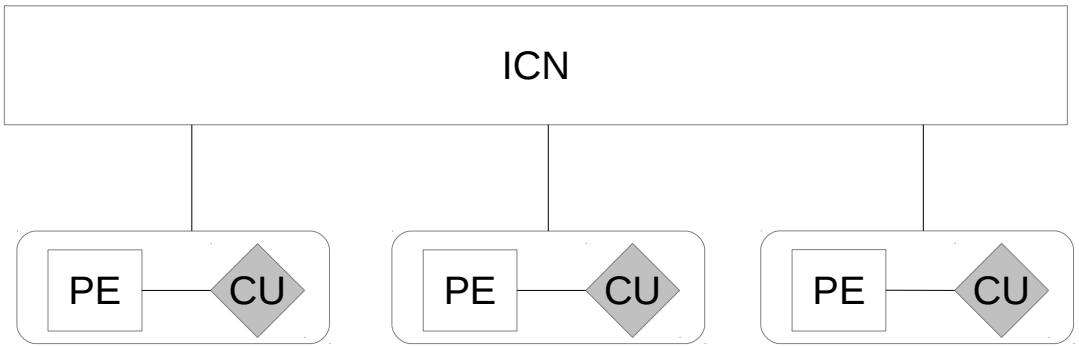
- Single Instruction Single Data (SISD)
- Single Instruction Multiple Data (SIMD)
- Multiple Instruction Multiple Data (MIMD)



La classification de Flynn

En fonction du flot d'instructions et de données

- Single Instruction Single Data (SISD)
- Single Instruction Multiple Data (SIMD)
- Multiple Instruction Multiple Data (MIMD)



L'Inter Connection Network

Accès à la mémoire

L'ensemble des mécanismes pour le transfert de données entre processeurs ou entre les processeurs et les modules de mémoire.

Espace d'adressage partagé

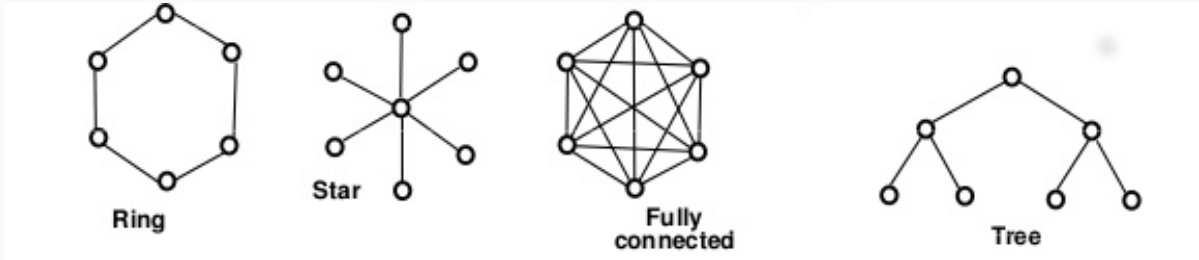
- Uniform Memory Access (UMA)
- Non Uniform Memory Access (NUMA)
    * Cache Coherent NUMA (ccNUMA)
    * noCache Coherent (ncNUMA)

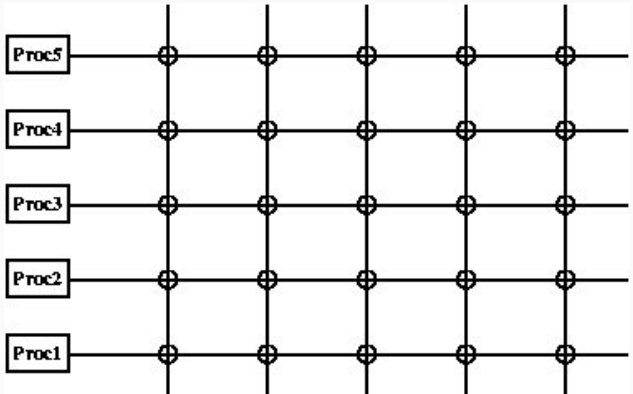
Espace d'adressage distribué (Message Passing Platform)

- Chaque processeur a son propre espace d'adressage
- Les échanges sont explicites par envoi/réception.

L'Inter Connection Network

- static pour du point-à-point, tous les éléments sont reliés entre eux
  * Completed Connected Networks (CCN)
  * Limited Connected Networks (LCN)
- dynamic en utilisant des switchs
  * exemple du crossbar



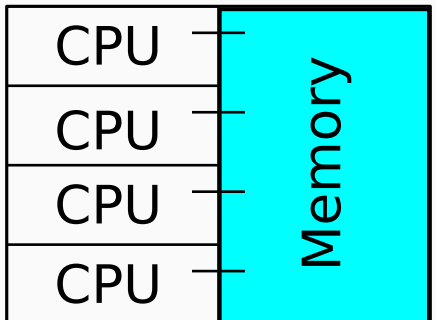


Classification simplifiée

- Architecture à mémoire partagée
- Architecture à mémoire distribuée
- Architecture hybride

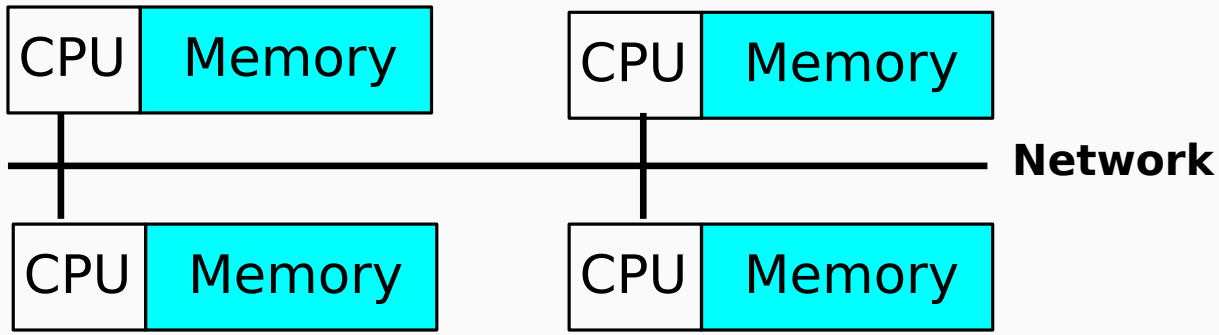
Classification simplifiée

- Architecture à mémoire partagée
  * Tous les processus partagent le même espace mémoire
  * Cela semble facile à programmer car on n'a pas à se soucier de ce que chaque processus peut voir
  * Cependant il faut gérer dans le code les accès concurrents en mémoire ce qui est complexe (synchronisations)
  * En pratique on a de grandes pertes de performances si les données ne sont pas bien gérées, c'est en fait très compliqué
- Architecture à mémoire distribuée
- Architecture hybride



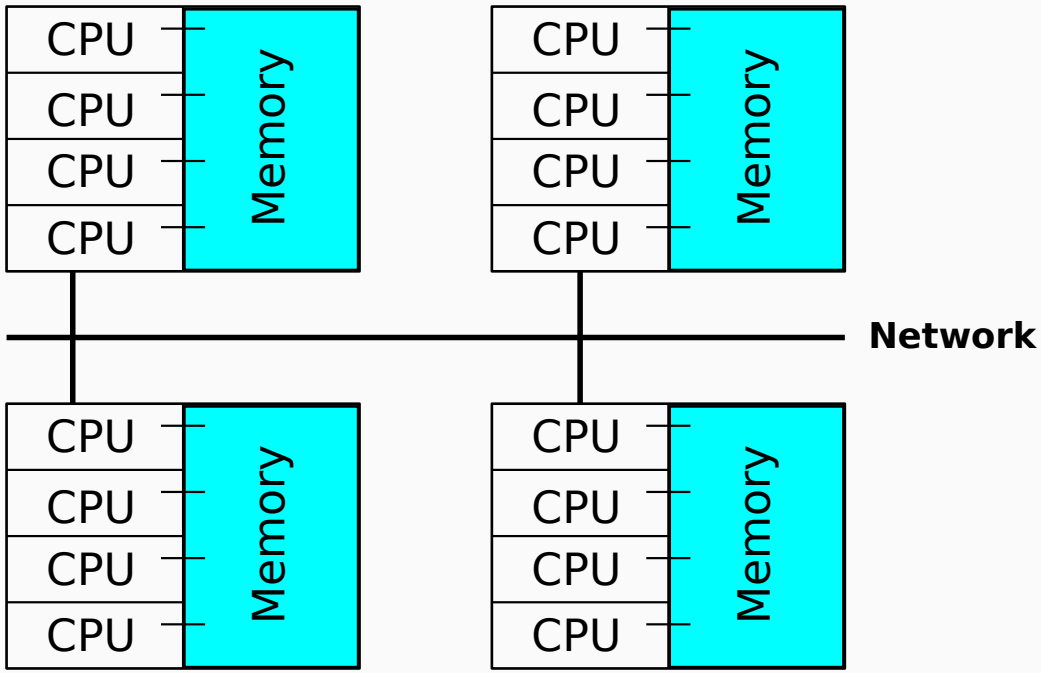
Classification simplifiée

- Architecture à mémoire partagée
- Architecture à mémoire distribuée
  * Chaque processus est indépendant en mémoire
  * Il faut gérer la distribution des données aux différents processus
  * Il faut éventuellement introduire des communications pour échanger des informations utiles entre processus
  * Le travail semble plus important, mais il est en réalité moins complexe dans des programmes de taille importante
- Architecture hybride



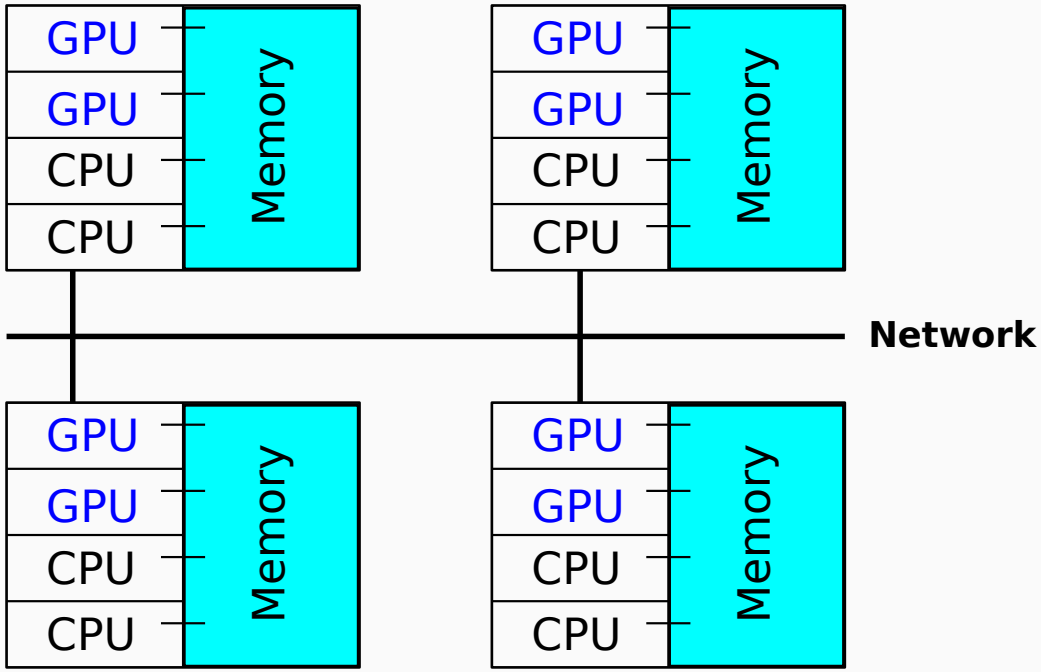
Classification simplifiée

- Architecture à mémoire partagée
- Architecture à mémoire distribuée
- Architecture hybride
  * C'est encore plus compliqué !



Classification simplifiée

- Architecture à mémoire partagée
- Architecture à mémoire distribuée
- Architecture hybride
    * On mélange la programmation sur architecture à mémoire partagée et distribuée !



Le présent et l'avenir c'est le parallélisme !

Comment obtenir plus de puissance ?

- Machines personnelles avec plus de processeurs et de cœurs
- Cluster : ensemble de machines homogènes et localisées
- Massively Parallel Processing(MPP) : Machine spécialisée à mémoire distribuée
- Grid : ensemble de ressources hétérogènes et dé-localisées (peut contenir des clusters)
- Cloud : un parc de machines, d'équipements de réseau et de logiciels maintenu par un fournisseur, que les consommateurs peuvent utiliser en libre service via un réseau informatique
- ...

L'algorithme des k-moyennes

Les entrées

- le nombre de classes k
- un ensemble de n objets

Les sorties

- Un ensemble de k classes d'objets qui minimise le critère des moindres carrés

$$E = \sum_{i=1}^{k} \sum_{p \in C_i} d(p, m_i)^2$$

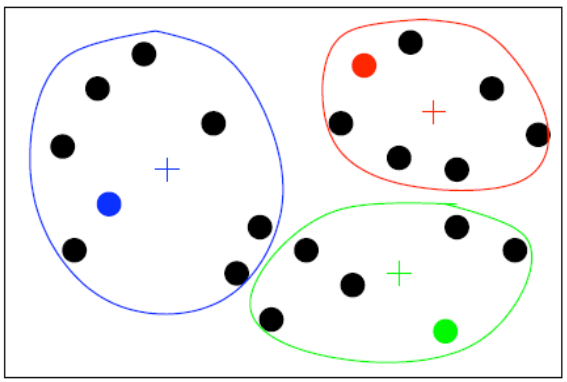
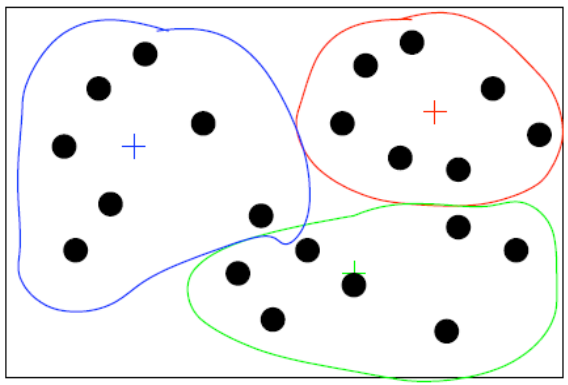
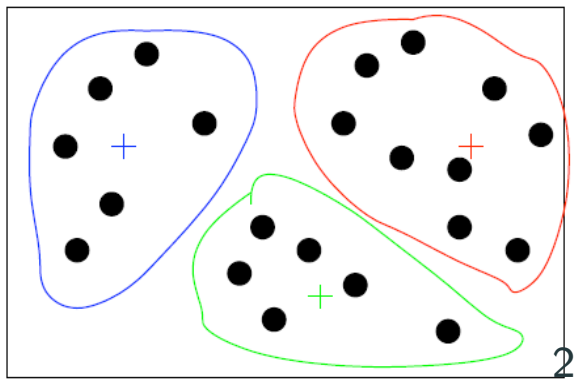
où $d(p, m_i)$ est la distance entre les points p et m_i le centre de la classe C_i.

Une application marketing

Algorithme des k-moyennes

- choisir arbitrairement k objets comme centres des classes
- répéter
  * affecter chaque objet à la classe dont il est le plus similaire au sens de la distance choisie.
  * calculer la valeur moyenne des objets pour chaque classe qui devient le nouveau centre

  jusqu'à convergence

Données numériques

Suppositions sur les données

- 6 millions de clients dans la base de données
- un client caractérisé par un tableau de 500 nombres flottants

Objectif

Partitionner la base de données en 60 classes distinctes à partir desquelles les profils types seront construits.

Suppositions sur l'algorithme des $k-$moyennes

- le temps de calcul de la distance entre un profil type et un client nécessite 1000 opérations flottantes
- l'algorithme converge en 200 étapes.

Suppositions sur la machine “séquentielle”

Caractéristiques

- un processeur 1 cœur
  * 1 milliard d'opérations sur des nombres flottants par seconde (1 GFlop/s)
- un disque dur
  * 3,33 millions de nombres flottants par seconde entre la mémoire vive et le disque dur en lecture et écriture
- la mémoire vive
  * 1 milliard de nombres flottants (64bits) en plus du système d'exploitation et de l'application (environ 8 Go)

Suppositions sur la machine “séquentielle”

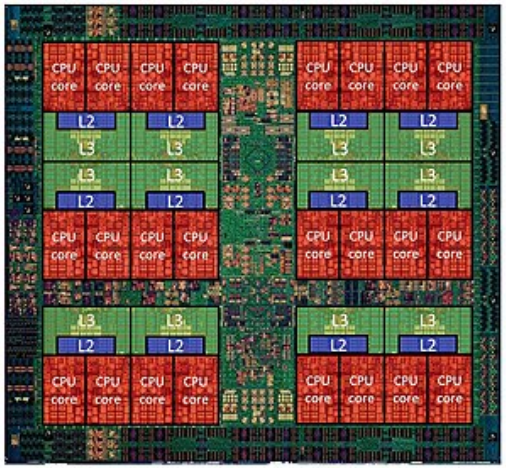
Caractéristiques

- FLOPS (Flop/s) : Nombre d’opérations à virgule flottante par seconde
  - FPU Floating-Point Unit
  - LINPACK comme référence
  - nombre de cœurs $\times$ fréquence $\times$ nombre de cycles pour une opération flottante

Processeurs multi-cœurs

Architecture à mémoire partagée

- Les processeurs actuels ont de 2 à 4 cœurs avec la possibilité d'avoir deux processeurs sur la même carte mère
- Ils partagent la mémoire vive
- Un programme classique (en Java, C, ADA, Ocaml) utilise un seul cœur, il faut une programmation spécifique pour qu'un programme utilise plusieurs cœurs





Grappe de calculs

Machine à mémoire distribuée

- Plutôt que d'avoir une machine avec plusieurs cœurs, on considère maintenant plusieurs machines connectées par un réseau
- Chaque processeur a sa propre mémoire qui ne peut être accédée directement par un autre processeur : on parle de machine à mémoire distribuée
- Un exemple : les grappes de PCs

Grappe : Configuration

Hypothèses

- La grappe est constituée de 8 machines séquentielles (mêmes caractéristiques)
- Le réseau est de type Gigabit Ethernet avec
  * une latence pour l'établissement d'une communication de $4 \times 10^{-5} s$
  * un débit pour une communication point-à-point de 10^7 float/s.
- Les données initiales se trouvent sur un seul disque.

Parallélisation

Questions à se poser

- Machine à mémoire partagée :
  * comment répartir les calculs ?
  * comment limiter ou gérer les accès concurrents aux données
- Machine à mémoire répartie :
  * comment répartir les données ?
  * comment répartir les calculs ?