

# **Calcul à haute performance**



## **Les supercalculateurs en France – Quelques témoignages vécus**

### **Raison d'être**

## ↳ L'implication forte de JvN dans les “computing instrument” est la **capacité de calcul**

- Tous les problèmes « intéressants » ont des formulations non linéaires ⇒ Il faut faire **beaucoup** de calculs
  - ✓ La bombe, et déjà la météo
- JvN analyse le nombre d'opérations requises pour un problème donné ⇒ Début des réflexions sur ce qui deviendra une science à part entière : la complexité algorithmique

# Mécanismes hardware

↪ **Vitesse de commutation des composants élémentaires ⇒ Très sensible au départ**

↪ **Parallélisme « naturel » de la machine**

- L'instruction est décodée par étape qui peuvent donc se chevaucher [4 niveaux dans la machine de l'IAS à Princeton]

✓ Ancêtre des " pipe lines"

↪ **Asynchronisme des opérations lentes**

- Entrées/Sorties, Impressions

✓ Machine Gamma 60 de Bull - CDC avec ses 10 processeurs d'I/O

↪ **Architecture générale de la machine et des bus**

↪ **Les " pipe lines" au sens moderne du terme**

✓ IBM 360-91

## ↪ Les physiciens inventent un langage qui deviendra FORTRAN (1954)

- Les notations à indices (matrices, tenseurs, etc.) conduisent à des expressions très redondantes qu'il ne faut pas traduire en directe ⇒ Gaspillage de ressources critiques
- Nécessité absolue d'optimiser le code généré ⇒ Début de l'histoire des compilateurs
  - ✓ La référence → le compilateur du Fortran H (Medlock, Lowry)
- L'alignement des données est en soi un facteur critique qui peut diviser la performance par 2 !!!
- L'organisation des données en mémoire est encore plus critique car cette fois il s'agit d'E/S ⇒ Nécessité d'optimiser les allocations mémoires pour activer les caches, l'enjeu est cette fois 1 ou 2 ordres de grandeur !!!
  - ✓ Grands tableaux, grandes matrices ± « creuses », 1ère base de données dès les années 60 (IMS/DL1, IDS, ...)

# Symbiose

↪ **Très tôt, tous les experts s'accordent sur la relation symbiotique qu'il y a entre le compilateur et l'architecture de la machine**

- “ Pipe lines ” de + en + sophistiqués
  - ✓ Cas des pipe lines « spéculatifs »
- Architecture VLIW
- Vectorisation, et autres structures ...

**La complexité de ces mécanismes rend leur programmation « à la main » virtuellement impossible**

**→ Confiance totale dans le compilateur et les outils de debug associés**