Architecture des Ordinateurs

Florent Hivert

26 septembre 2004

Introduction

Enjeux du cours :

- Culturel : comprendre comment fonctionne un ordinateur ;
- Technique : comprendre l'évolution des technologies ;
- Informatique : comprendre ce que fait un compilateur, comment il optimise le code et comment ce code est exécuté. On peut ainsi écrire du code plus rapide.

La préhistoire de l'informatique

1614, John Neper invente les logarithmes et la **règle à calculer** ;

1623, Wilhelm Schickard construit une machine mécanique en appliquant les idées de Neper ;

1642, Pascal présente une machine qui additionne et soustrait les nombres de 6 chiffres en base 10 : **la Pascaline**.

1672, amélioration de la Pascaline par Leibnitz un chariot mobile permet de faire les multiplications et les divisions.

Le XIX$^{\mathbf{e}}$ siècle

1805, Joseph Jacquard (d'après des idée de Falcon en 1728) : cartes perforées pour métiers à tisser : c'est le premier **programme**.

1822, Charles Babage essaye de faire une **Machine Analytique**

- machine pilotée par cartes perforées décrivant la séquence d'instructions (programme extérieur) ;
- notion de processeur, de mémoire (magasin), d'unité centrale (moulin) ;
- entrée et sortie par cartes perforées.

Programmes d'Ada Augusta. Ne fonctionnera jamais.

1854, **Machine à Différences** (Babage et Scheutz).

Le XIX$^{\text{e}}$ siècle (suite)

1854, Georges Bool, **Une étude des lois de la pensée**. Calcul des propositions, logique élémentaire ;

Calculateur statistique d'Hermann Hollerith pour les recensements.

Fondation en 1890 de **Tabulating Machine Company** qui devient en 1908 **International Business Machine**.

Le XX$^{\mathrm{e}}$ siècle : les Idées fondatrices

1930, Vannevar Bush : analyseur différentiel analogique, utilisé jusque dans les années 1960.

1938, Claude Shannon invente la **Théorie de l'information** : tout peut être représenté par des 0 et des 1 : c'est la numérisation.

1936, Alan Turing et les **Machines de Turing**. Ce que l'on peut calculer et ce qu'on ne peut pas.

Le XX$^{\text{e}}$ siècle : balbutiements

1938, Konrad Zuse, John Atanasoff, Georges Stibitz : machine à calculer éléctromécanique (Zuse : tubes à vide, jugée irréalisable par le gouvernement allemand).

1939-1944, Howard Aiken, machine éléctromécanique :
- Multiplication de nombres de 23 chiffres en 6 secondes ;
- Addition en 3 dixièmes de seconde.

Naissance de l'ordinateur : 1945

1945, John Eckert et John Mauchly construisent l'**ENIAC** (Electronic Numerical Integrator And Calculator) : 18000 tubes, 30 tonnes. Multiplication de nombre de 10 chiffres en 3ms.

1945, Von Neumann propose l'**EDVAC** (Electronic Discrete Variable Automatic Computer) : Problème de retards et de brevets.

1949, Maurice Wilkes construit l'**EDSAC** (Electronic Delay Storage Automatic Calculator).

Les idées de Von Neumann

- Machine **universelle** programmée ;
- Instructions numériques stockées en mémoire ;
- Instructions modifiables exécutées en séquence avec des ruptures de séquences (Sauts).

Les composants :

- unité arithmétique et logique (UAL binaire) ;
- unité de commande ;
- mémoire centrale ;
- unités d'entrée ;
- unités de sortie.

Actuellement, la plupart des ordinateurs sont des machines de Von Neumann, seules les technologies ont changé.

Proposition de définition de l'architecture de la machine

Trois niveaux possibles :

1 - Vue générale :	agencement des divers composants, relations, communications entre eux.
2 - Niveau macroscopique :	composition interne de chaque partie, jeu d'instructions.
3 - Niveau microscopique :	composants électriques / électroniques présentés sous leur aspect logique.

Ordinateurs de 1$^{\text{re}}$ génération, 1945–1953

Composants : relais, tubes à vides, résistances ;

Logiciels : langage machine seulement ;

1951, Eckert et Mauchly, construisent l'**UNIVAC**.

1953, IBM 701 (40 exemplaires) puis IBM 650 (1500 exemplaires).

Problème de fiabilité des tubes à vides...

Industrie de l'informatique, 2^{e} génération, 1953–1963

1948, John Bardeen, Walter Brattain et William Shockley découvrent **le transistor**.

Composants :	transistors, mémoire à tores de ferrite, imprimantes, bandes magnétiques.
Logiciels :	apparition des systèmes d'exploitation, langages évolués FORmula TRANslator (1957) COmmon Business Oriented Language (1959)

Apparition de l'industrie, IBM, DEC, HP, etc.

Industrie de l'informatique, 3^{e} génération, 1963–1975

Composants :	circuits intégrés,
Machine :	faible consommation énergétique, fiable, encombrement réduit.
Évolution :	Multiprocesseur, temps partagé, accès interactif, Apparition des réseaux Premiers problèmes de compatibilité entre machines.

Industrie de l'informatique, $4^{\text{ème}}$ génération, 1975– ?

Centaine de milliers puis millions de transistors sur une puce : quatrième génération

Composants :	Very Large Scale Integration Premier microprocesseur INTEL 4004 (1971)
Logiciel :	Traitement distribué, machine virtuelle réseau, base de données Convivialité
Évolution :	Parallélisme d'exécution (pipe-line, vectorisation) Ordinateurs personnels Augmentation en puissance

Conclusions

Innovations technologiques principalement.

Principes de Von Neumann toujours valables.

Seules évolutions notables : Parallélisme en général,
multiprocesseur,
pipe-line d'instructions

Forte évolution des puissances et des prix.

Les performances...

- Vues par l'utilisateur : temps de réponse $\neq$ temps CPU pur.
- Vues par le centre de calcul : débit de sortie (« throughput »)
  $=$ nombre de traitements effectués sur une certaine durée.

- **tous les composants concourent aux performances**,
  il faut rechercher un équilibre.

- **plus les composants sont petits, plus ils sont rapides** :
  - une seule puce contenant tout (de moins en moins vrai) ;
  - temps de traversé signal à l'intérieur d'un composant ;
  - localité et hiérarchie mémoire.

Quelques remarques sur l'évolution des performances

Loi de Moore : la vitesse des processeurs et la taille de la mémoire doublent tous les 18 mois.

Mais ce n'est pas dû seulement à la technologie : Évolution de l'Architecture.

Évolution plus rapide pour les micros que pour les super-ordinateurs, disparition des mini-ordinateurs.

Mesure des performances

* La seule mesure est le temps !

* Interviennent :
  - la fréquence d'horloge ;
  - nombre de cycle moyen par instructions ;
  - nombre d'instructions.

$$\text{Temps total} = \text{Nbr d'instructions} \times \text{Nbr de cycles moyen par instr} \\ \times \text{temps de cycle d'horloge}$$

Loi d'Amdahl

$$\text{Accélération} = \frac{\text{Temps sans la caractéristique}}{\text{Temps avec la caractéristique}}$$

Mais si l'on améliore une partie du dispositif (programme ou processeur) :

$$\begin{aligned}\text{Accélération}_{\text{Globale}} &= \frac{\text{Temps}_{\text{Ancien}}}{\text{Temps}_{\text{Nouveau}}} \\ &= \frac{1}{(1 - \text{Fraction}_{\text{Améliorée}}) + \frac{\text{Fraction}_{\text{Améliorée}}}{\text{Accélération}_{\text{Améliorée}}}}\end{aligned}$$

Retenir

Grande dépendance entre l'architecture et le compilateur.

Rendre rapide le cas courant.

Ne pas trop perdre de temps sur les cas rares.