

Systèmes d'exploitation et architecture des ordinateurs

FLORIAN LETOMBE
letombe@cril.univ-artois.fr
Bureau 105F

Le Cours

- ▶ Unité d'enseignement 1.2
- ▶ Module 1.23 : Outils et méthodes informatiques pour le multimédia
- ▶ Volume horaire : 24 h (6 h Cours, 6 h TD, 12 h TP)
- ▶ Objectifs :
 - ▶ comprendre le fonctionnement système, matériel et logiciel, d'un ordinateur dans son contexte de travail
- ▶ Pré-requis : aucun !!!

Contenu

- ▶ Architecture des ordinateurs
 - ▶ codage des informations (bits, octets, codage des nombres, codage ASCII, ...)
 - ▶ éléments de logique
 - ▶ matériel (unité centrale, processeur, bus, périphériques, ...)
 - ▶ gestion de la mémoire
 - ▶ principe du « boot » et BIOS
 - ▶ drivers et librairies
- ▶ Systèmes d'exploitation
 - ▶ principe et histoire des systèmes d'exploitation
 - ▶ systèmes de fichiers
 - ▶ gestion de tâches, notion de processus
 - ▶ les caches (processeur, mémoire, disque, ...)
 - ▶ les interfaces utilisateurs
 - ▶ notion de langages de commandes

Précisions

- ▶ Intervenant Cours & TDs :
 - ▶ F. Letombe
- ▶ Intervenants TPs :
 - ▶ J. Hondermarck
 - ▶ F. Letombe

- ▶ Modalités de contrôle de connaissances :
 - ▶ Théorique :

$$\max\left(\frac{DS + Examen}{2}; Examen\right)$$

- ▶ Pratique : 2 Examens TP
- ▶ Calcul de la moyenne :

$$Moyenne = \frac{2}{3} Théorique + \frac{1}{3} Pratique$$

Histoire de l'informatique

- La « préhistoire » (-3000 à 1940)

- Les premiers pas de l'informatique (1941 à 1968)

- D'autres dates et personnages importants

Architecture des ordinateurs

- Généralions d'ordinateurs

- Codage de l'information

- Architecture de von Neumann

- La structure matérielle d'un ordinateur

Systemes d'exploitation

- Définition

- Processus

- Mémoire

- Entrées/Sorties

- Systemes de fichiers

... Un tour de magie

Carte n°1

1	3	5	7	9	11	13	15
17	19	21	23	25	27	29	31
33	35	37	39	41	43	45	47
49	51	53	55	57	59	61	63

Carte n°2

2	3	6	7	10	11	14	15
18	19	22	23	26	27	30	31
34	35	38	39	42	43	46	47
50	51	54	55	58	59	62	63

Carte n°3

4	5	6	7	12	13	14	15
20	21	22	23	28	29	30	31
36	37	38	39	44	45	46	47
52	53	54	55	60	61	62	63

Carte n°4

8	9	10	11	12	13	14	15
24	25	26	27	28	29	30	31
40	41	42	43	44	45	46	47
56	57	58	59	60	61	62	63

Carte n°5

16	17	18	19	20	21	22	23
24	25	26	27	28	29	30	31
48	49	50	51	52	53	54	55
56	57	58	59	60	61	62	63

Carte n°6

32	33	34	35	36	37	38	39
40	41	42	43	44	45	46	47
48	49	50	51	52	53	54	55
56	57	58	59	60	61	62	63

Citation



« - *Est-ce que les histoires que vous racontez
ne vous empêchent pas de dormir ?*
- *Si, mais comme ce sont des histoires à
dormir debout, je récupère !* »

Raymond Devos

Extrait du sketch **Le vent de la révolte**

Histoire de l'informatique

La « préhistoire » (-3000 à 1940)

Les premiers pas de l'informatique (1941 à 1968)

D'autres dates et personnages importants

Architecture des ordinateurs

Génération d'ordinateurs

Codage de l'information

Architecture de von Neumann

La structure matérielle d'un ordinateur

Systèmes d'exploitation

Définition

Processus

Mémoire

Entrées/Sorties

Systèmes de fichiers

Avant notre ère

-3000 : Période de l'empereur Chinois **Fou-Hi** dont le symbole magique, **l'octogone à trigramme** contient les 8 premiers nombres représentés sous forme binaire par des traits interrompus ou non : 000 001 010 011 etc ...



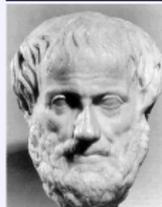
-2000 : Apparition au Moyen Orient du premier « outil » de calcul : **l'abaque**



-1000 : Invention du **boulier** en Chine



-300 : Le philosophe Grec **Aristote** définit dans son oeuvre ce qu'est **la logique** (ou Organon)



Les premières claculatrices mécaniques

1623 : Wilhelm Schickard invente une horloge calculante



1632 : L'Anglais Oughtred invente la Règle à calcul



1642 : Pascal met au point, pour aider son père collecteur des impôts à Rouen, la Pascaline



1679 : Gottfried Wilhelm von Leibniz découvre et met au point une arithmétique binaire



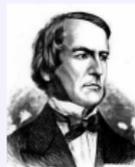
1694 : Leibniz invente une machine à calculer dérivée de la Pascaline mais capable de traiter les multiplications et division

1820 : Charles-Xavier Thomas de Colmar invente l'arithmomètre sur la base de la machine de Leibniz



Les grands noms

1854 : George **Boole** publie un ouvrage dans lequel il démontre que tout processus logique peut être décomposé en une suite d'opérations logiques appliquées sur deux états



1904 : Invention du premier tube à vide, **la diode** par John Ambrose **Fleming**



1937 : Alan M. **Turing** publie un document sur les nombres calculables et invente la **Machine de Turing** (puis le test de Turing en 1950)



1938 : Thèse de Claude E. **Shannon** qui le premier fait le parallèle entre les circuits électriques et l'algèbre Booléenne. Il définit le chiffre binaire : **bit** (Binary digiT)



Histoire de l'informatique

La « préhistoire » (-3000 à 1940)

Les premiers pas de l'informatique (1941 à 1968)

D'autres dates et personnages importants

Architecture des ordinateurs

Génération d'ordinateurs

Codage de l'information

Architecture de von Neumann

La structure matérielle d'un ordinateur

Systèmes d'exploitation

Définition

Processus

Mémoire

Entrées/Sorties

Systèmes de fichiers

Les premiers ordinateurs

1941 : Création du calculateur binaire **ABC** par John Atanasoff et Clifford Berry - premier calculateur à utiliser l'algèbre de Boole

1941 : Konrad **Zuse** met au point le **Z3**, le premier calculateur avec programme enregistré (premier véritable ordinateur)

1945 : John **Von Neumann** décrit l'EDVAC (Electronic Discrete Variable Automatic Computer) ⇒ **architecture** Von Neumann

1946 : Création de l'**ENIAC** (Electronic Numerical Integrator and Computer) par P. Eckert et J. Mauchly



Les premiers langages de programmation

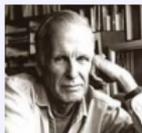
1950 : Invention de **l'assembleur** par Maurice V. Wilkes de l'université de Cambridge. Avant, la programmation s'effectuait directement en binaire



1955 : IBM lance **l'IBM 704** développé par Gene Amdahl, machine sur laquelle sera développé le langage FORTRAN



1957 : Création du premier langage de programmation universel, le **FORTRAN** (FORmula TRANslator) par John Backus d'IBM



1964 : Thomas Kurtz et John Kemeny créent le langage **BASIC** (Beginner's All-purpose Symbolic Instruction Code) au Dartmouth College pour leurs étudiants



1968 : Création du langage **PASCAL** par Niklaus Wirth



Histoire de l'informatique

La « préhistoire » (-3000 à 1940)

Les premiers pas de l'informatique (1941 à 1968)

D'autres dates et personnages importants

Architecture des ordinateurs

Génération d'ordinateurs

Codage de l'information

Architecture de von Neumann

La structure matérielle d'un ordinateur

Systèmes d'exploitation

Définition

Processus

Mémoire

Entrées/Sorties

Systèmes de fichiers

1964 : Création du code ASCII (American Standard Code for Information Interchange), normalisé en 1966 par l'ISO

```
!"#$%&'()*+,-./  
0123456789:;<=>?  
@ABCDEFGHIJKLMNO  
PQRSTUVWXYZ[\]^_  
`abcdefghijklmno  
pqrstuvwxyz{|}~
```

1965 : Gordon Moore écrit la première « **loi de Moore** » disant que la complexité des circuits intégrés doublera tous les ans



1969 : Ken Thompson et Dennis Ritchie mettent au point **UNIX** sur un Dec PDP 7

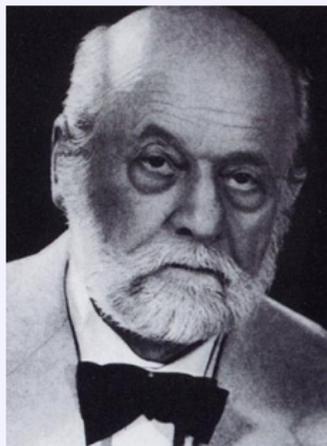


etc ...

Plus d'informations sur

<http://histoire.info.online.fr>

Citation



« *L'architecture, c'est ce qui fait les belles
ruines.* »
Auguste Perret

Objectif

Comprendre le fonctionnement d'un ordinateur à bas niveau

Système bancaire	Réservations aériennes	Navigateur Web	} Application
Compilateurs	Éditeurs	Interpréteur de commandes	
Système d'exploitation			} Programme système
Langage machine Micro-architecture Périphériques physiques			

} Matériels

Histoire de l'informatique

La « préhistoire » (-3000 à 1940)

Les premiers pas de l'informatique (1941 à 1968)

D'autres dates et personnages importants

Architecture des ordinateurs

Générations d'ordinateurs

Codage de l'information

Architecture de von Neumann

La structure matérielle d'un ordinateur

Systèmes d'exploitation

Définition

Processus

Mémoire

Entrées/Sorties

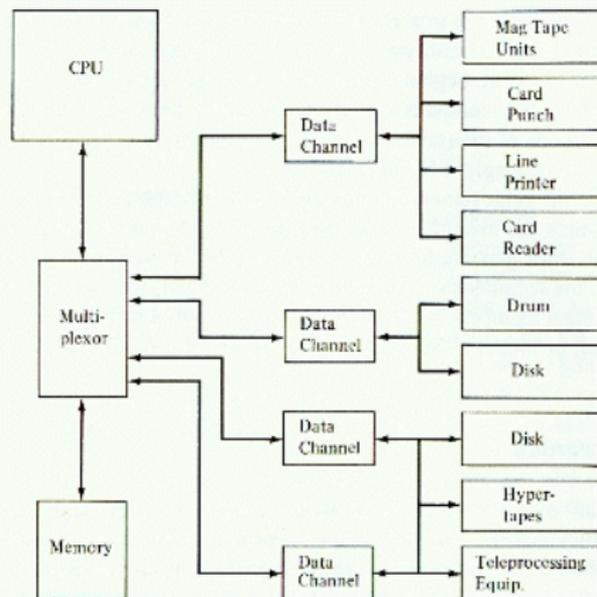
Systèmes de fichiers

1^{ère} génération (1945–1957) : les tubes à vide et les cartes enfichables

- ▶ Pas de système d'exploitation
- ▶ **ENIAC** (Electronic Numerical Integrator and Computer), premier ordinateur électronique numérique généraliste : 30 tonnes, 500 m^2 , >18000 tubes à vide, 5000 additions/s
- ▶ Principal inconvénient : **programmé manuellement** (positionner les commutateurs, brancher et débrancher les cables)
- ▶ La machine de von Neumann (EDVAC \Rightarrow IAS), *concept de programme enregistré*
- ▶ Grande lenteur/fragilité

2^{ème} génération (1958–1964) : les transistors et le traitement par lots

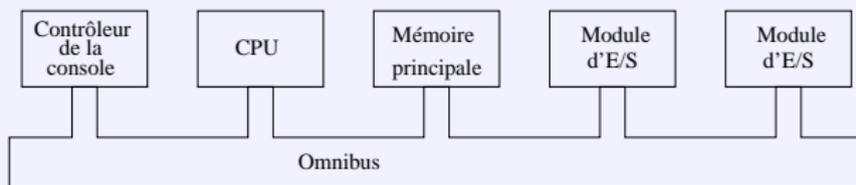
- ▶ Transistor : plus petit, moins cher, plus fiable
- ▶ Apparition des langages **FORTRAN** et **assembleur**
- ▶ Apparition des **systèmes d'exploitation** (SE) : FMS (Fortran Monitor System) et IBSYS (IBM 7094)
- ▶ IBM 7094 : architecture à **commutateur central**



An IBM 7094 configuration.

3^{ème} génération (1965–1971) : les circuits intégrés et la multiprogrammation

- ▶ Amélioration des coûts et des performances (circuits intégrés)
- ▶ Une famille d'ordinateurs compatibles entre eux
- ▶ Une architecture et un même jeu d'instructions
- ▶ Apparition de la **multiprogrammation** (partitionnement de la mémoire pour des tâches différentes)
- ▶ Inconvénient : système d'exploitation énorme et très complexe
- ▶ **Structure en bus** du PDP-8



4^{ème} génération (1972–1977) : les ordinateurs personnels

- ▶ Développement des **circuits LSI** (Large Scale Integration) : plus de 1000 composants sur une seule puce de circuits intégrés
- ▶ Ils ont la même architecture que les mini-ordinateurs, mais leur prix est beaucoup moins élevé
- ▶ Il existe deux systèmes d'exploitation principaux : **MS-DOS** (Microsoft Inc.) et **UNIX**
- ▶ Apparition du partage de temps, une variante de la multiprogrammation (chaque utilisateur possède un terminal en ligne)

5^{ème} génération (1978–????) : les ordinateurs personnels portables et de poche

- ▶ Apparition de réseaux d'ordinateurs individuels qui fonctionnent sous des systèmes d'exploitation en réseau ou des systèmes d'exploitation distribués
- ▶ Apparition des **PIC** (Personal Intelligent Communicator de chez Sony) et des **PDA** (Personal Digital Assistant, comme le Newton de chez Apple), grâce à l'intégration des composants et l'arrivée des systèmes d'exploitation de type « micro-noyau »
- ▶ Utiles pour les « nomades » et les systèmes de gestion des informations (recherche, navigation, communication)
- ▶ Utilisent la reconnaissance de caractère (OCR) et les modes de communication synchrone et asynchrone (mode messagerie)

Récapitulatif

Génération	Dates approximatives	Technologie	Performance (opérations par seconde)
1	1946–1957	Tube à vide	40 000
2	1958–1964	Transistor	200 000
3	1965–1971	Intégration à petite et moyenne échelle	1 000 000
4	1972–1977	Intégration à grande échelle	10 000 000
5	1978–????	Intégration à très grande échelle	100 000 000

Histoire de l'informatique

La « préhistoire » (-3000 à 1940)

Les premiers pas de l'informatique (1941 à 1968)

D'autres dates et personnages importants

Architecture des ordinateurs

Génération d'ordinateurs

Codage de l'information

Architecture de von Neumann

La structure matérielle d'un ordinateur

Systèmes d'exploitation

Définition

Processus

Mémoire

Entrées/Sorties

Systèmes de fichiers

Changement de base

- ▶ Habitude de travailler en base 10 (système décimal)
- ▶ 10 symboles distincts : les chiffres
- ▶ En base b , on utilise b chiffres

Nom de la base	b	Chiffres
Binaire	2	0, 1
Octal	8	0, 1, 2, 3, 4, 5, 6, 7
Décimal	10	0, 1, 2, 3, 4, 5, 6, 7, 8, 9
Hexadécimal	16	0, 1, 2, 3, 4, 5, 6, 7, 8, 9, A, B, C, D, E, F

Représentation des nombres entiers

Définition (*Forme générale d'un nombre entier en base b*)

$$\text{Nombre} = a_n a_{n-1} \dots a_1 a_0 = \sum_{i=0}^n a_i \times b^i$$

Exemple

La forme générale d'un nombre entier en décimal (base 10) est

$$\text{Décimal} = d_n d_{n-1} \dots d_1 d_0 = \sum_{i=0}^n d_i \times 10^i$$

Dans 10 ans ...

$$2016 = d_3 d_2 d_1 d_0 = \sum_{i=0}^3 d_i \times 10^i$$

avec $d_3 = 2$ (**poids fort**), $d_2 = 0$, $d_1 = 1$ et $d_0 = 6$ (**poids faible**)

Exemple détaillé de cette année

Binaire

$$\begin{array}{cccccccccccc}
 1 & & 1 & & 1 & & 1 & & 1 & & 0 & & 1 & & 1 & & 0 \\
 1 \times 2^{10} & + & 1 \times 2^9 & + & 1 \times 2^8 & + & 1 \times 2^7 & + & 1 \times 2^6 & + & 1 \times 2^5 & + & 1 \times 2^4 & + & 1 \times 2^3 & + & 1 \times 2^2 & + & 1 \times 2^1 & + & 1 \times 2^0 \\
 1024 & + & 512 & + & 256 & + & 128 & + & 64 & + & 0 & + & 16 & + & 0 & + & 4 & + & 2 & + & 0
 \end{array}$$

Octal

$$\begin{array}{cccc}
 & 3 & & 7 & & 2 & & 6 \\
 3 \times 8^3 & + & 7 \times 8^2 & + & 2 \times 8^1 & + & 6 \times 8^0 \\
 1536 & + & 448 & + & 16 & + & 6
 \end{array}$$

Décimal

$$\begin{array}{cccc}
 & 2 & & 0 & & 0 & & 6 \\
 2 \times 10^3 & + & 0 \times 10^2 & + & 0 \times 10^1 & + & 6 \times 10^0 \\
 2000 & + & 0 & + & 0 & + & 6
 \end{array}$$

Hexadécimal

$$\begin{array}{ccc}
 & 7 & & D & & 6 \\
 7 \times 16^2 & + & 13 \times 16^1 & + & 6 \times 16^0 \\
 1792 & + & 208 & + & 6
 \end{array}$$

Notations

Définition (*Notation d'indice*)

- ▶ Par défaut, un nombre est écrit en décimal (base 10)
- ▶ Un nombre α en base b se note

$$\alpha_b \text{ ou } (\alpha)_b$$

Exemple

$$\begin{aligned} 2006 &= 11111010110_2 \\ &= 3726_8 \\ &= (3726)_8 \\ &= 2006_{10} \\ &= 7D6_{16} \\ &= (7D6)_{16} \end{aligned}$$

Passage d'une base quelconque en décimal

Définition (*Méthode de conversion $b \rightarrow 10$*)

- ▶ Soit un nombre α écrit en base b comme suit

$$(\alpha)_b = (a_n a_{n-1} \dots a_1 a_0)_b$$

- ▶ Alors $(\alpha)_b = a_n \times b^n + a_{n-1} \times b^{n-1} + \dots + a_1 \times b^1 + a_0 \times b^0$

Exemple

$$\begin{aligned}
 b = 2 \text{ et } \alpha_b = 1010_2 &= 1 \times 2^3 + 0 \times 2^2 + 1 \times 2^1 + 0 \times 2^0 \\
 &= 1 \times 8 + 0 \times 4 + 1 \times 2 + 0 \times 1 \\
 &= 8 + 0 + 2 + 0 \\
 &= 10_{10}
 \end{aligned}$$

$$\begin{aligned}
 b = 16 \text{ et } \alpha_b = 3E8_{16} &= 3 \times 16^2 + 14 \times 16^1 + 8 \times 16^0 \\
 &= 3 \times 256 + 14 \times 16 + 8 \times 1 \\
 &= 768 + 224 + 8 = 1000_{10}
 \end{aligned}$$

Passage du décimal à une base quelconque

Définition (*Méthode de conversion* $10 \rightarrow b$)

- ▶ Soit un nombre α écrit en base 10
- ▶ Divisions successives de α par b
- ▶ Chacun des restes ainsi obtenus, du chiffre de poids faible au chiffre de poids fort, forme α converti en base b

Exemple

$$\begin{array}{r} \text{Quotient} \rightarrow 46 \\ \text{Reste} \rightarrow 0 \end{array} \left| \begin{array}{l} 2 \\ \hline 23 \end{array} \right. \leftarrow \text{base}$$

$$\begin{array}{r} 1 \left| \begin{array}{l} 2 \\ \hline 11 \end{array} \right. \\ \quad 1 \left| \begin{array}{l} 2 \\ \hline 5 \end{array} \right. \\ \quad \quad 1 \left| \begin{array}{l} 2 \\ \hline 2 \end{array} \right. \\ \quad \quad \quad 0 \left| \begin{array}{l} 2 \\ \hline 1 \end{array} \right. \\ \quad \quad \quad \quad 1 \left| \begin{array}{l} 2 \\ \hline 0 \end{array} \right. \end{array}$$

$$46_{10} = 101110_2$$

$$\begin{array}{r} 46 \\ 14 = E \end{array} \left| \begin{array}{l} 16 \\ \hline 2 \end{array} \right. \left| \begin{array}{l} 16 \\ \hline 0 \end{array} \right.$$

$$\begin{array}{c} \downarrow \\ 46_{10} = 2E_{16} \end{array}$$

Cas particuliers des bases 2, 8 et 16

Définition (*Méthode de conversion 2 → 8 (resp. 16)*)

- ▶ Chaque groupe de 3 (resp. 4) bits d'un nombre exprimé en binaire, en partant du bit de poids faible, peut être directement exprimé par un nombre de 0 à 7 (resp. 15)
- ▶ Il est parfois nécessaire d'ajouter un ou deux 0 (**non significatifs**) pour compléter un groupe de bits

Exemple

Hexadécimal		7		B		A		3				
Binaire	0	1	1	1	1	0	1	0	0	0	1	1
Octal				7		5		6		4		3

Cas particuliers des bases 2, 8 et 16 (suite et fin)

Définition (*Méthode de conversion 8 (resp. 16) → 2*)

Chaque chiffre octal (resp. hexadécimal) – 0 à 7 (resp. 15) – est remplacé par un groupe de 3 (resp. 4) bits de valeur équivalente

Exemple

Hexadécimal		7		B		A		3				
Binaire	0	1	1	1	1	0	1	0	0	0	1	1
Octal				5		6		4				3

Opérations arithmétiques de base

- ▶ Mêmes méthodes en base quelconque qu'en base 10
- ▶ Une retenue ou report apparaît quand on atteint ou dépasse la valeur b de la base

Exemple

- ▶ On veut calculer $123_4^2 = 123_4 \times 123_4$
- ▶ Or $123_4 = 1 \times 4^2 + 2 \times 4 + 3 \times 1 = (16 + 8 + 3)_{10} = 27_{10}$

▶ De plus

$$\left[\begin{array}{r} 27 \\ \times 27 \\ \hline 189 \\ + 540 \\ \hline 729 \end{array} \right]_{10} \quad \text{et} \quad \left[\begin{array}{r} 123 \\ \times 123 \\ \hline 1101 \\ + 3120 \\ + 12300 \\ \hline 23121 \end{array} \right]_4$$

- ▶ Vérification : $23121_4 = (512 + 192 + 16 + 8 + 1)_{10} = 729_{10}$

Le code ASCII

- ▶ **Codage** adopté comme standard dans les années 60 permettant de représenter les caractères sur 8 bits, soit 256 caractères possibles
- ▶ **ASCII** : « American Standard Code for Information Interchange » traduisez « Code Américain Standard pour l'Échange d'Informations »
- ▶ Grossièrement
 - ▶ Les codes 0 à 31 sont des **caractères de contrôle** ; ils permettent de faire des actions t.q. le retour à la ligne ou le bip (sonore ?)
 - ▶ Les codes 65 à 90 représentent **les majuscules**
 - ▶ Les codes 97 à 122 représentent **les minuscules** (il suffit donc de modifier le 6^{ème} bit pour passer de majuscules à minuscules, c'est-à-dire ajouter 32 au code ASCII en base décimale)

Histoire de l'informatique

La « préhistoire » (-3000 à 1940)

Les premiers pas de l'informatique (1941 à 1968)

D'autres dates et personnages importants

Architecture des ordinateurs

Généralités d'ordinateurs

Codage de l'information

Architecture de von Neumann

La structure matérielle d'un ordinateur

Systèmes d'exploitation

Définition

Processus

Mémoire

Entrées/Sorties

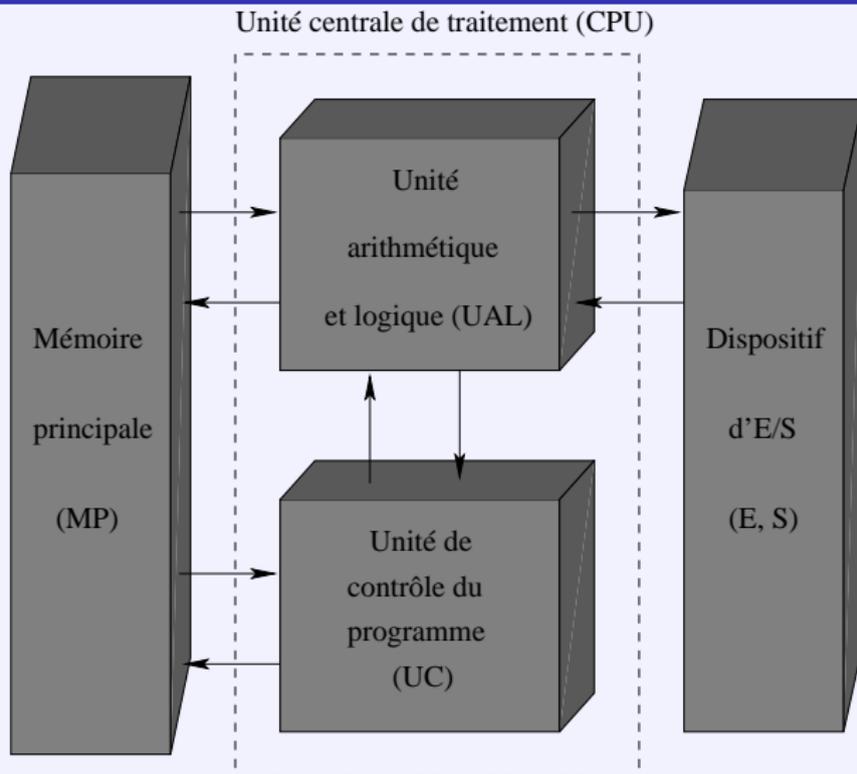
Systèmes de fichiers

Caractéristiques de l'IAS

La machine de von Neumann (EDVAC) \Rightarrow IAS (Institute for Advanced Studies) est composée de

1. une **mémoire principale** (MP) qui contient données et instructions
2. une **unité arithmétique et logique** (UAL) capable de fonctionner sur des données binaires
3. une **unité de contrôle** (UC) qui interprète les instructions en mémoire et en entraîne l'exécution
4. un **dispositif d'entrée et de sortie** (E, S) pris en charge par l'unité de contrôle

Structure de l'ordinateur IAS



Histoire de l'informatique

La « préhistoire » (-3000 à 1940)

Les premiers pas de l'informatique (1941 à 1968)

D'autres dates et personnages importants

Architecture des ordinateurs

Généralités d'ordinateurs

Codage de l'information

Architecture de von Neumann

La structure matérielle d'un ordinateur

Systèmes d'exploitation

Définition

Processus

Mémoire

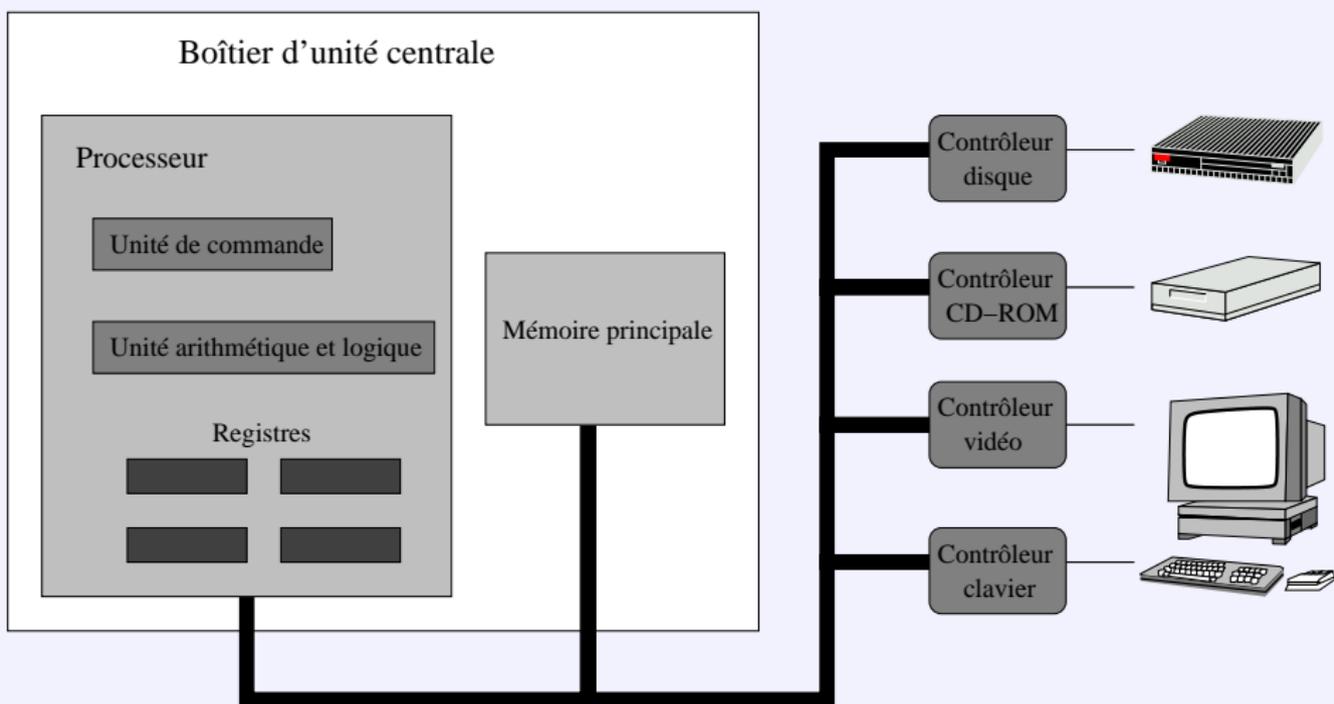
Entrées/Sorties

Systèmes de fichiers

Définitions

- ▶ Registres : mémoire haute vitesse qui se trouve dans le processeur
- ▶ Processeur : réunion de l'UAL, de l'UC et de registres
- ▶ Disque magnétique : plateau circulaire à face simple ou double équipé d'une surface magnétisable capable de stocker les données
- ▶ Unité centrale : boîte (tour par exemple) qui contient le processeur, la mémoire principale et le disque
- ▶ PC : réunion de l'unité centrale, du clavier et de la souris
- ▶ Instruction : traitement effectué à un instant donné par le système (action atomique)
- ▶ Programme : suite d'instructions effectuant un certain traitement (le nombre de programmes possibles est théoriquement infini)

Quelques composants d'un ordinateur personnel



Bus

La carte-mère

- ▶ **Élément constitutif principal** de l'ordinateur (en anglais « mainboard » ou « motherboard »)
- ▶ Socle permettant la connexion de l'ensemble des éléments essentiels de l'ordinateur
- ▶ Caractéristiques
 - ▶ le **facteur d'encombrement** : la géométrie, les dimensions, l'agencement et les caractéristiques électriques de la carte-mère
 - ▶ le **chipset** (traduisez jeu de composants ou jeu de circuits) : circuit électronique chargé de coordonner les échanges de données entre les divers composants
 - ▶ le **type de support de processeur** : de lui dépend le type de processeur
 - ▶ les **connecteurs d'entrée-sortie**

La carte-mère : composants intégrés

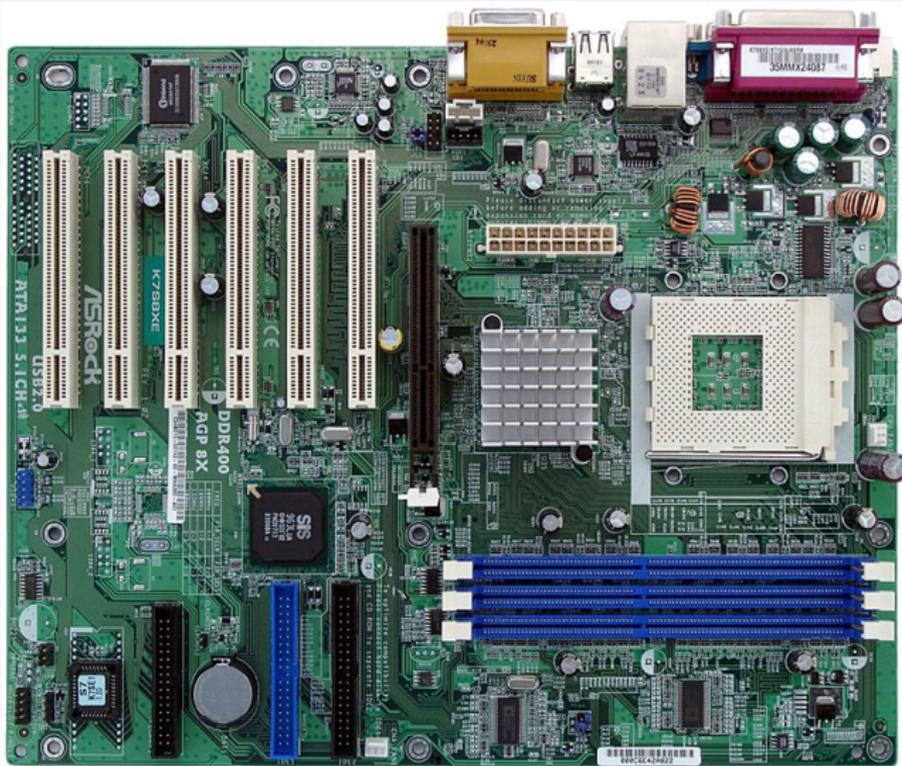
La carte-mère contient un certain nombre d'**éléments embarqués**, i.e. intégrés sur son circuit imprimé :

- ▶ Le **chipset** (ou pont) circuit qui contrôle la majorité des ressources
- ▶ L'horloge et la pile du CMOS
- ▶ Le **BIOS**
- ▶ Le bus système et les bus d'extension

En outre, les cartes-mères récentes embarquent généralement un certain nombre de **périphériques** multimédia et réseau pouvant être désactivés :

- ▶ carte réseau intégrée
- ▶ carte graphique intégrée
- ▶ carte son intégrée
- ▶ contrôleurs de disques durs évolués

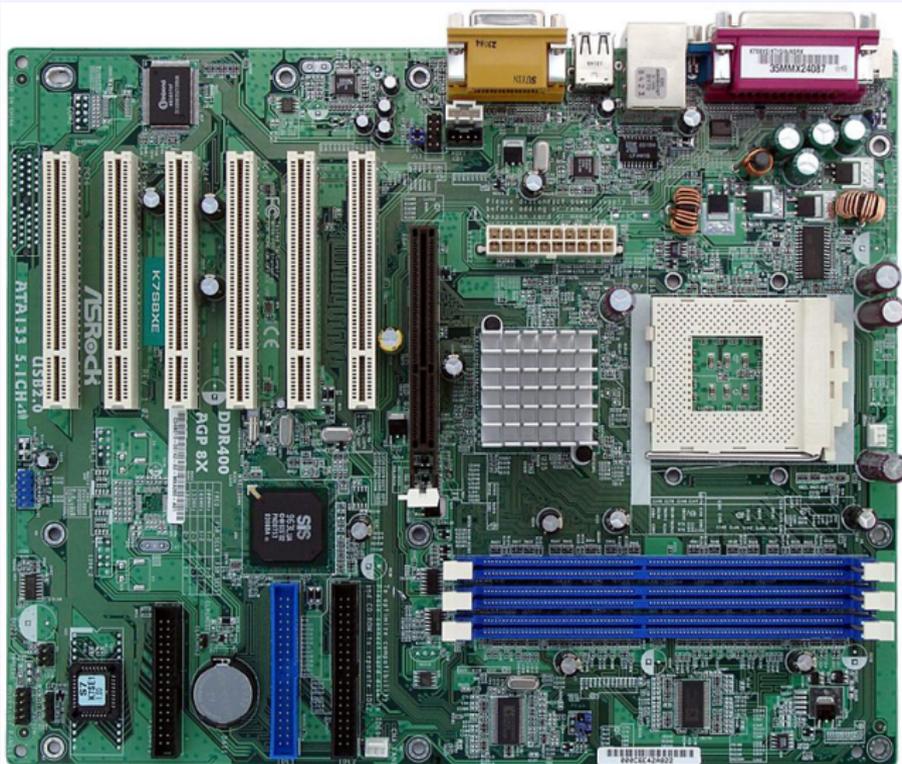
Concrètement : une carte-mère



L'horloge et la pile du CMOS

- ▶ L'horloge
 - ▶ **Horloge temps réel** (RTC = *Real Time Clock*) : circuit chargé de la synchronisation des signaux du système
 - ▶ Elle est constituée d'un **crystal de quartz** qui, en vibrant, donne des impulsions (**tops d'horloge**) afin de cadencer le système
 - ▶ **Fréquence de l'horloge** (exprimée en MHz) : nombre de vibrations du cristal par seconde, i.e. nombre de tops d'horloge émis par seconde
- ▶ Pile du CMOS
 - ▶ **CMOS** (*Complementary Metal-Oxyde Semiconductor*, ou BIOS CMOS) : circuit électronique qui conserve certaines informations sur le système t.q. l'heure, la date système et quelques paramètres essentiels du système
 - ▶ Le CMOS est continuellement alimenté par une **pile** (au format pile bouton) ou une batterie située sur la carte-mère

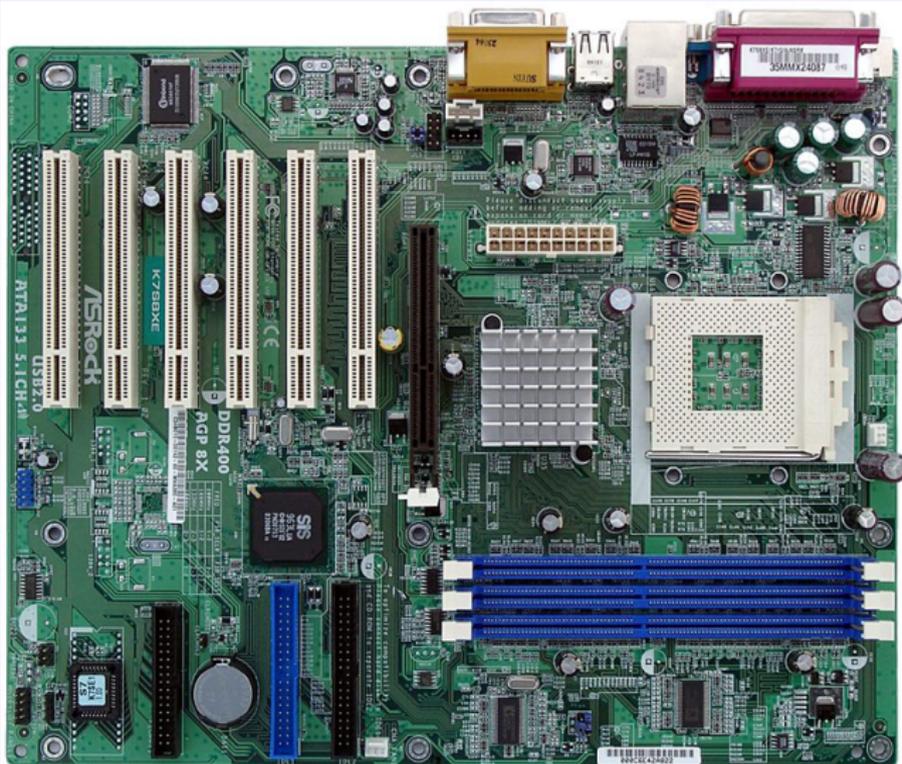
L'horloge et la pile du CMOS sur la carte-mère



Le BIOS

- ▶ **BIOS** (*Basic Input/Output System*) : programme basique servant d'interface entre le SE et la carte-mère
- ▶ Le BIOS est stocké
 - ▶ dans une **ROM** (mémoire morte) et utilise les données contenues dans le CMOS pour connaître la configuration matérielle du système
 - ▶ dans une **EEPROM** (mémoire modifiable par impulsions électriques, *flasher* = action de modifier l'EEPROM)
- ▶ Lorsque le système est mis sous-tension ou réamorcé (*Reset*), le BIOS fait l'inventaire du matériel présent dans l'ordinateur et effectue un test appelé **POST** (*Power-On Self Test*) afin de vérifier son bon fonctionnement
- ▶ La plupart des BIOS ont un « *setup* » (programme de configuration) qui permet de modifier la configuration basique du système

Le BIOS sur la carte-mère



Le processeur

- ▶ **Processeur** (CPU = *Central Processing Unit* soit Unité Centrale de Traitement) : cerveau de l'ordinateur cadencé au rythme d'une horloge interne (e.g. un ordinateur à 200 MHz possède une horloge envoyant 200 millions de battements par seconde)
- ▶ A chaque top d'horloge le processeur exécute une action, correspondant à une instruction ou une partie d'instruction
- ▶ L'indicateur ou **CPI** (Cycles Par Instruction) permet de représenter le nombre moyen de cycles d'horloge nécessaire à l'exécution d'une instruction
- ▶ La puissance du processeur = nombre d'instructions qu'il est capable de traiter par seconde exprimé en **MIPS** (Millions d'Instructions Par Seconde) =
$$\frac{\text{fréquence du processeur}}{\text{CPI}}$$

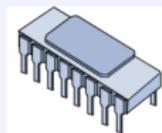
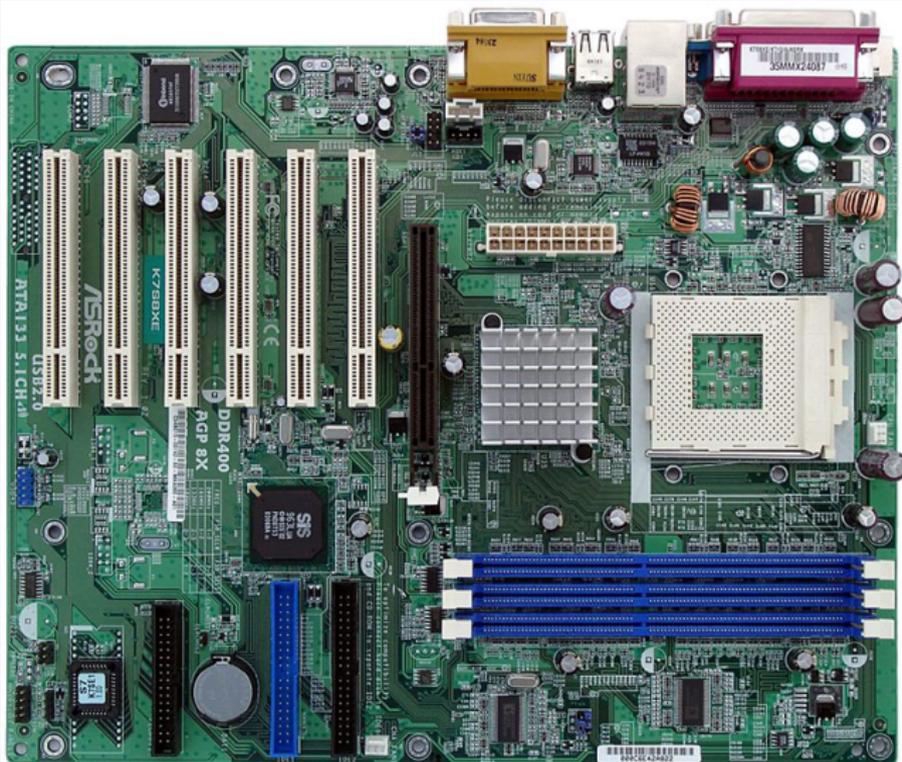


Figure: Intel 4004

Le processeur : unités fonctionnelles

- ▶ **Unité de contrôle** (*control unit*) : lit les données arrivant, les décode puis les envoie à l'unité d'exécution
 - ▶ séquenceur (ou bloc logique de commande) : synchronise l'exécution des instructions au rythme d'une horloge (envoi des signaux de commande)
 - ▶ compteur ordinal : contient l'adresse de la prochaine instruction à traiter
 - ▶ registre d'instruction : contient l'instruction en cours de traitement
- ▶ **Unité d'exécution** (ou unité de traitement) : accomplit les tâches que lui a données l'unité de contrôle
 - ▶ unité arithmétique et logique (UAL)
 - ▶ unité de virgule flottante (FPU = *Floating Point Unit*) : accomplit les calculs complexes non entiers que ne peut réaliser l'UAL
 - ▶ registre d'état : stocke des indicateurs sur l'état du système
 - ▶ registre accumulateur : stocke les résultats des opérations
- ▶ **Unité de gestion des bus** (ou unité d'E/S) : gère les flux d'informations entrant et sortant, en interface avec la mémoire vive du système

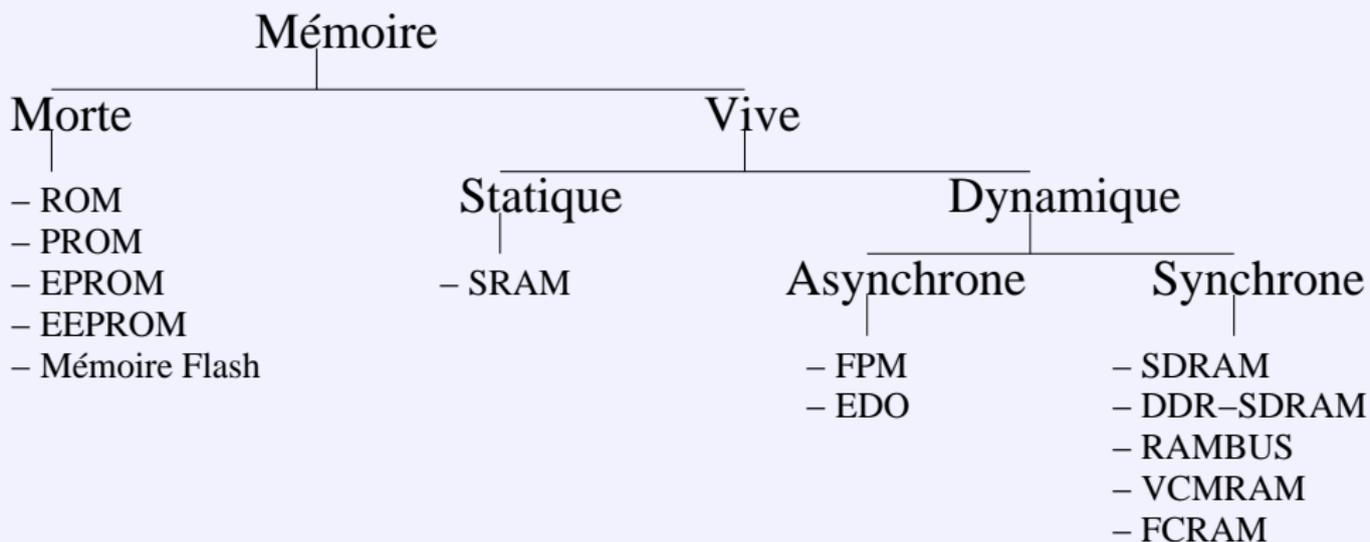
Le processeur sur la carte-mère



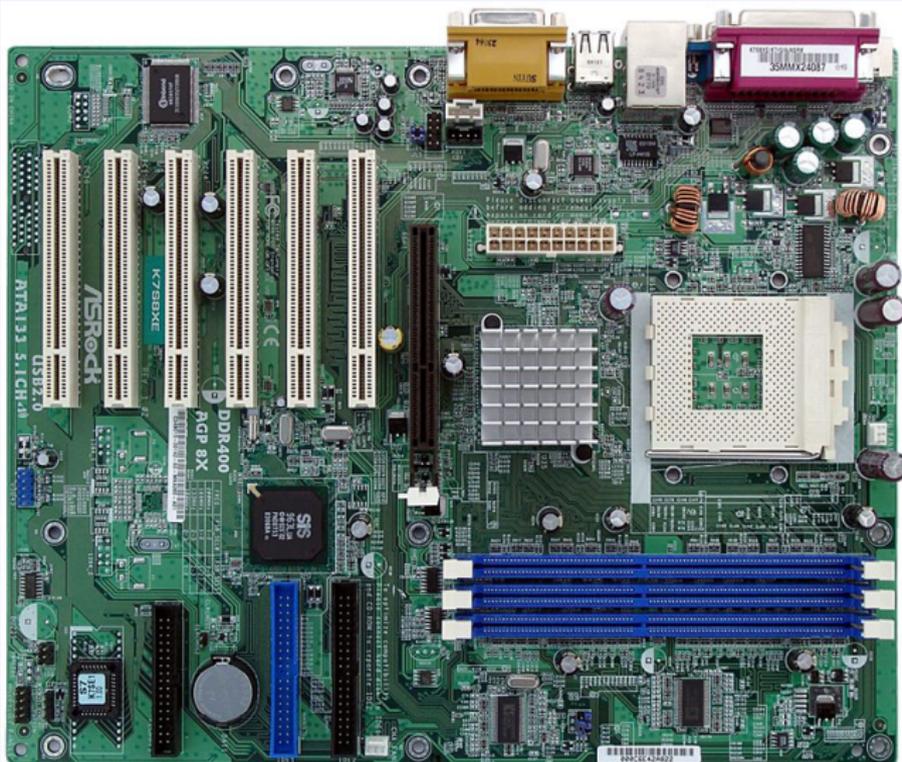
La mémoire

- ▶ Catégories de mémoires
 - ▶ la **mémoire centrale** (ou mémoire interne) permettant de mémoriser temporairement les données lors de l'exécution des programmes
 - ▶ la **mémoire de masse** (ou mémoire physique ou mémoire externe) permettant de stocker des informations à long terme
- ▶ Caractéristiques techniques
 - ▶ La **capacité** : volume global d'informations (en bits) que la mémoire peut stocker
 - ▶ Le **temps d'accès** : intervalle de temps entre la demande de lecture/écriture et la disponibilité de la donnée
 - ▶ Le **temps de cycle** : intervalle de temps minimum entre deux accès successifs
 - ▶ Le **débit** : volume d'information échangé par unité de temps, exprimé en bits par seconde
 - ▶ La **non volatilité** : l'aptitude d'une mémoire à conserver les données lorsqu'elle n'est plus alimentée électriquement

La mémoire (suite et fin)



La mémoire sur la carte-mère



Les périphériques d'E/S

Périphériques d'entrée

- ▶ clavier
- ▶ scanner

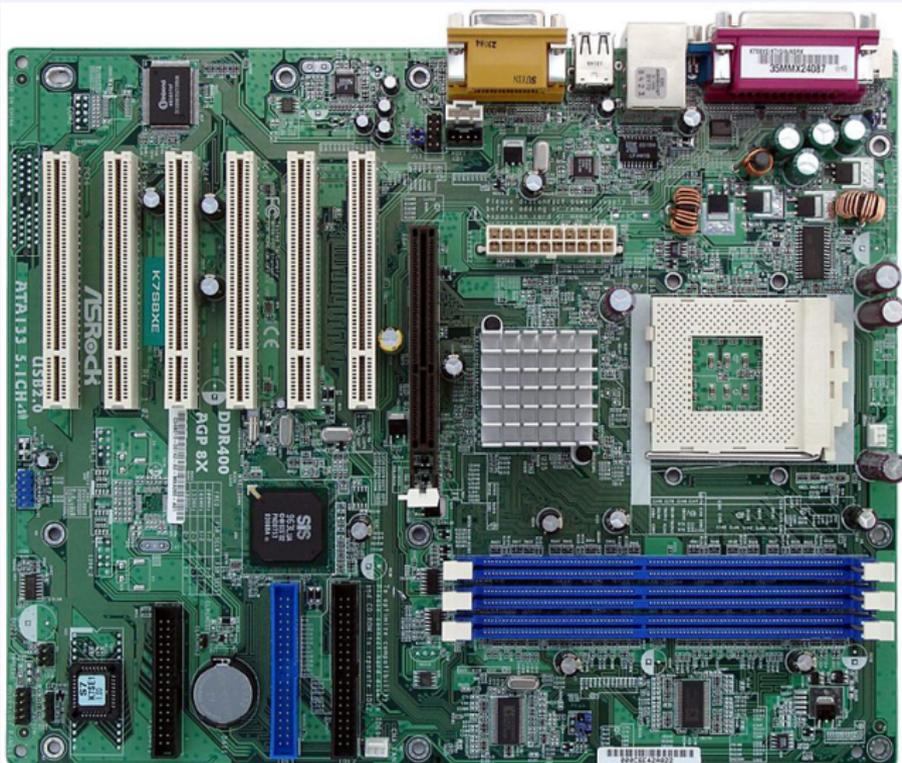
Périphériques de sortie

- ▶ écran
- ▶ imprimante

Périphériques d'entrée/sortie

- ▶ disque dur
- ▶ carte son

Les périphériques d'E/S sur la carte-mère

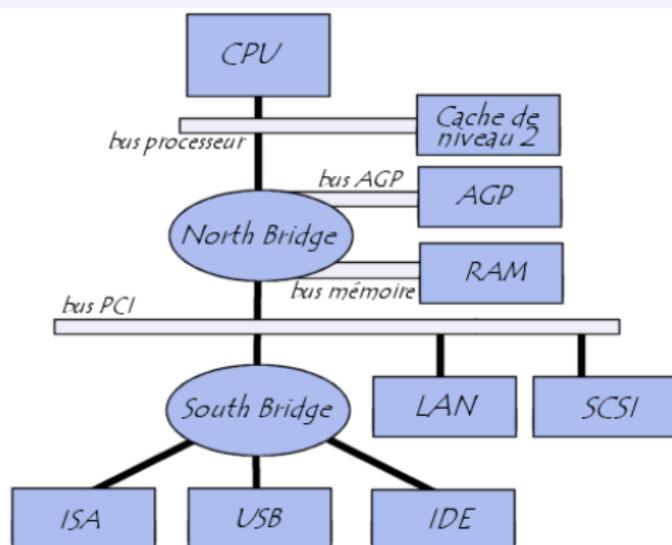


Les bus

- ▶ **Caractéristiques**
 - ▶ De sa largeur en bits et de sa fréquence dépend sa rapidité de la communication entre les unités de l'ordinateur
 - ▶ Un bus connecte l'unité centrale à sa mémoire principale (accès bus) et à la mémoire résidant sur les unités de contrôle des périphériques
 - ▶ Un bus permet de transférer des données entre la carte-mère et les périphériques qui s'y connectent
- ▶ Un **bus d'extension** permet d'étendre un système PC à l'aide de cartes, en permettant aux données de circuler entre la carte et l'unité centrale, e.g.
 - ▶ AGP pour les cartes graphiques
 - ▶ PCI pour les cartes d'extension
 - ▶ USB
 - ▶ Firewire
 - ▶ PCI-Express

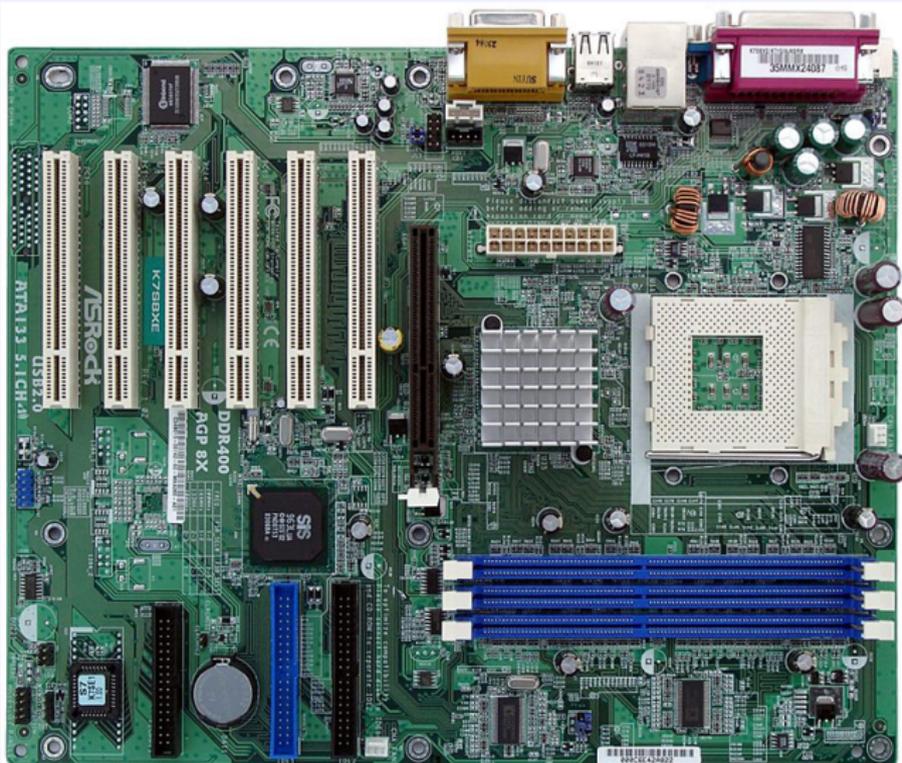
Les bus et le chipset

- ▶ **Chipset** (i.e. jeu de composants) : élément chargé d'aiguiller les informations entre les différents bus de l'ordinateur afin de permettre à tous les éléments constitutifs de l'ordinateur de communiquer entre eux
- ▶ Deux éléments sur la carte-mère

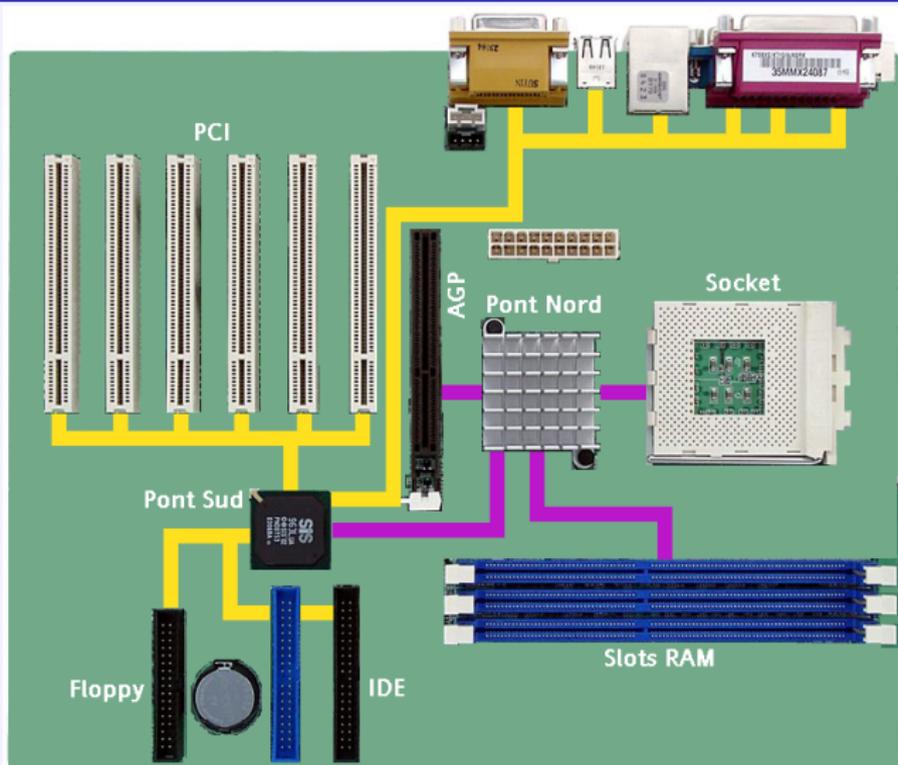


- ▶ **Pont Nord** : contrôle les échanges entre le processeur et la mémoire vive
- ▶ **Pont Sud** : gère les communications avec les périphériques d'entrée-sortie

Les bus sur la carte-mère



Les bus sur la carte-mère



Citation



*« Le capitalisme, c'est l'exploitation de
l'homme par l'homme ;
et le marxisme, c'est le contraire. »*

Henri Jeanson

Objectif

Pour qu'un ordinateur soit capable de faire fonctionner un programme informatique (appelé parfois application ou logiciel), la machine doit être en mesure d'effectuer un certain nombre d'opérations préparatoires afin d'assurer les échanges entre le processeur, la mémoire, et les périphériques

Système bancaire	Réservations aériennes	Navigateur Web	} Application
Compilateurs	Éditeurs	Interpréteur de commandes	
Système d'exploitation			} Programme système
Langage machine			
Micro-architecture			} Matériels
Périphériques physiques			

Histoire de l'informatique

La « préhistoire » (-3000 à 1940)

Les premiers pas de l'informatique (1941 à 1968)

D'autres dates et personnages importants

Architecture des ordinateurs

Génération d'ordinateurs

Codage de l'information

Architecture de von Neumann

La structure matérielle d'un ordinateur

Systèmes d'exploitation

Définition

Processus

Mémoire

Entrées/Sorties

Systèmes de fichiers

Définition

Définition (*Système d'exploitation*)

- ▶ Le système d'exploitation (noté SE ou OS, abréviation du terme anglais Operating System), est chargé d'assurer la liaison entre les ressources matérielles, l'utilisateur et les applications
- ▶ Deux tâches
 - ▶ Fournir à l'utilisateur une machine étendue ou virtuelle, plus simple à programmer
 - ▶ Gestion des ressources. Deux dimensions du partage (multiplexage)
 - ▶ temps
 - ▶ espace

Histoire de l'informatique

La « préhistoire » (-3000 à 1940)

Les premiers pas de l'informatique (1941 à 1968)

D'autres dates et personnages importants

Architecture des ordinateurs

Génération d'ordinateurs

Codage de l'information

Architecture de von Neumann

La structure matérielle d'un ordinateur

Systèmes d'exploitation

Définition

Processus

Mémoire

Entrées/Sorties

Systèmes de fichiers

Définition

Définition (*Processus*)

- ▶ C'est un programme en cours d'exécution
- ▶ Chaque processus possède
 - ▶ un espace d'adressage qui contient
 - ▶ le programme executable
 - ▶ ses données
 - ▶ sa pile
 - ▶ un ensemble de registres dont
 - ▶ le compteur ordinal
 - ▶ le pointeur de pile
 - ▶ d'autres registres matériels et informations nécessaires

Pseudo-parallélisme

- ▶ Les ordinateurs sont capables de faire plusieurs choses en même temps
- ▶ Le processeur bascule constamment d'un processus à l'autre : **multiprogrammation**
- ▶ Différence processus / programme

Création d'un nouveau processus

Evénements conduisant à la création d'un nouveau processus

- ▶ Initialisation du système
- ▶ Exécution d'un appel système de création de processus par un processus en cours
- ▶ Requête utilisateur sollicitant la création d'un nouveau processus
- ▶ Initiation d'un travail en traitement par lots

Fin d'un processus

- ▶ Arrêt normal (volontaire) (exit)
- ▶ Arrêt pour erreur (volontaire)
- ▶ Arrêt pour erreur fatale (involontaire)
- ▶ Le processus est arrêté par un autre processus (involontaire) (kill)

Hierarchisation des processus

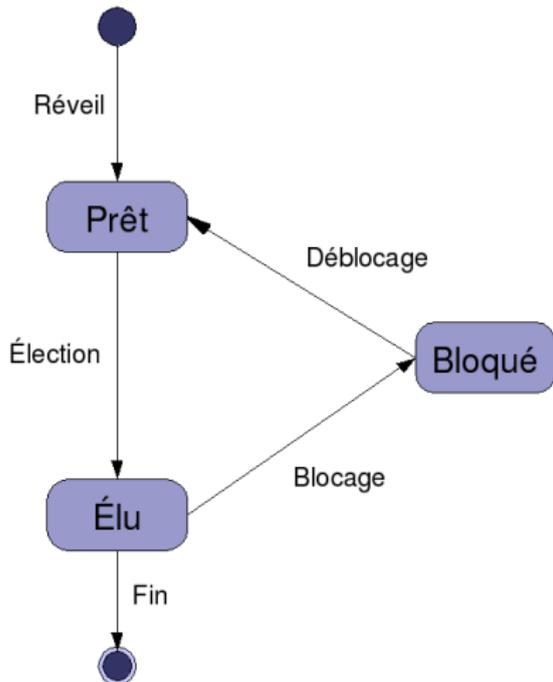
- ▶ Pas le cas sous Windows
- ▶ Sous UNIX
 - ▶ lorsqu'un processus en crée un autre, le père et l'enfant continuent d'être associés
 - ▶ l'enfant peut lui-même créer d'autres processus
 - ▶ formation d'une hiérarchie de processus (init)
 - ▶ Un processus et l'ensemble de ses descendants sont appelés un groupe de processus

États

Un processus peut prendre un de ces 3 états

- ▶ En cours d'exécution (le programme utilise le processeur)
- ▶ Prêt (exécutable, temporairement arrêté pour laisser un autre processus)
- ▶ Bloqué (ne peut pas s'exécuter tant qu'un événement externe ne se produit pas)

Le passage de *En cours* à *Prêt* et inversement est géré par l'**ordonnanceur** de processus



Histoire de l'informatique

La « préhistoire » (-3000 à 1940)

Les premiers pas de l'informatique (1941 à 1968)

D'autres dates et personnages importants

Architecture des ordinateurs

Génération d'ordinateurs

Codage de l'information

Architecture de von Neumann

La structure matérielle d'un ordinateur

Systèmes d'exploitation

Définition

Processus

Mémoire

Entrées/Sorties

Systèmes de fichiers

Gestion de la mémoire

- ▶ Hiérarchisation de la mémoire (cache, RAM, disque dur)
- ▶ Coordination de la manière dont sont utilisées les différentes mémoires
- ▶ Rôle du gestionnaire de mémoire
 - ▶ conserver la trace de la mémoire en cours d'utilisation ou pas
 - ▶ allouer la mémoire aux processus qui en ont besoin
 - ▶ gérer le va-et-vient (swapping) entre mémoire principale et disque

Monoprogrammation

Exemple

MS DOS : Un seul processus en mémoire à la fois

En pratique

- ▶ Partie de l'espace d'adressage réservée au système d'exploitation (ROM + SE chargé au démarrage)
- ▶ À la fin du programme, retour à l'interpréteur de commande qui demande le prochain programme à lancer

Multiprogrammation

- ▶ Facilite le développement de programmes en les fractionnant en processus indépendants
- ▶ Elle permet une maximisation de l'utilisation des ressources processeur
- ▶ Problème : comment organiser la memoire de la façon la plus efficace possible ?

Multiprogrammation avec partitions fixes

- ▶ Division de la mémoire en partitions (si possible inégales)
- ▶ Quand une tâche arrive, elle est placée dans une file d'attente
 - ▶ une file d'attente par partition de mémoire
 - ▶ une seule file d'attente pour toutes les partitions
- ▶ Réallocation
- ▶ Protection

Va et vient

- ▶ Mémoire insuffisante pour contenir tous les processus courants
- ▶ Nécessité de placer certains de ces processus sur le disque

Définition (*Va-et-vient* ou *swapping*)

Ramener régulièrement des processus sur le disque en mémoire centrale et inversement

Mémoire virtuelle

- ▶ La taille de l'ensemble formé par le programme, les données et la pile peut dépasser la capacité de mémoire disponible
- ▶ Le SE conserve les parties de programme en cours d'utilisation dans la mémoire principale, et le reste sur le disque
- ▶ La mémoire virtuelle permet :
 - ▶ d'augmenter le taux de multiprogrammation
 - ▶ de mettre en place des mécanismes de protection de la mémoire
 - ▶ de partager la mémoire entre processus

Histoire de l'informatique

La « préhistoire » (-3000 à 1940)

Les premiers pas de l'informatique (1941 à 1968)

D'autres dates et personnages importants

Architecture des ordinateurs

Génération d'ordinateurs

Codage de l'information

Architecture de von Neumann

La structure matérielle d'un ordinateur

Systèmes d'exploitation

Définition

Processus

Mémoire

Entrées/Sorties

Systèmes de fichiers

Entrées/Sorties

Le SE a la tâche importante de contrôler les périphériques d'entrées/sorties (E/S)

- ▶ Fonctions

- ▶ Emission des commandes vers les périphériques
- ▶ Interception des interruptions
- ▶ Gestion des erreurs

- ▶ But

- ▶ Fournir une interface simple entre les périphériques et le système
- ▶ Interface identique pour tous les périphériques

Les unités d'entrées/sorties

Deux catégories

- ▶ périphériques par bloc : informations stockées par blocs de taille fixe, chacun possédant sa propre adresse (e.g. : disque)
- ▶ périphériques par caractères : l'information circule sous la forme d'un flot de caractères, sans aucune structure de bloc (e.g. : clavier, imprimante, souris)

Deux parties dans une unité

- ▶ un composant mécanique, le périphérique (e.g. : disque)
- ▶ un composant électronique, le **contrôleur de périphérique** (e.g. : contrôleur IDE)

Communication

- ▶ Interface entre contrôleur et périphérique de très bas niveau
- ▶ Le contrôleur possède des registres qui permettent la communication avec le processeur
 - ▶ Écriture dans ces registres : le SE ordonne au périphérique de délivrer des données, d'en accepter ou d'effectuer une action donnée
 - ▶ Lecture : le SE peut connaître l'état du périphérique, savoir s'il est capable d'accepter une nouvelle commande
- ▶ Certains périphériques sont équipés d'un tampon de données que le SE peut lire ou écrire

Les interruptions

Pour permettre au processeur de réaliser d'autres opérations pendant qu'il attend la réalisation d'une E/S, on fait appel aux **interruptions**

L'accès direct à la mémoire (DMA)

- ▶ Disponible uniquement s'il y a un contrôleur DMA
- ▶ Le contrôleur DMA a accès au bus système sans dépendre du processeur → E/S programmée qui fait le travail du processeur
- ▶ Réduit le nombre d'interruptions

Les disques magnétiques

- ▶ Organisation en cylindres
- ▶ Chaque cylindre contient autant de pistes que de têtes empilées verticalement
- ▶ Les pistes sont divisées en secteurs

On appelle cette organisation **géométrie**

Sur les disques durs actuels, la géométrie spécifiée peut être différente du format physique réel

Histoire de l'informatique

La « préhistoire » (-3000 à 1940)

Les premiers pas de l'informatique (1941 à 1968)

D'autres dates et personnages importants

Architecture des ordinateurs

Génération d'ordinateurs

Codage de l'information

Architecture de von Neumann

La structure matérielle d'un ordinateur

Systèmes d'exploitation

Définition

Processus

Mémoire

Entrées/Sorties

Systèmes de fichiers

Stockage à long terme d'informations

- ▶ Enregistrement d'une grande quantité d'informations
- ▶ Informations conservées après la fin du processus qui les utilise (persistance)
- ▶ Plusieurs processus doivent pouvoir avoir accès simultanément à une information

Fichiers

- ▶ Mécanisme d'abstraction (l'utilisateur ne voit pas où et comment sont stockées les informations)
- ▶ Subdivision des fichiers par types en fonction de leur nature
 - ▶ typage fort : le type de fichier est défini par son extension (MS DOS)
 - ▶ typage déduit : les extensions des fichiers ne sont qu'indicatives, le système détermine la nature du fichier par inspection du contenu (UNIX)

Catalogues

- ▶ Nommés aussi **répertoires** ou **dossiers**
- ▶ Système à repertoires hiérarchiques
 - ▶ permet regroupement logique des fichiers
 - ▶ notion de chemin d'accès
 - ▶ chemin d'accès absolu (depuis la racine)
 - ▶ chemin d'accès relatif (depuis le répertoire courant)

Système de fichiers

- ▶ Disques divisés en partitions pouvant contenir différents systèmes de fichiers
- ▶ Secteur 0 du disque = Master Boot Record (MBR) qui comprend la table de partitions ; boot sur la partition marquée comme active
- ▶ L'organisation d'une partition varie fortement d'un système de fichiers à un autre ; cependant, présence d'un bloc de boot et souvent d'un superbloc qui contient les informations sur le type de système de fichiers
- ▶ Différentes méthodes d'implantation des fichiers (allocation contigüe, listes chaînées, ...).