Chapitre 1: Systèmes ordinés

Présentation pour

Java Software Solutions

Foundations of Program Design

Deuxième Edition

par John Lewis et William Loftus

Java Software Solutions est publié par Addison-Wesley

Focus du cours

- **Développement orienté-objet de logiciel**
  - **Résolution de problèmes**
  - **Implantation et design de programmes**
  - **Concepts orientés-objet**
    - **objets**
    - **classes**
    - **interfaces**
    - **héritage**
    - **polymorphisme**
  - **Graphique et interfaces usager graphiques**
  - **Le langage de programmation Java**

Systèmes ordinés

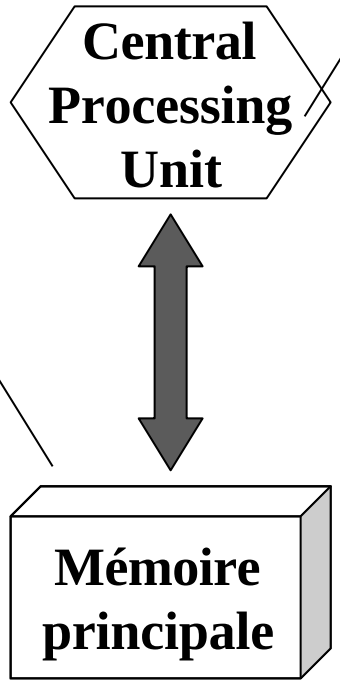
- **Nous devons en premier explorer les fondements du traitement informatisé**

- **Chapitre 1 se concentre sur :**
  - **Composantes d'un ordinateur**
  - **Interactions entre ces composantes**
  - **Stockage et manipulation d'information par l'ordinateur**
  - **Réseaux d'ordinateurs**
  - **Internet et le World-Wide Web**
  - **Programmation et les langages de programmation**
  - **Systèmes graphiques**

Hardware et Software

- **Hardware**
  - **Les parties physiques d'un ordinateur**
  - **clavier, moniteur, cables, chips**
- **Software**
  - **programmes et données**
  - **un *programme* est une série d'instructions**
- **Un ordinateur a besoin des deux: hardware et software**
- **Chacune est à proprement parler inutile sans l'autre**

CPU et mémoire principale



Processeur qui exécute les commandes du programme

Pentium 4 d'Intel
R12000 Mips
Processeur Sun Sparc

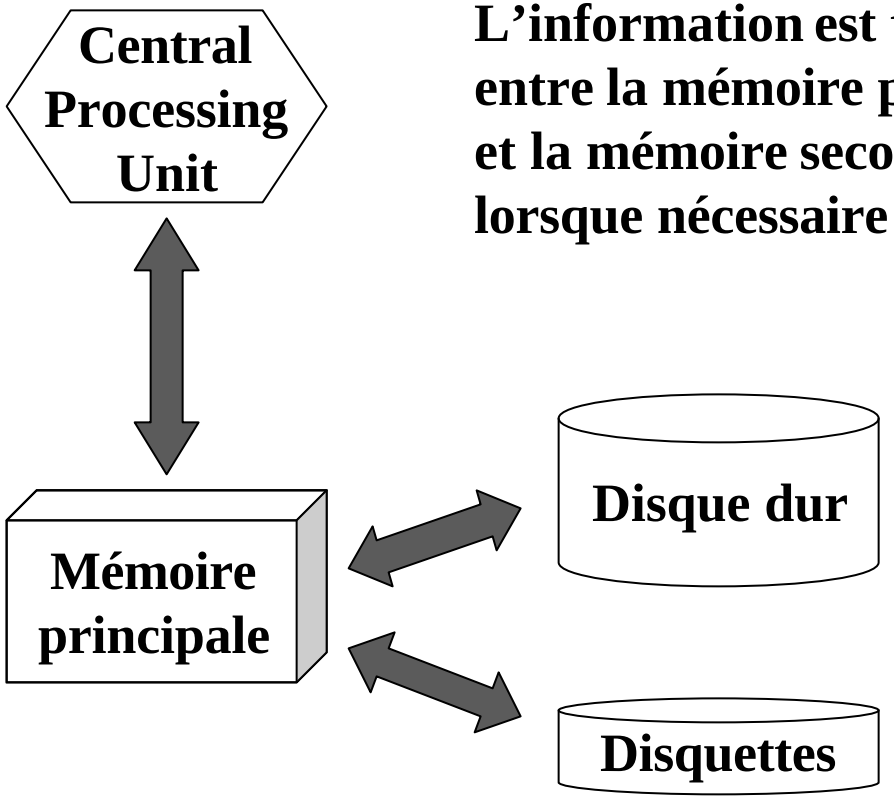
Stockage primaire pour les programmes et les données qui sont en utilisation active

Synonyme de RAM

Appareils de mémoire secondaire

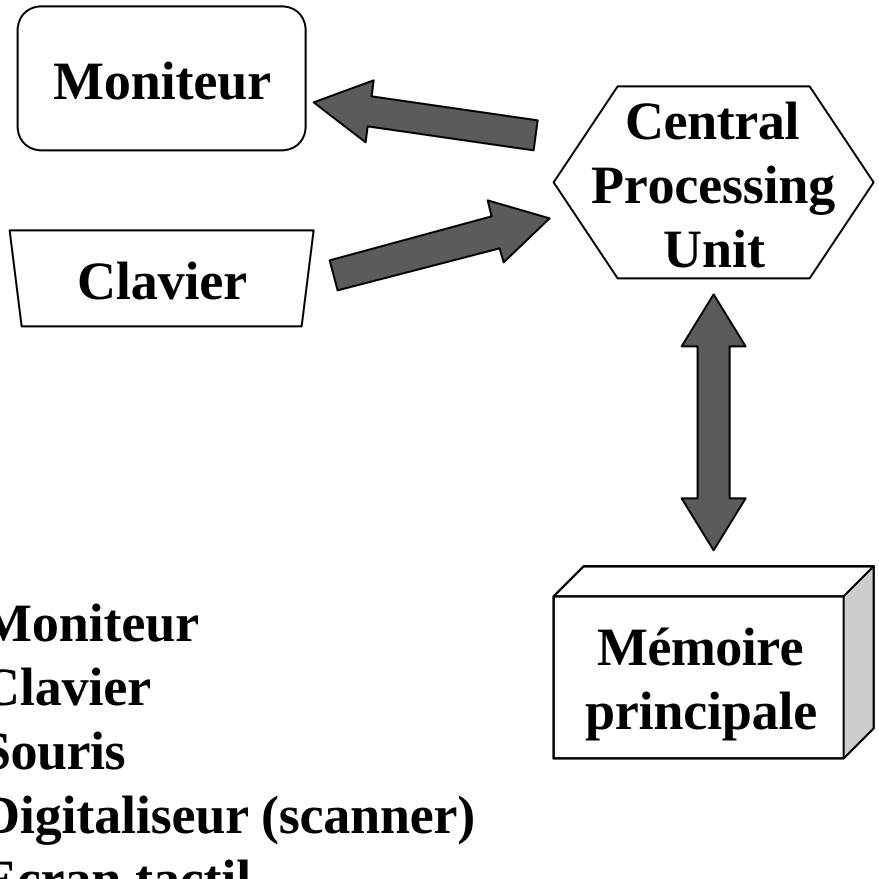
Les appareils de mémoire secondaire fournissent un stockage à long terme

L'information est transférée entre la mémoire principale et la mémoire secondaire lorsque nécessaire



Disques durs
Disquettes
Disquettes ZIP
CDs enregistrables
Cassettes

Appareils d'entrée / sortie (I/O)



Appareils I/O permettent une interaction avec l'usager

Moniteur
Clavier
Souris
Digitaliseur (scanner)
Ecran tactil

Catégories de software

- **Système d'exploitation**
  - **Contrôle toutes les activités de l'ordinateur**
  - **Fournit l'interface entre l'usager et l'ordinateur**
  - **Gère les ressources telles le CPU et la mémoire**
  - **Ex: Windows 98, Windows NT, Unix, Linux, Mac OS**
- **Programme d'application**
  - **Nom générique pour tout autre type de software**
  - **Ex: traitement de texte, compilateur/debugger, jeux**
- **La plupart des systèmes d'exploitation et des programmes d'application ont un interface usager graphique (GUI)**

Analogue vs. digital

- **Il existe deux façons de base de stocker et de traiter l'information :**

- ***Analogue***
  - **continue, en proportion directe avec les données représentées**
  - **musique sur un microsillon – une aiguille frotte dans le sillon qui est directement proportionnel au voltage envoyé au haut-parleur**

- ***Digital***
  - **L'information est brisée en morceaux, et chaque morceau est représenté séparément**
  - **musique sur un disque compact – le disque stocke les nombres qui représentent des niveaux spécifiques de voltages échantillonnés à différents niveaux**

Information digitale

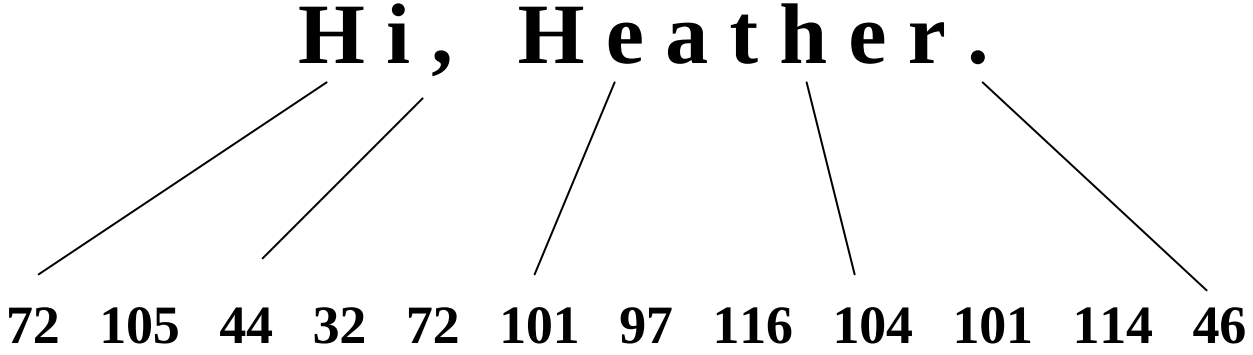
- **L'ordinateur stocke toute information sous forme digitale :**
  - **nombres**
  - **texte**
  - **graphiques et images**
  - **audio**
  - **vidéo**
  - **instructions d'un programme**

- **Toute information est *digitalisée* – i.e. séparée en morceaux et représentée par des nombres**

Représentation digitale de texte

- **Par exemple, chaque caractère est stocké comme un nombre, incluant les espaces, les chiffres et les caractères de ponctuation**

- **Les lettres majuscules et minuscules sont des caractères différents**



Nombres binaires

- **Une fois l'information digitalisée, elle est représentée et stockée en mémoire grâce à un *système de nombres binaires***
- **Un unique chiffre binaire (0 ou 1) est appelé un *bit***
- **Les appareils qui stockent et transfèrent l'information sont moins chers et plus fiables s'ils n'ont qu'à traiter deux états**
- **Un bit unique peut représenter deux états possibles, comme une ampoule électrique qui est allumée (1) ou éteinte (0)**
- **Des combinaisons de bits sont utilisées pour stocker des valeurs plus grandes**

Combinaisons de bits

1 bit	2 bits	3 bits	4 bits	
0	00	000	0000	1000
1	01	001	0001	1001
	10	010	0010	1010
	11	011	0011	1011
		100	0100	1100
		101	0101	1101
		110	0110	1110
		111	0111	1111

Chaque bit additionnel double le nombre de combinaisons possibles

Combinaisons de bits

- **Chaque combinaison peut représenter un item en particulier**
- **Il y a 2^N combinaisons avec N bits**
- **N bits sont requis pour représenter 2^N items uniquement**

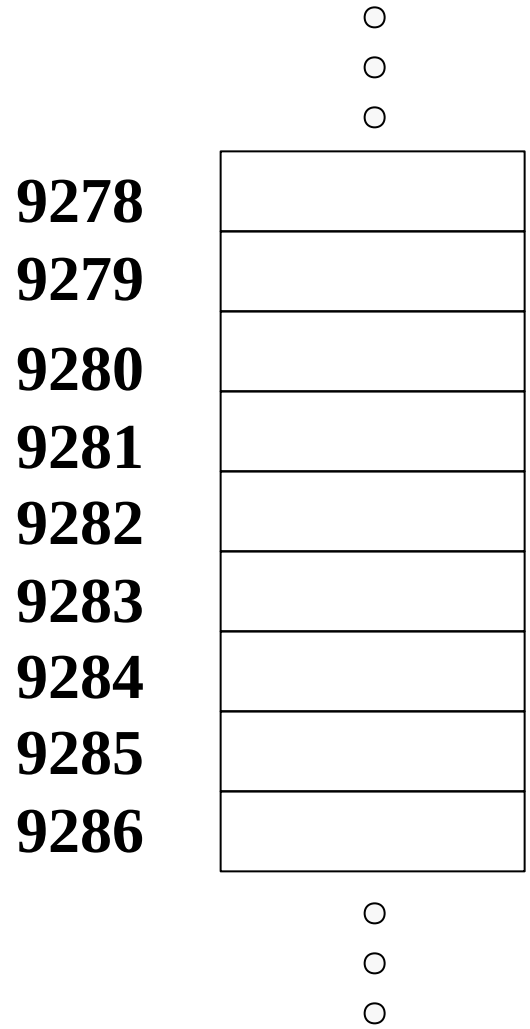
Combien d'items peuvent être représentés par

- **1 bit ?** $2^1 = 2$ **items**
- **2 bits ?** $2^2 = 4$ **items**
- **3 bits ?** $2^3 = 8$ **items**
- **4 bits ?** $2^4 = 16$ **items**
- **5 bits ?** $2^5 = 32$ **items**

Caractéristiques d'un ordinateur

- **Un ordinateur personnel pourrait avoir les caractéristiques suivantes :**
  - **Processeur de 600 MHz Pentium III**
  - **Mémoire de 256 MB RAM**
  - **Disque dur de 16 GB**
  - **CD ROM de vitesse 24x**
  - **Moniteur multimédia de 17" d'une résolution 1280 x 1024**
  - **Modem de 56 KB**
- **Qu'est-ce que tout ça signifie ?**

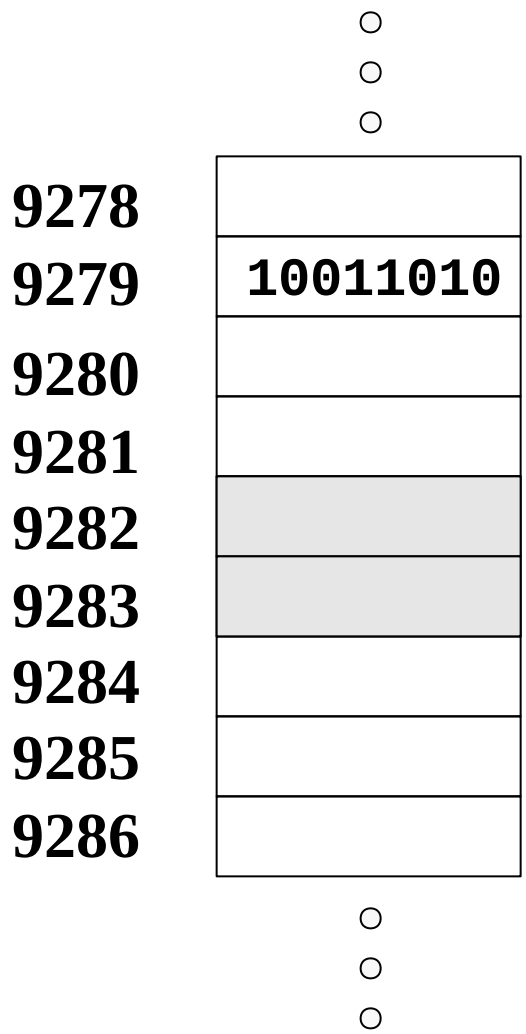
Mémoire



La mémoire principale est divisée en plusieurs régions mémoire (ou *cellules*)

Chaque cellule de mémoire a une *adresse* numérique, qui l'identifie uniquement

Stocker l'information



Chaque cellule mémoire stocke un ensemble de bits (habituellement 8 bits, ou un *octet* - *byte*)

Les valeurs plus grandes sont stockées dans des régions mémoire adjacentes

Capacité de stockage

- **Chaque appareil mémoire a une *capacité de stockage*, indiquant le nombre d'octets (bytes) il peut contenir**
- **Les capacités sont exprimées en diverses unités :**

Unité	Symbole	Nombre d'octets
kilooctet	**KO (*KB*)**	**2^{10} = 1024**
megaoctet	**MO (*MB*)**	**2^{20} (plus de 1 million)**
gigaoctet	**GO (*GB*)**	**2^{30} (plus de 1 billion)**
teraoctet	**TO (*TB*)**	**2^{40} (plus de 1 trillion)**

Mémoire

- **La mémoire principale est *volatile* - l'information stockée est perdue si l'électricité est interrompue**
- **Les appareils de mémoire secondaire sont *nonvolatils***

- **La mémoire principale et les disques sont des appareils à *accès direct* – l'information peut être accédée directement**
- **Les termes accès direct et accès aléatoire (random) sont souvent utilisés dans le même sens**
- **Un ruban magnétique est un appareil à accès séquentiel puisque ses données sont organisées dans un ordre linéaire – on doit lire toute l'information précédant l'information désirée pour y avoir accès**

RAM vs. ROM

- ***RAM*** **- Random Access Memory (accès direct)**
- ***ROM*** **- Read-Only Memory**

- **Les termes RAM et mémoire principale ont souvent la même signification**

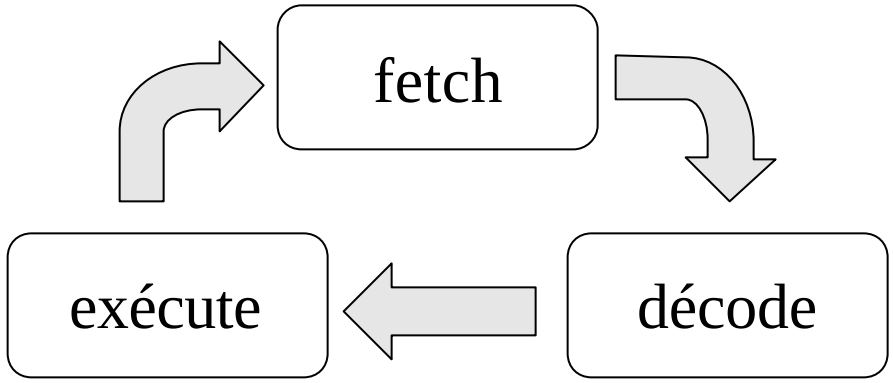
- **ROM peut apparaître sur une barrette mémoire, ou sur des appareils tels un CD ROM**

- **RAM et ROM sont des appareils à accès direct (random)!**
- **RAM pourrait aussi être appelé une mémoire Read-Write**

Le Central Processing Unit

- **Un CPU est aussi appelé un *microprocesseur***
- **Il suit continuellement le cycle *fetch-décode-exécute :***

Retrouve une instruction dans la mémoire principale

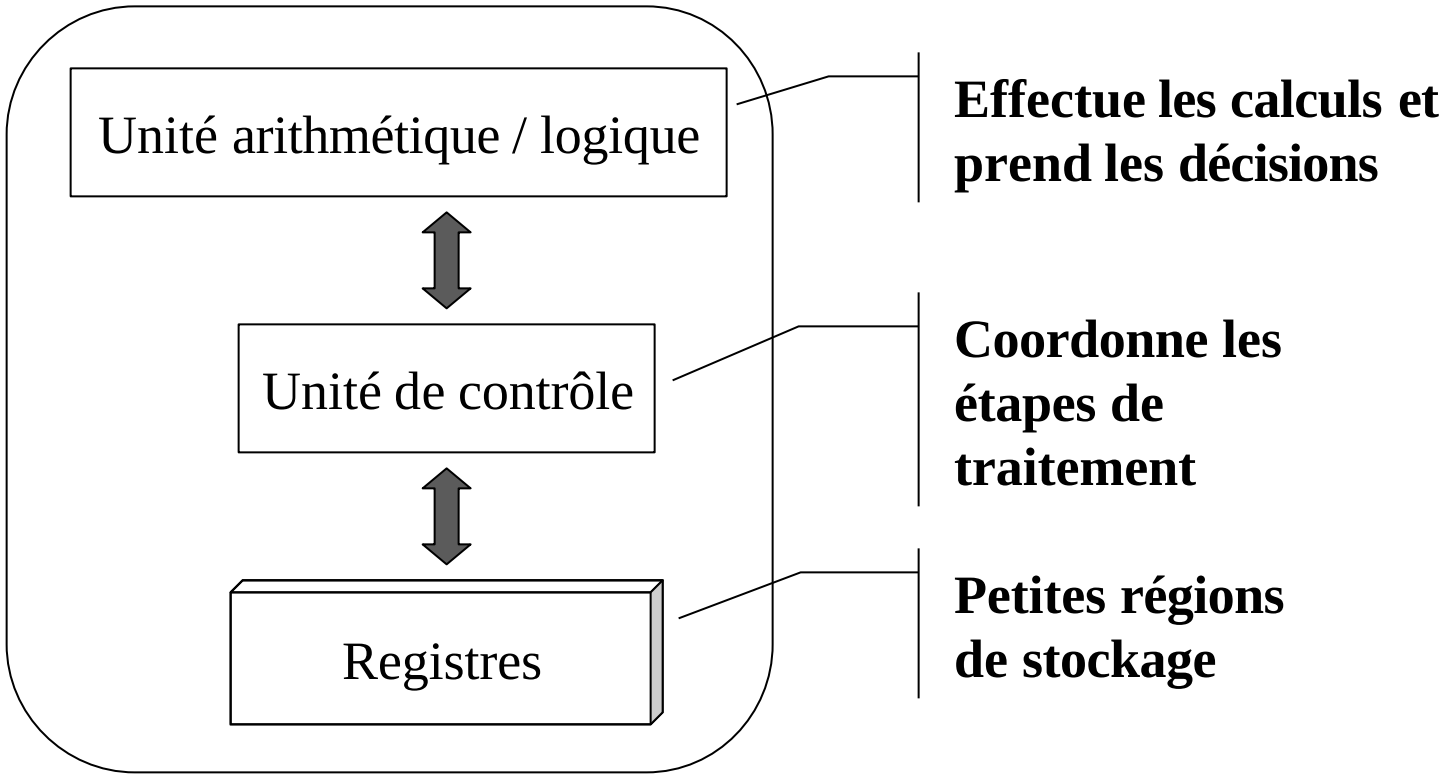


Exécute l'instruction

Identifie l'instruction

Le Central Processing Unit (CPU)

- **Le CPU contient:**



Le Central Processing Unit

- **La vitesse d'un CPU est contrôlée par *l'horloge du système***
- **L'horloge du système génère un pulse électronique à des intervals réguliers**
- **Les pulses coordonnent les activités du CPU**
- **La vitesse est mesurée en *megahertz* (MHz)**

Moniteur

- **La taille d'un moniteur (17") est mesurée diagonalement, comme pour un écran de télévision**
- **La plupart des moniteurs de nos jours ont des capacités *multimédia* : texte, graphique, vidéo, etc.**
- **Un moniteur a une certaine résolution maximale, indiquant le nombre d'éléments d'une image, appelés pixels, qui peuvent être affichés (tel 1280 par 1024)**
- **Une haute résolution (plus de pixels) produit des images plus précises**

Modem

- ***Les appareils de transfert de données*** **permettent d'envoyer et de recevoir de l'information entre ordinateurs**
- **Plusieurs ordinateurs sont équipés d'un *modem* qui permet de transmettre de l'information sur une ligne téléphonique**
- **Un appareil de transfert de données a un *taux maximum de transfert***
- **Un modem, par exemple, peut avoir un taux de transfert de 56,600 *bits par seconde* (bps)**

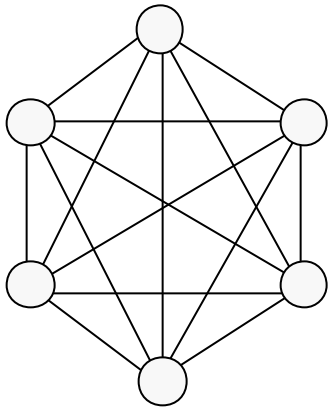
Réseaux

- **Un *réseau* est formé de deux ou plus ordinateurs interconnectés pour que les données et les ressources puissent être partagées**
- **La plupart des ordinateurs sont connectés d'une façon ou d'une autre à un type de réseau**
- **Chaque ordinateur a sa propre *adresse réseau* qui l'identifie uniquement parmi les autres**
- **Un *serveur de fichiers* est un ordinateur d'un réseau dédié au stockage des programmes et des données qui sont partagées par les usagers d'un réseau**

Connexions réseau

- **Chaque ordinateur dans un réseau pourrait être connecté à chaque autre ordinateur dans le réseau**
- **Ces connexions sont appelées *point-à-point***

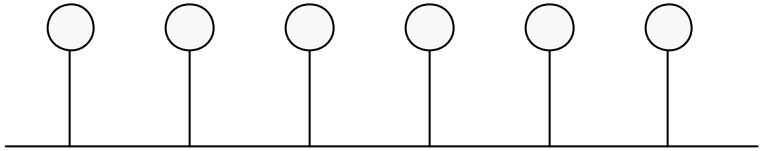
Ajouter un ordinateur requiert une nouvelle ligne de communication pour *chaque* ordinateur déjà dans le réseau



Cette technique n'est pas faisable pour plus de quelques machines rapprochées

Connexions réseau

- **La plupart des réseaux modernes partagent une seule ligne de communication**
- **Ainsi ajouter un nouvel ordinateur au réseau est relativement facile**

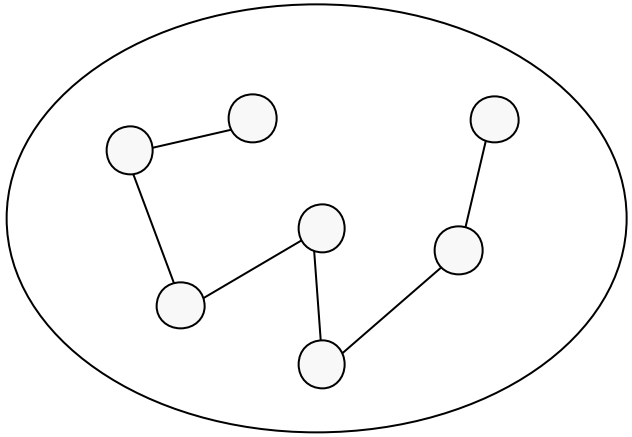


Les échanges sur le réseau doivent utiliser la ligne à tour de rôle, ce qui introduit des délais

Souvent l'information est brisée en parties, appelées *paquets*, envoyées à l'ordinateur récepteur qui les ré-assemblent

Réseau local *(local area)*

Un *réseau local* (LAN) couvre une petite distance et un petit nombre d'ordinateurs

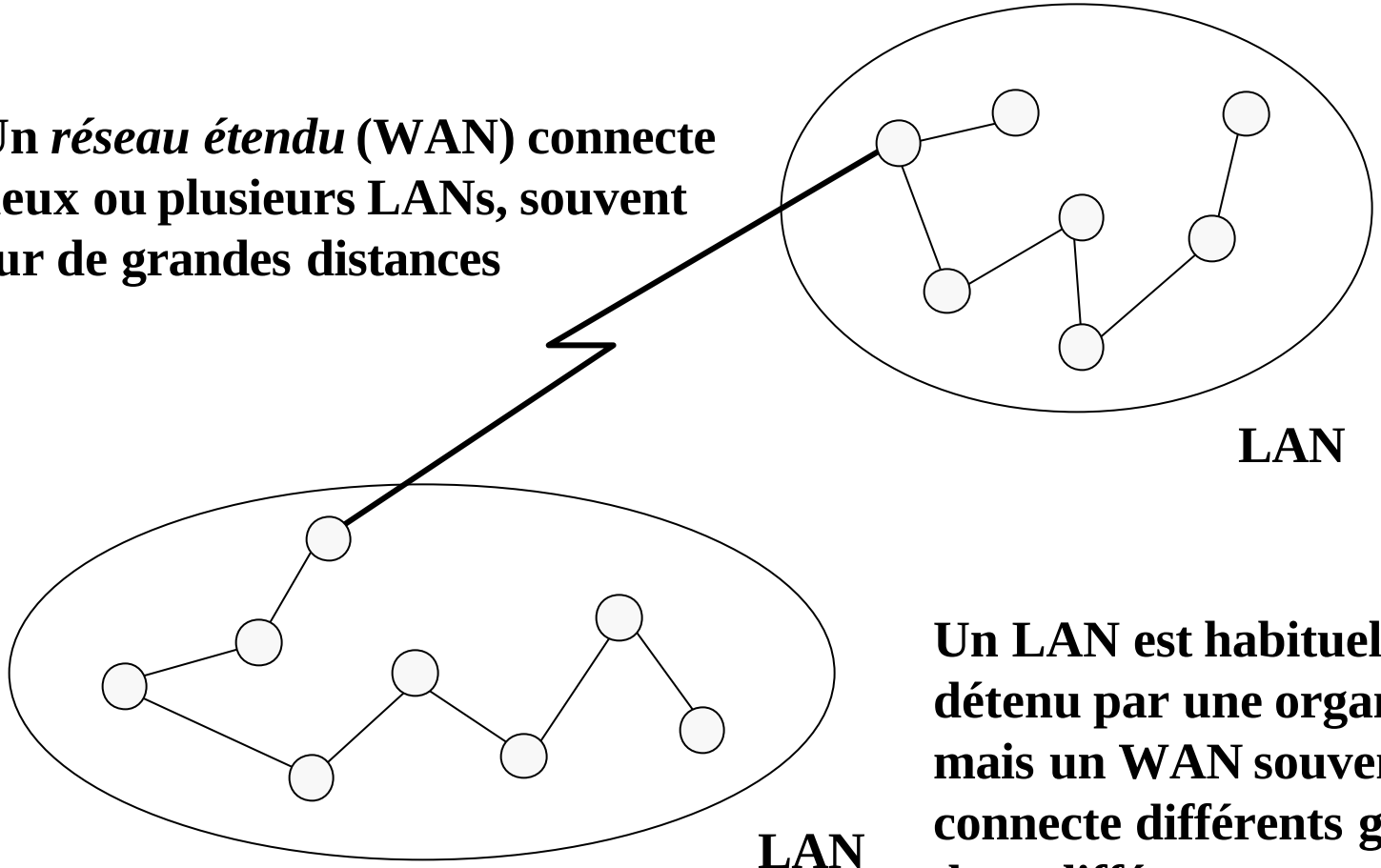


LAN

Un LAN connecte souvent les ordinateurs d'une seule pièce ou d'un édifice

Réseau étendu *(wide area)*

Un *réseau étendu* (WAN) connecte deux ou plusieurs LANs, souvent sur de grandes distances



Un LAN est habituellement détenu par une organisation, mais un WAN souvent connecte différents groupes dans différents pays

Internet

- ***L'internet*** **est un WAN qui couvre la planète entière**
- **Le mot internet provient du terme *internetworking*, qui implique une communication entre réseaux**
- **Il a commencé comme un projet du gouvernement américain, financé par le Advanced Research Projects Agency (ARPA), et était au début appelé ARPANET**
- **L'internet s'est rapidement étendu durant les années 1980 et 1990**
- **Moins de 600 ordinateurs étaient connectés à l'internet en 1983; maintenant il y en a plus de 10 million**

TCP/IP

- **Un protocole est un ensemble de règles qui déterminent comment les éléments communiquent entre eux**
- **Le software qui contrôle la communication internet suit une suite de protocoles appelés *TCP/IP***
- **Le *Internet Protocol* (IP) détermine le format de l'information alors qu'elle est transférée**
- **Le *Transmission Control Protocol* (TCP) contrôle comment les messages sont ré-assemblés et traite l'information perdue**

IP et les adresses internet

- **Chaque ordinateur sur l'internet a une *adresse IP* unique, telle :**

  `204.192.116.2`

- **La plupart des ordinateurs ont aussi un nom unique sur l'internet, qui est aussi appelé son *adresse internet* :**

  `renoir.villanova.edu`
  `kant.breakaway.com`

- **La première partie indique un ordinateur particulier (`renoir`)**
- **Le reste est le *nom de domaine*, indiquant l'organisation (`villanova.edu`)**

Noms de domaine

- **La dernière partie (le suffixe) de chaque nom de domaine habituellement indique le type de l'organisation:**

  `edu` - **institution éducationnelle**
  `com` - **compagnie commerciale**
  `org` - **organisation (à but non-lucratif)**
  `net` - **organisation basée sur le réseau**

Parfois le suffixe indique le pays :

`uk` - **United Kingdom**
`au` - **Australie**
`ca` - **Canada**
`se` - **Suède**

De nouvelles catégories de suffixes sont présentement considérées

Noms de domaine

- **Un nom de domaine peut avoir plusieurs parties**
- **Des noms de domaine uniques permettent à des ordinateurs individuels d'avoir le même nom localement**
- **Lorsqu'utilisée, une adresse internet est traduite en adresse IP par un software appelé le *Domain Name System* (DNS)**
- **Il n'y a pas une correspondance un-à-un entre les sections d'une adresse IP et les sections d'une adresse internet**

World-Wide Web

- **Le *World-Wide Web* permet d'accéder à plusieurs types d'information en utilisant un interface commun**
- **Un *browser* est un programme qui accède et présente l'information**
  - **texte, graphique, son, audio, vidéo, programmes exécutables**
- **Un document Web habituellement contient des liens vers d'autres documents Web, créant ainsi un environnement *hypermedia***
- **Le terme Web vient du fait que l'information n'est pas organisée d'une façon linéaire**

World-Wide Web

- **Les documents Web sont souvent définis en utilisant le *HyperText Markup Language* (HTML)**

- **L'information sur le Web est trouvée en utilisant un *Uniform Resource Locator* (URL):**

```
http://www.lycos.com
http://www.villanova.edu/webinfo/domains.html
ftp://java.sun.com/applets/animation.zip
```

- **Un URL indique un protocole (http), un domaine, et possiblement des documents spécifiques**

Résolution de problème

- **Ecrire un programme a pour but de résoudre un problème**

- **Les étapes générales pour résoudre un problème sont :**

  - **Comprendre le problème**
  - **Décomposer le problème en morceaux traitables**
  - **Faire le design d'une solution**
  - **Considérer les alternatives à la solution et la raffiner**
  - **Implanter la solution**
  - **Tester la solution et régler tout problème qui survient**

Résolution de problème

- **Plusieurs projets de software échouent parce que le développeur n'a pas réellement compris le problème à résoudre**
- **On doit éviter les hypothèses et clarifier les ambiguités**

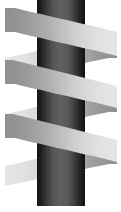
- **Lorsque les problèmes et leurs solutions grossissent, on doit organiser le développement en morceaux traitables**
- **Cette approche est fondamentale au développement de software**
- **Dans une approche *orientée-objet*, on sépare les solutions en morceaux appelés classes et objets**

Le langage de programmation Java

- **Un *langage de programmation* spécifie les mots et symboles qu'on utilise pour écrire un programme**

- **Un langage de programmation utilise un ensemble de règles qui spécifie comment les mots et les symboles peuvent être mis ensemble pour former des *énoncés de programme* valides**

- **Java a été créé par Sun Microsystems, Inc.**
- **Il a été introduit en 1995 et est devenu très populaire**
- **Java est un langage orienté-objet**



Structure d'un programme Java

- **Dans le langage de programmation Java :**
  - **Un programme est composé d'une ou de plusieurs *classes***
  - **Une classe contient une ou plusieurs *méthodes***
  - **Une méthode contient des *énoncés* (instructions) de programme**

- **Ces termes seront explorés en détail dans le cours**

- **Une application Java contient toujours une méthode appelée `main`**

- **Voir `Lincoln.java` (page 26)**

Structure d'un programme Java

```
//  commentaires sur la classe
public class MyProgram
{
}
```

Entête de la classe

Corps de la classe

Des commentaires peuvent être ajoutés presque n'importe où

Structure d'un programme Java

```
//  commentaires sur la classe
public class MyProgram
{

   //  commentaires sur la méthode
   public static void main (String[] args)
   {
   }

}
```

Entête de la méthode

Corps de la méthode

Commentaires

- **Les commentaires dans un programme sont aussi appelés *inline documentation***
- **Ils devraient être inclus pour expliquer le but du programme et décrire les étapes du traitement**
- **Ils ne modifient pas comment un programme s'exécute**
- **Les commentaires en Java peuvent prendre deux formes:**

```
// ce commentaire se rend jusqu'à la fin de la ligne

/*  ce commentaire continue jusqu'au symbole de
    terminaison, même sur plusieurs lignes     */
```

Identificateurs

- **Les *identificateurs* sont des mots qu'un programmeur utilise dans un programme**
- **Un identificateur peut être constitué de lettres, de chiffres, le caractère souligné (_), et le signe de dollar ($)**
- **Un identificateur ne peut pas commencer par un chiffre**
- **Java distingue entre les lettres majuscules et les minuscules, et ainsi `Total` et `total` sont des identificateurs différents**

Identificateurs

- **On choisit parfois nous-mêmes des identificateurs en écrivant un programme (tel `Lincoln`)**
- **On utilise parfois le code d'autres programmeurs, et alors on utilise les identificateurs qu'ils ont choisis (tel `println`)**
- **On utilise souvent des identificateurs spéciaux appelés des mots réservés qui ont une signification prédéfinie dans le langage**
- **Un mot réservé ne peut pas être utilisé pour signifier autre chose**

Mots réservés

- **Les mots réservés en Java sont :**

abstract	default	goto	operator	synchronized
boolean	do	if	outer	this
break	double	implements	package	throw
byte	else	import	private	throws
byvalue	extends	inner	protected	transient
case	false	instanceof	public	true
cast	final	int	rest	try
catch	finally	interface	return	var
char	float	long	short	void
class	for	native	static	volatile
const	future	new	super	while
continue	generic	null	switch	

Espace blanc

- **Les espaces, les lignes blanches et les tabulations sont appelés les *espaces blancs***
- **Un espace blanc est utilisé pour séparer les mots et les symboles dans un programme**
- **Tout espace blanc en extra est ignoré**
- **Un programme Java valide peut être formatté de plusieurs façons différentes**
- **Les programmes devraient être formattés pour améliorer la lecture, en utilisant une indentation uniforme**
- **Voir `Lincoln2.java` et `Lincoln3.java`**

Niveau d'un langage de programmation

- **Il y a quatre niveaux pour un langage de programmation :**
  - **Langage machine**
  - **Langage assembleur**
  - **Langage de haut niveau**
  - **Langage de quatrième génération**
- **Chaque type de CPU a son *langage machine* spécifique**
- **Les autres niveaux ont été créés pour faciliter à un humain l'écriture de programmes**

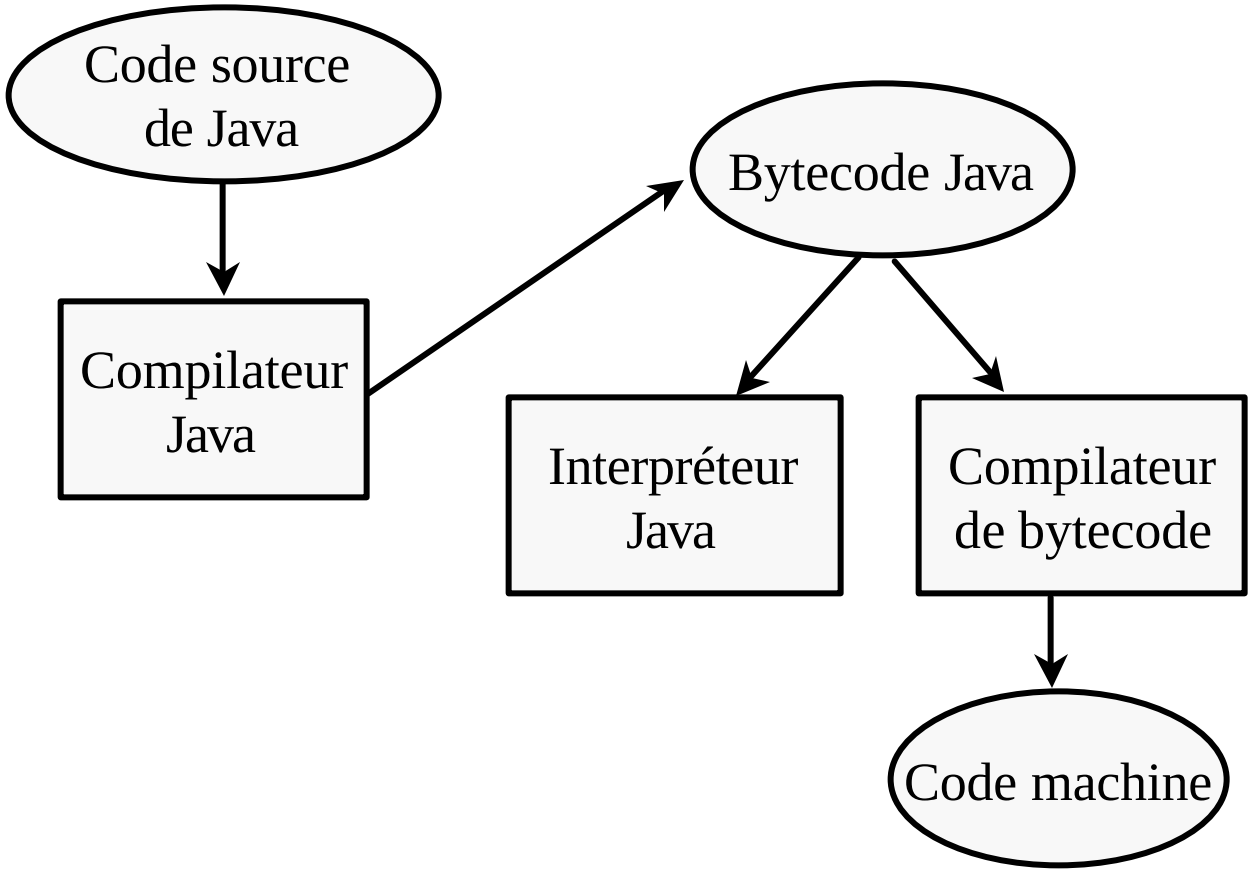
Langages de programmation

- **Un programme (code source) doit être traduit en langage machine avant d'être exécuté sur un type particulier de CPU**
- **Un *compilateur* est un outil software qui traduit le *code source* dans un langage cible spécifique**
- **Ce langage cible est souvent le langage machine pour un type particulier de CPU**
- **L'approche Java est un peu différente**

Traduction et exécution de Java

- **Le compilateur Java traduit le code source Java en une représentation spéciale appelée *bytecode***
- **Le bytecode Java n'est pas le langage machine pour un CPU traditionnel**
- **Un autre outil software, appelé un *interpréteur*, traduit le bytecode en langage machine et l'exécute**
- **Ainsi le compilateur Java n'est pas lié à une machine en particulier**
- **Java est considéré comme *architecture-neutral***

Traduction et exécution de Java



Environnements de développement

- **Il y a plusieurs environnements de développement pour écrire du software en Java :**
  - **Sun Java Software Development Kit (SDK)**
  - **Borland JBuilder**
  - **MetroWork CodeWarrior**
  - **Microsoft Visual J++**
  - **Symantec Café**

- **Quoique les détails de ces environnements diffèrent, le processus de base de compilation et d'exécution est essentiellement le même**

Syntaxe et sémantique

- **Les *règles de syntaxe* d'un langage définissent comment on peut mettre ensemble des symboles, des mots réservés et des identificateurs pour rendre un programme valide**

- **La *sémantique* d'un énoncé d'un programme définit ce que cet énoncé signifie (son but ou son rôle dans un programme)**

- **Un programme qui est syntaxiquement correct n'est pas nécessairement logiquement (sémantiquement) correct**

- **Un programme fera toujours ce qu'on lui dit de faire, pas nécessairement ce qu'on voudrait lui dire de faire**

Erreurs

- **Un programme peut avoir trois types d'erreurs**
- ***Erreurs de compilation* : le compilateur trouve les problèmes avec la syntaxe et les autres problèmes de base**
  - **Ex: variable non déclarée, assignation de types différents**
  - **S'il y a des erreurs de compilation, aucune version exécutable du programme n'est créée**
- ***Erreurs d'exécution* : un problème peut se produire lors de l'exécution du programme, comme en essayant de diviser par zéro, ce qui force le programme à se terminer anormalement**
- ***Erreurs logiques* : un programme peut s'exécuter, mais produire des résultats incorrects**

Introduction au graphique

- **Les dernières sections (1-2) de chaque chapitre du livre se concentrent sur des aspects graphiques**
- **La plupart des programmes informatiques ont des composantes graphiques**
- **Une image ou un dessin doit être digitalisé pour être stocké et traité sur un ordinateur**
- **Une image est décomposée en pixels, et chaque pixel est stocké séparément**

Représenter la couleur

- **Une image noir et blanc peut être stockée avec un bit par pixel (0 = noir et 1 = blanc)**
- **Une image couleur requiert plus d'information, et il existe plusieurs façons pour représenter une couleur donnée**
- **Par exemple, chaque couleur peut être représentée par un mélange de trois couleurs primaires : Rouge, Vert et Bleu**
- **En Java, chaque couleur est représentée par trois nombres entre 0 et 255 qui sont ensemble appelées une *valeur RGB***

Systèmes de coordonnées

- **Chaque pixel peut être identifié dans un système de coordonnées en deux dimensions**
- **Pour référer à un pixel dans un programme Java, nous utilisons un système de coordonnées dont l'origine est au coin supérieur gauche**