

M1105 : Bases des systèmes d'exploitation

Emmanuel Hainry

2013–2014

Contents

1 Introduction	1
1.1 Fonctions d'un ordinateur	3
2 Architecture d'un ordinateur	3
2.1 Composants matériels d'un ordinateur	3
2.2 Composants logiciels d'un ordinateur	3
2.2.1 Système	3
2.2.2 Pilotes	6
2.2.3 Applications	6
3 Manipuler un ordinateur	7
3.1 Système de fichiers	7
3.2 Droits	8
3.3 Gestion des processus	9

1 Introduction

Introduction

Objectifs :

- Définir informatique et ordinateurs
- Connaître le rôle des systèmes d'exploitation
- Savoir manipuler les systèmes de fichiers
- Automatisation de tâches élémentaires

Introduction

Compétences à acquérir :

- Connaître l'architecture des ordinateurs
- Savoir ouvrir/terminer une session
- Manipuler le système de fichiers (arborescence, droits)
- Maîtriser la notion de processus
- Comprendre et être capable de modifier des scripts simples

Un peu d'histoire

1642 Blaise Pascal crée la pascaline.

1800 Joseph-Marie Jacquard invente le métier à tisser.

1833 Charles Babbage conçoit la machine à différences.

1885 Hermann Hollerith crée une machine à cartes perforées.

1931 Kurt Gödel formule le théorème d'incomplétude.

1936 Alan Turing imagine une machine capable de réaliser les tâches d'un cervain humain.

1946 John P. Eckert et John Mauchly fabriquent l'ENIAC.

1946 John von Neumann conçoit EDVAC.

1972 Brian Kernighan et Dennis Ritchie : le langage C.

1977 Ordinateur personnel (Steve Wozniak chez Apple, Jack Tramiel chez Commodore, Steven Leininger chez Tandy).

1997 Assistant personnel (Apple, Palm) et smartphone (Nokia)

2010 Calcul dans les nuages (Amazon, Google)

Qu'est-ce qu'un ordinateur ?





- Une machine à calculer ?
- Un prouveur de théorèmes ?
- Un bidule relié au réseau ?
- Un truc qu'on peut programmer ?
- Un automate programmable universel

Qu'est-ce qu'un ordinateur

L'ordinateur est un appareil

- programmable
Un ordinateur se programme : on peut lui ajouter de nouvelles fonctionnalités.
- universel.
Toute fonctionnalité d'un ordinateur peut être ajoutée à mon ordinateur.

Ordinateur ou non ?

	 by Brad Montgomery	 ©Hewlett Packard	 by Glenn Fleishman	 by shawnc
Programmable	✓	✓	✗	✗
Universel	✗	✓	✓	✗

1.1 Fonctions d'un ordinateur

Fonctions d'un ordinateur

Les ordinateurs ont par définition trois fonctions :

- Calculer
- Gérer des données
- Communiquer

2 Architecture d'un ordinateur

2.1 Composants matériels d'un ordinateur

Matériel

Un ordinateur est constitué d'un certain nombre de composants matériels

- internes
 - Microprocesseur
 - Mémoire vive et mémoire morte
 - Carte mère
 - Carte graphique
 - Carte réseau
 - etc.
- périphériques (écran, imprimante, clavier, souris...)
- de gestion de données (internes ou externes)
 - disque dur
 - lecteur, graveur de DVD
 - clef USB, carte mémoire

2.2 Composants logiciels d'un ordinateur

Logiciel

Pour manipuler le matériel, l'ordinateur dispose d'un certain nombre de composants logiciels :

- le système d'exploitation
- les pilotes
- les applications

2.2.1 Système

Exemples de systèmes d'exploitation





Les logos proviennent de wikipedia, ils appartiennent à leurs propriétaires respectifs et sont utilisés à titre informatif.

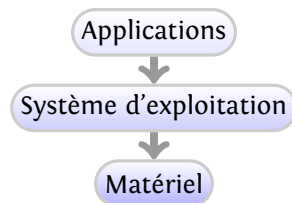


- Systèmes à destination des micro-ordinateurs vs des smartphones/tablettes.
- Interfaces graphiques peuvent être les mêmes.
- Applications identiques.

Système d'exploitation

- gère les ressources matérielles
- les rend disponibles pour les applications

Par exemple écriture sur le disque dur, envoi d'une information dans le câble réseau.



Système d'exploitation

- Windows¹, Mac OS X, GNU/Linux², NetBSD², Symbian OS...
- multi-utilisateur / mono-utilisateur
- multi-tâches / séquentiel

UNIX et dérivés

UNIX est une famille de systèmes d'exploitation définies par la norme Posix.

1969 Création Unix

1977 Version BSD (V5). BSD se ramifiera en openBSD, netBSD, freeBSD, PC-BSD, Dragonfly-BSD...

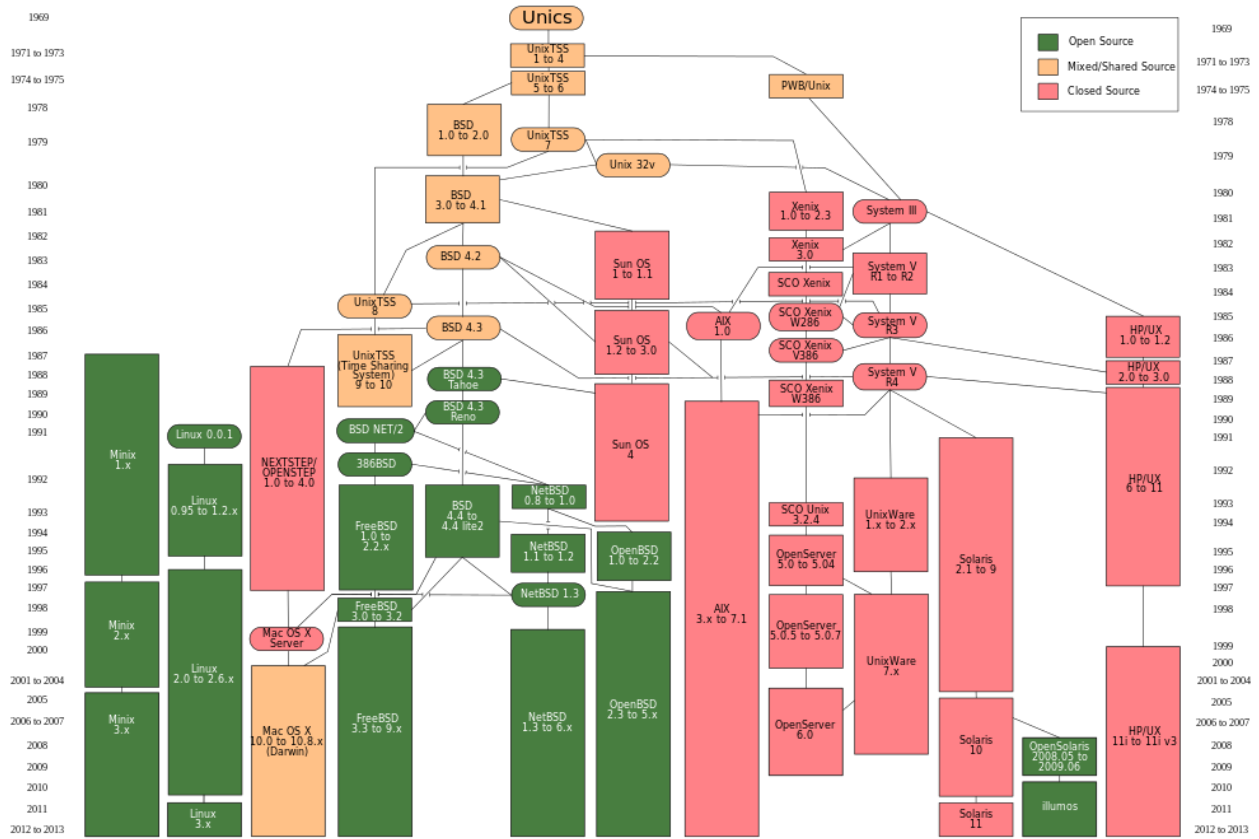
1991 Linux. Donnera naissance à Android.

1999 Mac OS X. Forké en iOS.

¹Gratuit pour les étudiants

²Libre et gratuit

Historique d'UNIX



POSIX (Portable Operating System Interface)

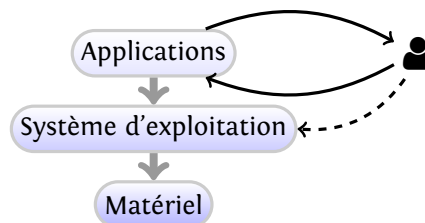
La norme Posix spécifie un certain nombre de commandes et de fonctions qui doivent être présentes dans un système.

Par exemple

- Comment créer et contrôler des processus.
- Reprise sur erreur.
- Accès fichiers.
- Commandes man, apropos, at, grep...

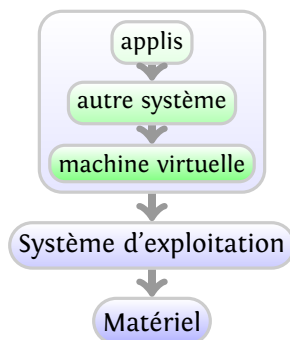
Système d'exploitation

Couche entre le matériel et les applications



Cas des machines virtuelles

Une machine virtuelle est une application se faisant passer pour un matériel à part entière.



- Migration d'un serveur sans rupture de service
- Test de la compatibilité ascendante sans casse
- Services indépendants simultanément accessibles

Note : les machines virtuelles sont différentes des émulateurs.

Machines virtuelles

- Exemples : VMWare, VirtualBox, Xen, Parallels, KVM...
- Application classique
- Choix du Système d'Exploitation à démarrer
- L'“image” du système peut être sauvegardée ou non après la session
- Protection contre les malwares
- Faire tourner des applications linux sous windows et inversement

2.2.2 Pilotes

Pilotes (drivers)

Les pilotes sont les logiciels dont se sert le noyau pour utiliser le matériel.

Chaque périphérique nécessite son pilote.

Les pilotes peuvent être intégrés au noyau, en être des modules ou être externes.

2.2.3 Applications

Applications

Les applications sont les programmes que l'utilisateur manipule.

Chacune a son utilité, sa configuration, les données qu'elle manipule.

UNIX : chaque programme ne fait qu'une seule chose mais il la fait bien.

Applications : exemple

Navigateur internet

- Plusieurs existent (Mozilla firefox, Microsoft Internet Explorer, Opera, Safari...)
- Plusieurs tâches simples : accès au réseau, téléchargement page html, interprétation html, affichage page, interaction utilisateur (clavier/souris)
- Des logiciels complexes
- Données : Configuration utilisateur, page web, images, plugins...

3 Manipuler un ordinateur

3.1 Système de fichiers

Système de fichier

- Sauvegarde sur disque linéaire (suite de 0 et de 1 cf. p. 10).
- Données apparaissent rangées dans des dossiers rangés dans des dossiers...
- Fichier : nom et dossier le contenant
- Dossier : nom et dossier le contenant

⇒ chemin complet : suite des dossiers puis nom du fichier
C:\Documents and Settings\Roger\Desktop\image.png
/home/roger/Desktop/image.png.

Système de fichiers

L'ensemble des fichiers et dossiers forme une arborescence.

```
.
|-- Downloads
|   |-- B1.pdf
|   |-- wien.ps
|   `-- zeno.jpg
|-- alire
|   |-- Athlon-Powersaving.gz
|   |-- BLUEBOOK.PDF
|   |-- Bash-prog.gz
|   |-- LN402chap1.pdf
|   |-- LN402debut.pdf
|   |-- MSMF_1984_2_13__1_0.pdf
|   |-- SDPP_1973-1974__15_2_A3_0.pdf
|   |-- SDPP_1975-1976__17_2_A6_0.pdf
|   |-- Troff
|   |   |-- cmus.pdf
|   |   |-- groff.pdf
|   |   |-- groff7.pdf
|   |   |-- groff_man.html
|   |   `-- screen.pdf
|   `-- langII.pdf
|-- cerisier.JPG
|-- comptes.xhb
|-- games.tc
|-- garrft.pdf
`-- temp
    |-- Hiragana Practice.pdf
    `-- texlive_port_doc-20070717.pdf
```

Représentation sur le disque

Un disque dur est constitué de plusieurs *plateaux*, chacun comprenant plusieurs *cylindres* découpés en *secteurs*.

Chaque secteur représente en général 512 octets.

Un fichier de plus de 512 octets sera donc réparti sur plusieurs secteurs (si possible contigus), la fin d'une partie est suivie par le numéro du secteur suivant.

Ainsi, indiquer le secteur où commence un fichier suffit à lire le fichier en entier.

Un dossier sera simplement un pseudo-fichier contenant les associations "nom de fichier"/"secteur initial". Dans l'exemple précédent, le dossier considéré contient 3 dossiers (Downloads, alire et temp) et 4 fichiers réguliers. Il pourra être représenté par

Downloads	123
alire	124
cerisier.JPG	137

comptes.xhb	155
games.tc	917
garrft.pdf	223
temp	199

Si je cherche le secteur 123, j'y trouverai le contenu de Downloads. Il s'agit d'un dossier, donc d'une liste de fichiers et dossiers, etc.

Arborescence

Dans l'exemple précédent, ils manquent deux choses :

- qu'est-ce qui est un dossier, qu'est-ce qui est un fichier.
- comment remonter dans l'arborescence

On ajoutera quelques informations (type de fichier, droits) en plus du nom et de la localisation.

On ajoutera aussi un "pointeur" vers le dossier parent.

Système de fichiers

La séparation de fichiers sur des secteurs qui ne se suivent pas est appelée la fragmentation. Elle ralentit la lecture des fichiers.

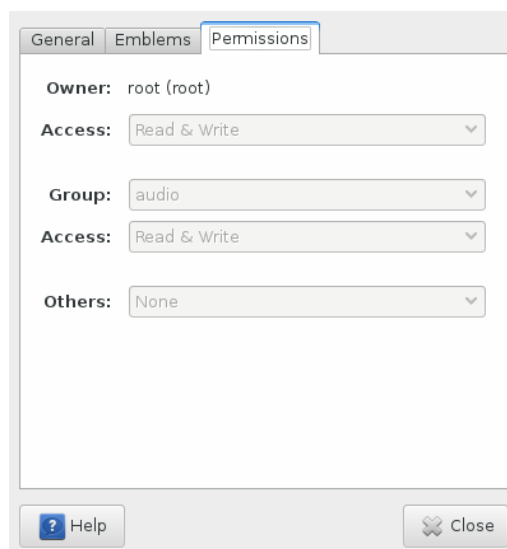
Il existe des stratégies différentes pour stocker les fichiers en évitant la fragmentation ⇒ différents systèmes de fichiers (FS)

- FAT12, FAT16, FAT32 : systèmes utilisés historiquement par Dos et compris par tous les SE modernes.
- NTFS : système utilisé par les Windows récents.
- ext2, ext3, ext4 : FS développé pour GNU/Linux.
- HFS, HFS+, HFSX : FS choisi par Mac OS X.
- NFS, CIFS, AFS : FS en réseau.
- ZFS : FS développé pour les très gros disques durs.

Fonctionnalités spécifiques : droits, fragmentation, noms longs, systèmes de fichiers à base de données, déduplication, FS spécialisés pour mémoire flash...

3.2 Droits

Droits



Droits (mode symbolique)

- rw- r-- ---

Droits (mode octal)

640

Cf. page 10.

3.3 Gestion des processus

Processus

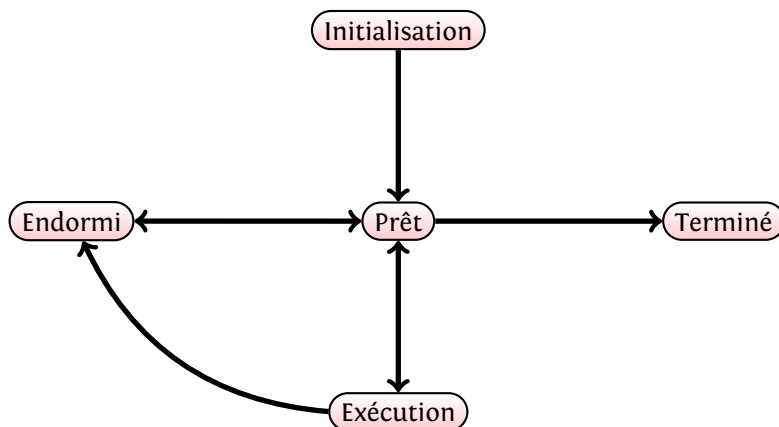
- Un processus est une tâche exécutable par l'ordinateur. Chaque application peut utiliser plusieurs processus.
- Un processus peut lui-même lancer des processus.
- Les processus s'exécutent en théorie en parallèle.
- Différents états : Exécution, Attente, terminé...

Analogie

Un cuisinier monoprocesseur prépare un plat (processus 1).
Il se blesse et donc interrompt le processus 1 pour lancer une tâche de soin (processus 2).
Pour se soigner, il lance une sous-tâche : désinfecter la plaie (processus 3).
Une fois le processus 2 achevé, il reprend le processus 1 où il l'avait laissé.

- Notion de priorité entre les processus.
- Contexte d'exécution, données, pointeur de programme conservés en cas d'interruption.
- Arbre des processus.

États d'un processus



Arborescence des processus

Chaque processus est lancé par un processus père et peut lancer des processus fils.

- Arborescence des processus.

Pour qu'un père termine, tous ses fils doivent se terminer (et pour qu'ils se terminent, tous leurs fils doivent se terminer, et pour...)

Dans le cas où un processus père meurt pour cause d'erreur, qu'arrive-t-il aux fils ?

- Cas des zombies

Il est possible d'interagir avec les processus (leur envoyer des signaux).

Références

Références

1. Étienne Samyn et Lionel Medini, « S'approprier son environnement de travail », <http://www.c2imes.org/>, 2007
2. Pierre Morvan, « Dictionnaire de l'informatique », Larousse, 1981
3. Moncef Mlouka et Bernard Hidoine, « Machines à calculer », http://interstices.info/jcms/c_15272/machines-a-calculer, 2006
4. William Gibson et Bruce Sterling, « La machine à différences », J'ai lu, 1999
5. Robert Harris, « Enigma », Pocket, 1995

Représentation binaire

- Conversion binaire/décimal

$$\begin{aligned} 10101101_2 &\Leftrightarrow 173_{10} \\ 2^7 + 2^5 + 2^3 + 2^2 + 2^0 &\Leftrightarrow 10^2 + 7 \times 10^1 + 3 \times 10^0 \end{aligned}$$

- Conversion binaire/octal

$$10\ 101\ 101 \Rightarrow 2\ 5\ 5$$

Représentation numérique des caractères

- ASCII (American Standard Code for Information Interchange)
- ISO-8859-15 (8-bit single-byte coded graphic character sets – Part 15: Latin alphabet No. 9)
- UTF-8 (UCS transformation format 8 bits)

Codage	taille	caractères
ASCII	7 bits	alphanumériques
ISO-8859-15	8 bits	é, à, ù, ç, œ
UTF-8	1, 2, 3 ou 4 octets	∞, ±, α, 北, ض

	ASCII	ISO-8859-15	UTF-8
A	100 0001	0100 0001	0100 0001
e	110 0101	0110 0101	0110 0101
é	-	1110 1001	1110 1001
北	-	-	0101 0011 0001 0111