



# Système d'exploitation

2017-2018

- 1 Systèmes d'exploitation
- 2 Démarrage d'un ordinateur moderne
- 3 Matériel et logiciel
- 4 Systèmes d'exploitations et applications
- 5 La philosophie UNIX
- 6 Les distributions



# Contenu du cours

- Qu'est-ce que c'est un système d'exploitation?
- Qu'est-ce qu'il se passe lors du démarrage d'un ordinateur?
- Remarques sur la frontière matériel-logiciel.
- Est-ce que les applications sont nécessairement spécifiques à un systèmes d'exploitation?
- La philosophie UNIX.
- Les distributions

- 1 **Systèmes d'exploitation**
- 2 Démarrage d'un ordinateur moderne
- 3 Matériel et logiciel
- 4 Systèmes d'exploitations et applications
- 5 La philosophie UNIX
- 6 Les distributions

# Deux sens de *système d'exploitation*

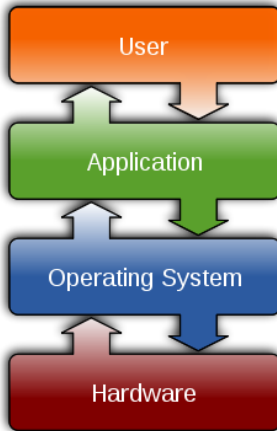
Angl : *Operating System* — OS

- 1 Dans le **sens stricte** : le logiciel de base qui s'occupe de la gestion des ressources d'une machine, et qui donne à un programmeur une vue abstraite des détails matériel d'une machine.  

  - ▶ On appelle ce logiciel aussi le **noyau** (angl. : *kernel*).
- 2 Dans le **sens populaire** : un noyau, plus un ensemble de logiciels qui viennent habituellement avec ce noyau (interfaces graphiques, suites bureautiques, éditeurs de texte, navigateurs, outils de programmation, ...).  


# Des exemples des systèmes d'exploitation

- Microsoft Windows (plusieurs)
- Mac OS (aujourd'hui dérivée de BSD)
- GNU/Linux
- BSD (Berkely Software Distribution — plusieurs variantes)
- GNU/Hurd
- OS/360, CP/M, OS/2, BS2000, Amiga OS, ...
- Sur des systèmes embarqués : iOS, Android (basé sur Linux), Firefox OS (basé sur Linux), Windows Mobile, Maemo, Symbian, ...

# La place d'un système d'exploitation



[https://en.wikipedia.org/wiki/Operating\\_system](https://en.wikipedia.org/wiki/Operating_system)

# Les deux fonctions d'un système d'exploitation



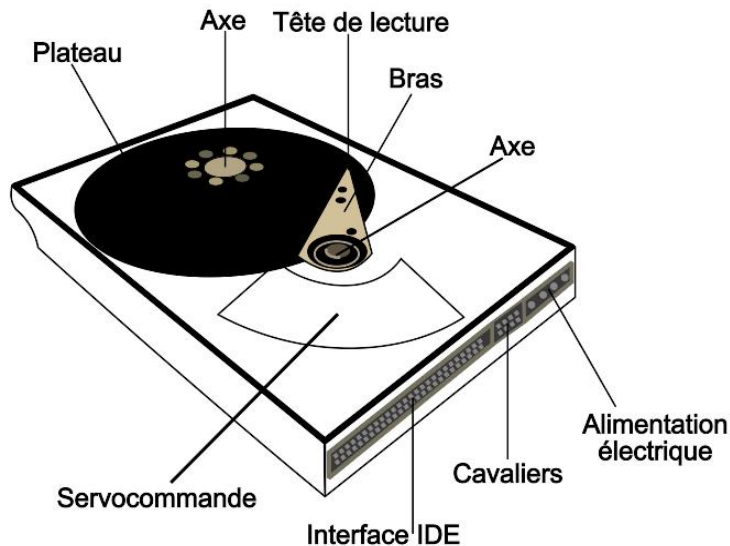
- Fournir à un programmeur une vue *abstraite* de la machine qui lui facilite la vie.
  - ▶ Par exemple : donner une interface commune pour écrire sur un fichier qui peut se trouver sur disque, ou disquette, ou sur le réseau.
- Gérer (l'accès à) des ressources communes entre plusieurs utilisateurs ou programmes
  - ▶ mémoire, disque, accès réseau, ...



# Noyau : Interface aux périphériques

- Interface uniforme pour les périphériques (disques, clefs USB, clavier, souris, ...).
- Pour un programmeur, le noyau est similaire à une bibliothèque : le noyau fournit des fonctions qu'un programmeur peut utiliser dans un programme (appels système, angl : *system calls*).
- Par exemple une fonction pour lire une partie d'un fichier, ou pour écrire sur un fichier, pour attendre qu'une touche de clavier soit pressée, ...
- L'implémentation des ces fonctions cache souvent les différences techniques entre différentes technologies (par exemple des disques IDE, SCSI, ATA, ...)

# Un disque dur (IDE)



### Definition (Fichier)

Fichier (angl. : *file*) : Une structure de données, qui existe indépendamment d'un programme utilisateur. Souvent sur un support persistant (disque, disquette, CD, ...) ou parfois dans la mémoire vive d'une machine.

- Normalement une structure **séquentielle**, par exemple une séquence de caractères (texte), un binaire (code exécutable).
- Il s'agit d'une **abstraction** : Vu d'un programmeur ou d'un utilisateur c'est une séquence, avec des actions possibles :
  - ▶ Lire
  - ▶ Remplacer le contenu
  - ▶ Ajouter du contenu à la fin du fichier
  - ▶ Se positionner à une certaine position dans le fichier

## Noyau : Fichiers (cont.)

- La **réalisation physique** est cachée : les morceaux du même fichiers peuvent être localisés à des endroits différents d'un disque, ou même exister sur des support différents (par exemple CD pour la version de base d'un fichier plus disque pour les parties modifiées par l'utilisateur).
- Les fichiers sont accessibles par leur **nom**
- Souvent **organisation hiérarchique** : répertoires et fichier réguliers.
- **Métadonnés** (angl. : *metadata*), en particulier attributs liés aux privilèges des utilisateurs.

# Noyau : Systèmes de fichiers (1)



Deux sens différents de *système de fichier* (angl. : *file system*) :

- 1 **Technique d'organiser**, de représenter sur un support, et de gérer des fichiers.
  - ▶ Exemples : *FAT*, *FAT32*, *NTFS* (DOS/Windows); *ext4*, *xf*s, *zfs* (Linux); *ISO9660* (CD et DVD).
- 2 Une **collection** de fichiers suivant une de ces techniques sur un disque (ou un autre support). Un disque est normalement *partitionné* (par l'administrateur du système), sur chaque partition on peut avoir un système de fichiers (sens 2).

## Noyau : Systèmes de fichiers (2)

Différences entre les systèmes de fichiers (sens 1) :

- Différences dans le support (CD/DVD, disque, réseau, ...)
- Limite dans la taille des fichiers.
- Performance pour un certain type d'applications.
- Robustesse contre des failles du matériel ou de l'alimentation électrique (!)

## Noyau : Système de fichiers (3)

Les problèmes de la conception d'un système de fichier :

- Support physique : blocs de disque d'une taille fixe
  - par exemple 0.5Ko = 512 caractères
- Comment gérer la succession de plusieurs blocs ?
- Optimisation pour les fichiers de grande taille ou des fichiers de petite taille ?
- Comment trouver des blocs de disque nécessaires pour une extension d'un fichier ?
- Réutilisation des blocs disque dans le cas de suppression d'un fichier.
- Robustesse.

## L'organisation des fichiers sur une machine

- en partie standardisée, au moins pour les répertoires « systèmes » (les répertoires qui sont accédés directement par les outils qui viennent avec le système d'exploitation)
- en partie libre : des répertoires dont le contenu peut être librement organisé, soit
  - ▶ par l'administrateur système
  - ▶ par les utilisateurs



# Filesystem Hierarchy Standard

- « Standard » établie par plusieurs grand groupes du monde UNIX.
- /boot répertoire spécifique au démarrage, aussi pour le système d'exploitation lui-même.
  - /bin répertoire pour les commandes de base.
  - /lib répertoire qui contient des bibliothèques de bases (par exemple pour le noyau).
  - /usr commandes, et fichiers de support, pas absolument nécessaires pour un fonctionnement minimal (peut être fournis par le réseau).
    - ▶ /usr/lib bibliothèques partagés
    - ▶ ...
  - /home contient les répertoires utilisateurs.
  - /etc fichiers de configuration du système
  - ...

<http://www.pathname.com/fhs/>

# Noyau : gérer des processus

## Definition (Processus)

Une instance d'un programme en cours d'exécution.

- Chaque processus a sa propre **zone mémoire** : pour son propre code, et sa mémoire de travail (par exemple les variables déclarées dans le programme).
- Les zones mémoire des processus différents sont séparées une de l'autre (en principe).
- À chaque moment, **un seul processus peut tourner sur un processeur**.
- Le système d'exploitation distribue le temps de calcul parmi les processus qui sont prêts à tourner (système **multi-tâche**).
- Un processus est interrompu quand son créneau est épuisé, ou quand il doit attendre un événement externe (par exemple une réponse d'un disque) : **changement de contexte**.
- Les processus peuvent avoir des priorités différentes.

# Exemple de processus sur une machine

System Monitor

Processes Resources File Systems

Load averages for the last 1, 5, 15 minutes: 0,49, 0,32, 0,27

Refresh View ▾

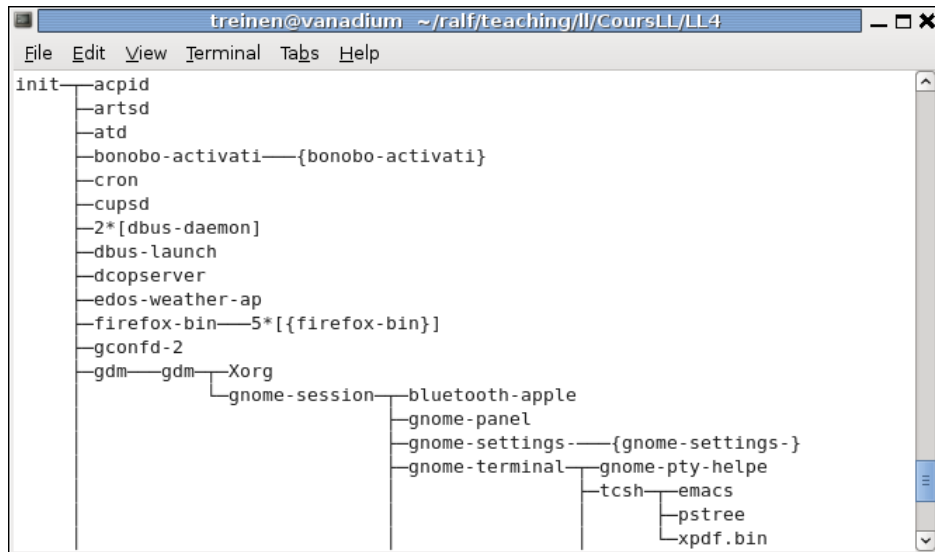
Process Name	Status	% CPU ▾	Nice	ID	Memory	Waiting Channel	Priority
gnome-shell	Running	4	0	3063	299,7 MiB	0	Normal
gnome-system-monitor	Running	2	0	6891	11,7 MiB	0	Normal
evolution	Sleeping	0	0	32536	33,9 MiB	0	Normal
evolution-calendar-factory	Sleeping	0	0	32484	50,8 MiB	0	Normal
evolution-addressbook-fac	Sleeping	0	0	32452	16,6 MiB	0	Normal
gvfsd-burn	Sleeping	0	0	31827	8,0 KiB	0	Normal
gvfsd-trash	Sleeping	0	0	31722	520,0 KiB	0	Normal
emacs	Sleeping	0	0	31710	153,9 MiB	0	Normal
applet.py	Sleeping	0	0	31708	2,9 MiB	0	Normal
tracker-miner-fs	Sleeping	0	19	31701	70,8 MiB	0	Very Low
nautilus	Sleeping	0	0	31696	43,8 MiB	0	Normal
tracker-store	Sleeping	0	0	31695	18,6 MiB	0	Normal
evolution-alarm-notify	Sleeping	0	0	31692	2,4 MiB	0	Normal
mission-control-5	Sleeping	0	0	31682	141,3 MiB	0	Normal
gvfsd-metadata	Sleeping	0	0	31675	6,8 MiB	0	Normal
evolution-source-registry	Sleeping	0	0	31655	2,6 MiB	0	Normal
gnome-shell-calendar-serv	Sleeping	0	0	31652	19,3 MiB	0	Normal
pulseaudio	Sleeping	0	0	31625	2,6 MiB	0	Normal
gsd-printer	Sleeping	0	0	31620	268,0 KiB	0	Normal

End Process

## Noyau : l'hierarchie de processus

- Un processus peut créer un nouveau processus.
- Hiérarchie de processus, structure d'un arbre.
- Un processus peut se terminer : fin régulier d'un programme, erreur d'exécution, signal envoyé par un autre processus.
- *Démon* : processus qui est lancé directement par le système et qui réagit à des requêtes (par exemple des requêtes réseau), par exemple un serveur web, agent de transport de mail, ...

# L'arbre de processus sur une machine



# Gestion de la mémoire

- Tout processus a sa propre zone de mémoire
- Utilisée pour
  - ▶ le programme (exécutable)
  - ▶ les données utilisées par le programme
- Le noyau gère des zones mémoire pour les processus différents
- Pas nécessairement continue dans la mémoire physique
- Peut être temporairement déplacée sur disque

## Noyau : utilisateur multiples

- Utilisateur différents avec des privilèges différents.
- Privilèges : accès à des fichiers, utilisation du réseau, de la carte son, du graveur DVD, . . . .
- Les processus lancés par un utilisateur “ héritent ” ses privilèges.
- Privilèges particuliers pour les démons.
- Administrateur système, ou *super user*, avec des privilèges maximales (aussi utile quand il y a une seule personne physique qui utilise une machine).

## Noyau : gestion des connexions réseau

- *Réseau* : Structure décentralisée et complexe, pas sous contrôle des utilisateurs.
- Internet : basé sur une hiérarchie de logiciels appelé la *pile de protocoles TCP/IP* qui réalisent des fonctionnalités de plus en plus complexes.
- Le système d'exploitation fournit une vue d'une connexion réseau qui ressemble à un fichier (fonctions ouvrir/fermer, lire et écrire).



# L'avenir des systèmes d'exploitation

- **Micro-noyaux** : factoriser les tâches d'un noyau en un noyau minimal, et des programmes autonomes. On sait depuis longtemps que c'est en principe la bonne façon d'écrire un noyau, mais qui a pour l'instant donné que des systèmes expérimentaux (Mach, GNU Hurd, Minix 3, ...)
- **Virtualisation** : Faire tourner un système complet sur une simulation logicielle d'une machine (machine virtuelle). C'est également une vieille idée (OS/360 des années 60) qui est aujourd'hui de nouveau utilisé pour cloisonner des applications.

# Outline

- 1 Systèmes d'exploitation
- 2 Démarrage d'un ordinateur moderne**
- 3 Matériel et logiciel
- 4 Systèmes d'exploitations et applications
- 5 La philosophie UNIX
- 6 Les distributions

# Le démarrage d'un ordinateur : BIOS

## BIOS : *Basic Input/Output System*

- Stocké sur un circuit<sup>1</sup> de la carte mère
- Configurable par un menu lors du démarrage du système.
- Détermine ou chercher le système à démarrer (quel disque, lecteur CD ou DVD, clef USB, disquette, par le réseau, ...)
- Ici : démarrage d'un disque dur (par exemple le premier).
- Chaque disque contient une zone particulière qui est destinée au démarrage du système. Lancement du binaire qui se trouve dans cette zone du disque indiqué par le BIOS.

---

1. plus précisément : sur une mémoire non volatile, mais remplaçable

# Le démarrage d'un ordinateur : BIOS

## BIOS : *Basic Input/Output System*

- Stocké sur un circuit<sup>1</sup> de la carte mère
- Configurable par un menu lors du démarrage du système.
- Détermine ou chercher le système à démarrer (quel disque, lecteur CD ou DVD, clef USB, disquette, par le réseau, ...)
- Ici : démarrage d'un disque dur (par exemple le premier).
- Chaque disque contient une zone particulière qui est destinée au démarrage du système. Lancement du binaire qui se trouve dans cette zone du disque indiqué par le BIOS.

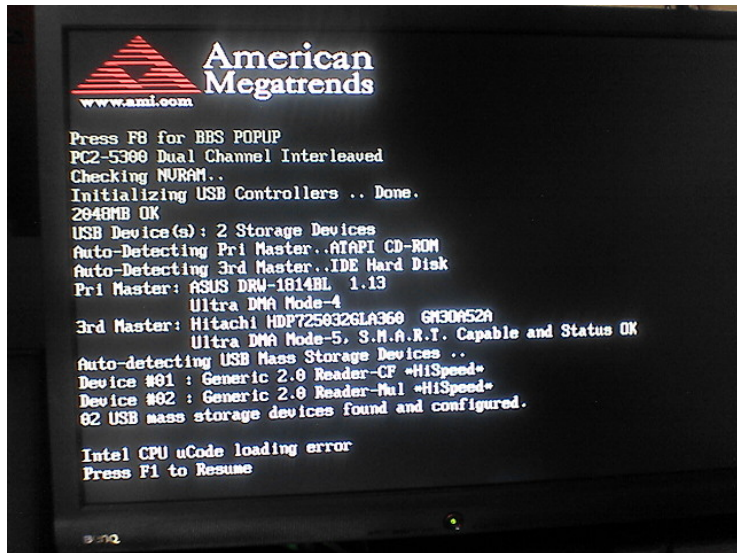
## Plus récemment : UEFI

- même principe, beaucoup plus de fonctionnalités
- notamment : la capacité de lire un système de fichiers

---

1. plus précisément : sur une mémoire non volatile, mais remplaçable

# Démarrage d'un BIOS

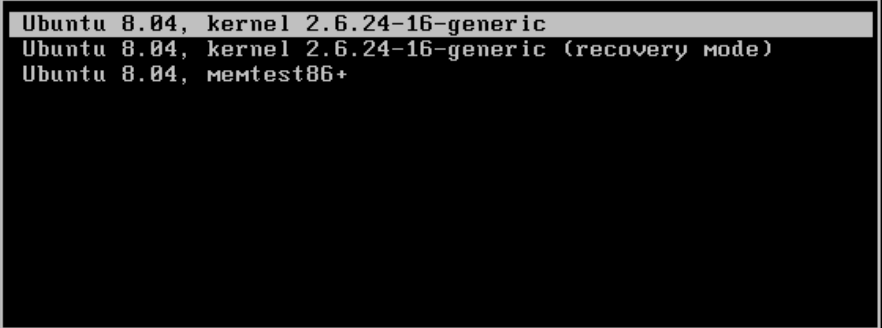


# Le démarrage d'un ordinateur : le chargeur

Lancement du système propre, ou (souvent dans le cas de systèmes multi-boot) d'un **chargeur** (angl. : *bootloader*) :

- Logiciel qui réside sur un support comme disque dur, clef USB, . . . .
- Peut être modifié ou configuré par l'utilisateur, l'installation de ce logiciel nécessite d'outils spécialisés car le chargeur réside sur une zone spéciale du disque.
- Ce logiciel détermine (éventuellement après un choix fait par l'utilisateur) quel système d'exploitation lancer.
- Le chargeur connaît une liste de système d'exploitation installés (par exemple Windows, GNU/Linux, BSD), avec le disque et l'endroit sur le disque où se trouve le noyau.
- Exemples (GNU/Linux) : LILO, Grub

# Le chargeur GRUB



```
Ubuntu 8.04, kernel 2.6.24-16-generic
Ubuntu 8.04, kernel 2.6.24-16-generic (recovery mode)
Ubuntu 8.04, memtest86+
```

Use the ↑ and ↓ keys to select which entry is highlighted.  
Press enter to boot the selected OS, 'e' to edit the  
commands before booting, or 'c' for a command-line.

[https://en.wikipedia.org/wiki/File:GRUB\\_screenshot.png](https://en.wikipedia.org/wiki/File:GRUB_screenshot.png)

## Démarrage d'un ordinateur : GNU/Linux

- Réserve des zones mémoire pour la gestion des ressources du système.
- Lancement d'un premier processus `init` qui sert comme racine de l'arbre de processus.
- `init` crée (par exemple) des processus qui attendent qu'un utilisateur entre son login et mot de passe, puis (quand authentification de l'utilisateur réussie) se transforment dans un interpréteur de commande avec les privilèges de l'utilisateur.
- Quand un utilisateur se déconnecte : son processus d'interpréteur de commande est détruit. `init` s'en rend compte et crée de nouveau un processus qui attend qu'un nouvel utilisateur s'identifie.



# Démarrage GNU/Linux : montage systèmes de fichiers

```
295
audit(1216470015.968:3): policy loaded auid=4294967295 ses=4294967295
INIT: version 2.86 booting
        Welcome to Red Hat Enterprise Linux Server
        Press 'I' to enter interactive startup.
Setting clock (utc): Sat Jul 19 05:20:22 MST 2008      [ OK ]
Starting udev:                                         [ OK ]
Loading default keymap (us):                          [ OK ]
Setting hostname rhce-prep.example.com:              [ OK ]
No devices found
Setting up Logical Volume Management:
    No volume groups found
                                                    [ OK ]

Checking filesystems
/: clean, 4871/263232 files, 72321/263056 blocks
/home: clean, 117/130560 files, 27384/522080 blocks
/var: clean, 1165/130560 files, 65117/522080 blocks
/dev/md0: clean, 12/883872 files, 45604/883456 blocks
/usr: clean, 81733/524288 files, 427747/524120 blocks
/boot: clean, 33/66264 files, 24068/265040 blocks
                                                    [ OK ]

Remounting root filesystem in read-write mode:       [ OK ]
Mounting local filesystems:                          [ OK ]
Enabling local filesystem quotas:                    [ OK ]
```

# Démarrage GNU/Linux : lancement de services

```
Enabling /etc/fstab swaps: [ OK ]
INIT: Entering runlevel: 3
Entering non-interactive startup
Applying Intel CPU microcode update: [ OK ]
Checking for hardware changes [ OK ]
Bringing up interface eth0:
Determining IP information for eth0... done. [ OK ]

Starting auditd: [ OK ]
Starting restorecond: [ OK ]
Starting system logger: [ OK ]
Starting kernel logger: [ OK ]
Starting irqbalance: [ OK ]
Starting mcstransd: [ OK ]
Starting portmap: [ OK ]
Starting setroubleshootd: [ OK ]
Starting NFS statd: [ OK ]
Starting mdmonitor: [ OK ]
Starting RPC idmapd: [ OK ]
Starting system message bus: [ OK ]
Starting Bluetooth services: [ OK ]
Mounting other filesystems: [ OK ]
Starting PC/SC smart card daemon (pcscd): [ OK ]
Starting hidd: [ OK ]
```

# Outline

- 1 Systèmes d'exploitation
- 2 Démarrage d'un ordinateur moderne
- 3 Matériel et logiciel**
- 4 Systèmes d'exploitations et applications
- 5 La philosophie UNIX
- 6 Les distributions

# La frontière matériel/logiciel

- ... n'est pas toujours aussi nette qu'on peut le croire.
- Le BIOS est en principe un logiciel, mais il est installé dans une mémoire non volatile circuit, et souvent considéré (erronément) comme du matériel.
- Microcode pour la programmation des unités centrales (CPU).
- Machines virtuelles : Simulation d'un ordinateur par un logiciel.
- Micrologiciels (angl : *firmware*) pour contrôler le matériel
  - ▶ p.ex. disques intelligents qui déplacent automatique de secteurs défectueux
  - ▶ le BIOS est aussi un exemple de micrologiciel

# Outline

- 1 Systèmes d'exploitation
- 2 Démarrage d'un ordinateur moderne
- 3 Matériel et logiciel
- 4 Systèmes d'exploitations et applications**
- 5 La philosophie UNIX
- 6 Les distributions

## Est-ce que les applications sont spécifiques à un OS?

- *En principe* souvent pas.
- Exemple : Navigateur Firefox disponible pour plusieurs OS
- Exemple : Le noyau du système d'exploitation des Mac est passé d'un système propriétaire de Apple à une variante d'un OS libre (BSD Unix).
- En pratique : il y a autour d'un certain noyau de OS une certaine culture de logiciels, interfaces, etc.
- Une autre limitation : installation de nouveaux logiciels ne doit pas être en conflit avec les logiciels installés habituellement avec l'OS installé.

# Outline

- 1 Systèmes d'exploitation
- 2 Démarrage d'un ordinateur moderne
- 3 Matériel et logiciel
- 4 Systèmes d'exploitations et applications
- 5 La philosophie UNIX**
- 6 Les distributions

# UNIX : Coopération entre logiciels indépendants

- Plusieurs logiciels qui sont chacun autonome mais qui communiquent, par exemple par un mécanisme de tuyau (tampon pour le transfert de données entre applications différentes).
- Exemple en UNIX : Afficher une liste triée de tous les fichiers dans un répertoire :
  - ▶ La commande `ls` produit la liste de fichiers
  - ▶ La commande `sort` trie une liste de mots
  - ▶ Combinaison des deux : `ls | sort`
- *Glue Languages* : langages de programmation conçus pour le traitement de données textuelles et pour la liaison entre applications. P.ex. Perl, Python, ...



# La philosophie UNIX

Doug McIlroy (inventeur des tuyaux en UNIX) :

*This is the Unix philosophy : Write programs that do one thing and do it well. Write programs to work together. Write programs to handle text streams, because that is a universal interface.*

# Programmes qui manipulent des programmes

- Compilateur : traduire code source en code machine
- Générateur de parseur : construit le code source d'une partie d'un compilateur à partir d'une description formelle de la syntaxe d'un langage de programmation.
- Générateur du code source pour une interface.
- Analyseur de code : Assurer des propriétés de code, comme pas de débordement de la mémoire (il y a des limitations théoriques de cette approche).
- Vérification de code par rapport à une spécification formelle.
- Test automatique.

# Outline

- 1 Systèmes d'exploitation
- 2 Démarrage d'un ordinateur moderne
- 3 Matériel et logiciel
- 4 Systèmes d'exploitations et applications
- 5 La philosophie UNIX
- 6 Les distributions**

# Qu'est-ce que c'est une distribution (GNU/Linux)?

- 1 Un **installateur** : premier démarrage d'un ordinateur sous Linux, détection du matériel, choix des pilotes nécessaires, partition des disques, configuration réseau, installation d'un système de base, finalement installation de paquets.  
Normalement sur support amovible (disquette, CD, DVD, clé USB), ou par réseau.
- 2 Beaucoup de **paquets** : en Debian environ 30.000 paquets disponible pour l'architecture amd64.  
Paquets pour des composants essentiels (noyau Linux, outils de base), des librairies, des applications, ...

# Choisir et installer des paquets

(haut niveau, à-la “appstore”)

The screenshot displays the Ubuntu Software Center interface. At the top, there's a navigation bar with icons for 'All Software', 'Installed', and 'History', along with a search bar. Below this, the main content area is divided into sections: 'What's New' and 'Recommended For You'. The 'What's New' section features a grid of software packages with icons, names, categories, and prices. The 'Recommended For You' section has a button to 'Turn On Recommendations' and a note about Canonical's recommendation system. At the bottom, there's a 'Top Rated' section with a grid of highly-rated software packages. A sidebar on the left lists various software categories like Accessories, Books & Magazines, Developer Tools, etc. The bottom of the interface shows a total of 46295 items.

Ubuntu Software Center

Navigation: All Software, Installed, History

Search:

**What's New**

Icon	Name	Category	Price
	Warrior Form Builder	Graphic Interface D...	US\$ 4.99
	Pinball Games	Games	Free
	Tico Puzzles	Puzzles	Free
	Electricity Consump...	Accessories	Free
	Packet Sender	Developer Tools	Free
	BtResourceSearch(##...	Accessories	Free
	ZHackers: Volume T...	Books & Magazines	US\$ 2.99
	CubeCraft Games	Games	Free
	Mahjong Epic 2	Board Games	Free

**Recommended For You**

To make recommendations, Ubuntu Software Center will occasionally send to Canonical a list of software currently installed.

**Top Rated**

Icon	Name	Category	Rating	Count	Price
	GParted Partition ...	Themes & Tweaks	★★★★★	(453)	Free
	Geany	IDEs	★★★★★	(395)	Free
	VLC media player	Sound & Video	★★★★★	(1647)	Free
	GeoGebra	Mathematics	★★★★★	(66)	Free
	Stellarium	Astronomy	★★★★★	(399)	Free
	Ubuntu restricted e...	System	★★★★★	(316)	Free
	FileZilla	File Sharing	★★★★★	(837)	Free
	Synaptic Package ...	Themes & Tweaks	★★★★★	(285)	Free
	7zip	Accessories	★★★★★	(461)	Free
	Midnight Comm...	Accessories	★★★★★	(36)	Free
	SMPlayer	Sound & Video	★★★★★	(336)	Free
	Hugin PTO generator	Graphics	★★★★★	(54)	Free

46295 items

(bas niveau, beaucoup plus de détails)

File Edit Package Settings Help

Reload Mark All Upgrades Apply Properties

All
Amateur Radio
Communication
Communication (contrib)
Communication (non free)
Cross Platform
Cross Platform (contrib)
Cross Platform (non free)
Databases
Debug
Debug (contrib)
Debug (non free)
Development
Development (contrib)
Development (non free)
Documentation

Sections
Status
Origin
Custom Filters
Search Results
Architecture

S	Package	Installed Version	Latest Version	Description
<input type="checkbox"/>	libxft2	2.1.0-2	2.1.0-2	debugging informations for libxft
<input type="checkbox"/>	libxft-dev	2.1.0-2	2.1.0-2	debugging informations for libxft
<input type="checkbox"/>	libxftgui4-4	4.8.1-5	4.8.1-5	Basic GUI C functions for Xfce4
<input type="checkbox"/>	libxftgui4-4-dbg	4.8.1-5	4.8.1-5	debugging informations for libxft
<input type="checkbox"/>	libxftgui4-dev	4.8.1-5	4.8.1-5	Development files for libxftgui4
<input type="checkbox"/>	libxftconf-0-2	4.8.1-1	4.8.1-1	Client library for Xfce4 configure i
<input type="checkbox"/>	libxftconf-0-2-dbg	4.8.1-1	4.8.1-1	debugging informations for libxft
<input type="checkbox"/>	libxftconf-0-dev	4.8.1-1	4.8.1-1	Development files for libxftconf
<input type="checkbox"/>	libxfce4util-dev	1.5.0-4	1.5.0-4	X11 miscellaneous 'fixes' extensio
<input checked="" type="checkbox"/>	libxfce4util3	1.5.0-4	1.5.0-4	X11 miscellaneous 'fixes' extensio
<input type="checkbox"/>	libxfce4util3-dbg	1.5.0-4	1.5.0-4	X11 miscellaneous 'fixes' extensio
<input checked="" type="checkbox"/>	libxfce4util3:i386	1.5.0-4	1.5.0-4	X11 miscellaneous 'fixes' extensio
<input type="checkbox"/>	libxfce4util-dev	1.1.4.5-2	1.1.4.5-2	X11 font rasterisation library (dev
<input checked="" type="checkbox"/>	libxfce4util1	1.1.4.5-2	1.1.4.5-2	X11 font rasterisation library
<input type="checkbox"/>	libxfce4util1-dbg	1.1.4.5-2	1.1.4.5-2	X11 font rasterisation library (deb

No package is selected.

42128 packages listed, 2099 installed, 0 broken. 0 to install/upgrade, 0 to remove

# Encore plus de distributions ...



debian / getting debian

[About Debian](#) [Getting Debian](#) [Support](#) [Developers' Corner](#)

## Getting Debian

Debian GNU/Linux is distributed [freely](#) over Internet. You can download all of it from any of our [mirrors](#). The [Installation Manual](#) contains detailed installation instructions.

If you simply want to install Debian, these are your options:

### [Download a small installation image](#)

**Use Internet to download additional files during installation.**

These small "netinst" images can be downloaded quickly and should be

### [Download large installation images](#)

**Useful when the install target has no Internet connection.**

The CD/DVD images can be downloaded using [HTTP/FTP](#), [BitTorrent](#), or [Jigdo](#).

[Download](#) > [Overview](#) [Cloud](#) [Server](#) [Desktop](#) [Ubuntu 桌面版](#)

## Get Ubuntu

Ubuntu is completely free to download, use and share.



### Ubuntu Desktop >

Download Ubuntu desktop and replace your current operating system whether it's Windows or Mac OS, or, run Ubuntu alongside it.

# Il y a plein de distributions ...

 **Linux Mint**  
from freedom came elegance


HomeCommunity**Download**ServicesProjectAbout


Linux Mint 16Linux Mint DebianAll versionsDocumentationBuy CDs


## Download Linux Mint 16 Petra




### Information

Our latest release is Linux Mint 16, codename "Petra".



 [Donate](#) [Participate](#) [Download](#)



**Follow us**

**Sponsors**

Download opensUSE 13.1

switch to **Development Version**

### Installation Medium



#### 4.7GB DVD (also suitable for USB stick)

Contains a large collection of software for desktop or server use.  
Suitable for installation or upgrade.

**Download Method** ?

**Type of Computer** ?

**Documentation**

### Getting Help

[Download Help](#)  
[Burn CD/DVD Image\(s\)](#)  
[Installation Guides](#)  
[Official Manuals](#)  
[Package Repositories](#)  
[Reporting Bugs](#)

### Sponsored by