

110 Définition et rôles

INF3173

Principes des systèmes d'exploitation

Jean Privat

Université du Québec à Montréal

Hiver 2021

Plan

- 1 Définition des systèmes d'exploitation
- 2 Rôles d'un système d'exploitation
- 3 Bibliothèques et démons

Définition des systèmes d'exploitation

Définition des systèmes d'exploitation

Questions essentielles

- C'est quoi ?
- À quoi ça sert ?
- Comment c'est fait ?

Problèmes

- Pas de définition formelle parfaite
 - Pas de liste précise des rôles, des tâches ou des composantes
- Ça évolue en fonction des besoins et des ressources

Tentative de définition

Un **système d'exploitation** est une **couche logicielle** qui sert d'intermédiaire entre les **utilisateurs** et les ressources **matérielles** de l'ordinateur et qui offre un **environnement** d'exécution aux **programmes** qui se veut efficace, robuste et utilisable.

Composantes d'un ordinateur

Grossièrement, un ordinateur comporte...

Du « matériel »

Processeur, mémoire, disques, périphériques, etc.

Un « système d'exploitation »

Qui semble (*un mal*) nécessaire pour

- Configurer le matériel
- Installer et exécuter des applications

Exemples: Windows, Debian, MacOS, Android

Des « programmes d'application »

Les vrais logiciels utiles à l'utilisateur

Système d'exploitation : deux points de vue

Clé en main (avec programmes système)

- C'est le point de vue grand public
- Inclut toute sorte de programmes : shells, interfaces graphiques, utilitaires de base, outils de configuration, etc.

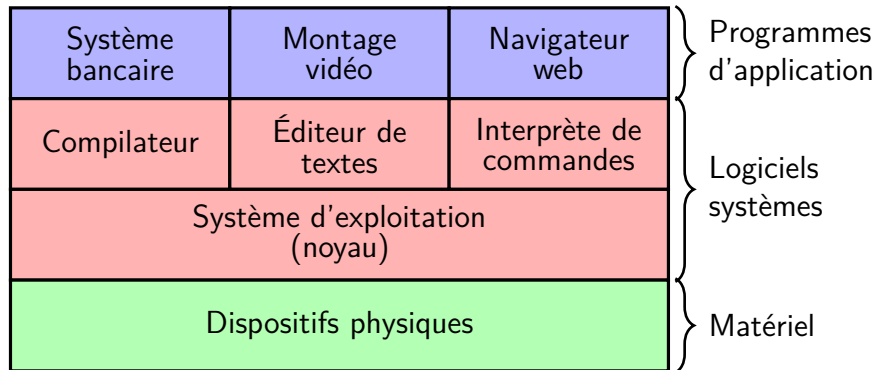
Espace disque recommandé pour une installation normale

- **Debian 10** \approx 10Go d'espace disque
- **Windows 10** \approx 32Go d'espace disque

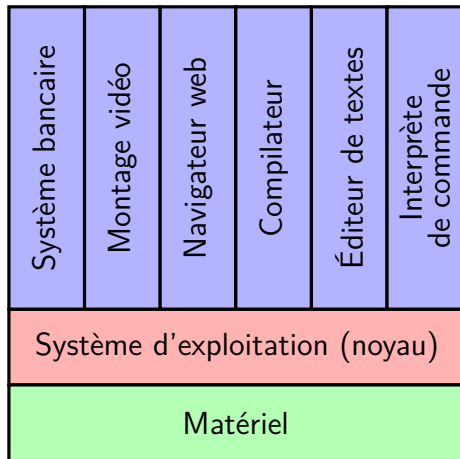
Noyau seul

- C'est le point de vue du cours
- Seulement la couche logicielle toujours en cours d'exécution
- Exemple: Linux (qui servira d'exemple dans le cours)
- Espace disque: Debian linux-image-5.7.0-2-amd64 \approx 300 Mo

Composantes d'un ordinateur : vision de l'utilisateur



Composantes d'un ordinateur : notre vision



Vision du SE = notre vision

- Une seule catégorie de programmes
- Pas d'accès direct entre les programmes et le matériel

Composantes d'un système d'exploitation

Qu'a-t-on le droit de mettre dans un système d'exploitation ?

1998 Microsoft face à la justice américaine

- Abus de position dominante à cause des restrictions empêchant la désinstallation d'Internet Explorer.
- [Source Wikipedia](#)

2020 Microsoft Edge impossible à désinstaller

- [Source](#)

Rôles d'un système d'exploitation

SE = Couche d'abstraction

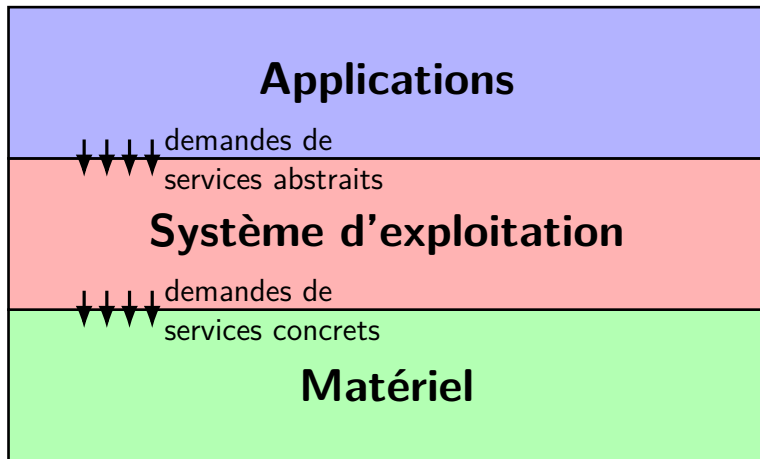
Abstrait la machine

- Cache certains détails que l'utilisateur n'a pas à connaître pour exploiter la machine
- Présente à l'utilisateur une machine virtuelle facile à utiliser et à programmer
- Offre toute sorte de services abstraits: gestion des fichiers, communication entre programmes, etc.

Connaît

- Connaît les détails internes intimes de la machine
- Utilise les services concrets (matériels) de la machine

SE = Couche d'abstraction



SE = Couche d'abstraction

Pour les applications ?

- Pas besoin d'être spécifique à chaque matériel possible
- Y compris du matériel qui n'existe pas encore
- Mais peuvent devenir spécifique à un système d'exploitation
- Développer des applications portables entre différents systèmes d'exploitation est plus difficile

Pour les matériels ?

- Développement de **pilotes** (*driver*) spécifiques au système d'exploitation
- Mais tous les systèmes d'exploitation sont pas égaux
- Version de système d'exploitation non maintenu, matériel discontinué, effort de développement non rentable

On y reviendra

SE = Couche d'abstraction

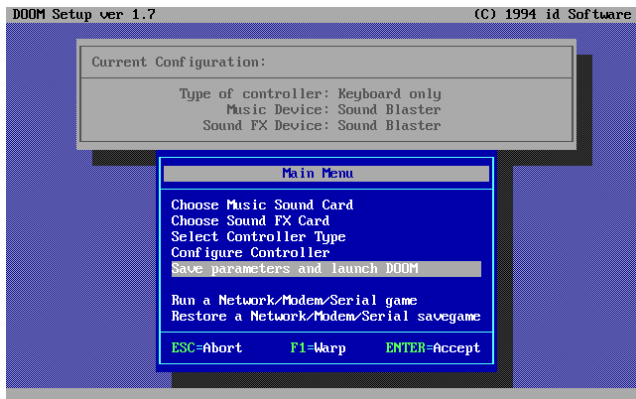
Source de Linux v5.7

```
$ du -sh * | sort -hr | head
626M    drivers
134M    arch
50M    Documentation
45M    include
42M    fs
40M    sound
39M    tools
33M    net
10M    kernel
5,9M    lib
```

Plus de **70%** du code source dans drivers et arch.

Question. Expliquer la commande Unix

Jouer à Doom sur PC dans les années 90



- DOS n'offrait pas d'abstraction pour les cartes son
- Chaque jeu vidéo devait gérer entièrement un ensemble fixé de cartes sons
- L'utilisateur voyait des détails matériels: IRQ, DMA, etc.

SE = Gestionnaire de ressources

Répartir efficacement les ressources limitées

- Temps CPU
 - Mémoire
 - Périphérique (ex. disques, réseau, imprimante, webcam, etc.)
- Quelles politiques adopter ?
- Quels mécanismes mettre en œuvre ?

Faire cohabiter pacifiquement plusieurs processus et utilisateurs (et matériels)

- Protection mémoire
 - Gestion des droits
 - Respect des répartitions des ressources
- Comment imposer ça ?

Gestion des processus

- Création et destruction des processus
- Décider de l'attribution processeur aux processus
- Suspendre et continuer les processus
- Permettre la synchronisation et la communication des processus

Gestion de la mémoire

- Répartir la mémoire entre les processus
- Gérer l'espace libre et les demandes de mémoire
- Décider du passage en mémoire distante

Gestion des fichiers

- Création, manipulation et destruction des fichiers et répertoires
- Gestion de l'espace disque libre
- Gestion des droits

Gestion des périphériques

- Gestion de la mémoire, cache, tampons, IRQ, DMA
- Pilotes spécifiques
- Gestion de l'énergie (batterie)
- Répartition des ressources entre processus (bande passante disque, réseau, etc.)

Bibliothèques et démons

Exclusivité du SE ?

Besoins

- Abstraire du matériel
- Offrir des services
- Gérer des ressources

Des approches existent déjà

- Bibliothèques logicielles
- Services, démons, serveurs

Qu'est-ce qui rend les systèmes d'exploitation **différents** ?

Bibliothèques logicielles

C'est quoi?

Composantes logicielles prêtes à l'usage par des programmes

- Exemple: bibliothèque cryptographique
- Compilées ou non, statiques ou dynamiques (.so, .dll)
- Offrent une interface abstraite aux programmes (API/ABI)

Avantages

- Permet de factoriser du **bon** comportement
- Mise à jour indépendante des bibliothèques dynamiques partagées

Services, démons, serveurs

C'est quoi?

Processus s'exécutant en arrière-plan qui répondent à des requêtes

- Servent aussi à gérer l'activité de périphériques
- Exemple: serveur d'impression
- Offrent des services abstraits via des mécanismes de communication entre processus

Avantages

- Permet de sous-traiter du **bon** comportement
- Mise à jour indépendante des logiciels

Exclusivité du SE

Qu'est-ce qui rend les systèmes d'exploitation différents ?

Les privilèges

Le système d'exploitation a le monopole de privilèges exclusifs

- Tout accès au matériel passe nécessairement par lui
- Toute allocation de ressource à un logiciel sera respectée
- Sauf, bien sûr, si le système d'exploitation autorise des formes de contournement

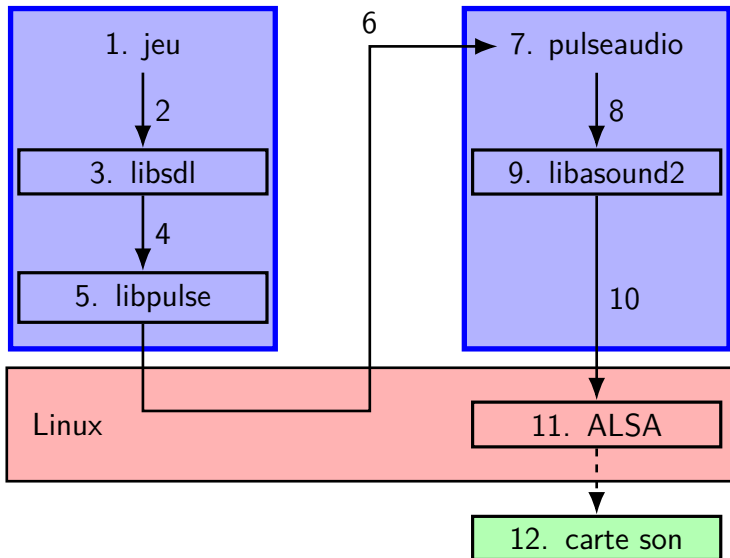
En pratique

Bibliothèques et services vont souvent encapsuler des services système pour harmoniser, simplifier (ou complexifier) les choses

- En informatique, on aime résoudre les problèmes en ajoutant un nouveau niveau d'abstraction

Exemple: le son

- 1 Un jeu vidéo a besoin de son
- 2 Il utilise la bibliothèque `libsdl`
- 3 `libsdl` expose une API portable entre différents systèmes
- 4 `libsdl` peut utiliser `libpulse` pour jouer les sons
- 5 `libpulse` simplifie la communication avec `pulseaudio`
- 6 via des services systèmes d'IPC (*interprocess communication*)
- 7 `pulseaudio` est un serveur de son (démon) qui gère la configuration complexe des éléments audio de l'ordinateur
- 8 `pulseaudio` utilise la bibliothèque `libasound2`
- 9 qui simplifie l'utilisation des services de son de Linux
- 10 cela via des services système dédiés
- 11 `ALSA` (*Advanced Linux Sound Architecture*) est un morceau (sous-système) du noyau Linux
- 12 qui peut accéder à la carte son



Version simplifiée sans la libc ni libdl.

Exemple: le son

Le système d'exploitation isole les processus

Un processus ne peut agir directement

- sur un périphérique
- sur un autre processus

→ il doit passer par le système d'exploitation

Questions

Un processus pourrait-il...

- Utiliser directement `libpulse` sans passer par `libSDL` ?
- Faire de l'IPC directement avec `pulseaudio` sans passer par `libpulse` ?
- Communiquer directement avec `pulseaudio` sans passer par des services du noyau d'IPC ?
- Utiliser les services noyau d'ASLA sans passer `libasound2` ?
- Agir sur la carte son sans passer par des services du noyau ?

Analogie : Le SE est le gouvernement de l'ordinateur

Il ne sert à rien en soi

- Pas directement utile à l'utilisateur

Il permet la cohabitation *pacifique* entre

- Les différents programmes
- les différents utilisateurs
- les différents matériels

Il possède les capacités d'imposer cette cohabitation

- Il a le monopole de privilèges particuliers

Mécanismes et politiques

Reste une question : comment on fait ça ?

- Mécanismes : quels sont les moyens à mettre en oeuvre
- Politiques : quelles sont les règles à appliquer

Séparer les deux questions permet plus de souplesse

- D'une part fournir des mécanismes de base
 - D'autre part concevoir
- il est possible d'adapter les politiques sans devoir tout refaire

Philosophie Unix

Le plus souvent:

- Le noyau
- Expose des mécanismes simples ou élémentaires
- Les programmes et les administrateurs système
- Définissent des politiques
- Les implémentent à l'aide des mécanismes