

# Système d'exploitation : introduction

---

**Julien Forget**

Université Lille 1  
École Polytechnique Universitaire de Lille  
Cité Scientifique  
59655 Villeneuve d'Ascq

---

**GIS3**

2012 - 2013



# Qu'est-ce qu'un Système d'Exploitation ?

## Definition

Un Système d'Exploitation (Operating System) est un ensemble de programmes qui :

- Met à disposition les ressources matérielles de l'ordinateur ;
- Sert d'interface entre l'utilisateur et l'ordinateur ;
- Fait abstraction des spécificités complexes du matériel.

## Différentes classes de SE

Grande diversité suivant le type de matériel ou d'application :

- SE **mono** vs **multi-tâche** : partage du temps de calcul du processeur entre plusieurs programme ;
- SE **mono** vs **multi-utilisateur** : plusieurs utilisateurs peuvent accéder à l'ordinateur simultanément ;
- SE **distribué** : le SE gère plusieurs ordinateurs simultanément et répartit l'utilisation des ressources sur ce réseau ;
- SE **embarqué** : SE dédié à l'utilisation sur des ressources plus limitées (mémoire, capacité de calcul) ;
- SE **temps-réel** : assure des temps de réponse prédictibles (respect des échéances temporelles).

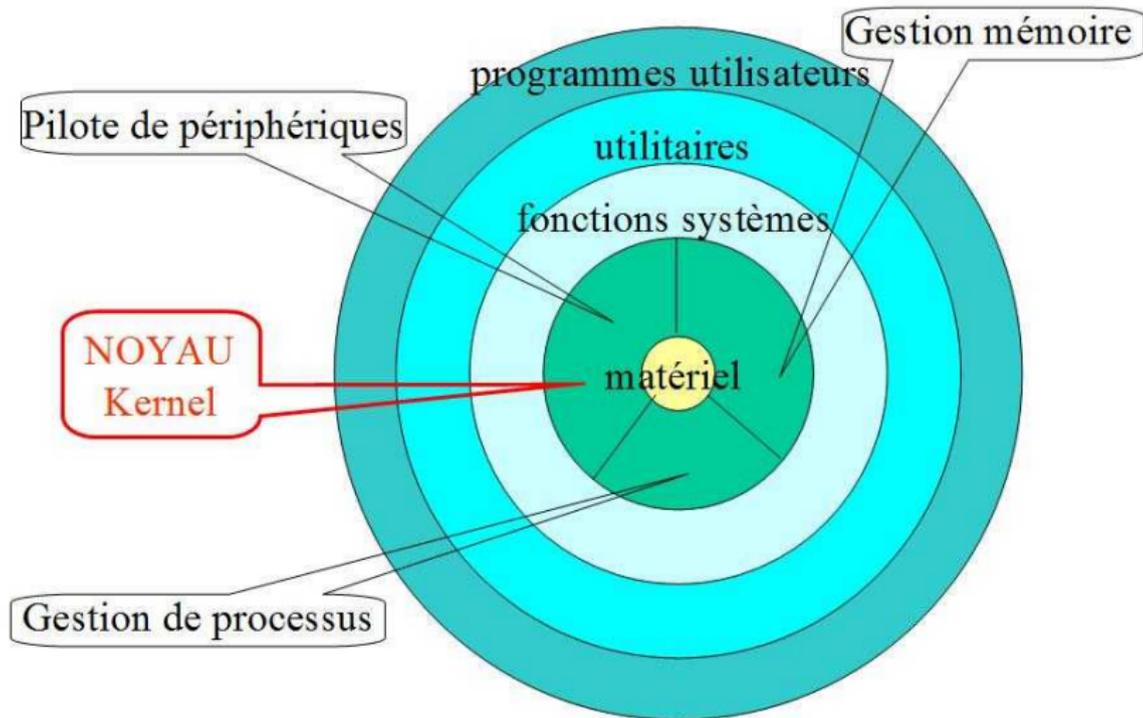
# UNIX

- 1971 : première distribution. Multi-tâche, multi-utilisateur ;
- 1973 : première version portable en C (défini pour l'occasion) ;
- Projet GNU (1983) : objectif de développer un SE **libre** ;
- Linux (1991) : un noyau UNIX libre développé par Linus Torvald  
⇒ premier OS complet GNU/Linux libre ;
- Distributions GNU/Linux : Debian, Ubuntu, Mandriva...
- Mac OS X (1999) : dédié aux ordinateurs Macintosh ;
- Android (2007) : embarqué ;
- RTAI, RTLinux : temps-réel.

# DOS/Windows

- MS-DOS (1981) : pour le 1er PC d'IBM. Mono-tâche, mono-utilisateur ;
- Windows 1.0 (1985) : multi-tâche, mono-utilisateur
- Windows NT (1993) : multi-tâche, multi-compte mais un seul utilisateur simultanément ;
- Windows CE (1996) : embarqué ;
- Windows 7 (2009) : multi-tâche, multi-compte.

# Structure d'un SE



## Structure d'un SE (2)

- **Le bloc matériel** (couche physique, *hardware*) : ressources matérielles (processeur, mémoires, périphériques, ...)
- **Le noyau** : gère l'accès aux processeurs, la mémoire, le contenu des disques, contrôle les échanges avec les différents périphériques ;
- **Fonctions systèmes** : bibliothèque standard d'appels système ;
- **Utilitaires** : interpréteurs de commandes (*shell*), éditeurs, compilateurs, gestionnaire de fenêtres et de bureau, etc.

# Plan du cours

Nous verrons :

- 1 Langage de commandes ;
- 2 Gestion de processus ;
- 3 Gestion mémoire ;
- 4 Gestion de disque.

# Langages de commandes

**Objectif** : (ré)-apprendre à se servir efficacement d'un terminal de commandes (Shell) :

- Utilisation de la ligne de commande ;
- Commandes classiques ;
- Scripts.

# Gestion de processus

**Objectif** : comprendre comment sont exécutés les programmes :

- Exécution multi-tâche ;
- Ordonnancement ;
- Gestion de processus communicants.

# Gestion mémoire

**Objectif** : comprendre comment la mémoire est allouée aux processus :

- Séparer la mémoire de différents processus ;
- Le va-et-vient entre mémoire et disque (swap) ;
- La pagination ;
- La segmentation.

# Gestion disque

**Objectif** : comprendre comment sont stockées les données sur des supports non volatiles (disque dur, clé USB, etc) :

- La vision utilisateur : fichiers, répertoires, droits d'accès ;
- La vision système : organisation des données sur le disque.

# Organisation de l'enseignement

- 20h de cours (Julien Forget) ;
- 22h de TP (Nathalie Devesa, Julien Forget, Laure Gonnord, Thomas Vantrois) ;
- 2 heures d'Examen sur table : tous documents papier autorisés, tous dispositifs portables interdits ;
- 2h de contrôle TP : tous documents papier autorisés.

# Bibliographie

Quelques références bibliographiques (cf Bibliothèque Universitaire) :

- *Systèmes d'exploitation*, Andrew Tanenbaum ;
- *Unix, Linux et les systèmes d'exploitation*, Michel Divay ;
- *Le noyau Linux*, Daniel Bovet et Marco Cesati.