

# Systemes d'exploitation

Introduction

# L'équipe

---

- Cours:
  - Alexandru Radovici
- TP & Devoir:
  - Cristiana Andrei
- Ressources
  - Răzvan Deaconescu, Mihai Carabaş et l'équipe USO et SO de ACS

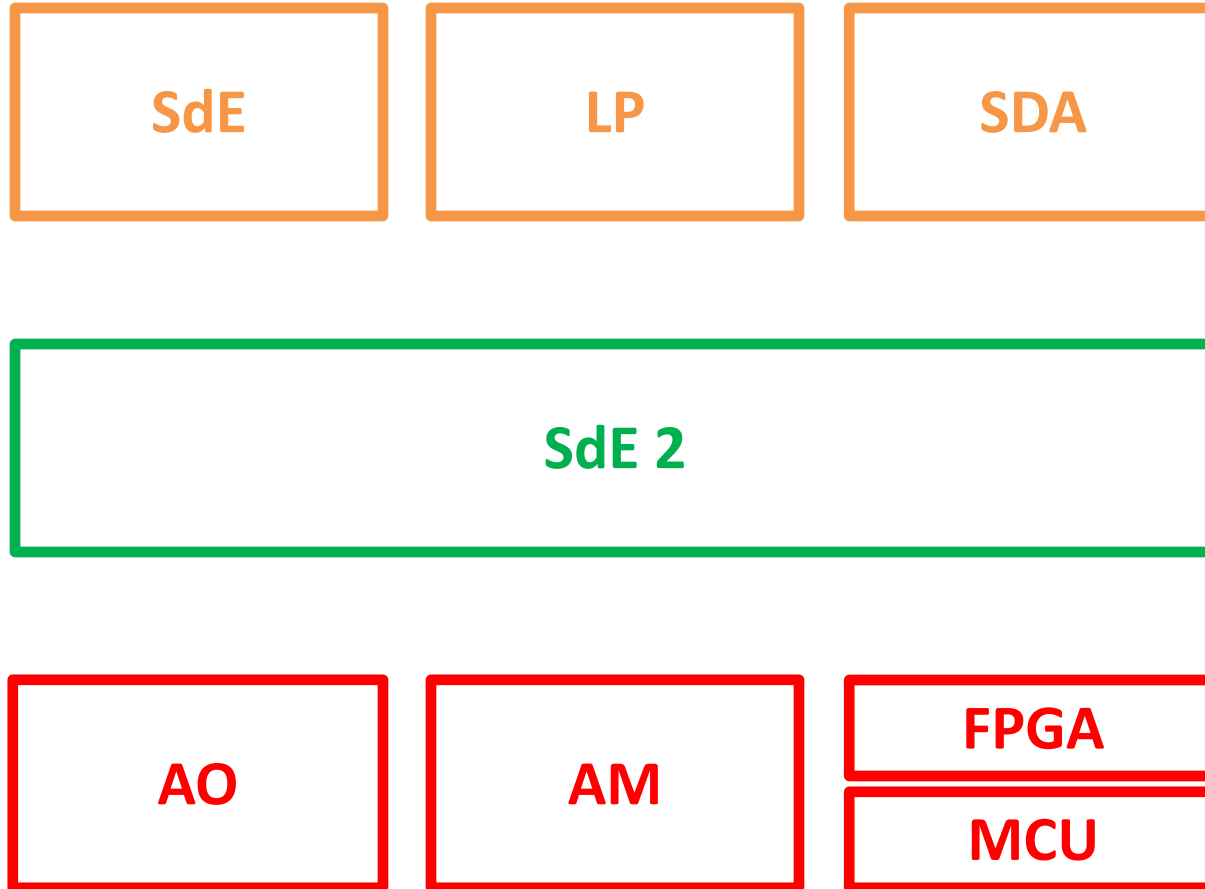
# Règles du cours

---

- Nous vous conseillons de venir au cours
- Si vous venez en classe, vous devez respecter ces règles
  - soyez attentif et **posez le plus de questions possible**

# Les courses de systèmes

---



# Deux parcours

---

## SdE 2 - Normal

- Cours
  - 14
- Programmation
  - Programmation en Java
  - POO
  - Pointeurs
- Langage utilisée
  - Rust
- Note maximale
  - 10

## SdE 2 - Simplifiée

- Cours
  - 5
- Utilisation
  - Linux
- Langage utilisée
  - bash
- Note maximale
  - 6

# Bibliographie

---

**Andrew Tanenbaum**, *Modern Operating Systems*,  
*2<sup>nd</sup> Edition*

**A. Silberschatz, P. Baer Galvin, G. Gagne**, *Operating Systems Concepts*, 9<sup>th</sup> Edition

**Steve Klabnic, Carol Nichols**, *The Rust Programming Language*, <https://doc.rust-lang.org/book/>

# Ressources pour le cours SdE2

---

- Site web: <http://ocw.cs.pub.ro/courses/sde2/>
- GitHub issues: <https://github.com/UPB-FILS-SdE2/questions/issues>
- Diapositives de cours
- Catalogue et calendrier Google
- La bibliographie

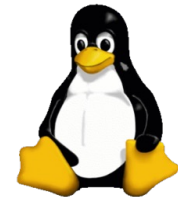
# Vous avez besoin

---

- Compte sur Github
  - nom real et photo real



- *Linux*
  - Windows Subsystem for Linux
  - Machine Virtuelle (VirtualBox, wmware)





## Cours

- 14 courses
- diapositives
- bibliographie
  - Très important de lire

## TP

- 12 TP
- Linux
- *Programmation en Rust*
- C'est important de collaborer avec vos collègues

## Contenu

- Intro en Rust
- Mini Shell
- Planificateur
- ELF Loader
- Serveur web

## Développent

- Linux ou *Windows Subsystem for Linux*
- 8-20 heures par une devoir
  - Test des devoirs avec github classroom
- Questions sur Github Issues

**Les devoirs sont individuelles**

# Vous saurez

---



**The Rust  
Programming  
Language**



Linux

# Examen

---

- Semestre
    - 5 devoirs (8p)
  - Session d'examen
    - épreuve écrite (6p)
  - Note
    - > 5 passer
    - 10 présence TP
    - 1.5 moyenne de TP ou > 5p en devoirs
- \* *l'utilisation de matériel est nécessaire*

- **Vous n'avez pas de devoirs copiés**
- **Vous avez au moins 11 présences au TP**

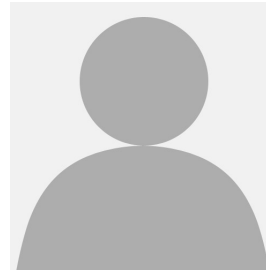
# Hall of Fame



Irina Bardu  
2021



Student



Student



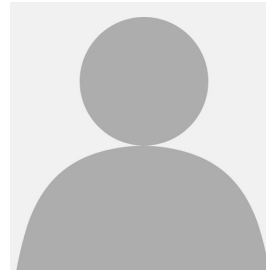
Student



Student



Student



Student



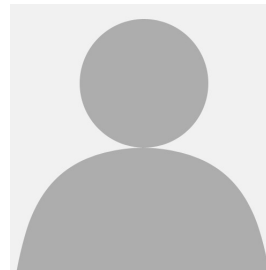
Student



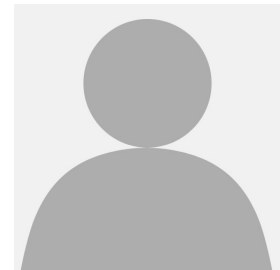
Student



Student



Student



Student

## Understanding Computer Technology

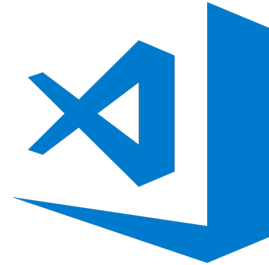


# Outils logiciels recommandés

---



GitHub Codespaces



Visual Studio Code



Chrome ou Edge



Ubuntu Linux



Windows 10  
Windows Subsystem for Linux



VirtualBox





- Américain
- Physicien
- Livre de SdE
- Minix OS
  - Mini-OS
  - Intel Management Engine

- Quelques mots sur SdE
- Introduction en Systèmes d'exploitation
- Sujets

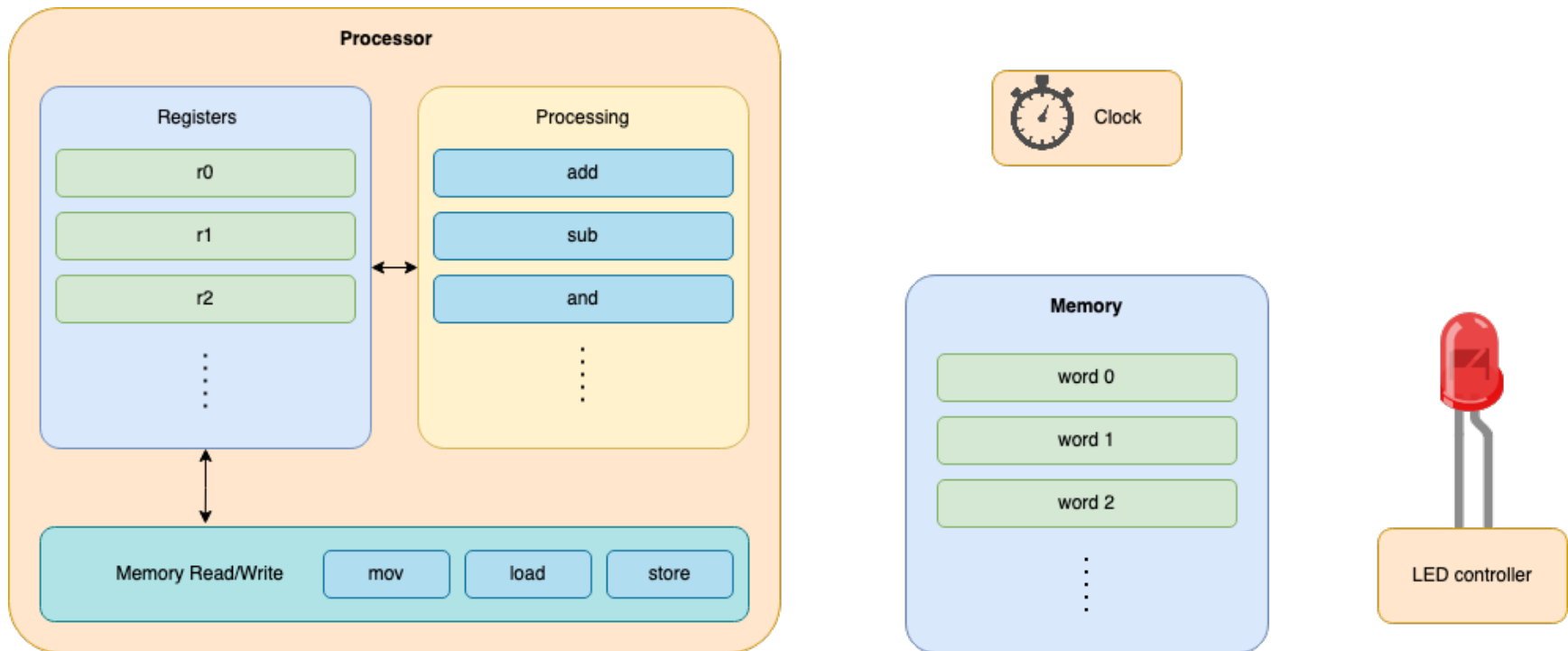


# Bibliographie pour aujourd'hui

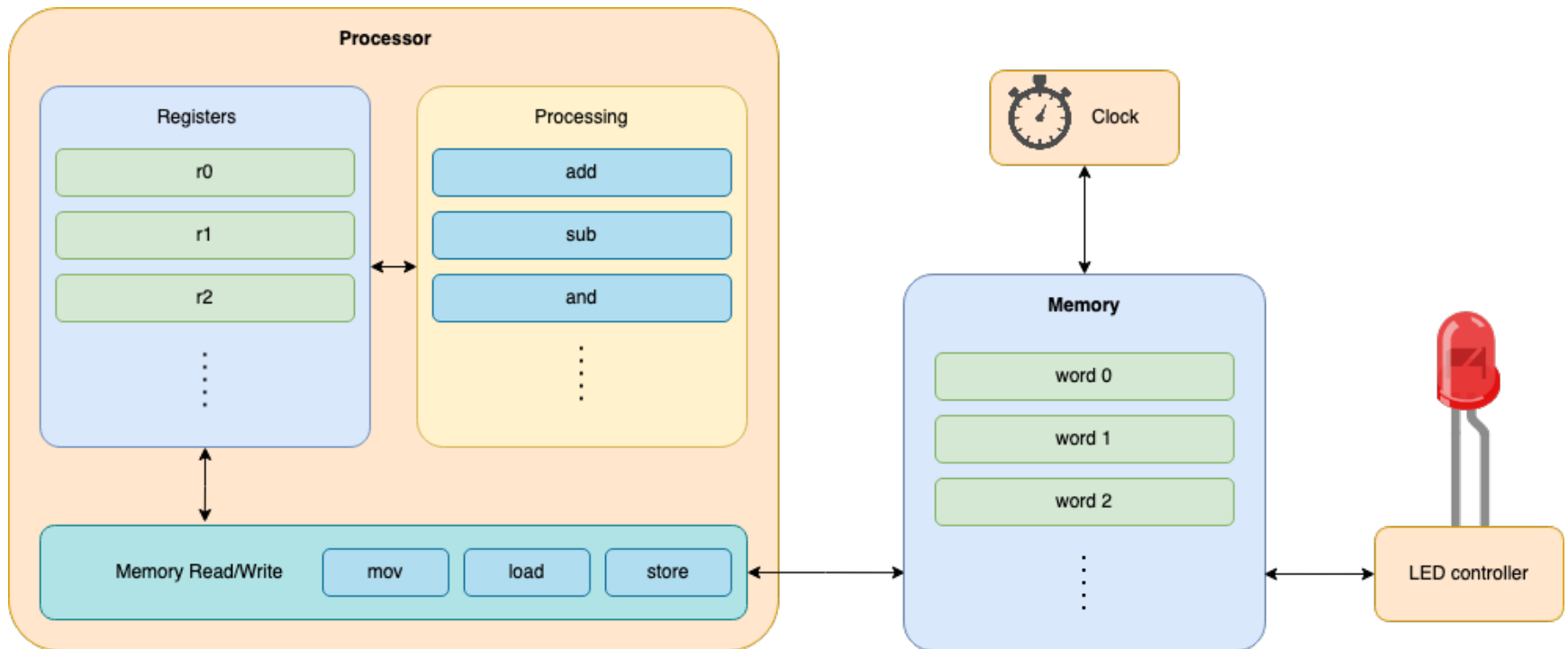
---

- Modern Operating Systems
  - Chapitre 1
- Operating Systems Concepts
  - Chapitre 1

# Systeme Informatique



# Systeme Informatique





# MATERIEL

# Materiel

---

- **Processeur (CPU)**
  - exécute le code (instructions)
- **Mémoire** du travail (RAM)
  - mémorise les données et le code pour les programmes
- **Bus** des données
  - fait la connexion entre le CPU, le RAM et les périphériques E/S
- **Périphériques** (entrée/sorties - E/S, *input/output* - I/O)
  - communication avec l'extérieur: utilisateur, autres systèmes, etc.
- Espace de **stockage** (disque, flash, ROM, NVRAM)
  - programmes (à partir de laquelle les processus seront faits)
  - données pour les processus
  - information pour les utilisateurs (fichiers)

## Microcontrôleur

- Fréquence basse (MHz)
- Petit espace de stockage (Mo)
- RAM limitée (Ko)
- Faible consommation d'énergie
- Pas cher

## L'ordinateur

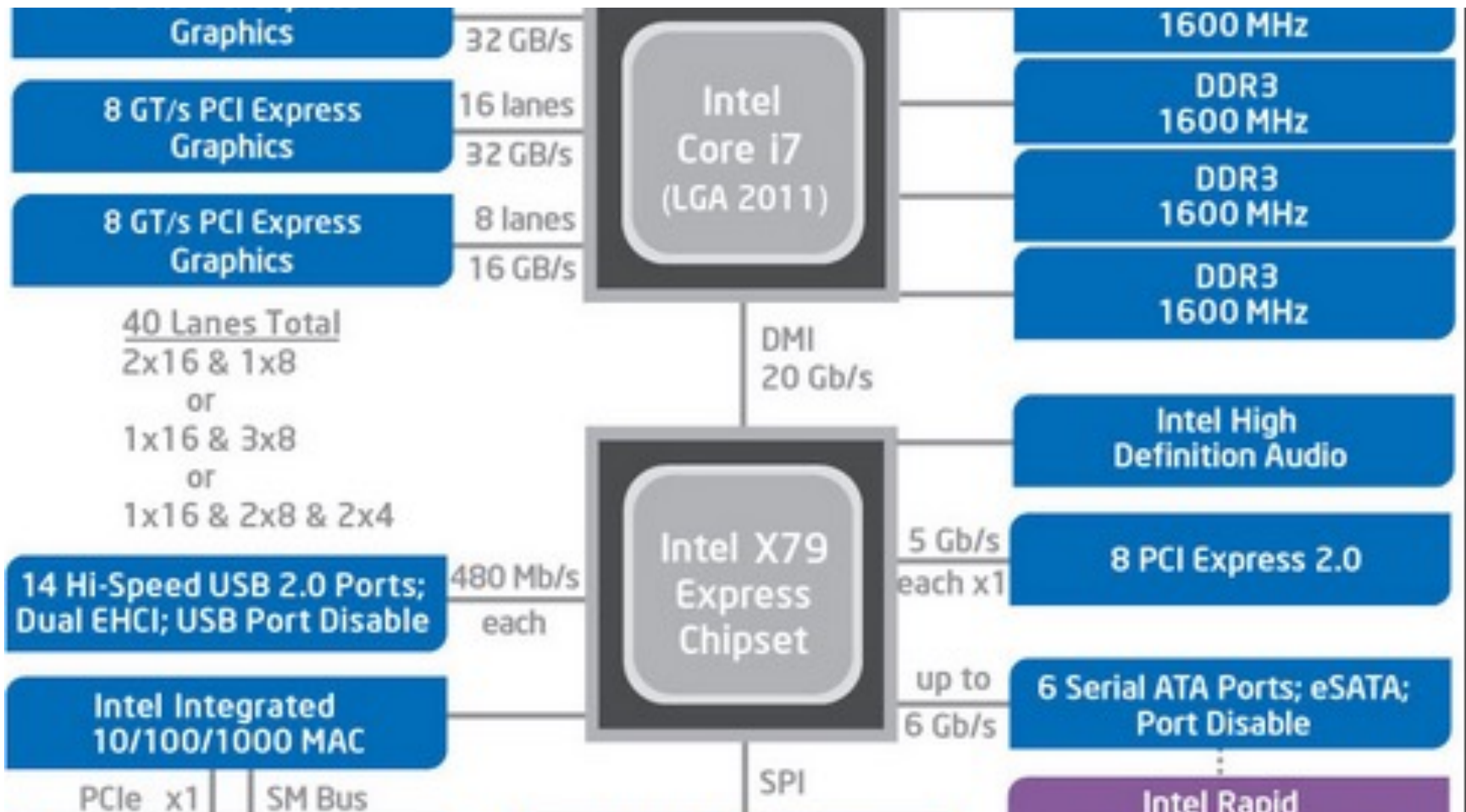
- Fréquence élevée (GHz)
- Grand espace de stockage (dizaines de Go)
- Grande RAM (Go)
- Consommation d'énergie plus élevée
- Chers



# PC



# PC





# SYSTEME D'EXPLOITATION

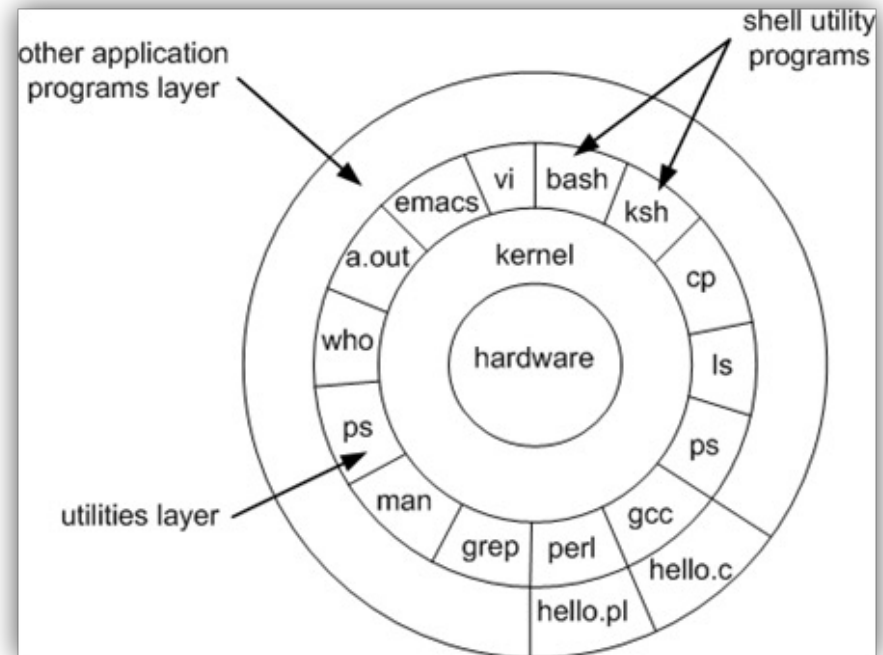
# Qu'est-ce qu'un SE?

---

- un programme
- vue de haut en bas: extension de la machine physique
- vue de bas en haut: gestionnaire des ressources physiques
- écrit en C ou Rust
- transparent pour l'utilisateur (*il marche*)

# La structure de SE

- **Noyau (kernel)**
  - gérer les ressources physiques
  - fournit une interface standard pour les applications
- **Programmes de base**
  - fournit une interaction utilisateur avec le noyau et le matériel
  - exemples: creation de fichiers, access des donnes sur la resaeu



# Windows

---

- Microsoft
- le plus utilisé pour les systèmes de bureau
- La version plus récente: Windows 10 (pour PC, tablettes et mobiles)
- Windows Server 2012 R2 (pour serveurs)



Windows

# macOS

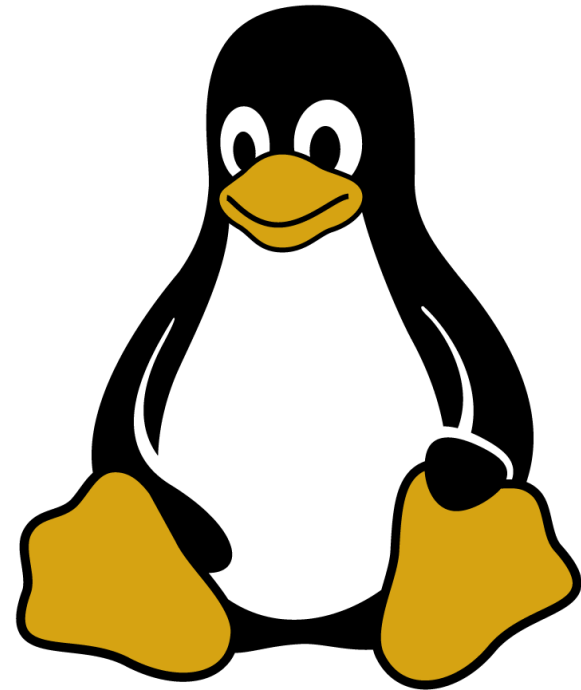
---

- Apple
- Seulement pour les systèmes Mac (Mac Pro, Mac Mini, MacBook)
- Approximative 9-10% pour Desktop
- La version plus récente: macOS 11 “Big Sur”



Mac<sup>TM</sup>OS

- Linus Torvalds et Greg Kroah-Hartman (personnes principales)
- open source
- distributions de Linux
- approximative 1-2% pour Desktop
- Utilisé pour les serveurs
- Utilisé pour dispositifs intégrée et mobiles (Android et autres)





# Autres SE

---



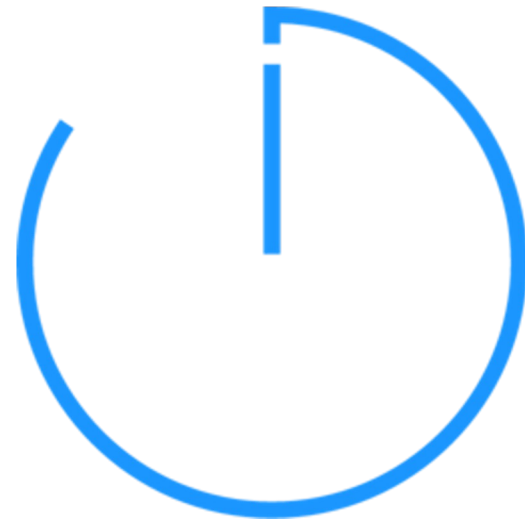
# Zephyr

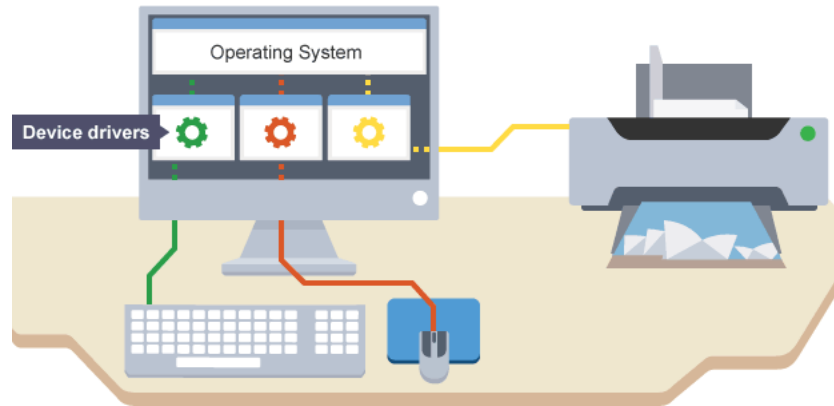
- Open source
- Linux en miniature
  - pour les dispositifs
- Binaire unique
- Lancé en 2016
- [zephyrproject.org](http://zephyrproject.org)



# Tock

- Open source
- Mini système
  - pour les dispositifs
  - développé à Rust
- Lancé en 2016
- [www.tockos.org](http://www.tockos.org)

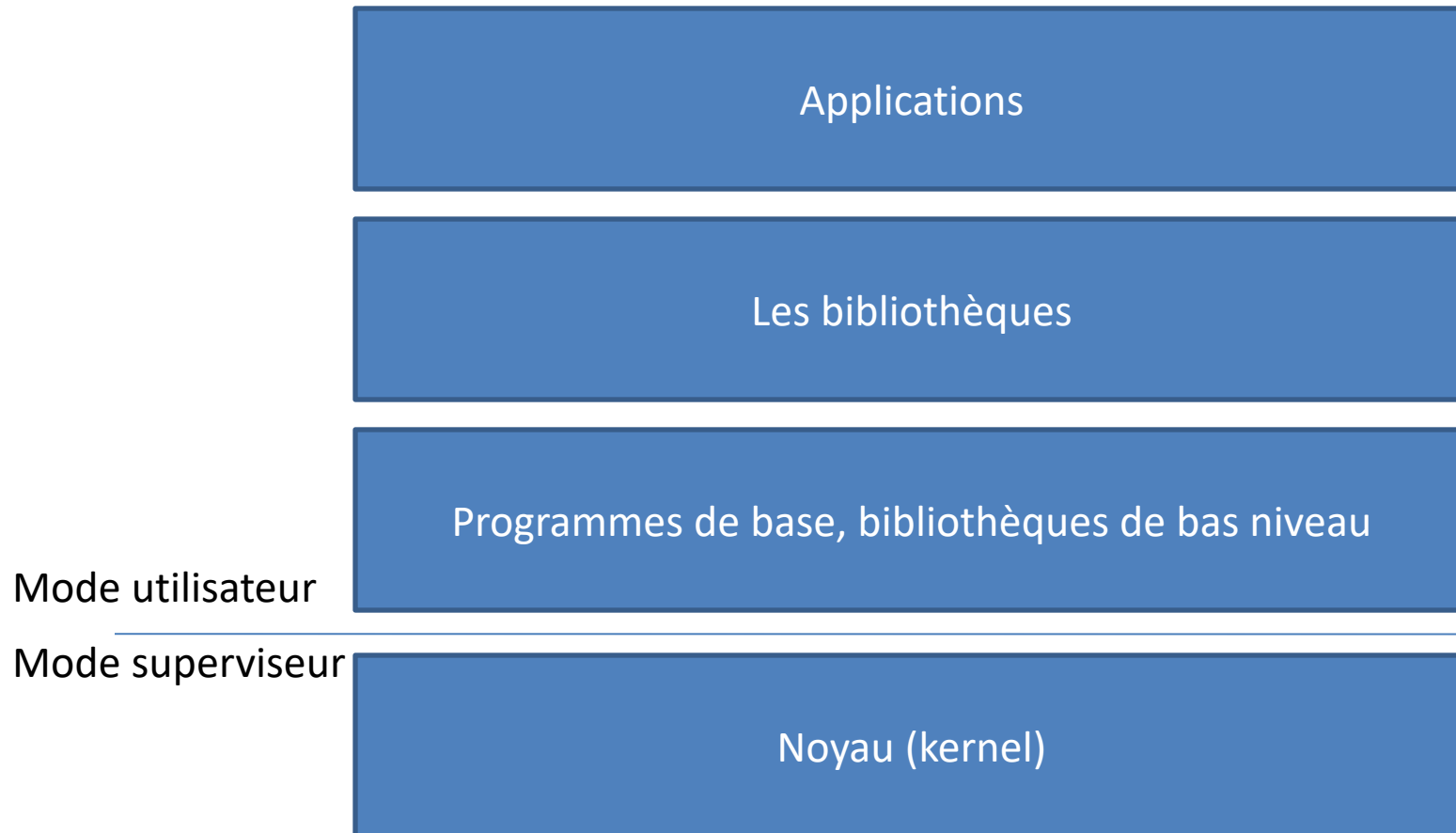




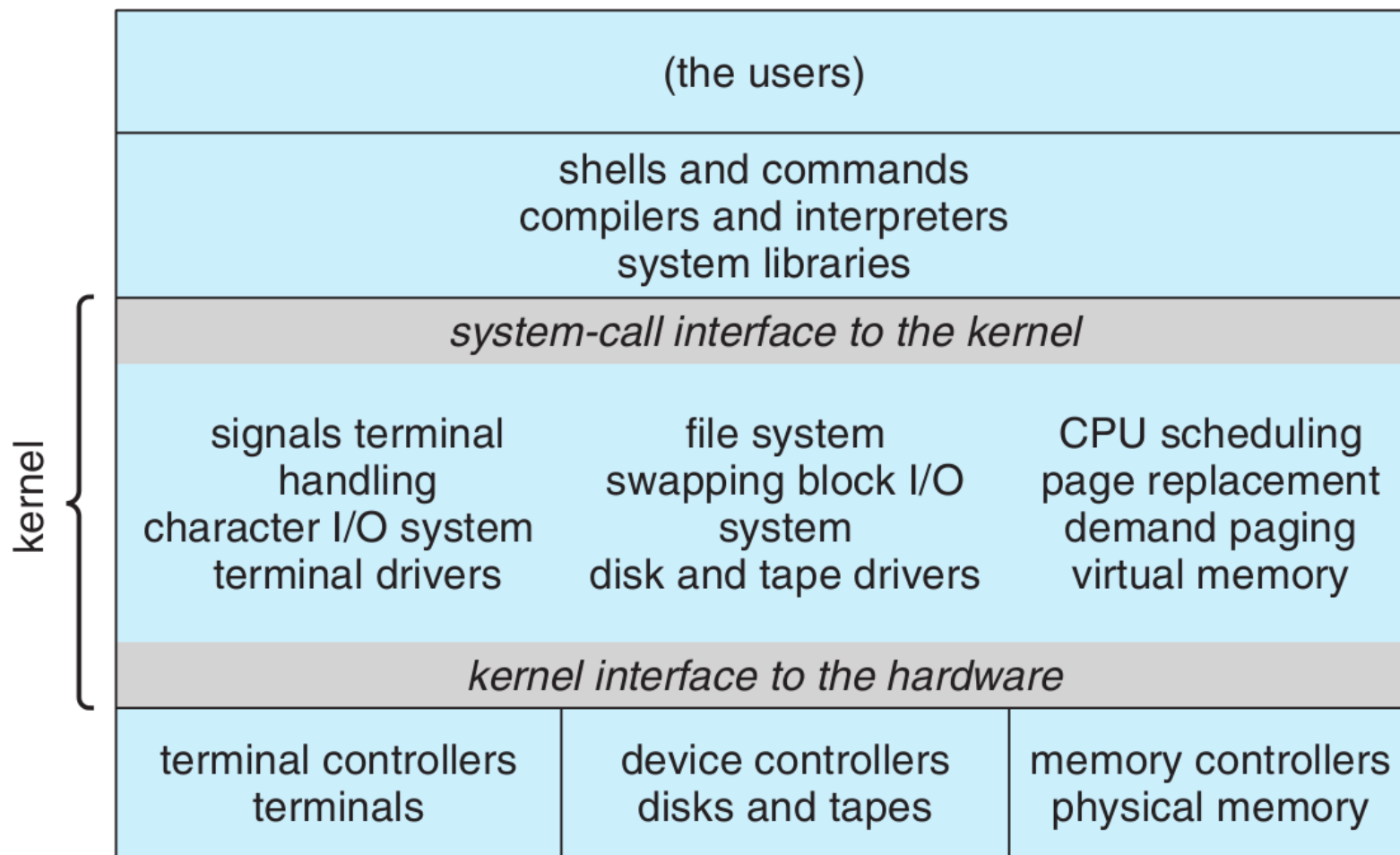
# STRUCTURE DE SE

# La pile de systèmes informatiques

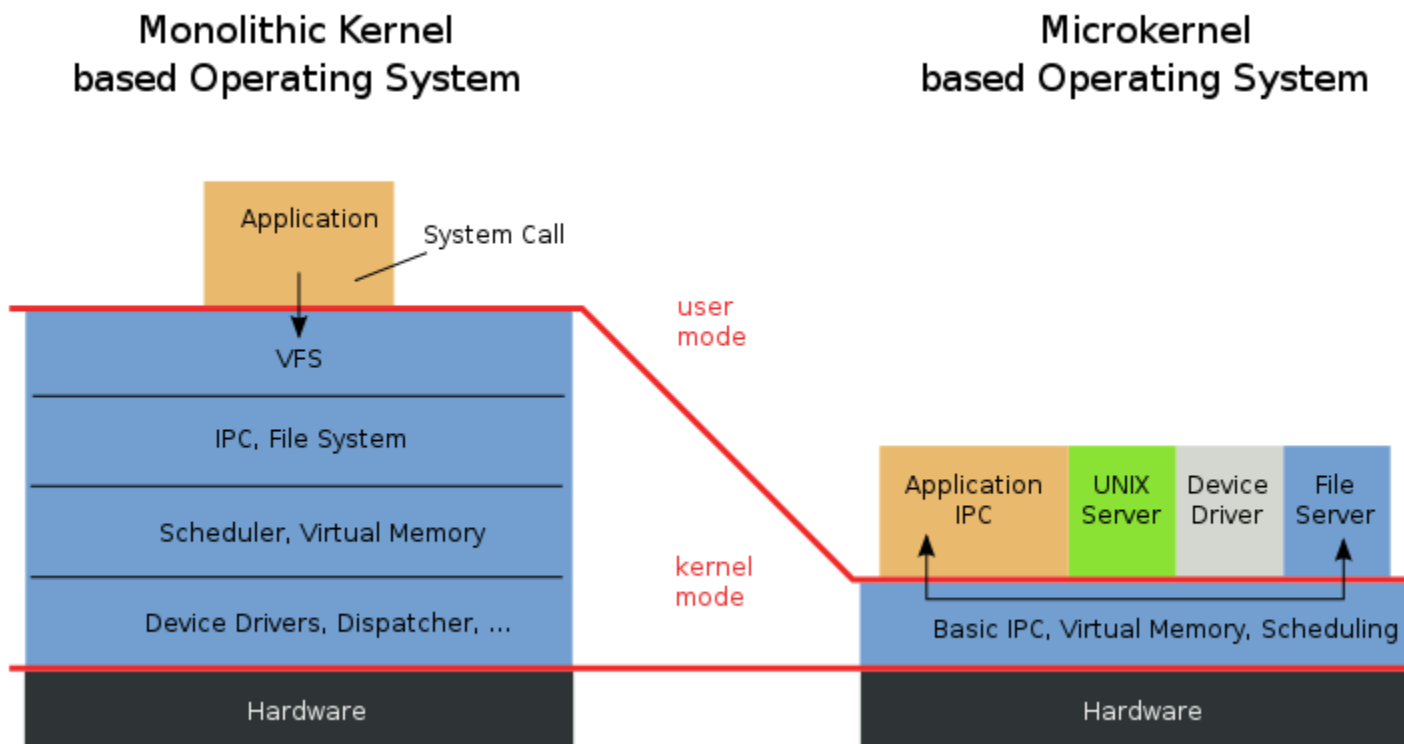
---



# La structure de SE (monolithique)



# SE monolithique vs. SE micronoyau



<https://en.wikipedia.org/wiki/Microkernel>

# SE monolithique vs. SE micronoyau

---

## Monolithique

- Efficient
- Communication par appel de fonction entre les composants
- Moins flexible
- Grand TCB (*Trusted Computing Base*) - design moins sécurisé

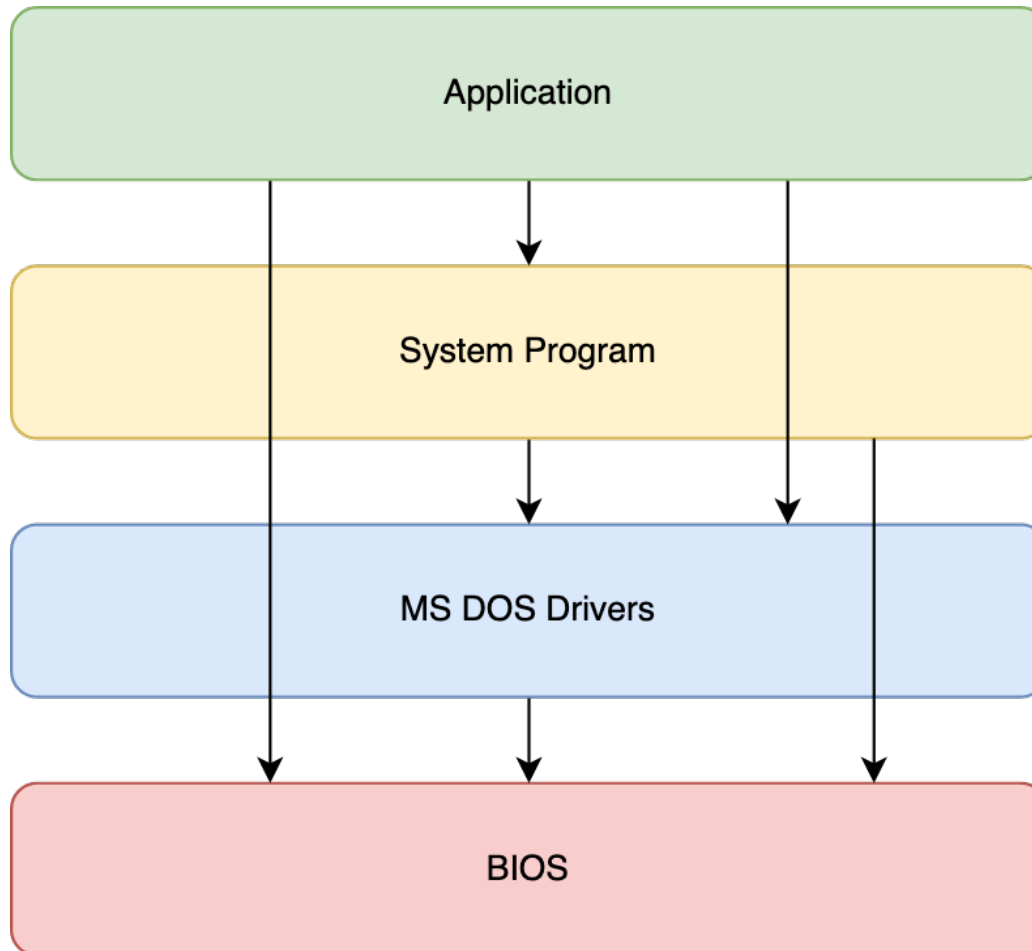
## Micronoyau

- est plus lent (communication entre services)
- plus modulaire
- Réduite TCB (design plus sécurisé)

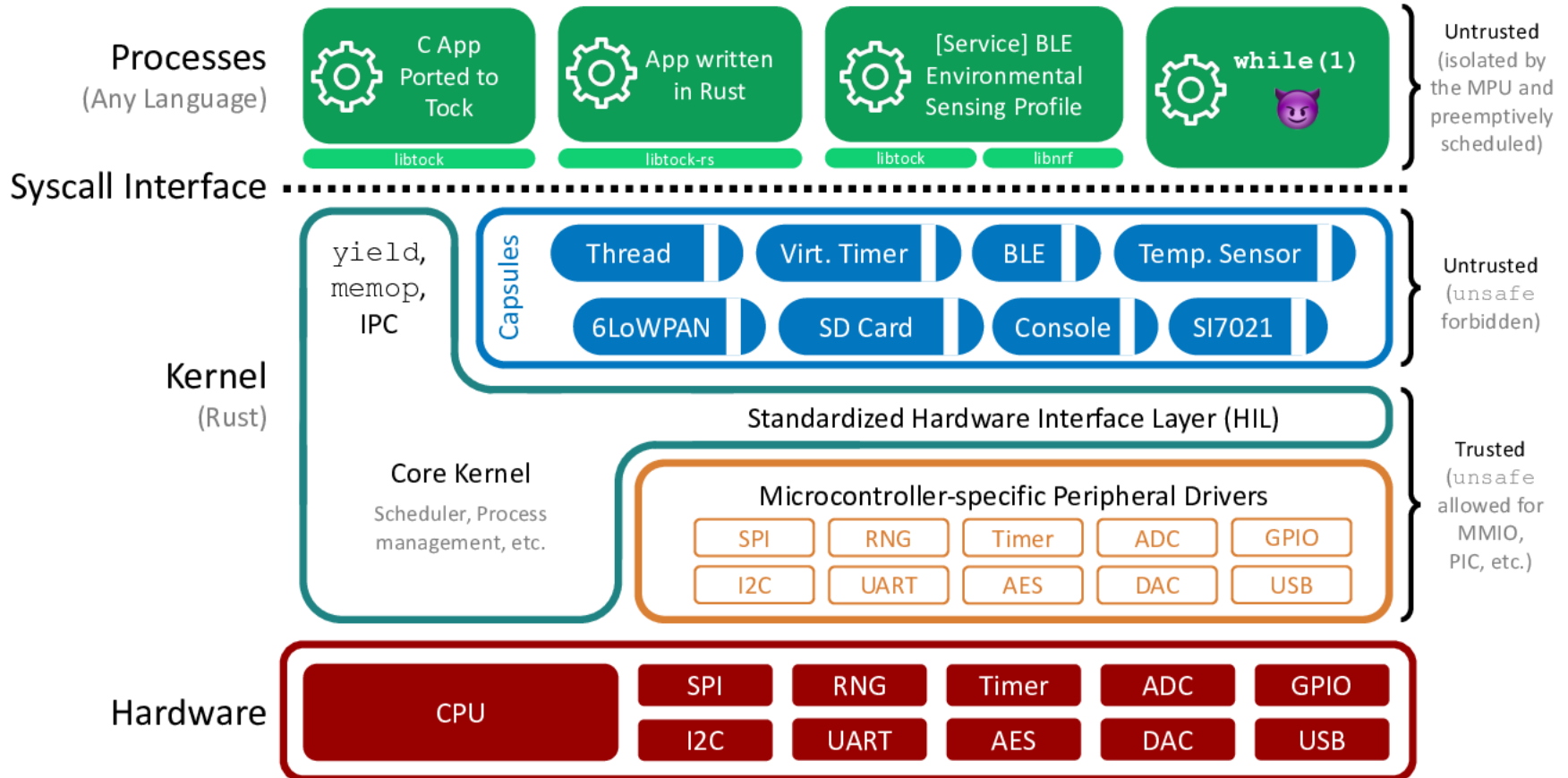


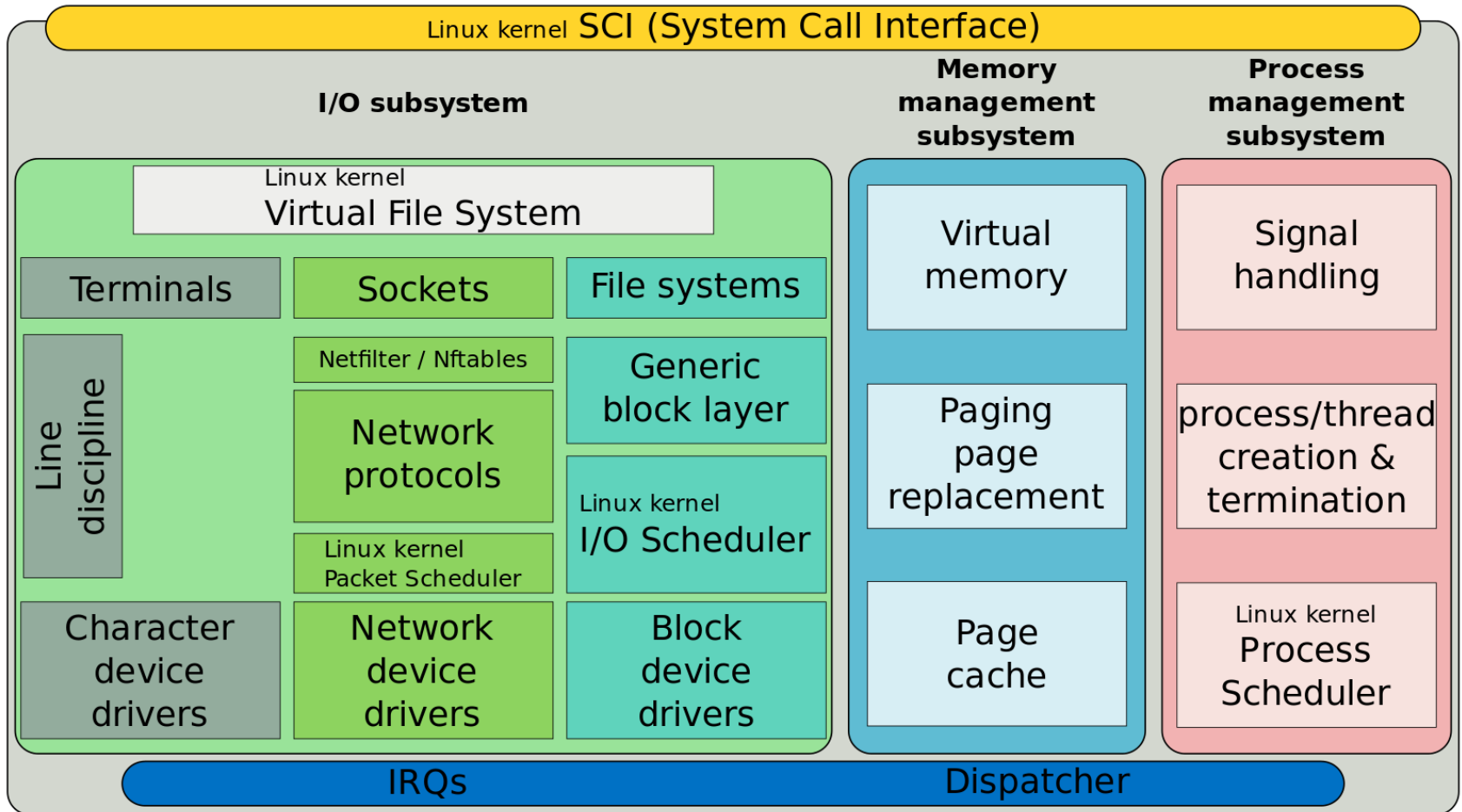
# MS DOS

---



# Tock





# Ressources pour le cours SdE2

---

- Site web: <http://ocw.cs.pub.ro/courses/sde2/>
- GitHub issues: <https://github.com/UPB-FILS-SdE2/questions/issues>
- Diapositives de cours
- Catalogue et calendrier Google
- Les machines virtuelles
- La bibliographie

- SE
- Noyau (*kernel*)
- Mode superviseur
- Mode utilisateur
- CPU
- Mémoire
- Bus de données
- Processus
- Mémoire virtuelle
- Espace utilisateur
- Espace noyau
- Appel de système
- Noyau monolithique
- Micronoyau
- Programmes de base
- Interface Utilisateur

# Questions

---

