

# Introduction au développement logiciel

## Programmation système

---

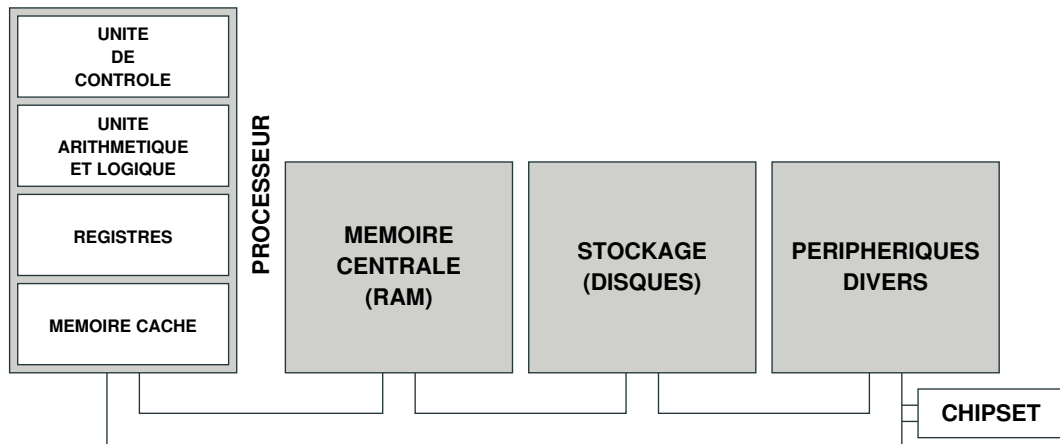
Georges-André Silber, Centre de recherche en informatique (CRI)

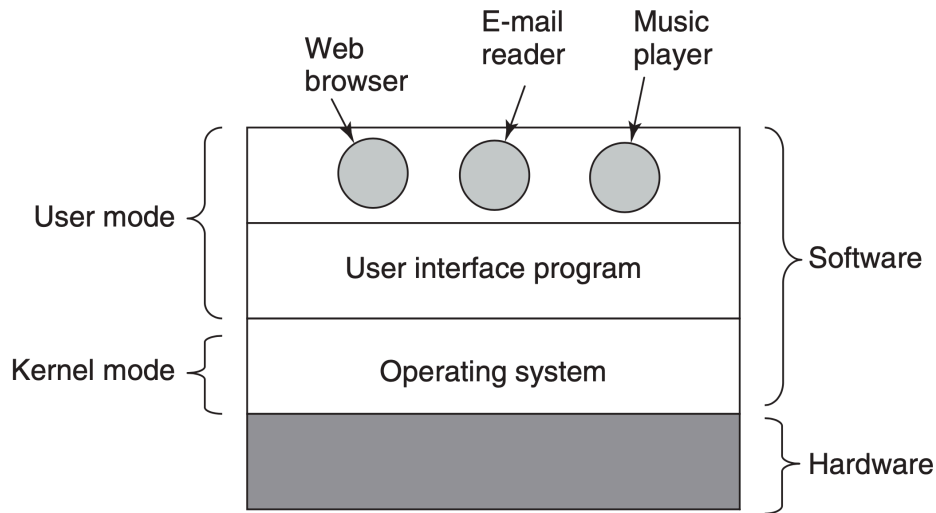
Mines Paris — PSL, décembre 2025

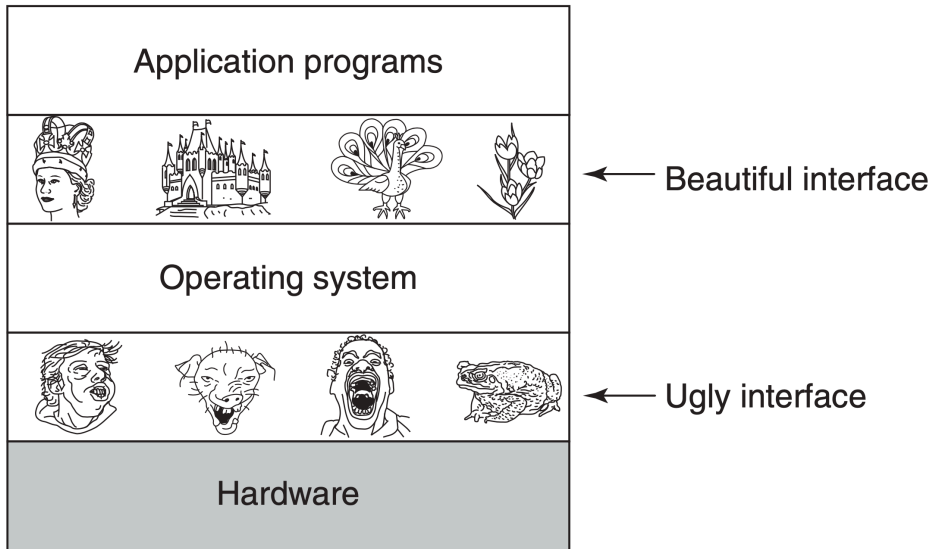
- Programmation à bas niveau, proche du matériel
- Systèmes d'exploitation, pilotes de périphériques
- Émulateurs et virtualiseurs
- Création de distributions (OS complets)
- Utilisation de langages "proches de la machine"
- C, Rust

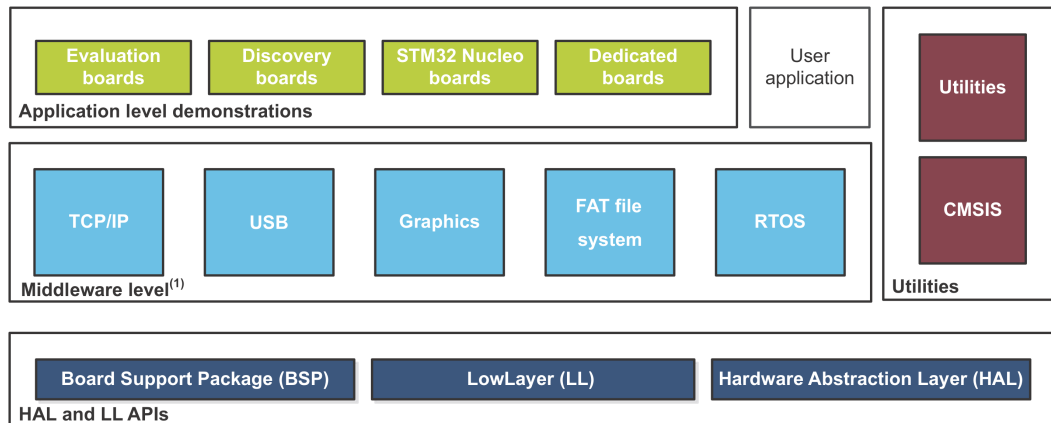
***low-level language*** *A primitive programming language in which each line of code needs, but never gets, 20 lines of comment.*

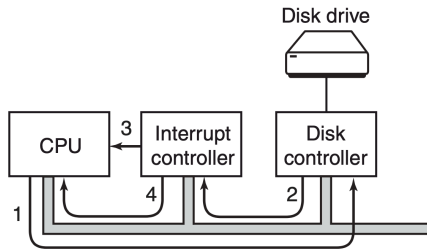
— *The Devil's Dp Dictionary*



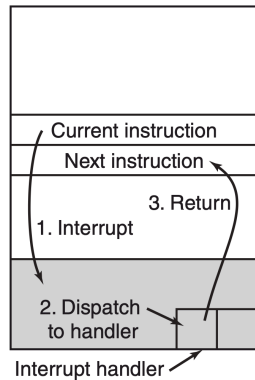




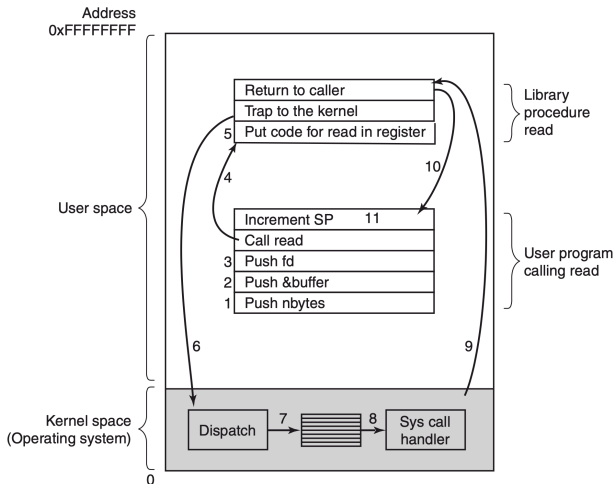




(a)

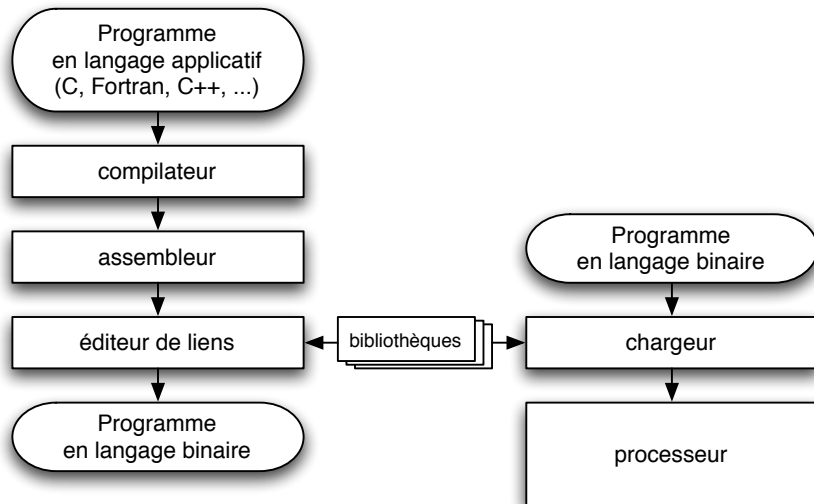


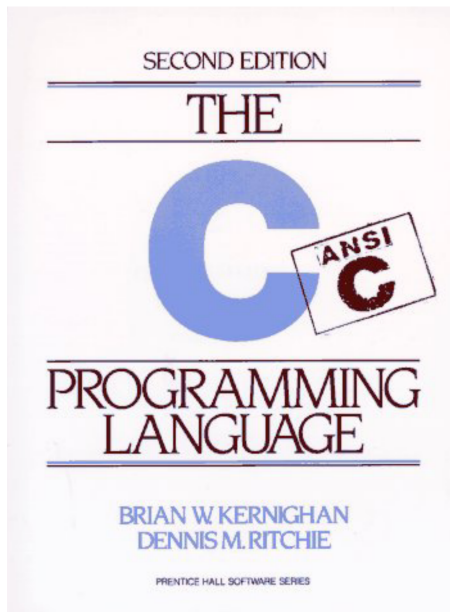
(b)



- Appels système Linux
- <https://filippo.io/linux-syscall-table/>
- `man syscall`
- `man syscalls`
- `man 2 intro`

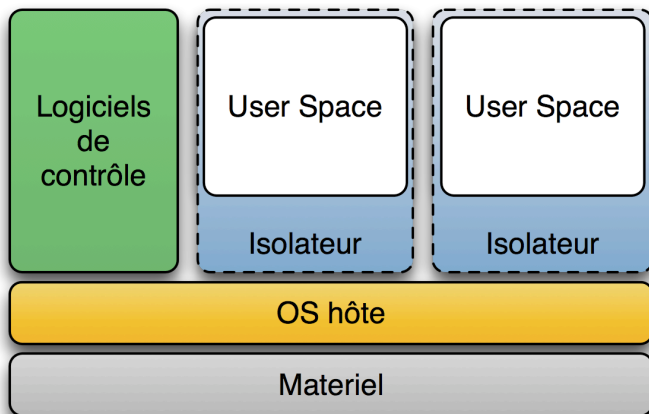




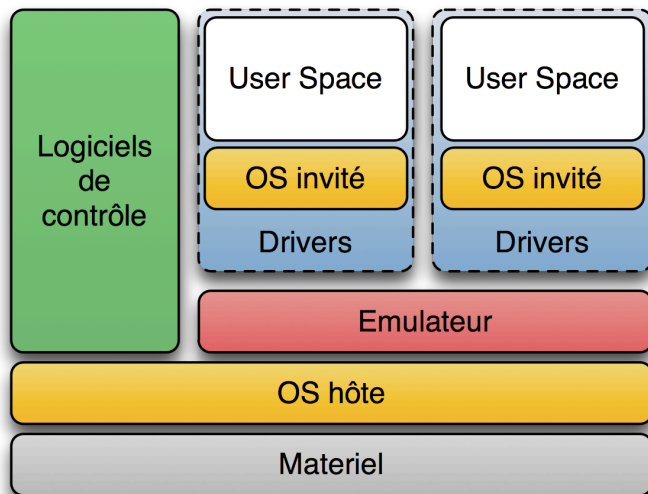


- Dennis Ritchie
- Brian Kernighan
- 1972 → C23 (2024?)
- <https://godbolt.org>

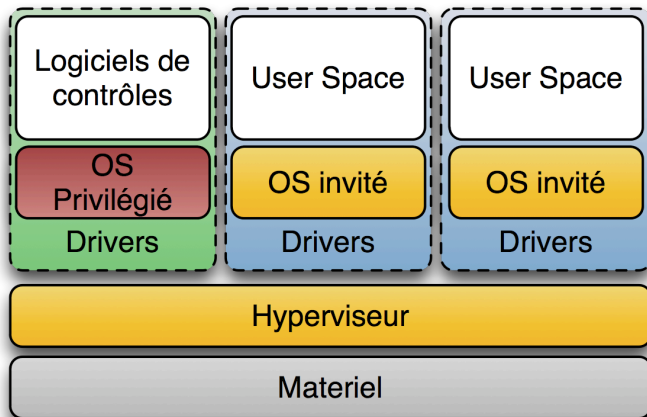
- chroot, Jails de FreeBSD
- *cgroups* et *namespaces* de Linux : LXC, LXD, Docker



- VirtualBox, QEMU



- Xen, KVM





- Firmware en mémoire Flash
- Accès bas niveau à un périphérique
- Lecture du MBR (512 octets)
- Bootloader + partition disque
- Chargement bootloader niveau 2
- Chargement binaire plus compliqué
- etc...