

# Interfaccia dei file (UNIX)

Principi:

**uniformità** utilizzata per file, IPC e I/O, con le stesse chiamate.

**open before use** consente al sistema di controllare i permessi e preparare la comunicazione (strutture kernel space ed eventualmente inizializzazione della periferica). Permette anche di implementare accesso esclusivo su alcuni dispositivi.

**orientata al byte** le operazioni operano su buffer di byte, anche per dispositivi che lavorano a blocchi.

**buffer kernel** permette all'applicazione e al dispositivo di effettuare operazioni a velocità diverse.

**chiusura esplicita** per liberare le risorse.

Chiamate di sistema:

**open** apre un file, con la possibilità di crearlo se non esiste, e restituisce il file descriptor. È importante che il test di esistenza, la creazione e l'apertura siano atomici. **creat** esiste per motivi storici.

**close** chiude un fd. Quando un processo termina tutti i suoi file sono chiusi automaticamente.

**read/write** legge/scrive un certo numero di byte in/da un buffer fornito dall'utente.

**stat** restituisce informazioni su un file.

**unlink** cancella un file, che però non viene effettivamente rimosso finché ci sono processi con un fd associato ancora aperto.

**pipe** crea un buffer kernel e restituisce due fd per quel buffer, il primo RO e l'altro WO. Utile per IPC. Le scritture fino a PIPE\_BUF byte ( $\geq 512$ , su Linux 2048) sono atomiche.

**dup** copia il fd specificato nel primo fd libero. **dup2(from, to)** permette di specificare la destinazione. Utile per modificare stdin, stdout e stderr prima di una **exec**.