

5 – Memory mapped files

mickael.hoerdt@hesge.ch

Introduction

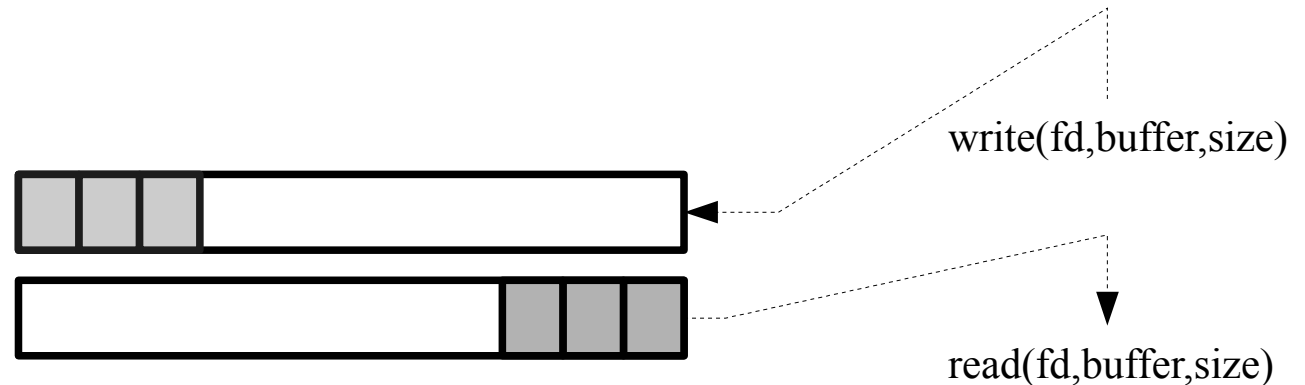
Les fichiers mappés en mémoire

Appels système de mappage de fichiers en mémoire

Exemple

Introduction

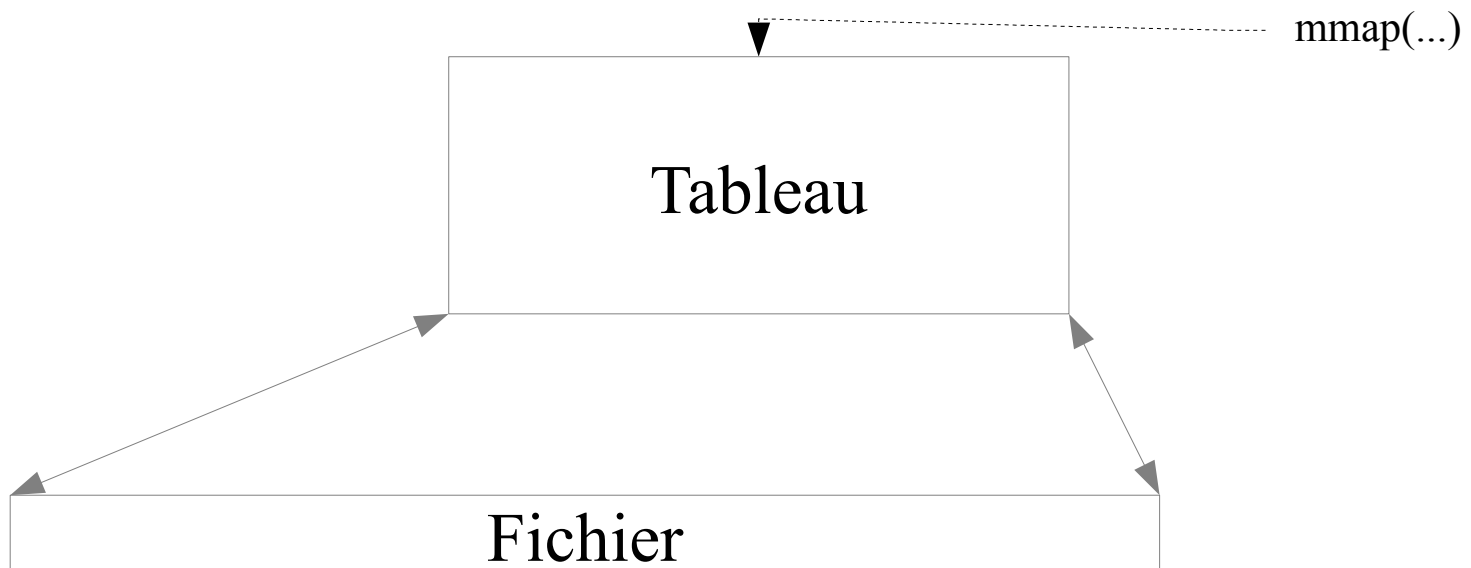
Les fichiers sous Unix sont vus comme des flux d'octets nommés par leur path , accessibles via un buffer de lecture/écriture.



Cette abstraction est peu adaptée aux fichiers stockés de grande taille : recopie coûteuse du contenu dans un buffer, coût des appels système `read()/write()`, nécessité de faire des aller-retour avec la tête de lecture...

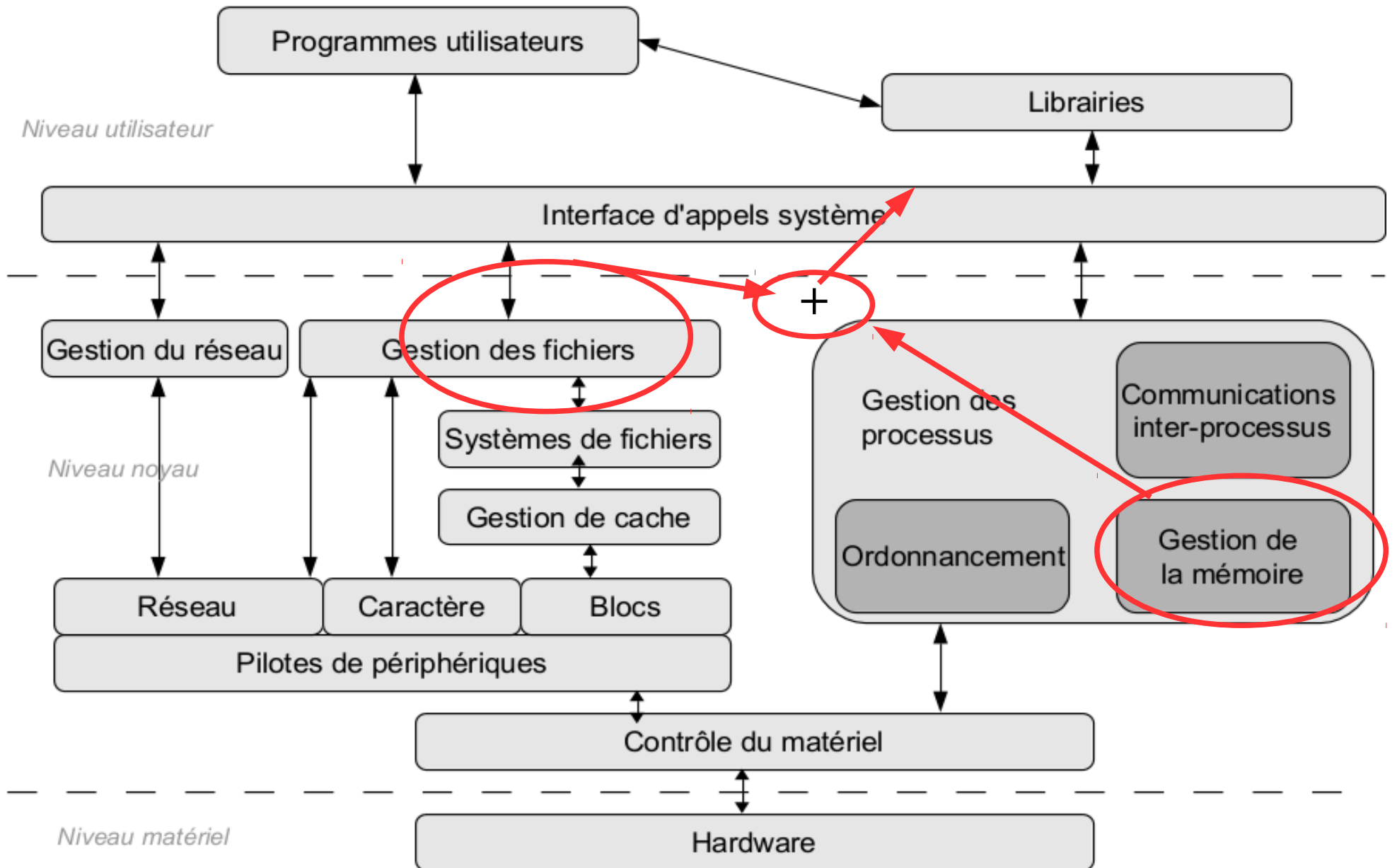
Introduction

Une autre abstraction, beaucoup plus adaptée : les fichiers mappés en mémoire :



Une fois le mapping effectué par l'appel système `mmap`, le fichier est vu par l'application comme un tableau à une dimension dont le contenu est accessible en lecture/écriture avec les primitives usuelles d'accès aux tableaux en C.

Rappels sur la vue d'ensemble



L'appel système mmap()

```
void *addr = mmap(0, len, protec, flags, fd, offest ) ;
```

Addr permet de préciser où a lieu le mapping exactement, mais on le fixe à zéro dans la Plupart des cas, laissant le système déterminer l'adresse pour nous.

Len est la longueur du mapping, pour un fichier cela correspond usuellement à sa taille.

Prot indique les protection désirée pour l'accès mémoire, on peut poser les flags suivant :

```
PROT_NONE  
PROT_READ  
PROT_WRITE  
PROT_EXEC
```

Fd est un descripteur de fichier ouvert au préalable, flags indique divers flags spécifique au mapping, pour mapper un fichier régulier, aucun flags particulier n'est nécessaire, si le fichier est accédé par plusieurs processus et qu'on veut partager le mapping, MAP_SHARED doit être posé

Si l'appel réussit :

- Un pointeur sur vers la région mappé est renvoyé.

Si l'appel échoue :

- Renvoie MAP_FAILED, perror() indique la cause de l'erreur

Exemple

```
Int *map ;

If ( (fd = open(FILEPATH, O_RDWR | O_CREAT | O_TRUNC, (mode_t)0600) ) < 1)
{
    perror("Ouverture du fichier");
    exit(EXIT_FAILURE);
}

If ( (map = mmap(0, FILESIZE, PROT_READ | PROT_WRITE, MAP_SHARED, fd, 0)) == MAP_FAILED)
{
    close(fd);
    perror("Error mmaping the file");
    exit(EXIT_FAILURE);
}

for (i = 1; i <= FILESIZE/(sizeof int); ++i)
{
    map[i] = i;
}

if (munmap(map, FILESIZE) == -1) {
    perror("Error munmap");
}

close(fd) ;
```

Note Importante : La zone mappée doit pré-exister
Dans le fichier avant d'appeler mmap() sinon, l'effet sera le
même que lorsqu'on sort d'un tableau en C : segfault