

Systèmes centralisés : TD4

2) Duplication et redirection

Un descripteur de fichier ouvert peut être recopié/affecté à un autre descripteur

- la commande `dup` affecte le descripteur fourni en paramètre au plus petit descripteur non utilisé

```
int dup (int descripteur)
```

descripteur affecté

descripteur à copier

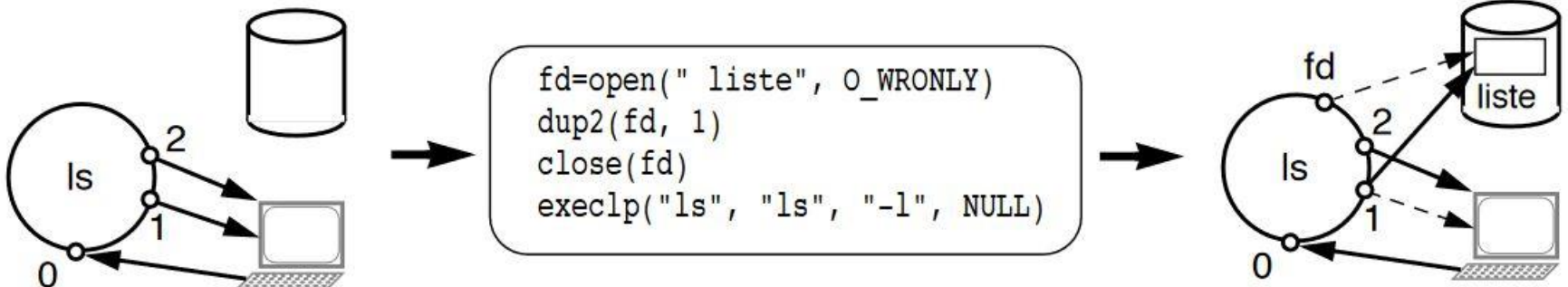
- la commande `dup2` réalise l'affectation explicite entre descripteurs

```
int dup2 (int descripteur_copié, int descripteur_affecté)
```

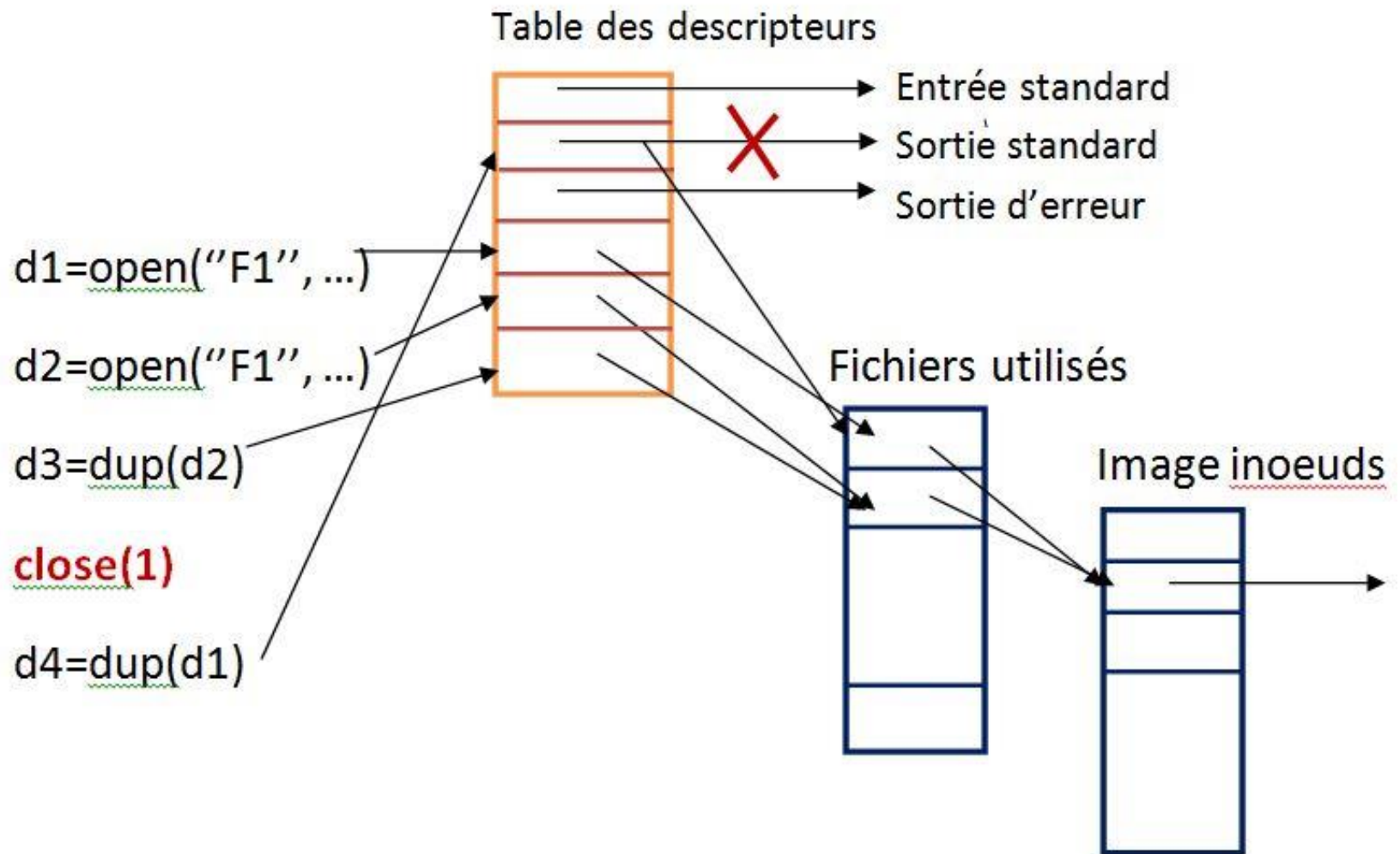
Si le descripteur affecté désignait déjà un fichier, celui-ci est fermé

Utilisation fréquente : réalisation des redirections

Problème : implanter la commande `"ls -l > liste"`



Systèmes centralisés : TD4



Systèmes centralisés : TD4

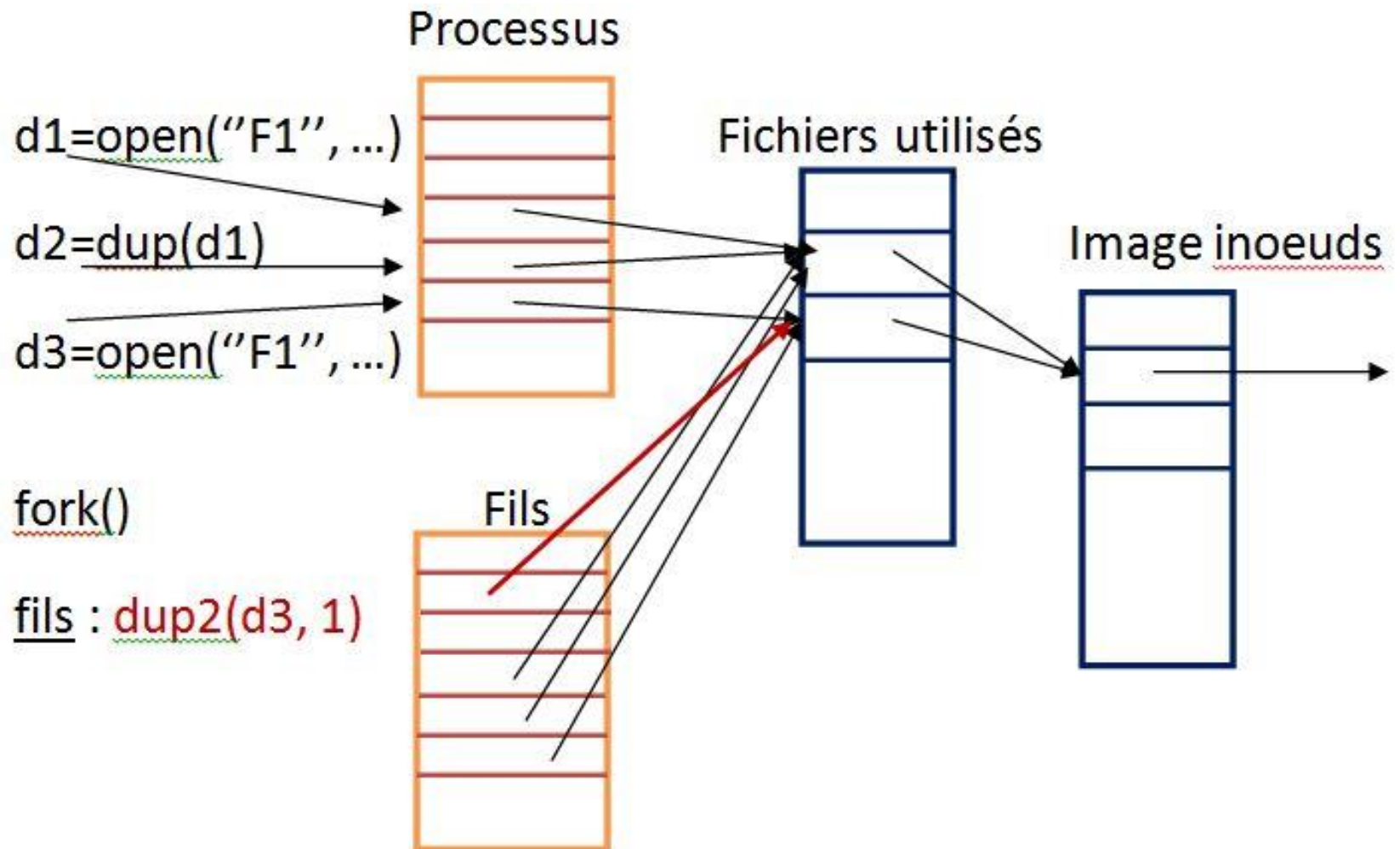
➤ **int dup2 (int desc, int nv_desc)**

- permet de forcer nv_desc comme synonyme de desc : nv_desc accède à la même ressource que desc
- si nv_desc était un descripteur ouvert alors il est préalablement fermé

Illustration : test_dup.c

```
int main(int argc, char* argv[]) {
    int ret, desc, sv_stdout;
    sv_stdout = dup(1);
    printf("dup stdout : %d\n", sv_stdout);
    desc = open("fic_dup2", O_WRONLY | O_CREAT |
        O_TRUNC, 0640);
    ret = dup2(desc, 1);
    printf("Test dup2 %d\n", ret); fflush(stdout);
    close(desc);
    dup2(sv_stdout, 1);
    close(sv_stdout);
    printf("stdout recupere \n");
}
```

Systèmes centralisés : TD4



Systèmes centralisés : TD4

➤ Exercice

Ecrire un programme qui réalise la commande UNIX cp (cp source destination) :
en utilisant la commande cat.
le fichier destinataire de la copie soit uniquement accessible en lecture pour tous les utilisateurs (groupe et autres)

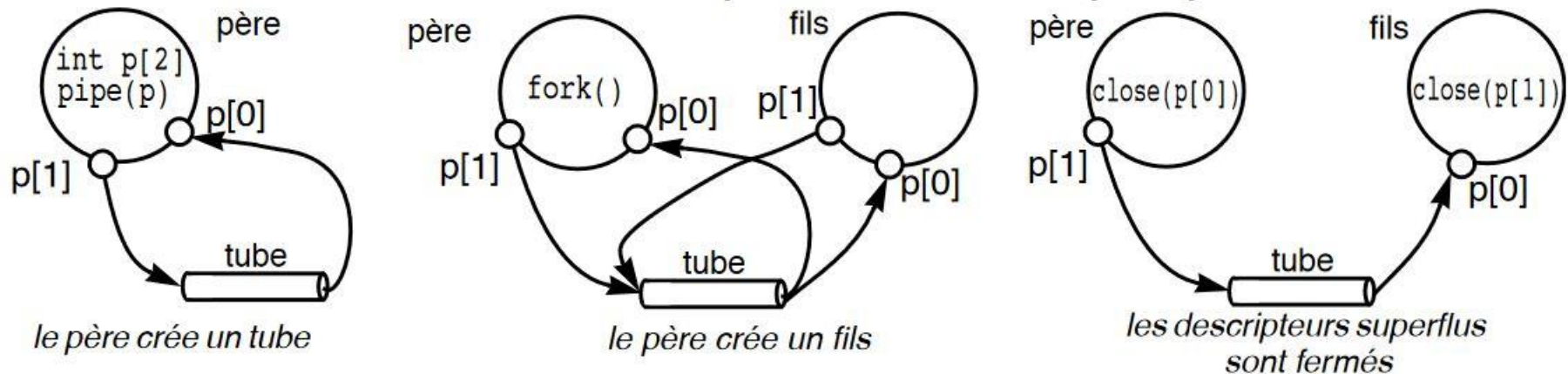
Systèmes centralisés : TD4 - Fichiers

3) Tubes (commande pipe)

Les tubes permettent d'échanger des flots de données entre processus ayant un ancêtre commun

- un tube est un tableau (T) de 2 descripteurs (fichiers)
 - ◊ le premier élément du tableau (T[0]) est la *sortie* du tube
 - ◊ le second élément (T[1]) est l'*entrée* du tube
- le flot de données transite de l'entrée du tube à la sortie du tube, dans l'ordre FIFO :
les données sont transmises en écrivant dans l'entrée (T[1]) et en lisant dans la sortie (T[0]) du tube
- un tube est un tampon de taille bornée :
 - ◊ une lecture dans un tube vide est bloquante
 - ◊ une écriture dans un tube plein est bloquante
 - ◊ une lecture dans un tube vide, sans écrivain potentiel renvoie 0
 - ◊ une écriture dans un tube sans lecteur potentiel provoque l'envoi de SIGPIPE

Schéma élémentaire de construction de processus communiquant par tubes



Systèmes centralisés : TD4

➤ Exercice

Définir deux processus devant respectivement :

- (a) écrire dans un tube les nombres 1 à N (processus 1, producteur) ;
- (b) lire les nombres écrits dans ce tube, en faire la somme puis afficher sur la sortie standard le résultat (processus 2, consommateur).

Le consommateur ne connaît pas la valeur de N.

Une fois cette version implantée, on souhaite la compléter de la manière suivante :

le processus 2 écrit maintenant son résultat final dans le tube.

On ajoute par ailleurs un processus qui lit ce résultat et l'affiche sur la sortie standard. Quel(s) problème(s) pose(nt) cette spécification ?

Systèmes centralisés : TD4

➤ Exercice

Ecrire un programme qui utilise le mécanisme de pipes pour réaliser la commande :

```
who | grep nom_utilisateur | wc -l
```