

Système partie 1

Pastille 7 - Wait

Vincent Colotte - N. de Rugy-Altherre

Attente père/fils

wait et *waitpid*

```
#include <sys/types.h>
#include <sys/wait.h>
prototypes :
    pid_t wait(int *pStatut);
    pid_t waitpid(pid_t $pid$, int *pStatut$,
```

wait : prend en compte la terminaison d'un des fils, récupère l'information sur sa terminaison dans *pStatut* puis efface l'entrée de ce fils dans la table des processus (la valeur renvoyée est le numéro du fils).

waitpid : sélectionne un processus-fils en particulier et attend la terminaison ou la suspension de ce fils.

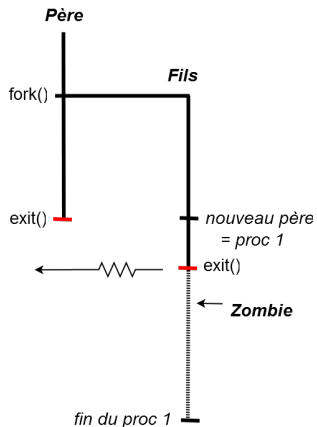
Attente père/fils

Macros pré-définies sur *pStatus*

- nature de la terminaison et la valeur de retour de *exit*.
- Exemple : `WEXITSTATUS(*pStatus)` donne l'entier transmis par *exit* ou *return* du fils
(attention à allouer le pointeur *pStatut* avant de l'utiliser)

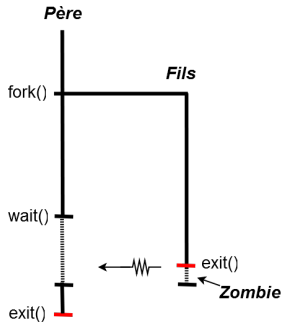
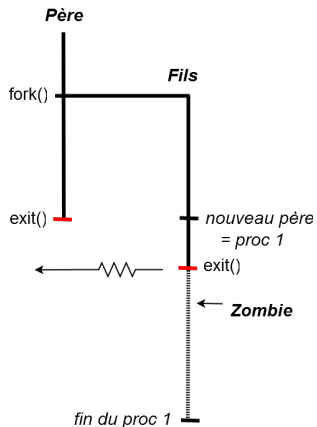
Attente père/fils

Wait



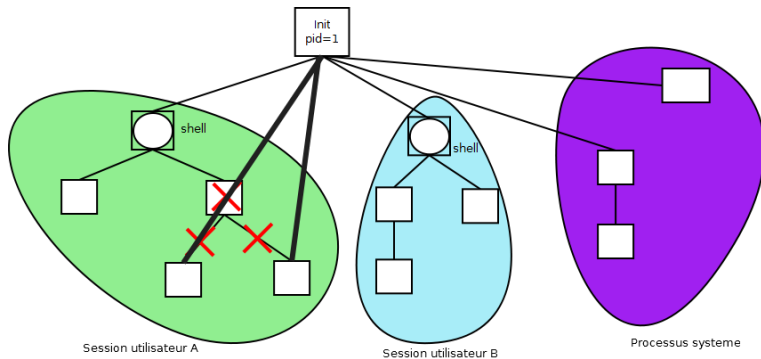
Attente père/fils

Wait



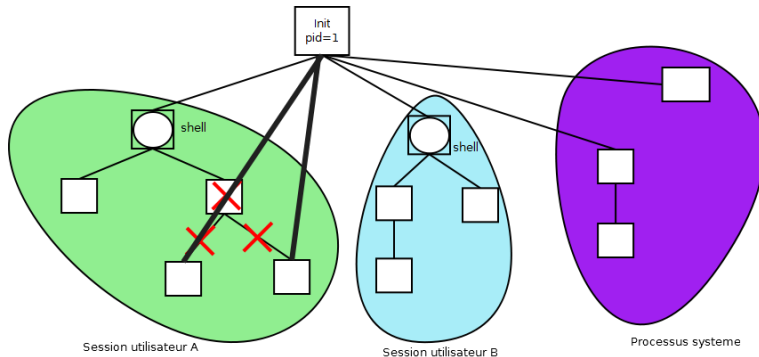
Rattachement au processus initial

PID = 1



Rattachement au processus initial

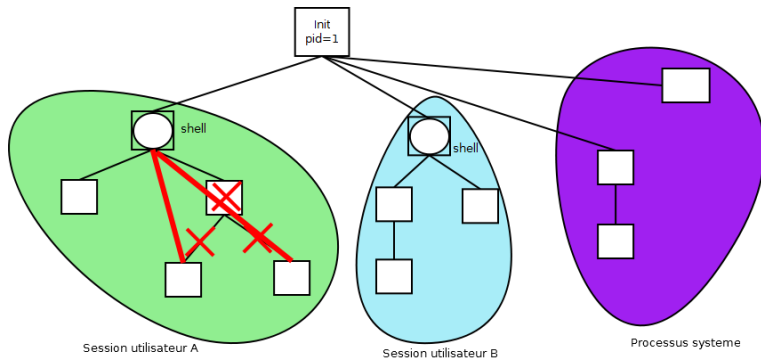
PID = 1



* >Ubuntu 14.04 : Rattachement au processus Leader de la session !

Ubuntu > 14.04

PID=... leader session!!!



Exercice

Exercice

Écrire un programme dont le père fait un `sleep(1)`, le fils un `sleep(5)` puis un `exit`. Le père doit attendre le fils avant de se terminer.

Indiquer les terminaisons par des affichages.