

# **Cours UNIX**

## **Les processus**

<http://www.chez.com/brunogarcia/Unix/>

Jean-Claude Bajard

IUT - université Montpellier 2

# Gestion des processus

Le système effectue trois types de tâche

- le partage du temps CPU (time sharing).
- la synchronisation des entrées-sorties (sync)
- le contrôle de la mémoire virtuelle (swap)

# Caractéristiques d'un processus

La commande **ps** affiche les processus courants, plusieurs options sont utiles : a pour voir tous les processus, x pour voir ceux non contrôlés par un terminal, u pour voir les propriétaires ....

- PID Identificateur numérique unique de chaque processus. (par ordre croissant)
- PPID PID du père du processus. Par défaut, tous les processus ont un père. Il n'existe que 2 exceptions
  - \* le processus init qui est le premier lancé
  - \* les orphelins, c'est à dire les processus dont le père est mort. Ils sont le plus souvent récupérés par init

## Mode d'exécution

- R Running Le processus est actif et exécute des instructions de type utilisateur
- Z Zombie Ce processus est mort, mais son père est encore actif et n'a pas fait de mise à jour. Un processus Zombie n'occupe pas de ressources systèmes
- W Waiting Processus en attente d'une ressource (fichier, mémoire, sémaphore etc...)
- D Dying En cours de mort, le plus souvent, ce genre de processus est en train d'écrire un fichier core
- S Swapped Ce processus est actuellement inactif et " swappé " sur le disque.
- S Stopped Le processus a été stoppé par l'envoi du signal stop, par exemple CTRL-Z

# Taille mémoire

- La taille statique est liée à l'exécutable au moment de son lancement:
  - \* la taille du code lié statiquement
  - \* les données statiques
  - \* la pile initiale
- La taille dynamique e :
  - \* la taille du code lié dynamiquement ( .so)
  - \* Les augmentations de taille de pile si celle ci est dynamique
  - \* Les données allouées dynamiquement sur le tas

# Temps CPU

Il y a trois durées d'exécution différentes pour un processus

- Wall clock intervalle séparant la date de lancement du processus de l'heure courante.
- utilisateur: nombre de millisecondes affectées à votre processus sur la CPU (toute instruction non privilégiée)
- système : nombre de millisecondes affectées à votre processus sur la CPU pour l'exécution d'instructions systèmes ou privilégiées. Ces instructions font partie du noyau du système. Elles concernent tous les appels système de bas niveau, tel que l'écriture physique sur le disque etc (non interruptible)

la durée CPU totale est la somme de la durée utilisateur et de la durée système.

# Divers

## le répertoire /proc

Il contient un répertoire par processus contenant diverses informations dont le fichier /proc/num\_pid/status contient toutes les informations

## Priorité

Conditionne l'affectation de la CPU aux processus.

- **nice** L'utilisateur a la possibilité d'influencer la priorité de l'un de ces processus

# Contrôle des processus

une commande lancée dans un shell peut être :

- Stoppée par un "Ctrl-z"
- placée en arrière plan : background **bg**
- replacée en avant plan : foreground **fg**
- arrêtée par
  - \* un "Ctrl-c" si elle est en avant plan
  - \* la commande **kill**, sinon



# Commandes utiles

- **jobs** affiche les commandes lancées dans le shell courant
- **ps** (-auxl) affiche les processus gérés par le système
- **top** donne l'activité du système classée par occupation du cpu
- **kill** envoie un signal à un ou plusieurs processus
  - \* **kill -9 pid** tue le processus dont le pid est donné
  - \* **kill -9 %job** tue le job (numéro obtenu avec la commande jobs)
  - \* **kill -15 pid** tue en terminant les accès en lecture-écriture

## Lancement différé

- **batch** dès que la charge du système le permet
- **at** différé une seule fois à une date précise ( -l liste des commandes) (atrm pour supprimer)
- **crontab** répétitif et régulier dans le temps grâce au démon **cron**
  - \* **crontab -e** pour éditer
  - \* **man 5 crontab** pour connaître le format (mn h dm m ds)

# Redirections

**stdin** entrée standard (clavier) **stdout** sortie standard (écran) **stderr** sortie erreurs

- < permet de rediriger l'entrée standard depuis un fichier
- > permet de rediriger la sortie standard vers un fichier
- 2> permet de rediriger la sortie des erreurs vers un fichier
- >& ou &> permet de rediriger la sortie des erreurs vers la sortie standard
- >> permet de rediriger la sortie standard en fin d'un fichier
- | permet de rediriger la sortie standard dans l'entrée d'une autre commande