

Ordonnement des processus

Christophe Viroulaud

Terminale - NSI

Archi 02



Les processus

Définition

Création d'un processus

Ordonnement

Le chef d'orchestre

L'ordonnement
(scheduling)

Quelques algorithmes
d'ordonnement

Un processeur ne peut exécuter qu'une seule instruction à la fois. Pourtant sur un ordinateur, il est possible d'écouter de la musique tout en surfant sur le web.

Les processus

Définition

Création d'un processus

Ordonnement

Le chef d'orchestre

L'ordonnement
(scheduling)

Quelques algorithmes
d'ordonnement

Découvrir les mécanismes pour exécuter plusieurs tâches
en même temps sur une machine.

1. Les processus

1.1 Définition

1.2 Création d'un processus

2. Ordonnement

Les processus

Définition

Création d'un processus

Ordonnement

Le chef d'orchestre

L'ordonnement
(scheduling)

Quelques algorithmes
d'ordonnement

À retenir

Un **programme** est un fichier stocké en mémoire (disque dur) et qui ne fait rien.

Un **processus** est l'exécution d'un programme dans la mémoire vive.

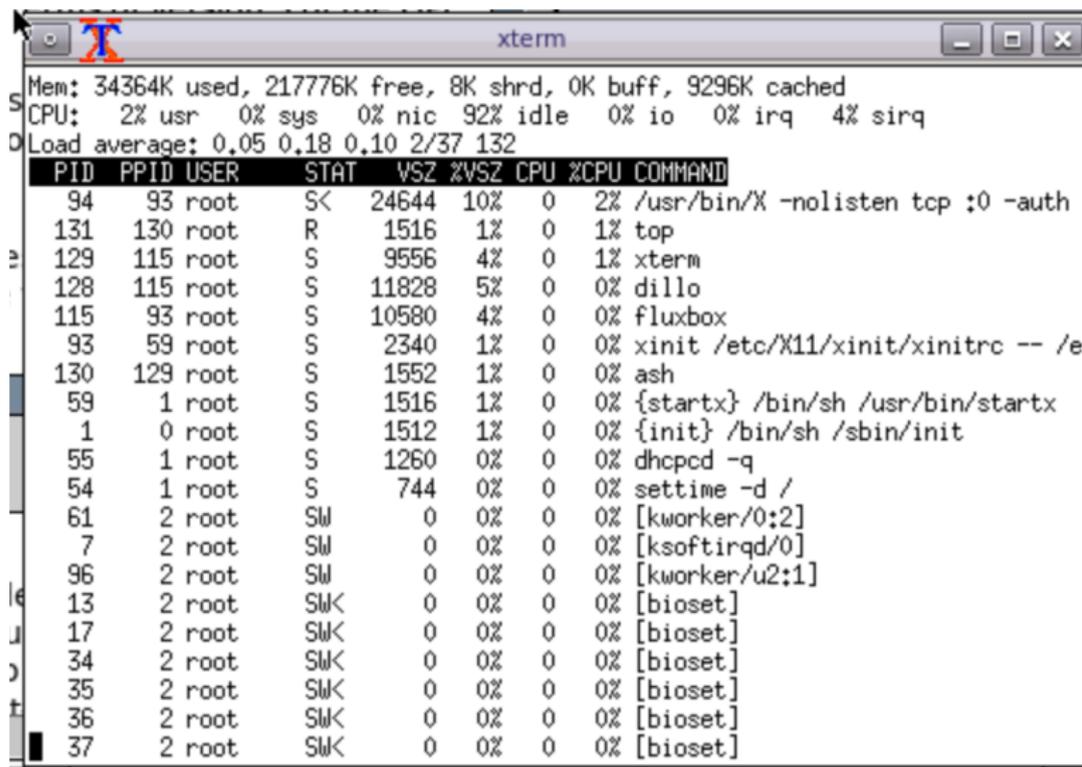
Activité 1 :

1. Se rendre sur l'émulateur de système GNU/Linux <https://tinyurl.com/linuxemul>
2. Faire un clic-droit sur le bureau et paramétrer le *keyboard mapping*.
3. Faire un clic-droit sur le bureau et ouvrir le navigateur Dillo.
4. Faire un clic-droit sur le bureau et ouvrir un terminal.
5. Écrire la commande

```
1 top
```

Code 1 – Visualiser les processus en cours

6. Utiliser la combinaison de touche **Ctrl+c** pour stopper la surveillance des processus.



Mem: 34364K used, 217776K free, 8K shrd, 0K buff, 9296K cached
CPU: 2% usr 0% sys 0% nic 92% idle 0% io 0% irq 4% sirq
Load average: 0.05 0.18 0.10 2/37 132

PID	PPID	USER	STAT	VSZ	%VSZ	CPU	%CPU	COMMAND
94	93	root	S<	24644	10%	0	2%	/usr/bin/X -nolisten tcp :0 -auth
131	130	root	R	1516	1%	0	1%	top
129	115	root	S	9556	4%	0	1%	xterm
128	115	root	S	11828	5%	0	0%	dillo
115	93	root	S	10580	4%	0	0%	fluxbox
93	59	root	S	2340	1%	0	0%	xinit /etc/X11/xinit/xinitrc -- /e
130	129	root	S	1552	1%	0	0%	ash
59	1	root	S	1516	1%	0	0%	{startx} /bin/sh /usr/bin/startx
1	0	root	S	1512	1%	0	0%	{init} /bin/sh /sbin/init
55	1	root	S	1260	0%	0	0%	dhcpcd -q
54	1	root	S	744	0%	0	0%	settime -d /
61	2	root	SW	0	0%	0	0%	[kworker/0:2]
7	2	root	SW	0	0%	0	0%	[ksoftirqd/0]
96	2	root	SW	0	0%	0	0%	[kworker/u2:1]
13	2	root	SW<	0	0%	0	0%	[bioset]
17	2	root	SW<	0	0%	0	0%	[bioset]
34	2	root	SW<	0	0%	0	0%	[bioset]
35	2	root	SW<	0	0%	0	0%	[bioset]
36	2	root	SW<	0	0%	0	0%	[bioset]
37	2	root	SW<	0	0%	0	0%	[bioset]

Figure 1 – Un processus est crée pour lancer le navigateur Dillo.

Les processus

Définition

Création d'un processus

Ordonnement

Le chef d'orchestre

L'ordonnement
(scheduling)Quelques algorithmes
d'ordonnement

Les processus

Définition

Création d'un processus

Ordonnement

Le chef d'orchestre

L'ordonnement
(scheduling)

Quelques algorithmes
d'ordonnement

1. Les processus

1.1 Définition

1.2 Création d'un processus

2. Ordonnement

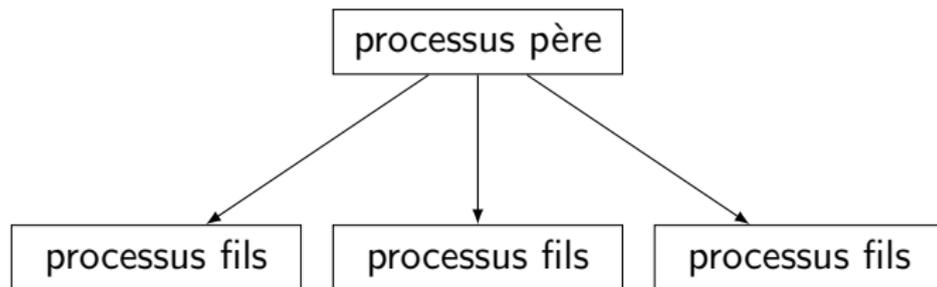


Figure 2 – Un processus est crée à partir d'un processus père :
c'est un **fork**.

À retenir

- ▶ Chaque processus possède un identifiant unique, le **PID** (**P**rocess **ID**entifier).
- ▶ Chaque processus possède un (et un seul) parent, le **PPID** (**P**arent **P**rocess **ID**entifier).

Remarque

Au démarrage de la machine un premier processus spécial (**init**) est lancé.

Les processus

Définition

Création d'un processus

Ordonnement

Le chef d'orchestre

L'ordonnement
(scheduling)

Quelques algorithmes
d'ordonnement

Activité 2 :

1. Noter le PID du processus de Dillo.
2. Noter son PPID.
3. Ce processus père possède-t-il d'autres fils ?

Les processus

Définition

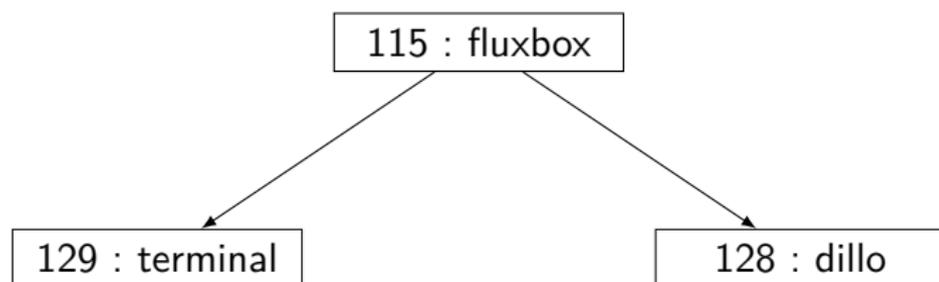
Création d'un processus

Ordonnement

Le chef d'orchestre

L'ordonnement
(scheduling)

Quelques algorithmes
d'ordonnement



Remarque

Les PID peuvent varier.

Activité 3 :

1. Afficher la liste de tous les processus :

```
1 ps
```

2. Retrouver le PID du processus de Dillo. Tuer le processus avec l'instruction :

```
1 kill numéro_PID
```

3. Que se passe-t-il si on tue un processus père ?

Les processus

Définition

Création d'un processus

Ordonnement

Le chef d'orchestre

L'ordonnement
(scheduling)

Quelques algorithmes
d'ordonnement

- ▶ Si un processus père est tué, ses fils sont tués également.
- ▶ On ne peut tuer le processus 1.

Observation

D'autres informations sont mentionnées :

- ▶ nom de l'utilisateur qui a créé,
- ▶ utilisation du CPU,
- ▶ état (STAT) :
 - ▶ R en cours d'exécution.
 - ▶ T processus stoppé.
 - ▶ I processus endormi ($>20s$).
 - ▶ S processus endormi ($<20s$).
 - ▶ Z processus zombie.
 - ▶ D processus non interruptible.
 - ▶ W processus swappé (échangé) sur disque.

Les processus

Définition

Création d'un processus

Ordonnement

Le chef d'orchestre

L'ordonnement
(scheduling)

Quelques algorithmes
d'ordonnement

Activité 4 : Exécuter une fonction dans l'interpréteur de commande (console, shell, bash¹) impose la création d'un nouveau processus. Une *fork bomb* est une forme d'attaque par déni de service.
Écrire la commande ci-dessous dans le terminal :

```
1 f(){ f|f& };f
```

1. Plus d'informations sur [l'histoire de l'interpréteur de commandes](#)

Les processus

Définition

Création d'un processus

Ordonnement

Le chef d'orchestre

L'ordonnement
(scheduling)

Quelques algorithmes
d'ordonnement

- ▶ `f()` définition de la fonction
- ▶ `{f | f&}` corps de la fonction : la fonction s'appelle récursivement `f`, envoie `|` le résultat de l'appel vers l'entrée de la même fonction `f` et masque le processus `&`
- ▶ `;` fin de la déclaration de la fonction
- ▶ `f` appel principal

Les processus

Définition

Création d'un processus

Ordonnement

Le chef d'orchestre

L'ordonnement
(scheduling)

Quelques algorithmes
d'ordonnement

1. Les processus

2. Ordonnement

2.1 Le chef d'orchestre

2.2 L'ordonnement (scheduling)

2.3 Quelques algorithmes d'ordonnement

Les processus

Définition

Création d'un processus

Ordonnancement

Le chef d'orchestre

L'ordonnancement
(scheduling)

Quelques algorithmes
d'ordonnancement

Plusieurs processus sont en cours simultanément.



Mais le processeur ne peut exécuter qu'une seule instruction
à la fois.



Le processeur travaille donc *en temps partagé* : il bascule
constamment d'un processus à l'autre.

1. Les processus

2. Ordonnement

2.1 Le chef d'orchestre

2.2 L'ordonnement (scheduling)

2.3 Quelques algorithmes d'ordonnement

Les processus

Définition

Création d'un processus

Ordonnement

Le chef d'orchestre

**L'ordonnement
(scheduling)**

Quelques algorithmes
d'ordonnement

L'ordonnancement (scheduling)

Les processus

Définition

Création d'un processus

Ordonnancement

Le chef d'orchestre

**L'ordonnancement
(scheduling)**

Quelques algorithmes
d'ordonnancement

- ▶ Un processeur dont la fréquence d'horloge est de 3,2 GHz exécute 3,2 milliards de cycles par seconde.

L'ordonnancement (scheduling)

- ▶ Un processeur dont la fréquence d'horloge est de 3,2 GHz exécute 3,2 milliards de cycles par seconde.
- ▶ Lors d'un cycle, le processeur exécute une tâche d'un processus.

Les processus

Définition

Création d'un processus

Ordonnancement

Le chef d'orchestre

**L'ordonnancement
(scheduling)**

Quelques algorithmes
d'ordonnancement

À retenir

L'ordonnanceur (scheduleur) sélectionne le prochain processus prêt (*Ready*) qui sera exécuté par le processeur. Son objectif est d'obtenir un *temps de traitement moyen* le plus court possible.

Les algorithmes d'ordonnement peuvent être classés en deux catégories :

- ▶ **Non pré emptif** : Sélectionne un processus, puis le laisse s'exécuter jusqu'à ce qu'il bloque (soit sur une Entrée/Sortie, soit en attente d'un autre processus) où qu'il libère volontairement le processeur. Exemple :
 - ▶ First Come First Served,
 - ▶ Shortest Job First.

Les processus

Définition

Création d'un processus

Ordonnement

Le chef d'orchestre

L'ordonnement
(scheduling)

Quelques algorithmes
d'ordonnement

Les algorithmes d'ordonnement peuvent être classés en deux catégories :

- ▶ **Non pré emptif** : Sélectionne un processus, puis le laisse s'exécuter jusqu'à ce qu'il bloque (soit sur une Entrée/Sortie, soit en attente d'un autre processus) où qu'il libère volontairement le processeur. Exemple :
 - ▶ First Come First Served,
 - ▶ Shortest Job First.
- ▶ **Pré emptif** : Sélectionne un processus et le laisse s'exécuter pendant un délai déterminé.
 - ▶ Round Robin,
 - ▶ Shortest Job First.

1. Les processus

2. Ordonnement

2.1 Le chef d'orchestre

2.2 L'ordonnement (scheduling)

2.3 Quelques algorithmes d'ordonnement

Les processus

Définition

Création d'un processus

Ordonnement

Le chef d'orchestre

L'ordonnement
(scheduling)

**Quelques algorithmes
d'ordonnement**

- ▶ **First Come First Served** : Une fois que le CPU a été alloué à un processus, celui-ci le garde jusqu'à ce qu'il décide de le libérer.
- ▶ **Shortest Job First** : Quand le CPU est disponible, il est assigné au processus qui possède le prochain cycle le plus petit.
- ▶ **Round Robin** : Chaque processus a une petite unité de temps appelée *quantum* (en général de 10 à 100 ms). L'ordonnanceur parcourt la file d'attente des processus prêts et alloue le CPU à chaque processus pendant un quantum.

Les processus

Définition

Création d'un processus

Ordonnancement

Le chef d'orchestre

L'ordonnancement
(scheduling)

Quelques algorithmes
d'ordonnancement

Imaginons 5 processus qui doivent s'exécuter sur un nombre de cycles déterminés :

- ▶ P1 : 10 cycles,
- ▶ P2 : 1 cycle,
- ▶ P3 : 2 cycles,
- ▶ P4 : 1 cycle,
- ▶ P5 : 5 cycles.

Les processus arrivent dans cet ordre dans la file d'attente de l'ordonnanceur.

Les processus

Définition

Création d'un processus

Ordonnement

Le chef d'orchestre

L'ordonnement (scheduling)

Quelques algorithmes d'ordonnement

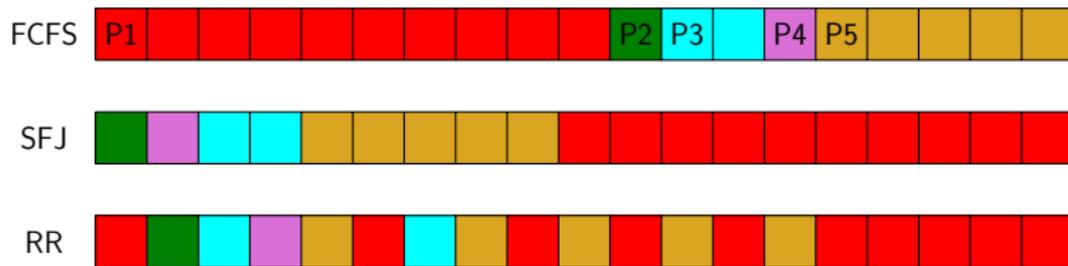


Figure 3 – Exécution des algorithmes d'ordonnement.

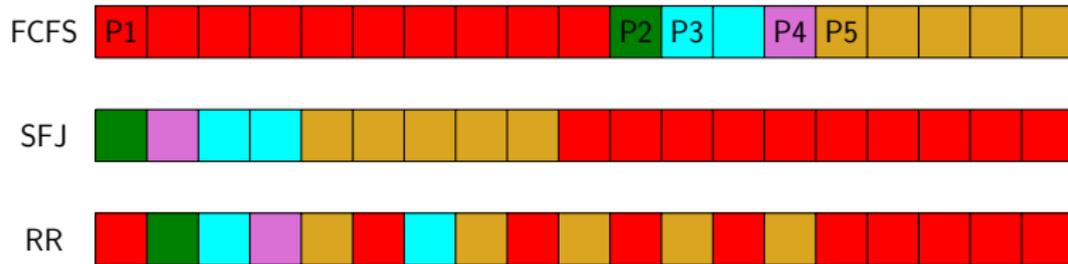


Figure 4 – Exécution des algorithmes d'ordonnement.

Algorithmes	P1	P2	P3	P4	P5	Moyennes
FCFS	10	11	13	14	19	13,4
SFJ	19	1	4	2	9	7
RR	19	2	7	4	14	9,2

Tableau 1 – Temps de présence des processus dans le système, en fonction de l'algorithme d'ordonnement utilisé.

Les processus

Définition

Création d'un processus

Ordonnement

Le chef d'orchestre

L'ordonnement
(scheduling)Quelques algorithmes
d'ordonnement

Remarques

- ▶ En pratique, un système d'exploitation (GNU/Linux) utilise plusieurs systèmes d'ordonnement (First Come First Served, Round Robin...)
- ▶ Des systèmes d'ordonnement par priorité complètent ce fonctionnement : chaque nouveau processus reçoit une priorité, modifiable par plusieurs paramètres.