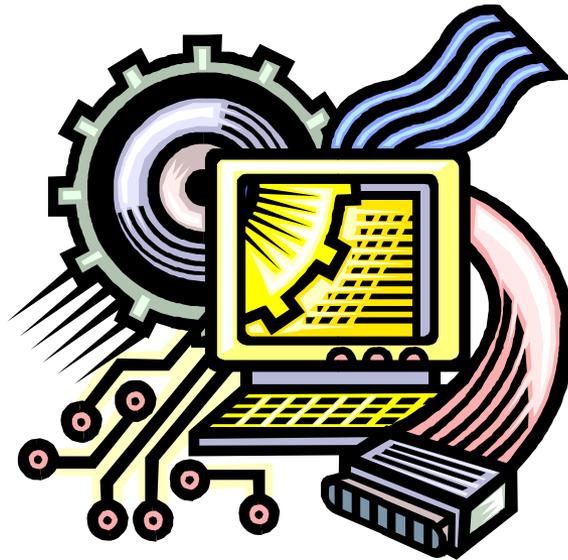


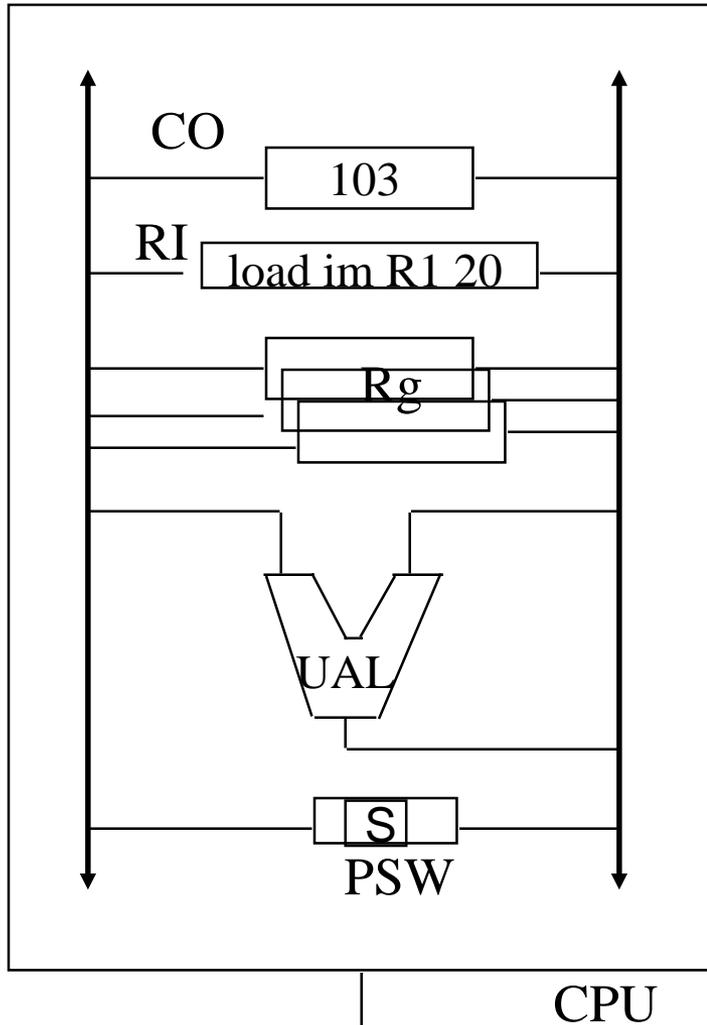
Processus- Ordonnancement



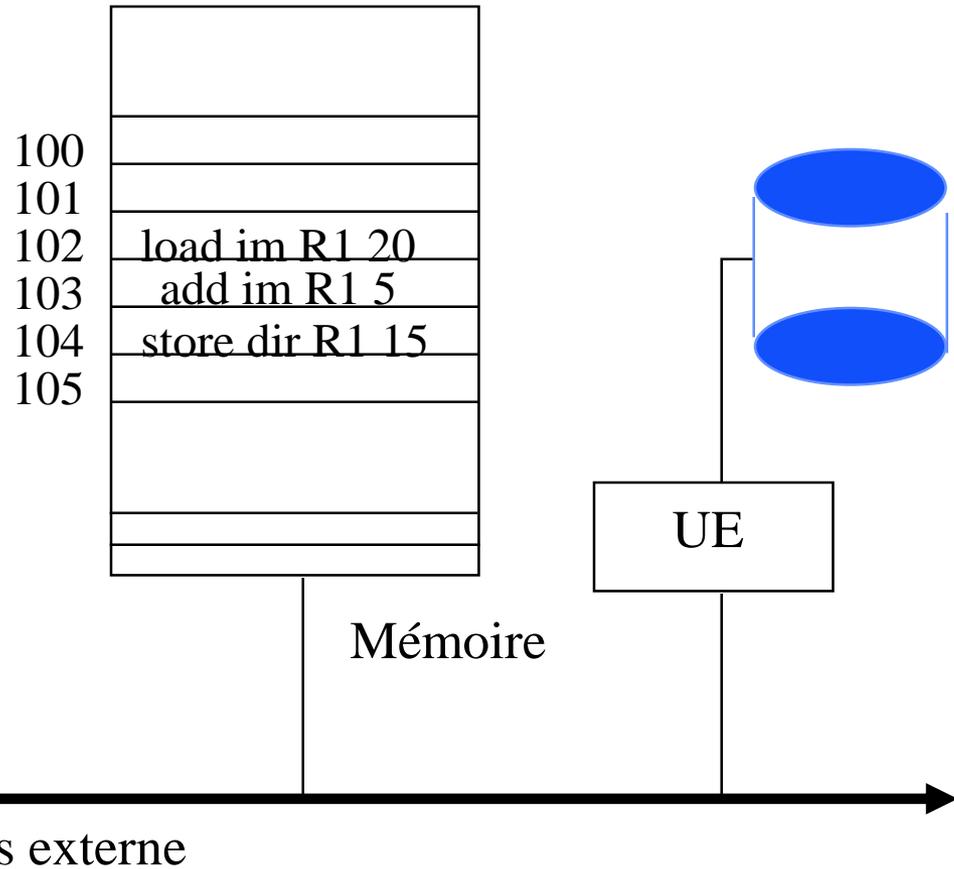
Processus

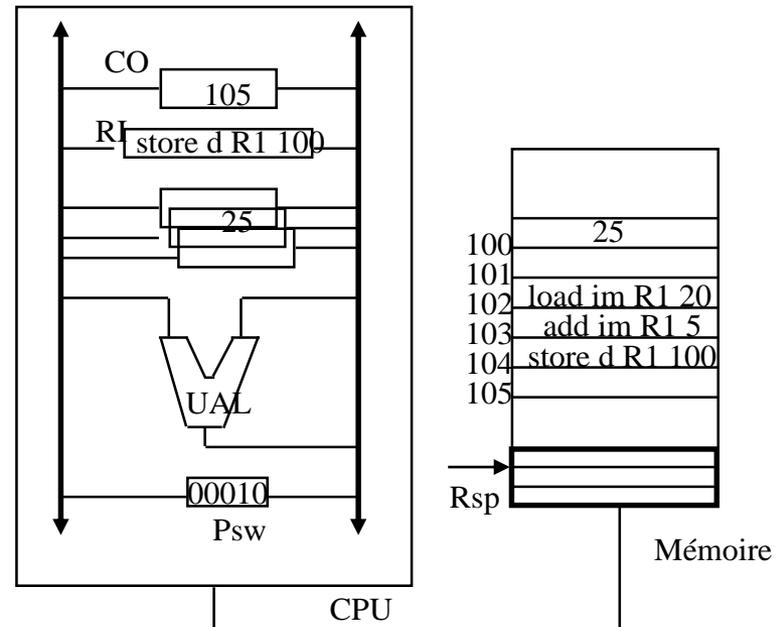
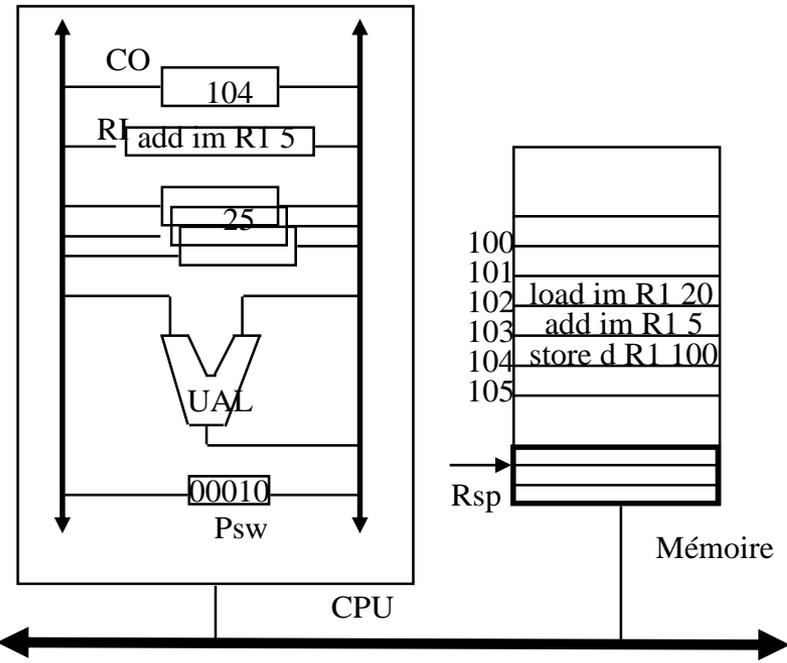
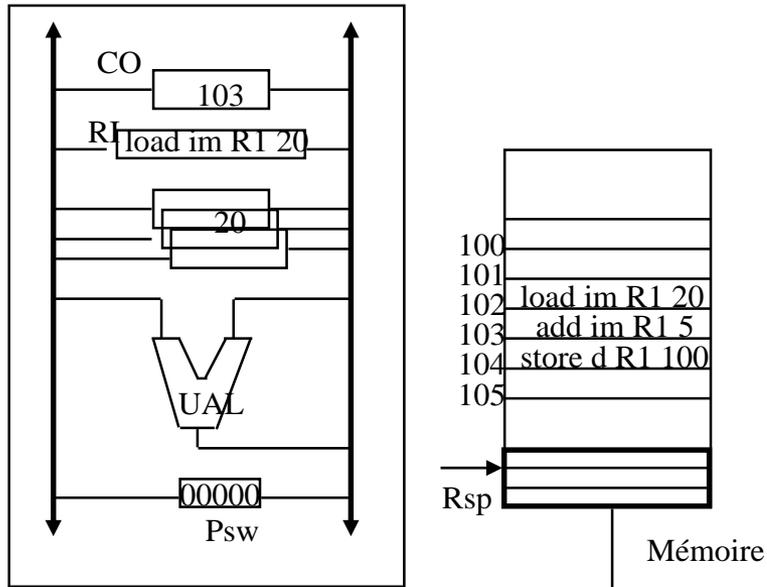
Un processus est une exécution de programme

Notion de processus



- ✓ CO : compteur ordinal
- ✓ PSW : registre d'état
- ✓ RI : registre instruction



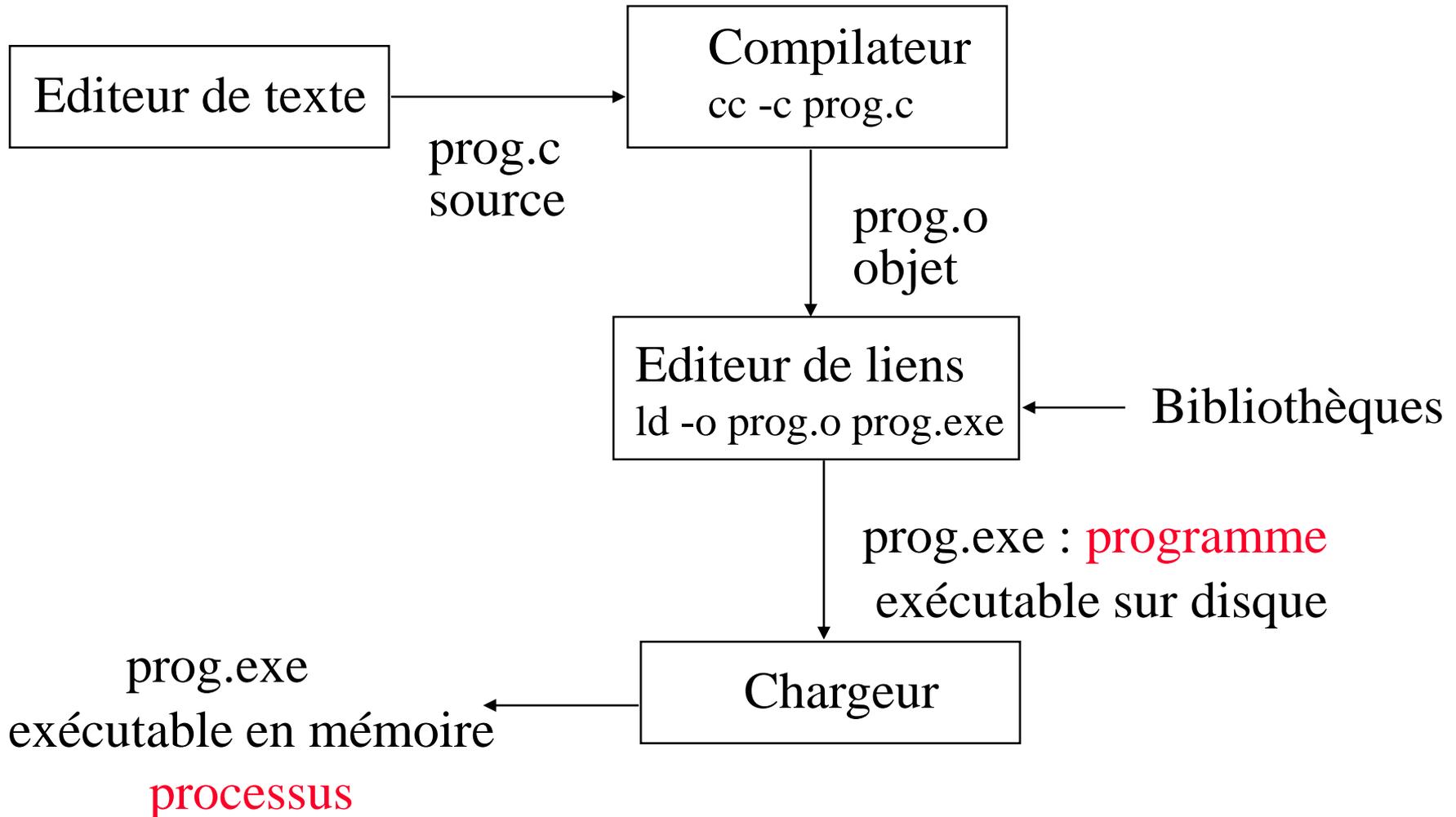


Notion de processus

- **Définition**

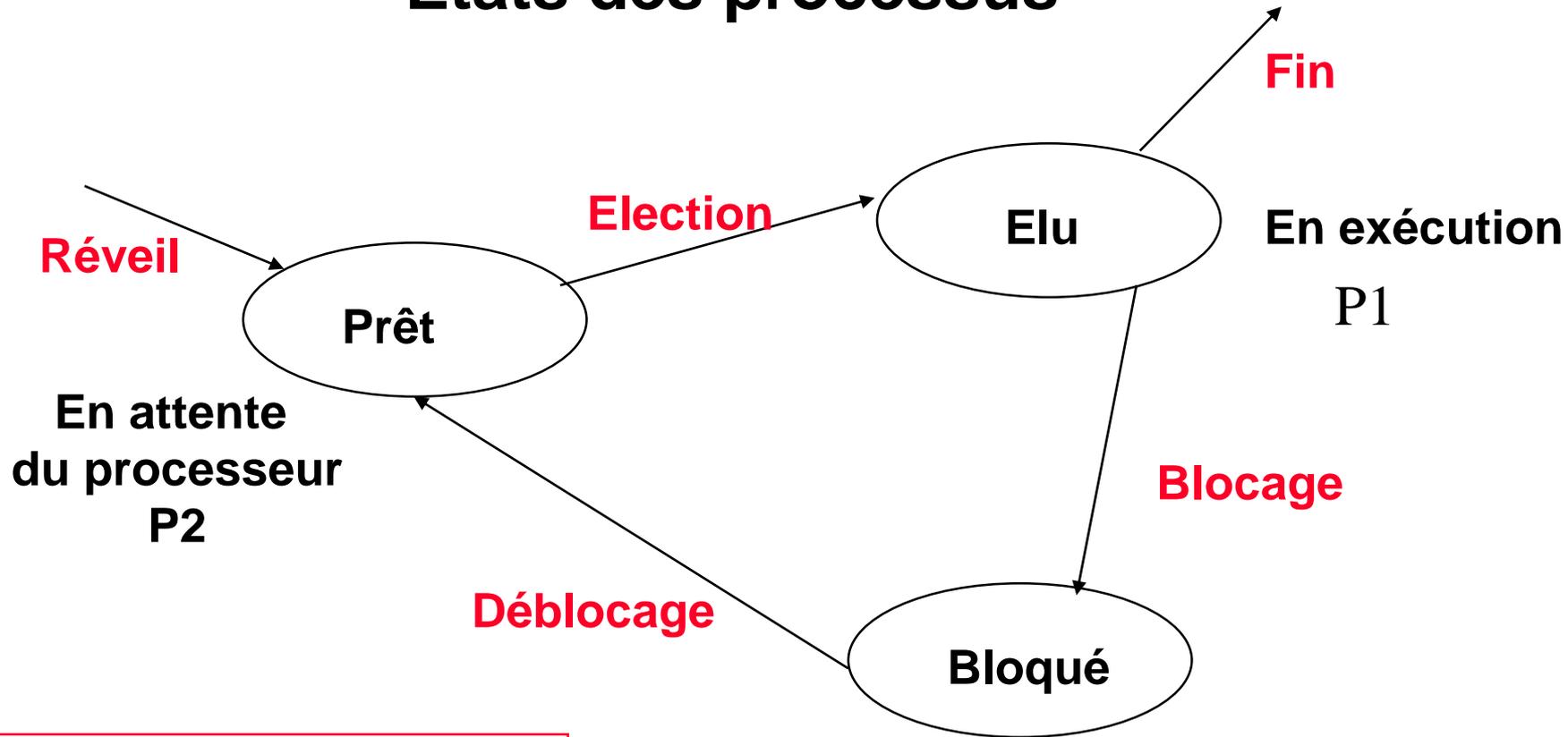
- **Un processus est un programme en cours d'exécution auquel est associé un environnement processeur (CO, PSW, registres généraux) et un environnement mémoire appelés contexte du processus.**
- **Un processus est l'instance dynamique d'un programme et incarne le fil d'exécution de celui-ci**
- **Un processus évolue dans un espace d'adressage protégé**

Du programme au processus



Systeme multiprocessus

Etats des processus



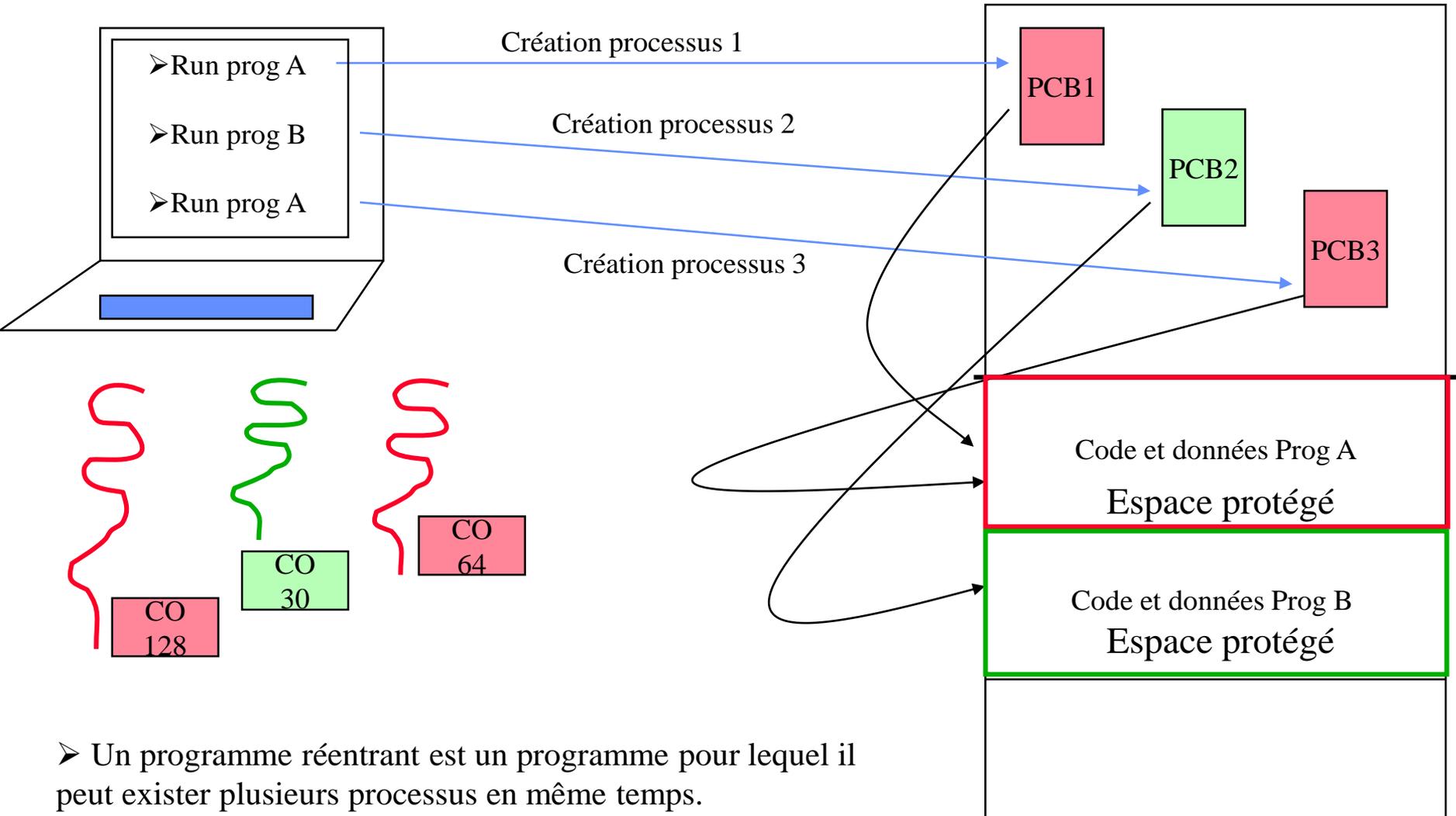
Une ressource désigne toute entité (matérielle, logicielle) dont a besoin un processus pour s'exécuter (processeur, variable)

bloc de contrôle de processus PCB

identificateur processus
état du processus
compteur instructions
contexte pour reprise (registres et pointeurs, piles,..)
pointeurs sur file d'attente et priorité(ordonnancement)
informations mémoire (limites et tables pages/segments
informations de comptabilisation et sur les E/S, périphériques alloués, fichiers ouverts,..

Bloc de contrôle de processus ou PCB

Notion de processus programme réentrant



➤ Un programme réentrant est un programme pour lequel il peut exister plusieurs processus en même temps.

Ordonnancement

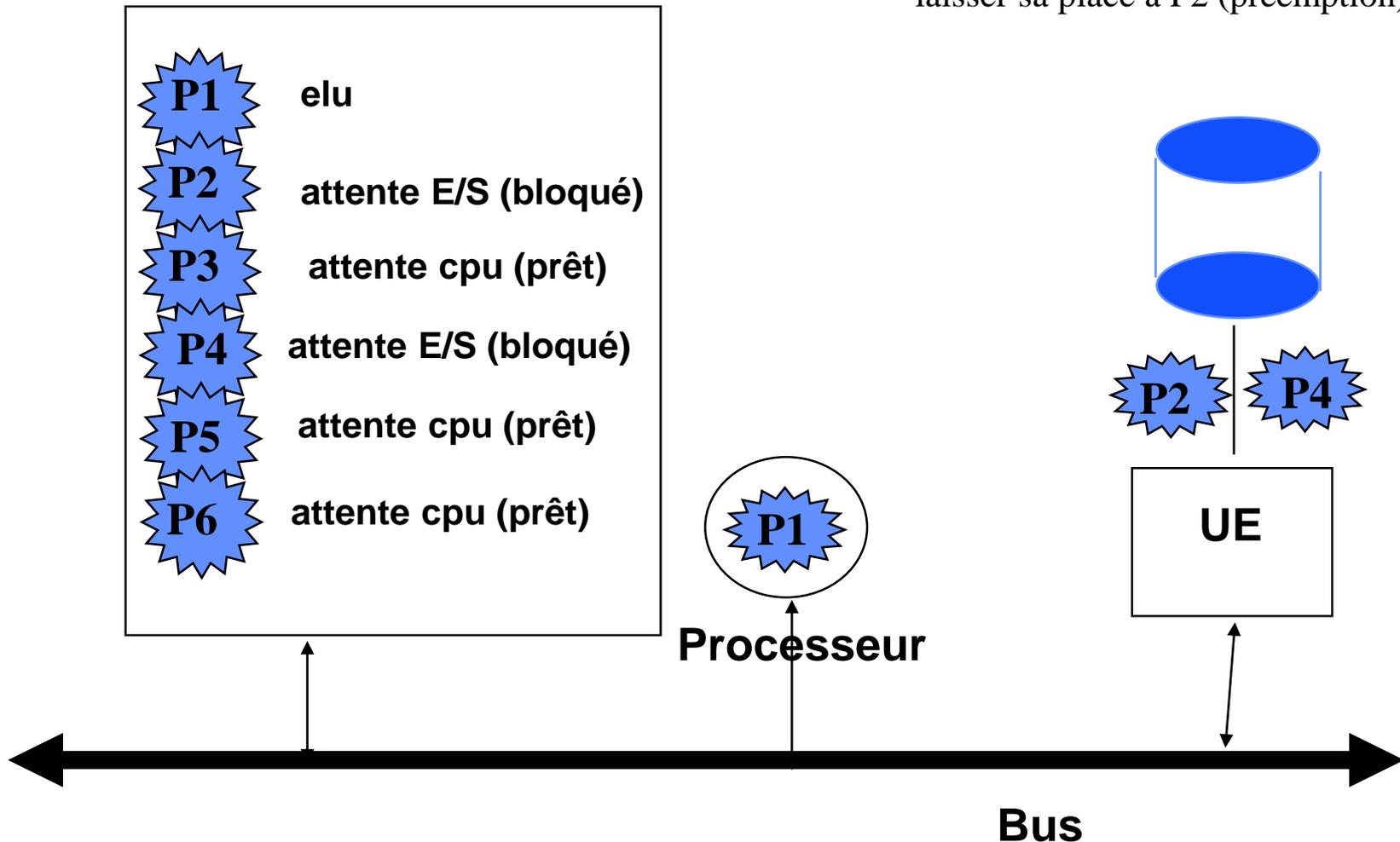
Ordonnancement dans un système multiprocessus

Cette fonction planifie l'exécution des processus

Systeme multiprocessus

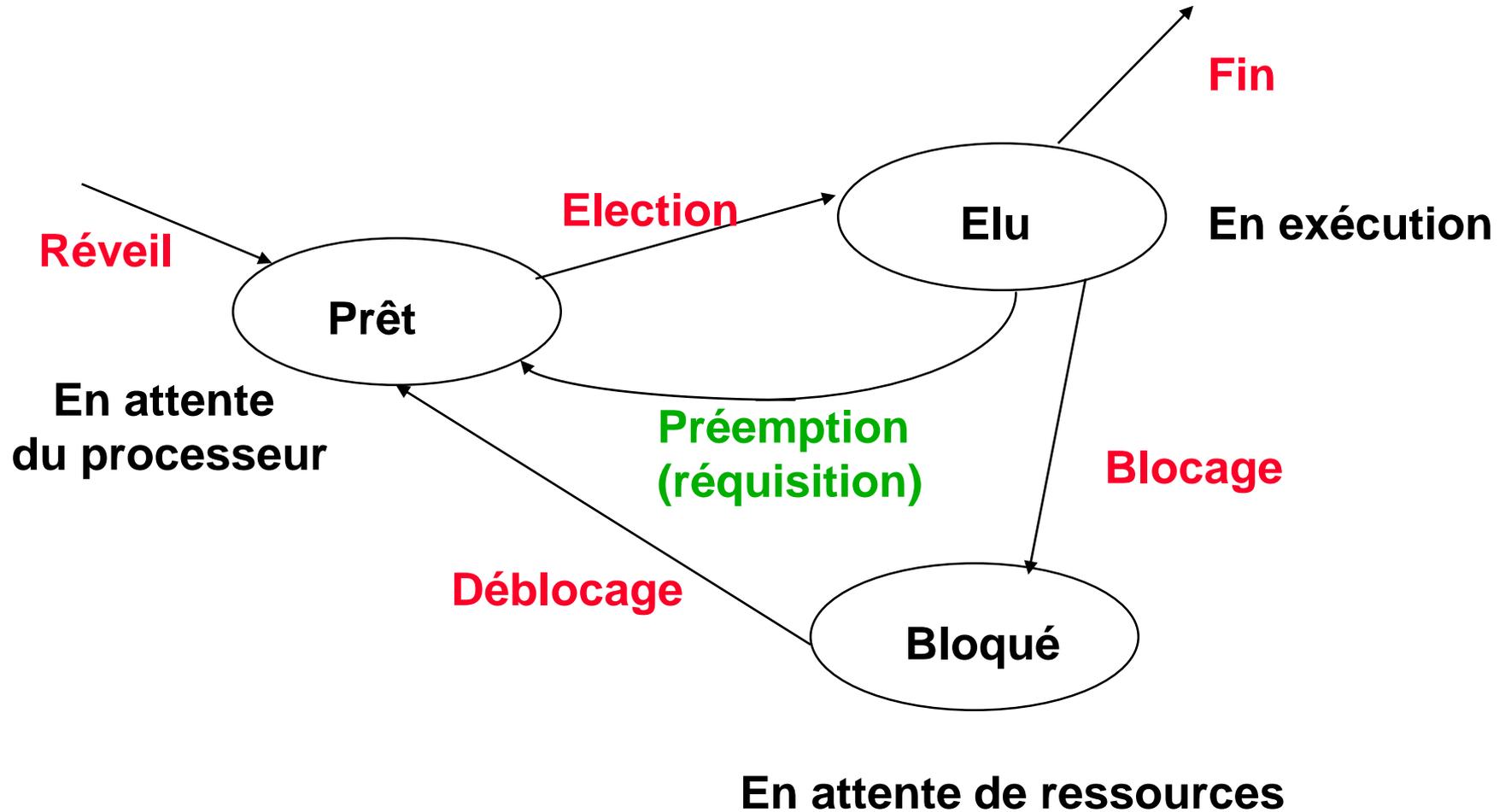
P1 se termine : quel processus exécuter
parmi les processus prêts P3, P5, P6
P2 achève son entrées-sortie et devient prêt :
P1 doit-il poursuivre son exécution ou
laisser sa place à P2 (préemption) ?

Mémoire Centrale



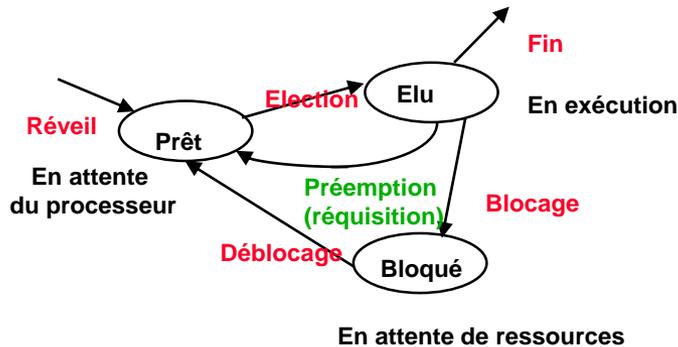
Systeme multiprocessus

Etats des processus



Systeme multiprocessus

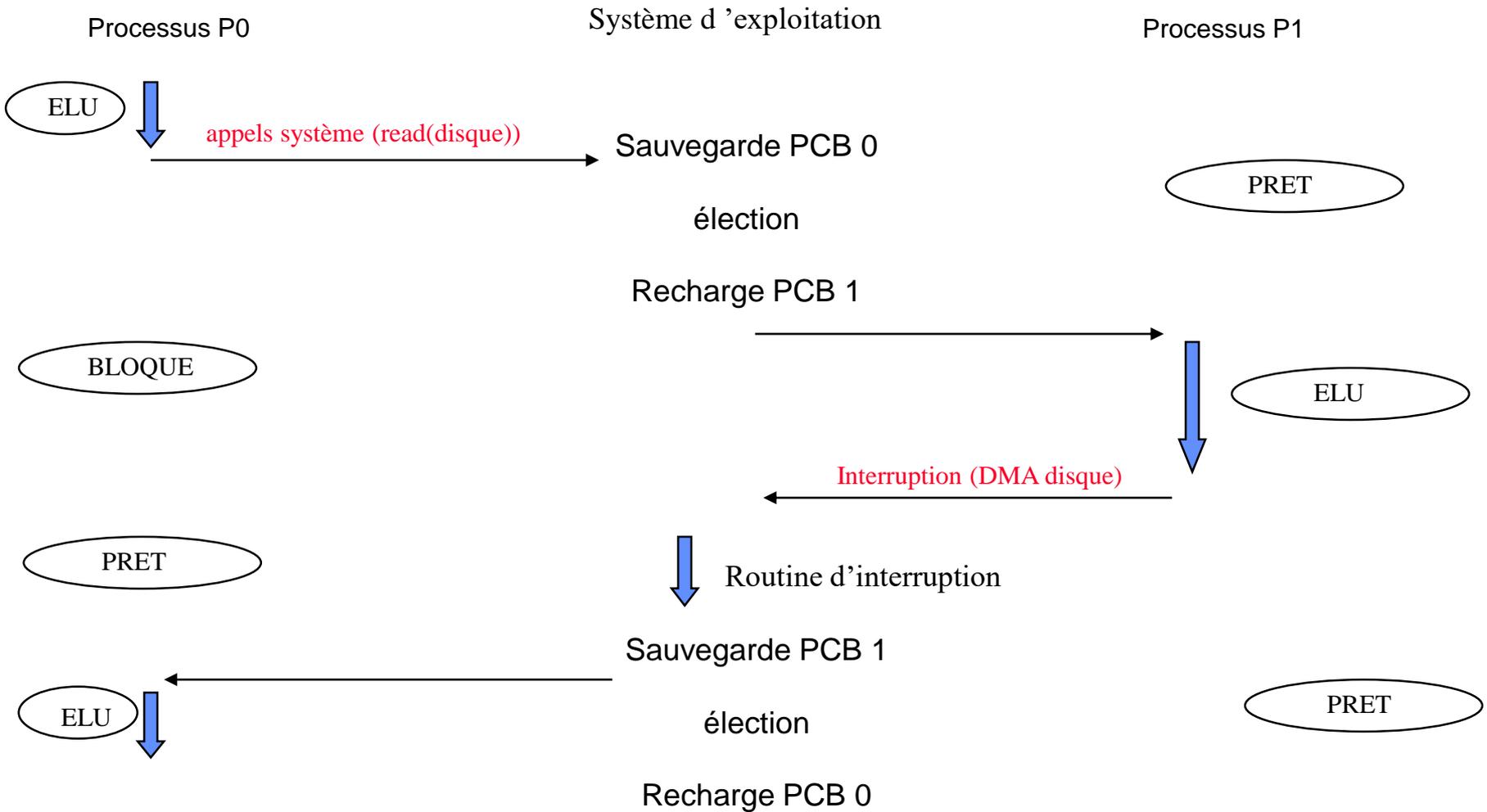
Etats des processus



- **Election** : allocation du processeur
- **Préemption** : réquisition du processeur
 - ordonnancement non préemptif : un processus élu le demeure sauf s 'il se bloque de lui-même
 - ordonnancement préemptif : un processus élu peut perdre le processeur
 - s 'il se bloque de lui-même (état bloqué)
 - si le processeur est réquisitionné pour un autre processus (état prêt)

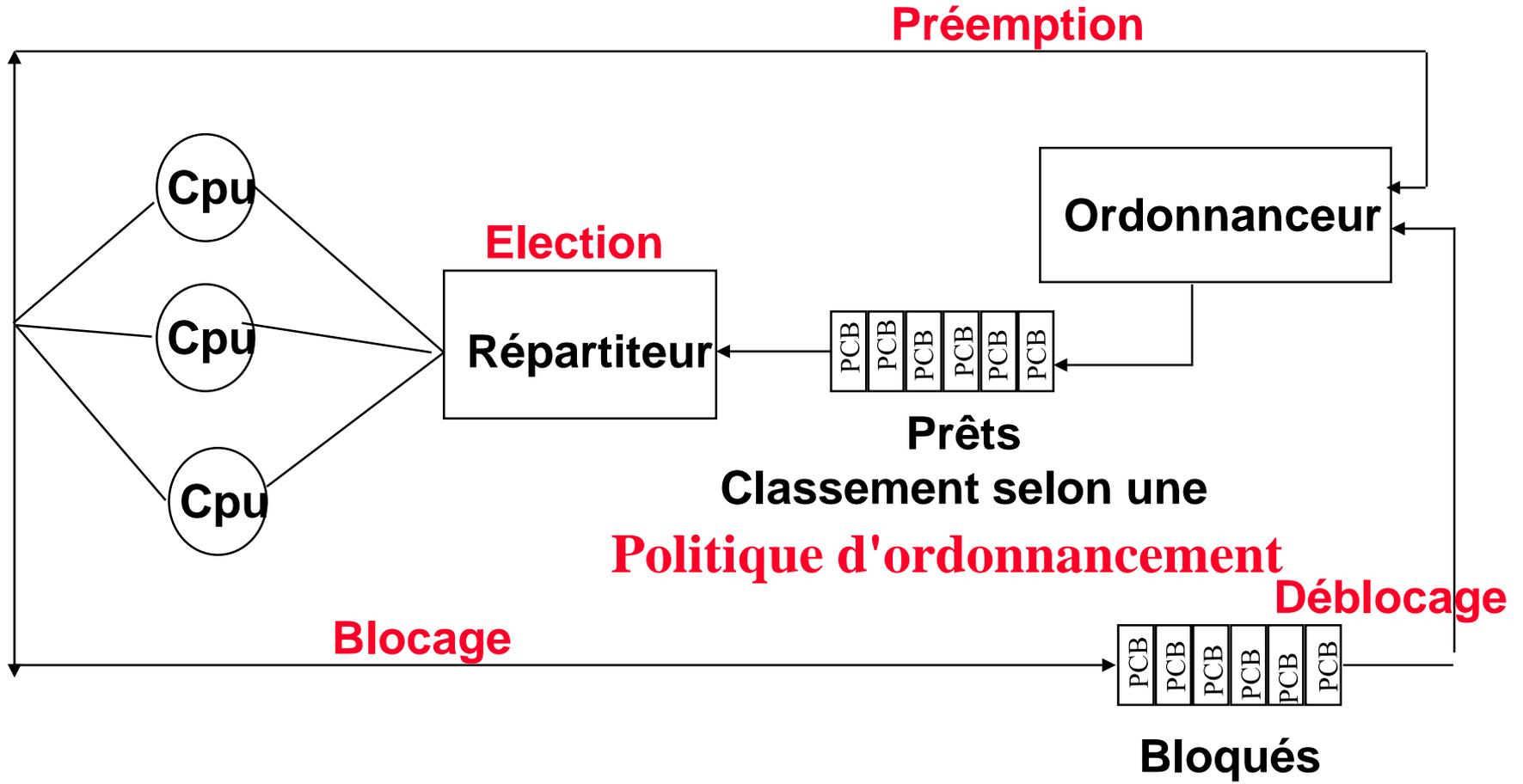
Systeme multiprocessus

Ordonnancement



Systeme multiprocessus

Ordonnanceur et repartiteur

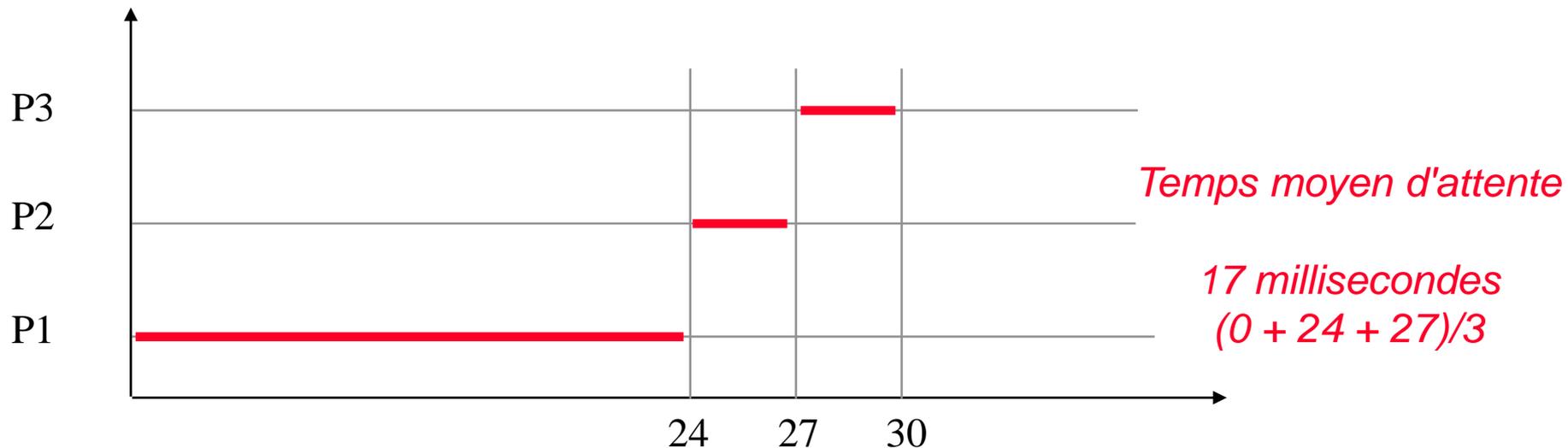
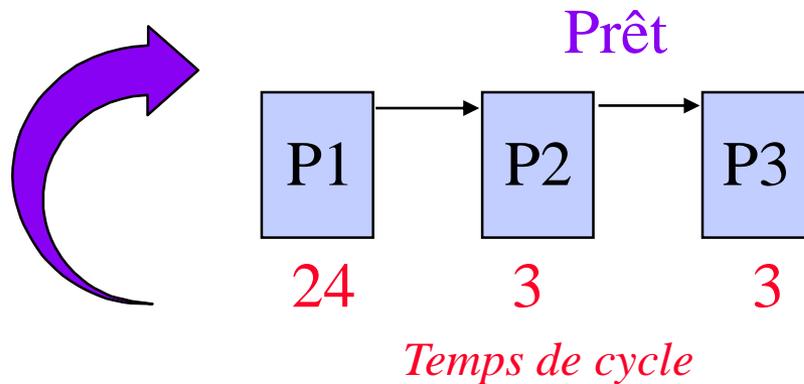


Politiques d'ordonnancement

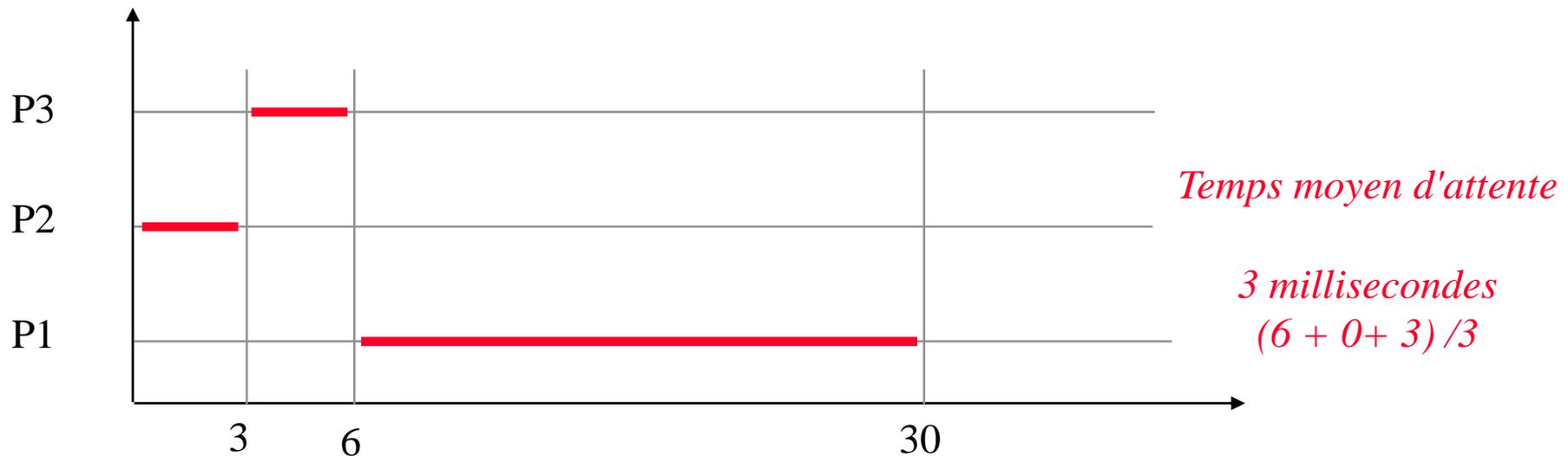
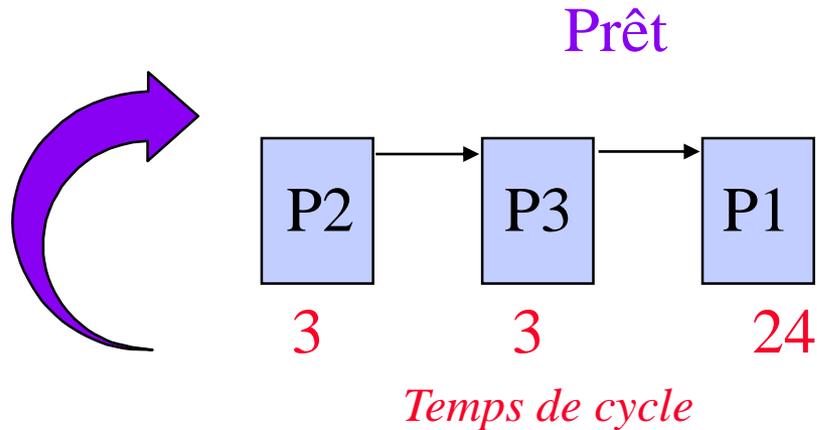
- **Premier arrivé, premier servi**
 - **FIFO, sans réquisition**
- **Par priorités constantes**
- **Par tourniquet (round robin)**
- **Par files de priorités de priorités constantes multiniveaux avec ou sans extinction de priorité**

Algorithme : Premier Arrivé Premier Servi

- FIFO, sans réquisition



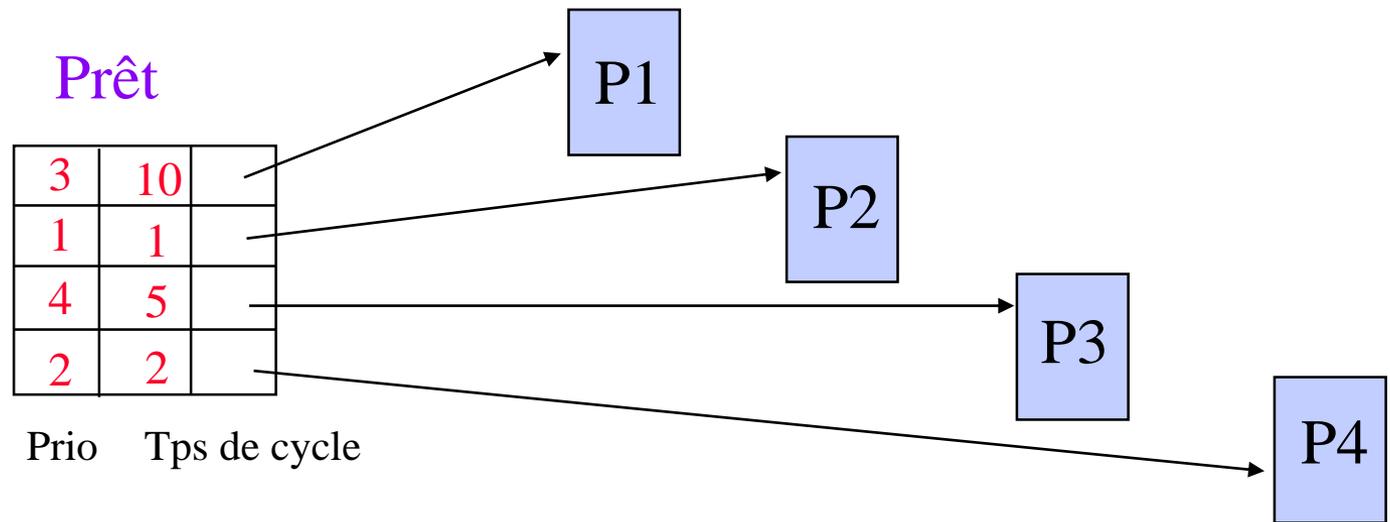
Algorithme : Premier Arrivé Premier Servi



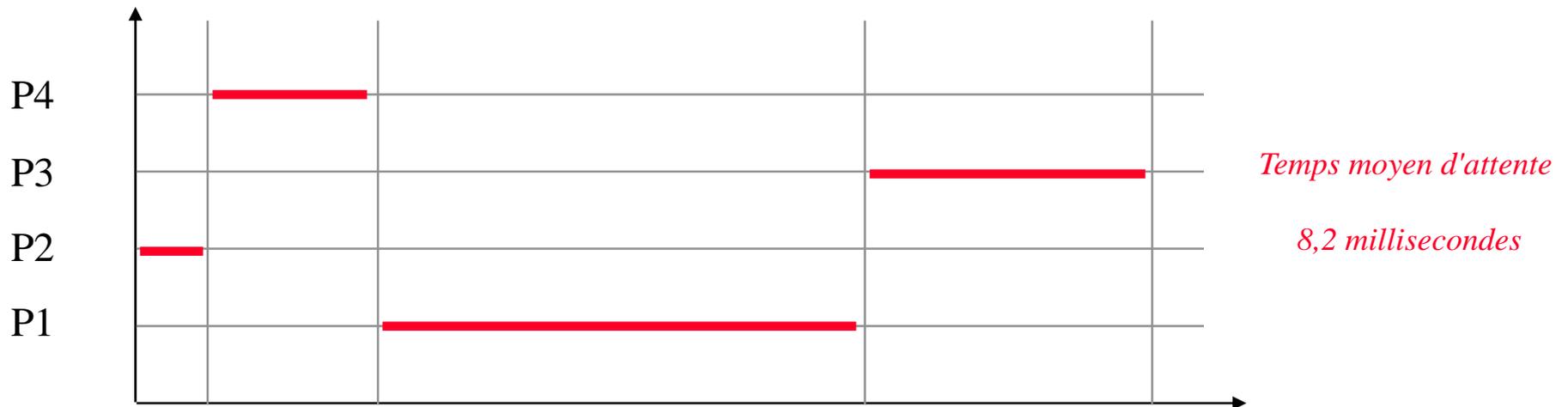
Politiques d'ordonnancement

- **Premier arrivé, premier servi**
- **Par priorités constantes**
 - **chaque processus reçoit une priorité**
 - **le processus de plus forte priorité est élu**
 - **Avec ou sans réquisition**
- **Par tourniquet (round robin)**
- **Par files de priorités de priorités constantes multiniveaux avec ou sans extinction de priorité**

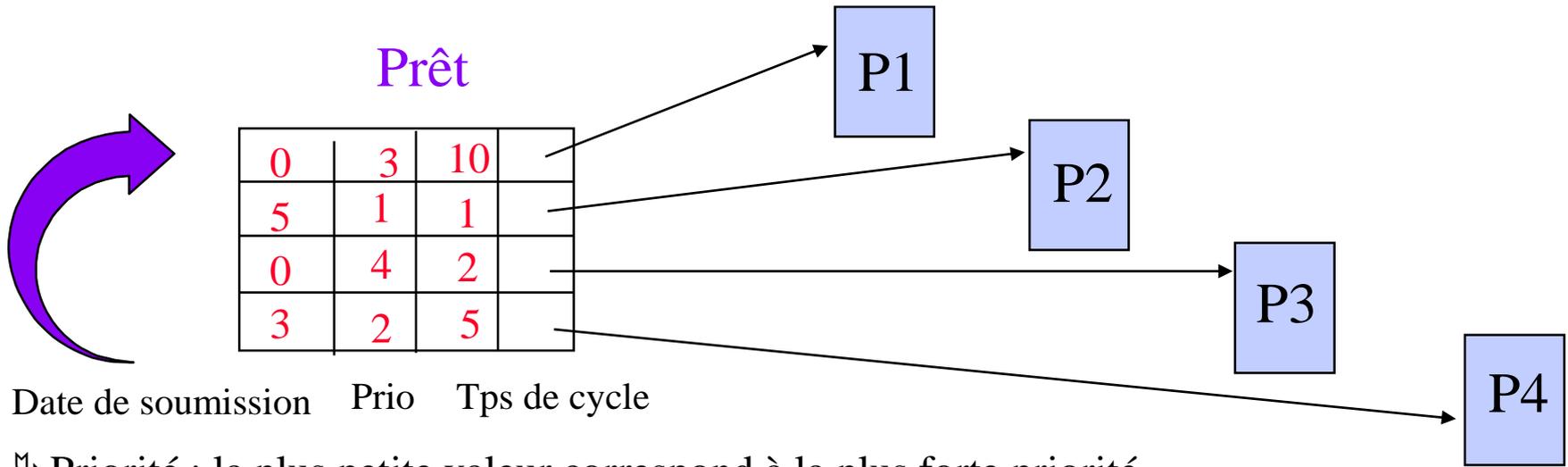
Algorithme : avec priorités



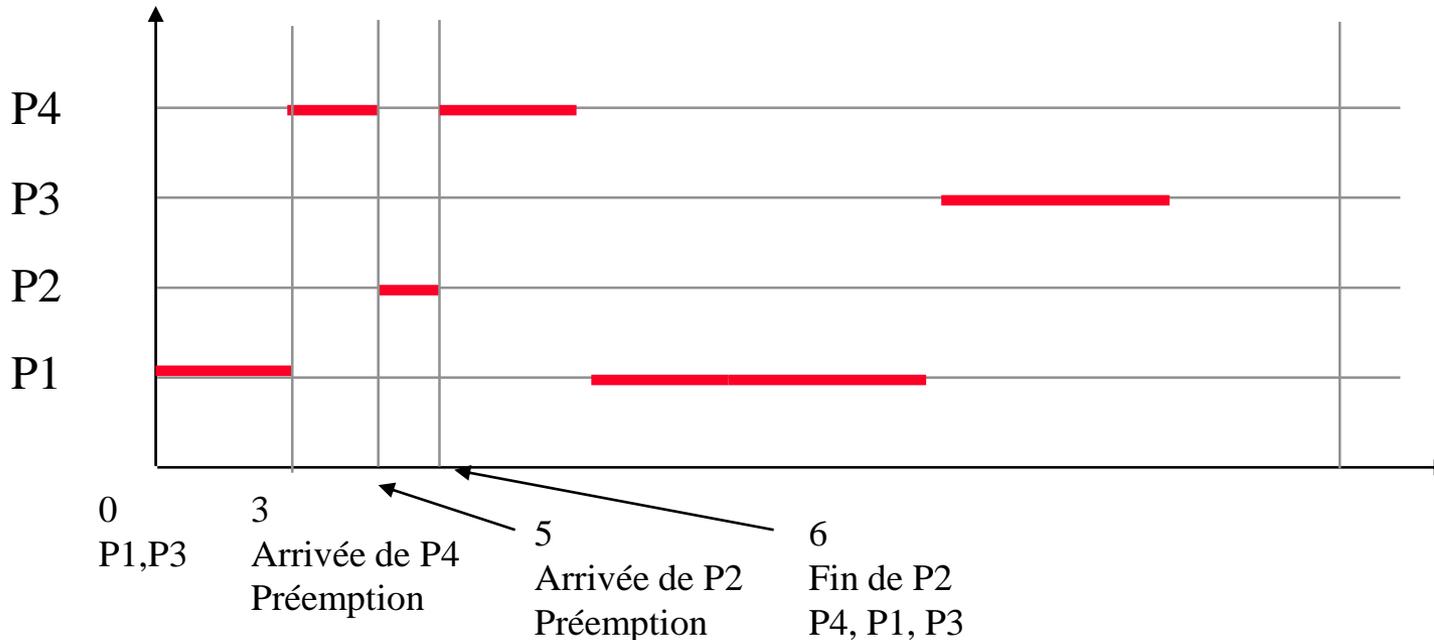
↳ Priorité : la plus petite valeur correspond à la plus forte priorité



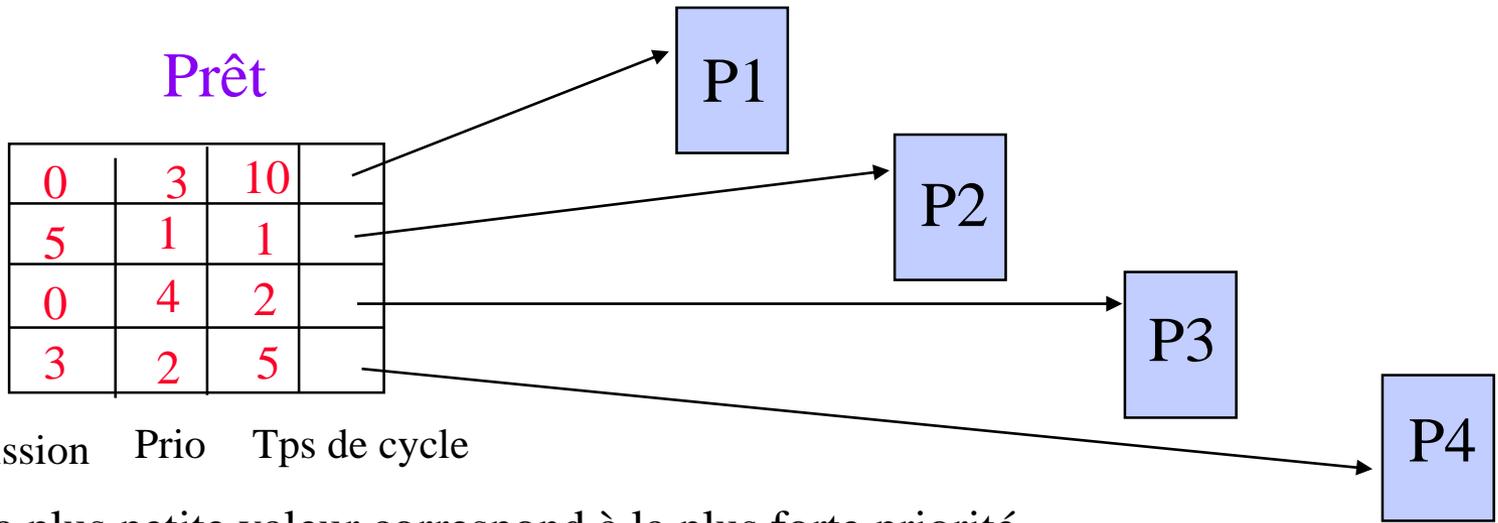
Algorithme : avec priorités préemptif



↳ Priorité : la plus petite valeur correspond à la plus forte priorité

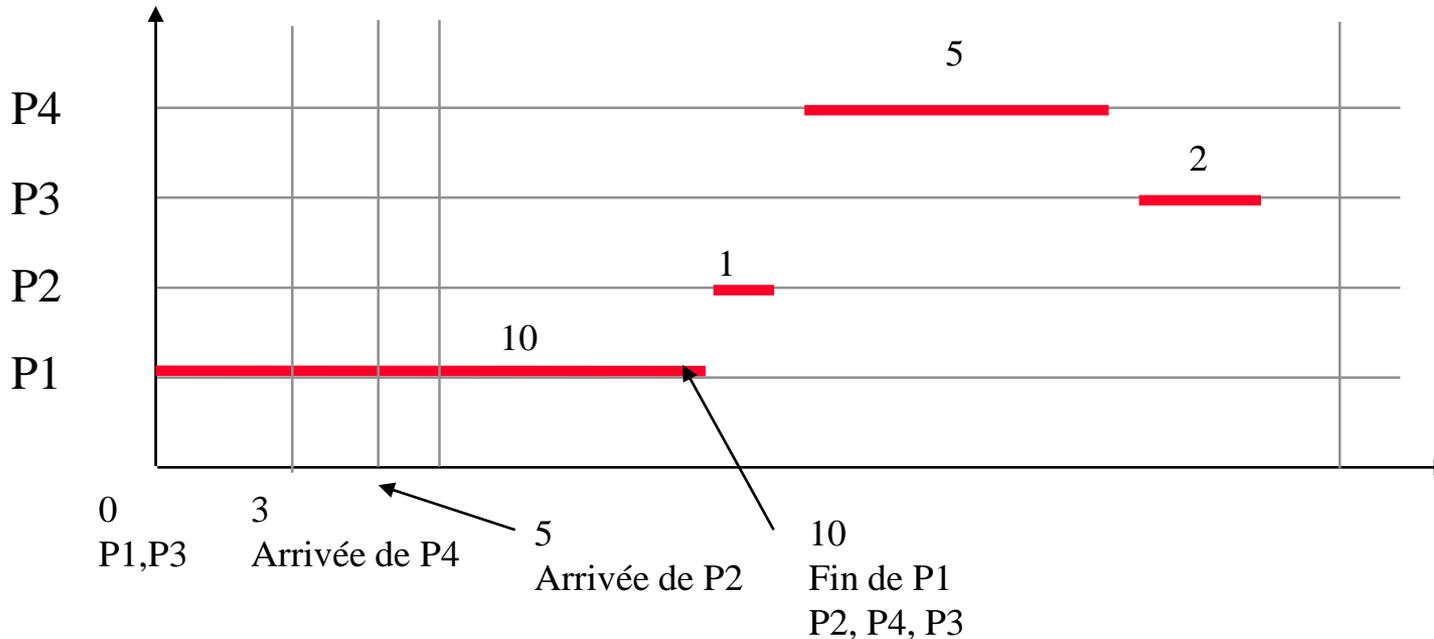


Algorithme : avec priorités non préemptif

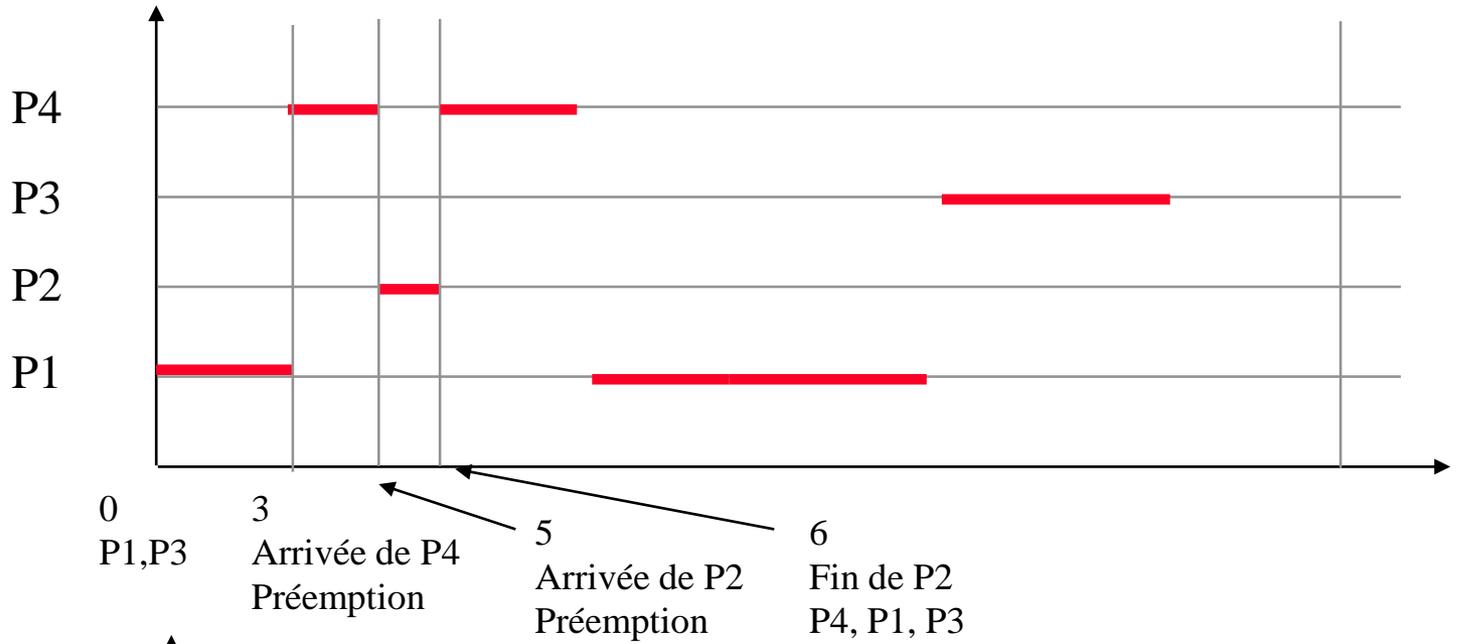


Date de soumission Prio Tps de cycle

↳ Priorité : la plus petite valeur correspond à la plus forte priorité

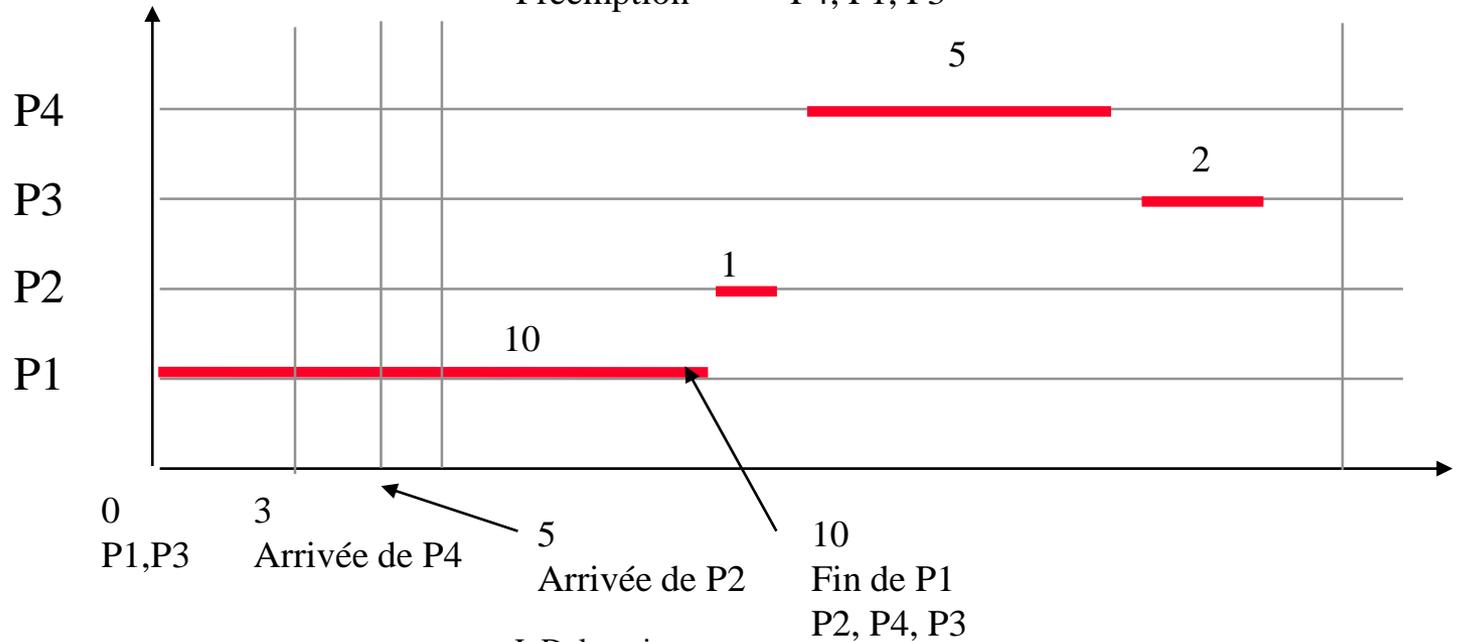


P
R
E
E
M
P
T
I
F



N
O
N

P
R
E
E
M
P
T
I
F

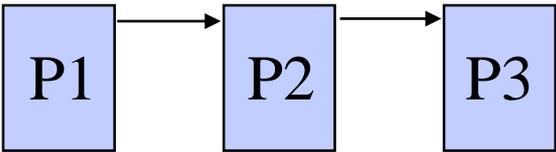


Politiques d'ordonnancement

- Premier arrivé, premier servi
- Par priorités constantes
- **Par tourniquet (round robin)**
 - Définition d'un quantum = tranche de temps
 - Un processus élu s'exécute au plus durant un quantum; à la fin du quantum, préemption et réinsertion en fin de file d'attente des processus prêts
- Par files de priorités de priorités constantes multiniveaux avec ou sans extinction de priorité

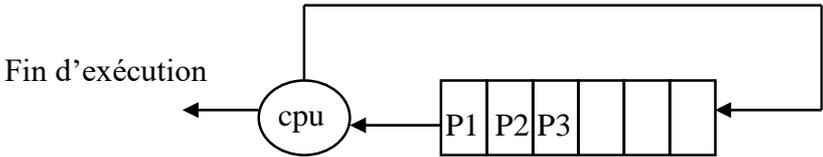
Algorithme : tourniquet

Prêt

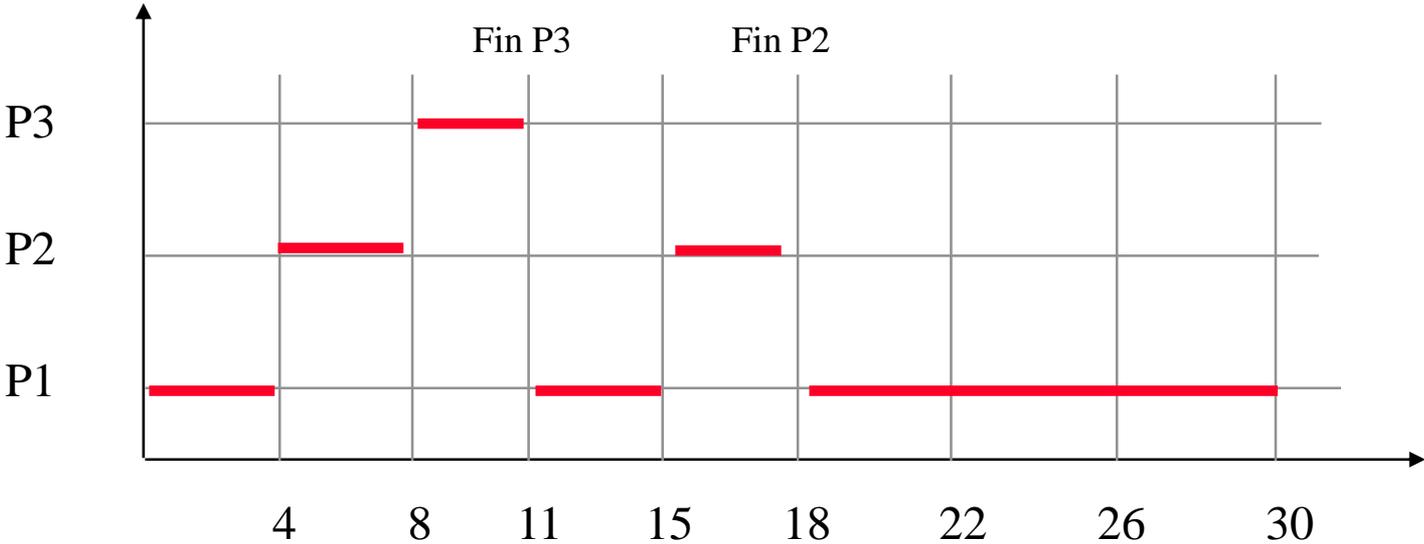


20 7 3

Temps de cycle



Quantum = 4



P1	P2	P3	P1	P2	P1
P2	P3	P1	P2	P1	
P3	P1	P2			

Politiques d'ordonnancement

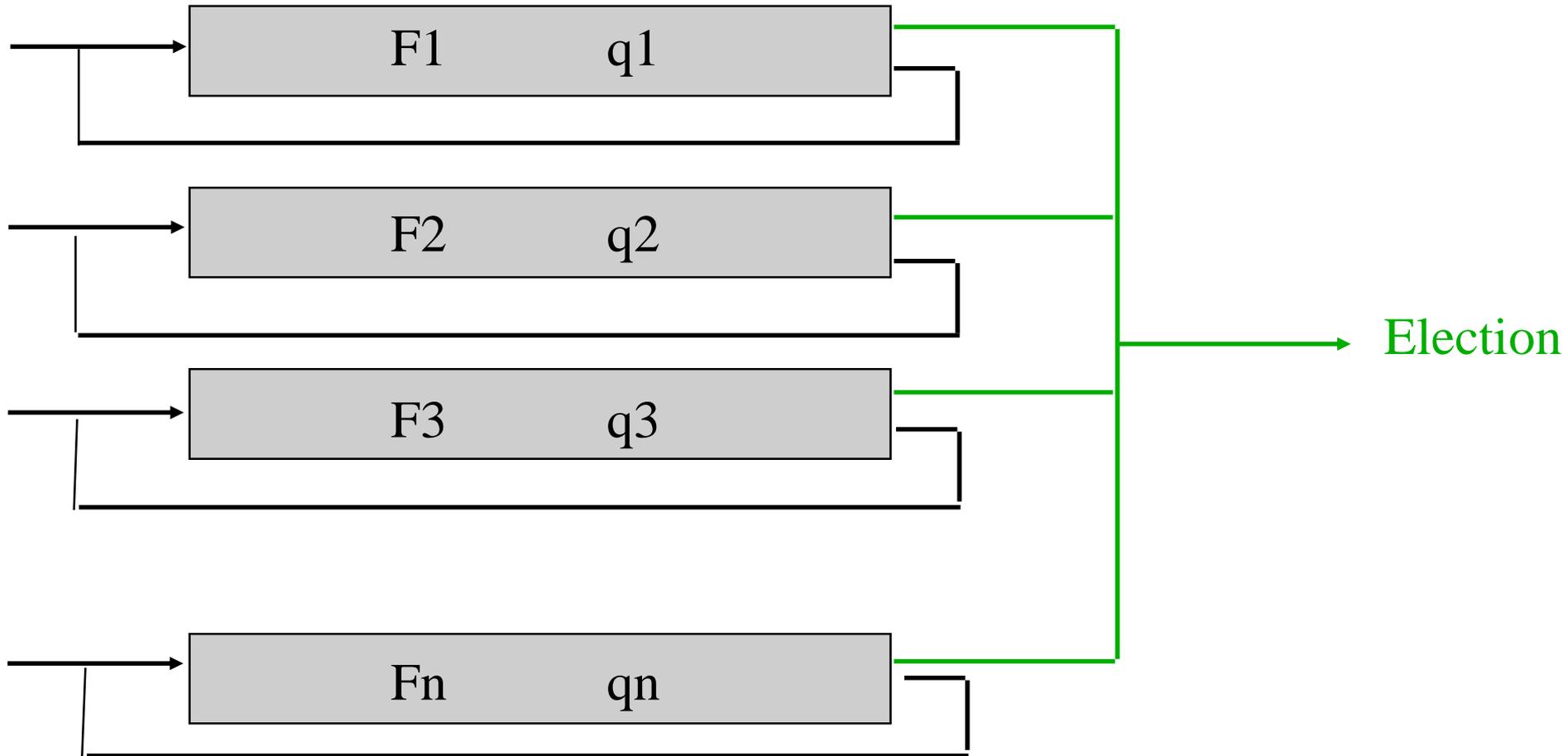
- **Premier arrivé, premier servi**
- **Par priorités constantes**
- **Par tourniquet (round robin)**

- **Par files de priorités de priorités constantes multiniveaux avec ou sans extinction de priorité**
 - **chaque file est associée à un quantum éventuellement différent**
 - **sans extinction : un processus garde toujours la même priorité**
 - **avec extinction : la priorité d'un processus décroît en fonction de son utilisation de la cpu**

Algorithme : multifeiles sans extinction

Prêt

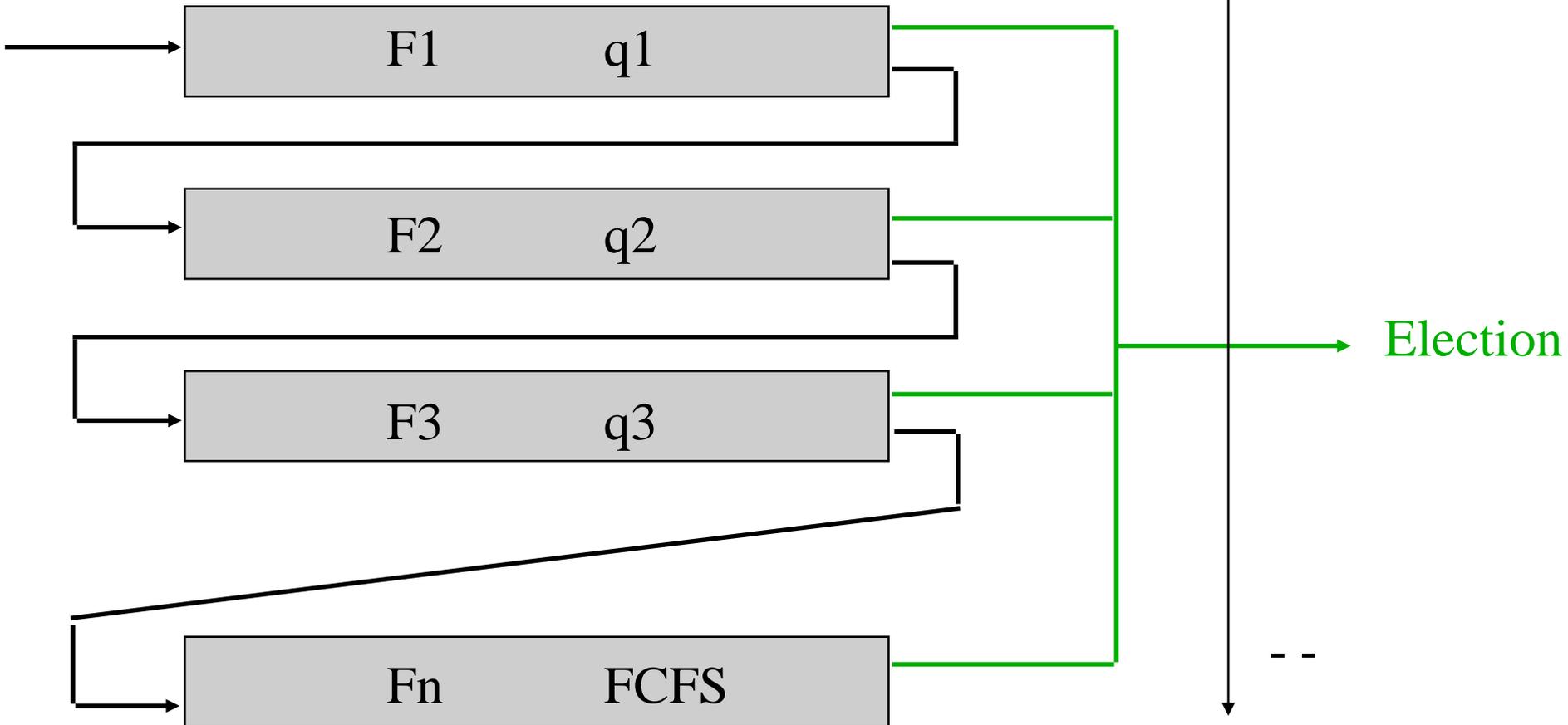
Arrivée



Algorithme : multifeiles avec extinction

Prêt

Arrivée



Politiques d'ordonnancement

Politique d'ordonnancement	Commentaire
FIFO	Pénalise les processus de court temps d'exécution
Priorité	Risque de famine des processus de faible priorité
Tourniquet	Valeur de quantum par rapport au coût des commutations de contexte

Ordonnancement dans le système LINUX

Systeme multiprocessus

Ordonnancement LINUX

Le système Linux est un gestionnaire de processus.

Il offre des services aux processus

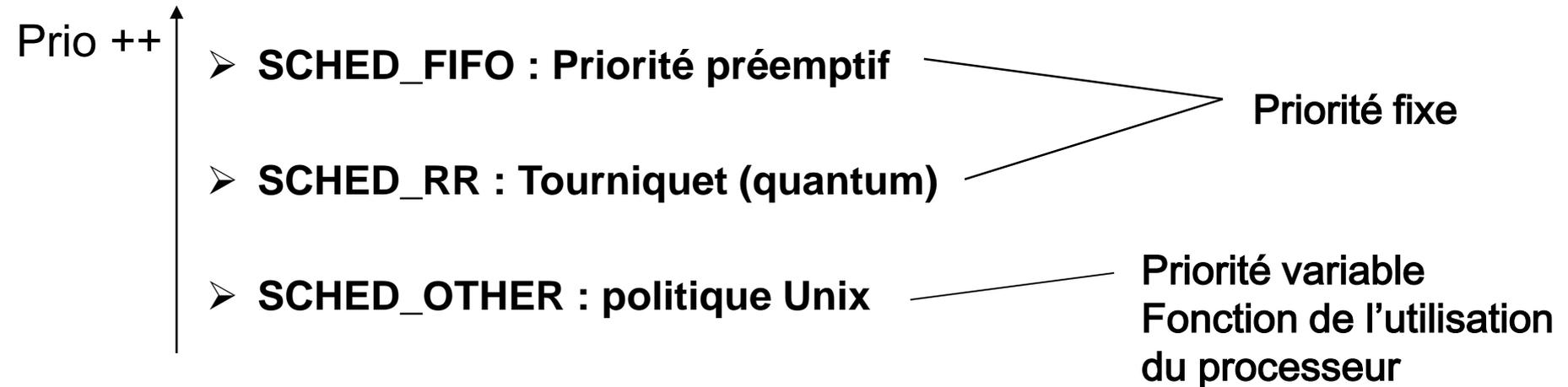
Il ne comporte pas à proprement parler de processus qui exécutent son code.

Ce sont les processus utilisateurs qui en passant en mode noyau exécutent le code du système

L'ordonnancement est lancé à chaque fois qu'un processus utilisateur s'apprête à repasser en mode utilisateur depuis le mode noyau et la variable noyau `need_resched = true`.

Ordonnancement : système LINUX

- Trois classes d'ordonnancement (norme POSIX) :



A l'instant t, le système élit (fonction GOODNESS du noyau)

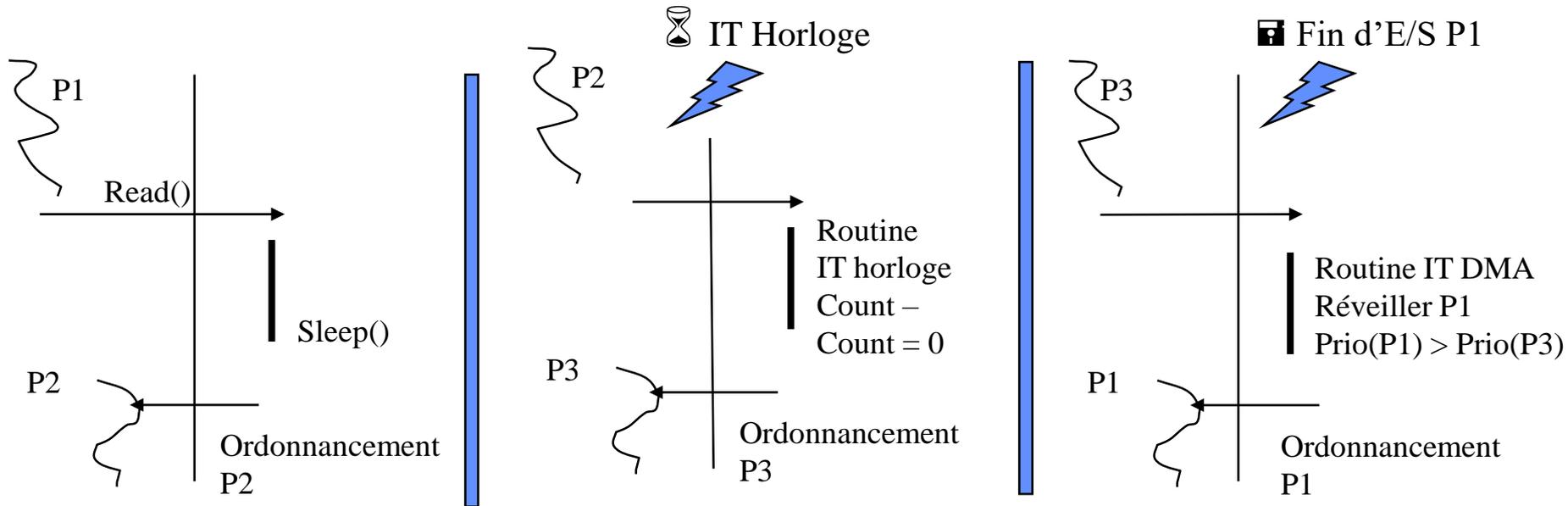
- Le processus **SCHED_FIFO** de plus forte priorité qui s'exécute jusqu'à sa fin ou jusqu'à préemption par un processus FIFO plus prioritaire
- Le processus **SCHED_RR** de plus forte priorité pour un quantum
- Le processus **SCHED_OTHER** de plus forte priorité

Systeme multiprocessus

Ordonnancement LINUX

Positionnement de need_resched :

- Un processus passe en mode bloqué ;
- Un processus a épuisé son quantum;
- Un processus temps réel plus prioritaire est réveillé.



Ordonnancement dans le système LINUX

