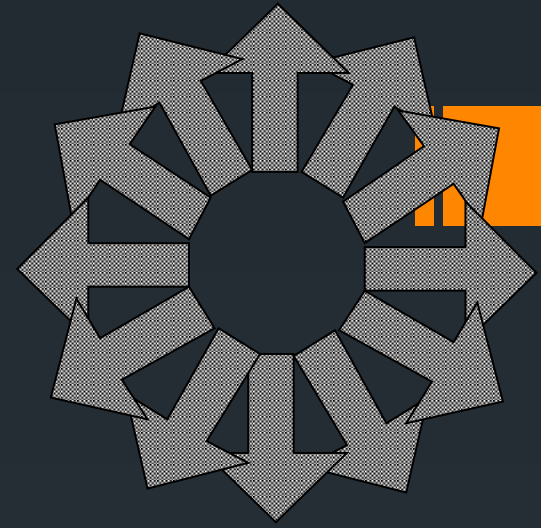


# Systemes d'exploitation



- La gestion des tâches

# Systemes complexes

- 1 Composantes du système
- 2 Notions de tâche et de processus
- 3 Modes de multiprogrammation
- 4 Gestion de l'unité centrale



# 1. Les composantes du système

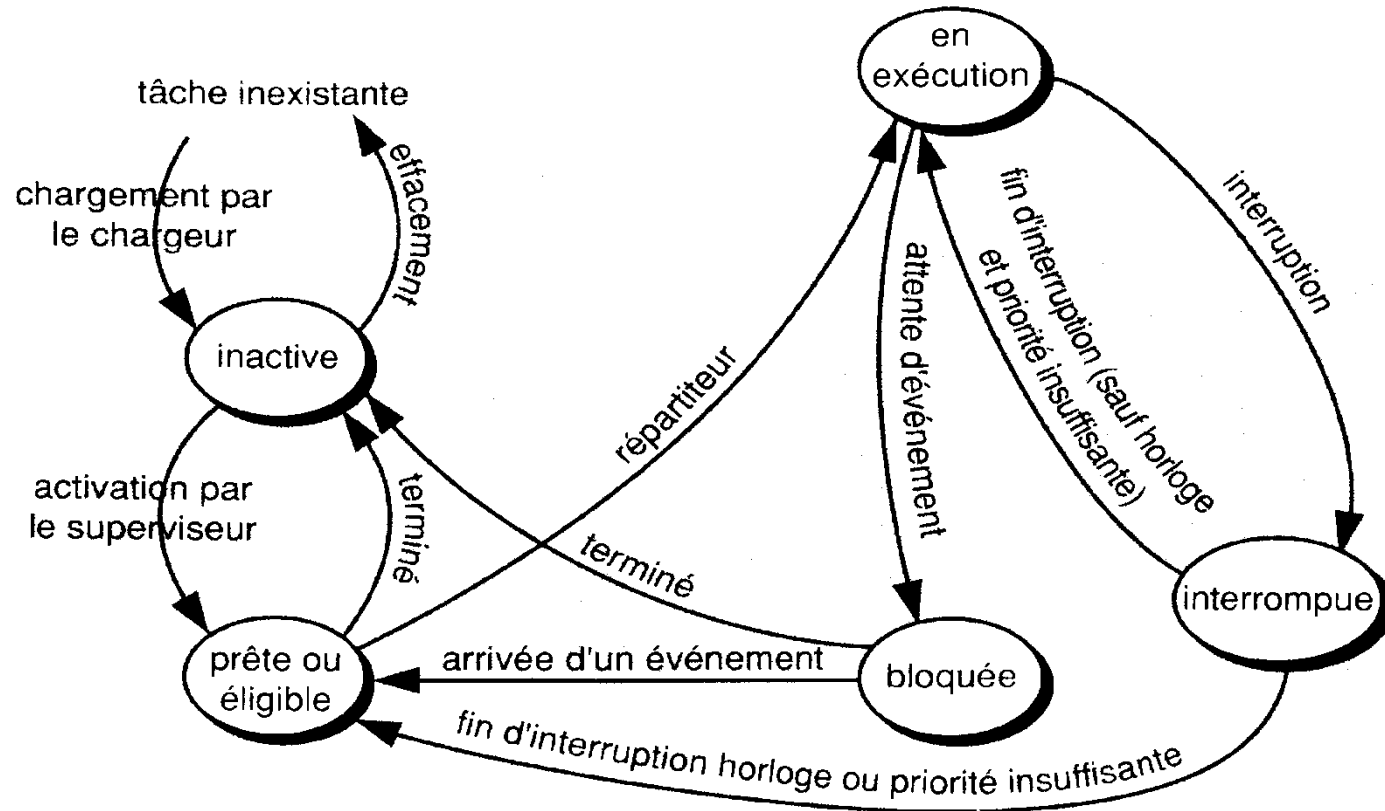
- Interprète de commandes
- Ensemble de requêtes (*les primitives*)
- Programme de gestion des interruptions
- Tables du système
- + **Superviseur** ← *ce qui est ajouté pour un système multitâche*
- + **Répartiteur** ← *ce qui est ajouté pour un système multitâche*

## 2.1 Tâches et processus : définitions

- **Tâche :**  
suite d'instructions en mémoire constituant un ensemble logique en cours d'exécution ou susceptible de s'exécuter.
- **Processus :**  
occurrence de programme en cours d'exécution ou attendant son exécution.
- **Tâche vit et meurt :**  
engendrée par commande, programme, sous-programme, programme de gestion des interruptions.
- **Un programme >>>> plusieurs tâches !**

**Tâche = Processus** (*selon le contexte*)

## 2.2 Tâches et processus : États d'une tâche

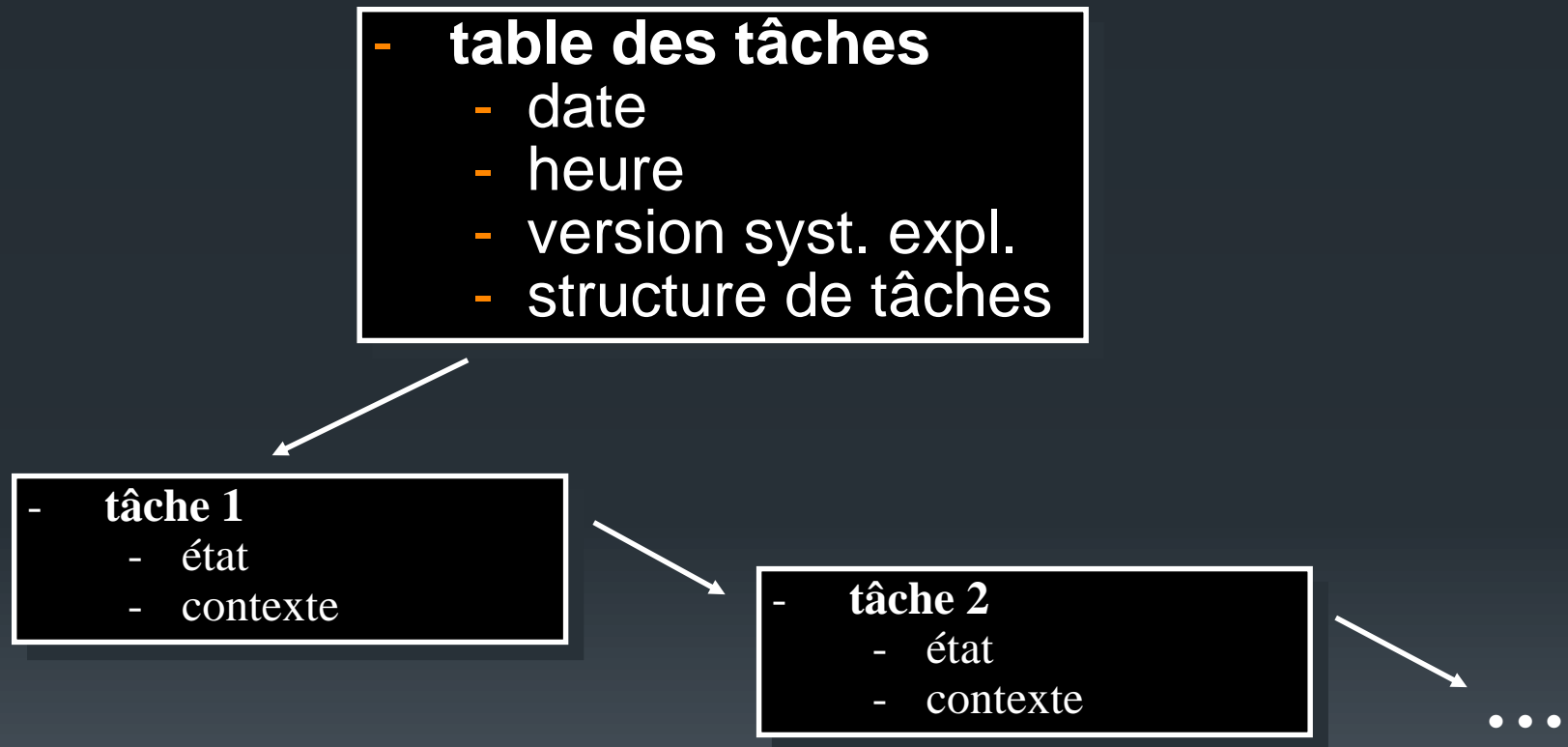




# État d'une tâche

- **En exécution**
    - Dispose de l'UCT, ses instructions qui s'exécutent
  - **Interrompue**
    - Programme plus prioritaire qui s'exécute
  - **Prête**
    - Candidate, mais une autre tâche s'exécute
  - **Bloquée**
    - Attend un évènement
  - **Inactive**
    - N'a pas encore débutée, ou a terminé
- \*\* l'état est géré par le superviseur**

## 2.3 Tâches et processus : représentation d'une tâche



## 2.4 Tâches et processus : Notions de priorité



**superviseur attribue la priorité**

- ressources requises
- période écoulée
- importance de l'utilisateur

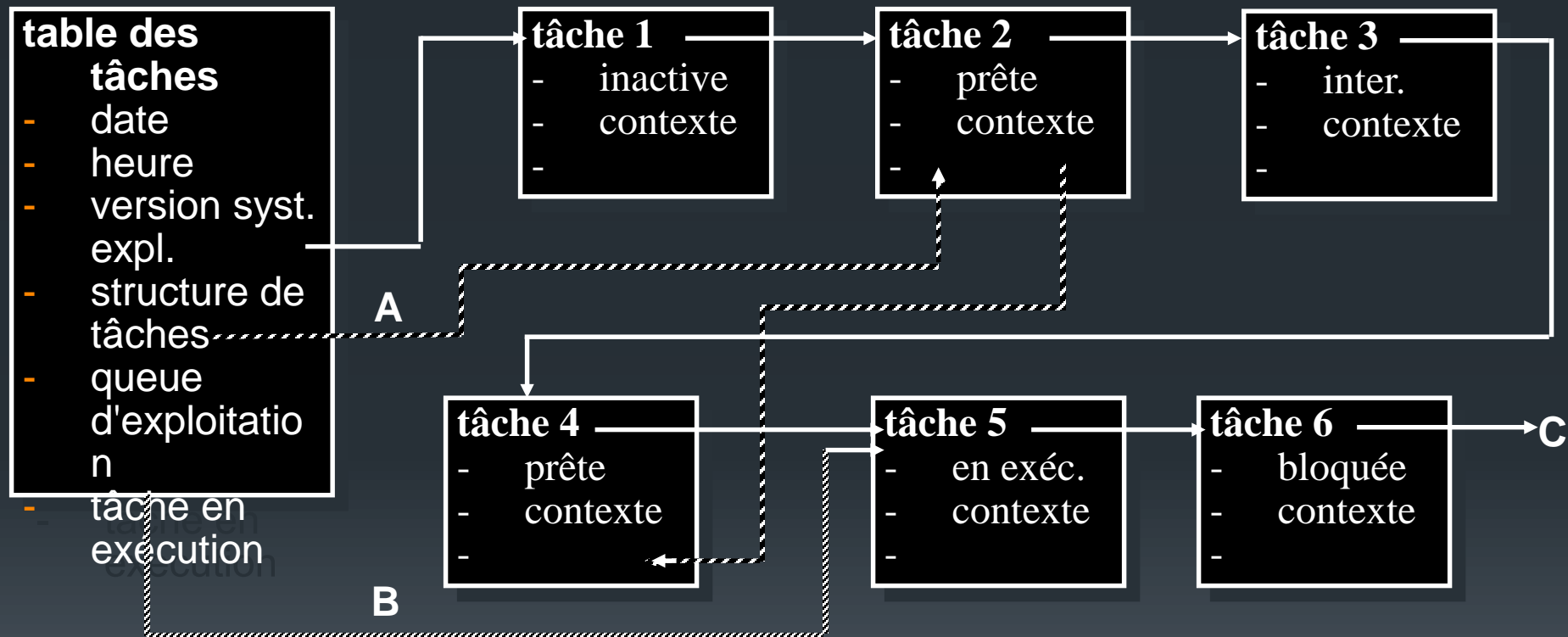
**répartiteur sélectionne tâche éligible**





## 2.4 Tâches et processus : Notions de priorité (queue)

- Queue d'exploitation :



## 2.5 Tâches et processus :

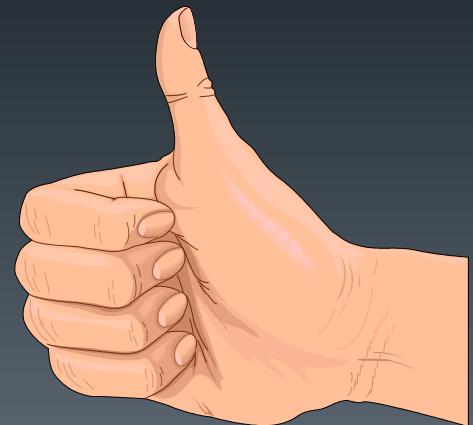
### Relations entre les tâches

- **Indépendantes**
  - les résultats ou données de l'autre ne sont pas requis
  - les ressources sont locales.
- **Parallèles**
  - elles partagent une même ressources.
- **Concurrentes**
  - elles sont mutuellement exclusive:  
utilisent une même ressource non partageable,
- **Coopérantes**
  - a besoin du résultat d'une autre pour continuer,

## 2.6 Tâches et processus : Synchronisation des tâches

- **Utilisation de sémaphores :**
  - variable entière initialisée au nombre de tâches qui peuvent accéder simultanément à une ressource;
  - requêtes :
    - P (S)      Wait — Puis-je utiliser une ressource ?
    - V(S) Signal — Vas-y !

où S est la variable sémaphore
- **Exemples :**
  - pour des tâches concurrentes:
    - S(M) M initialisée à 1
    - section critique (*qu'est-ce que la section critique ?*)
  - pour des tâches coopérantes
    - S(C) C initialisée à 0



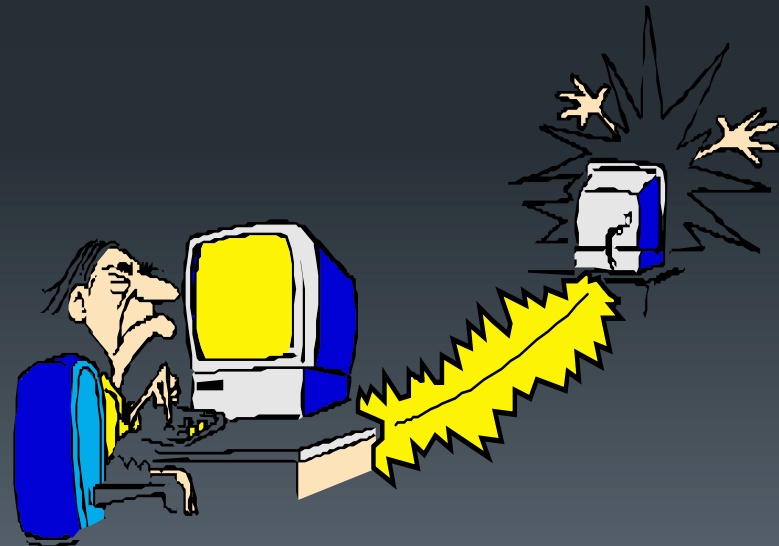
## 2.6 Tâches et processus :

### Synchronisation des tâches

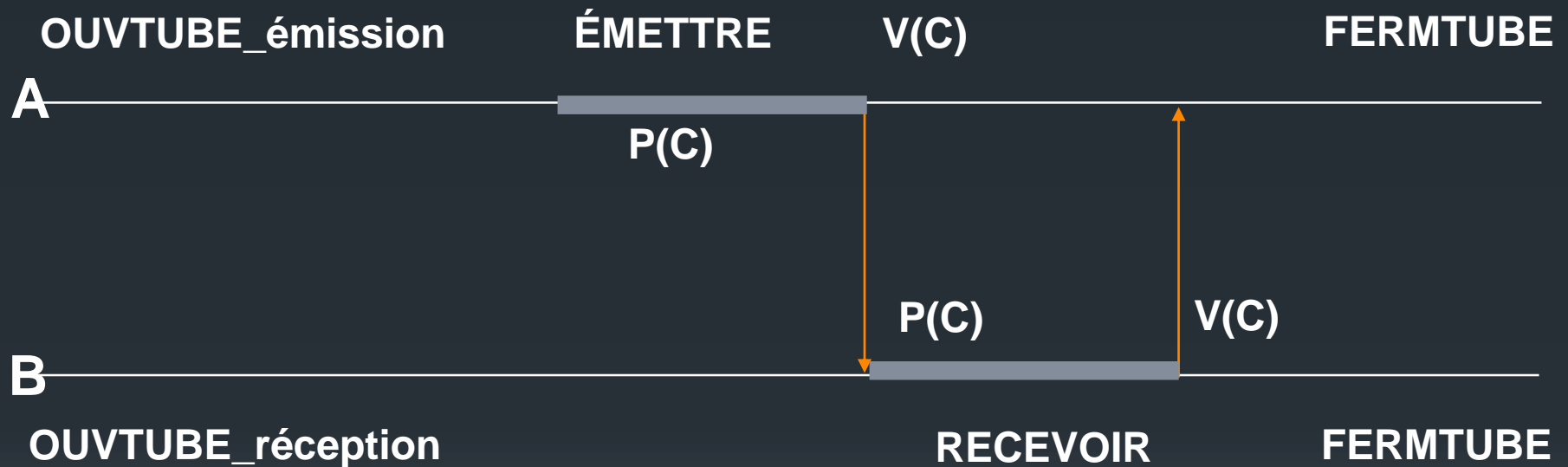
### Les Tubes

#### ■ Utilisation de tubes (*Pipe*)

- tampon qui permet à deux tâches de communiquer, généralement unidirectionnel
- création d'un tube
  - OUVTUBE\_réception
  - OUVTUBE\_émission
- émission d'un message
- réception d'un message
- suppression d'un tube
  - FERMTUBE\_réception
  - FERMTUBE\_émission



## 2.6 Tâches et processus : Synchronisation des tâches Les tubes



## 2.6 Tâches et processus : Synchronisation des tâches L'interblocage

- Interblocage ou étreinte mortelle ! (*deadlock*)

### Programme de la tâche A

...  
P(SA)  
...  
P(SB)  
...  
V(SB)

### Programme de la tâche B

...  
P(SB)  
...  
P(SA)  
...

# 3. Modes de multiprogrammation

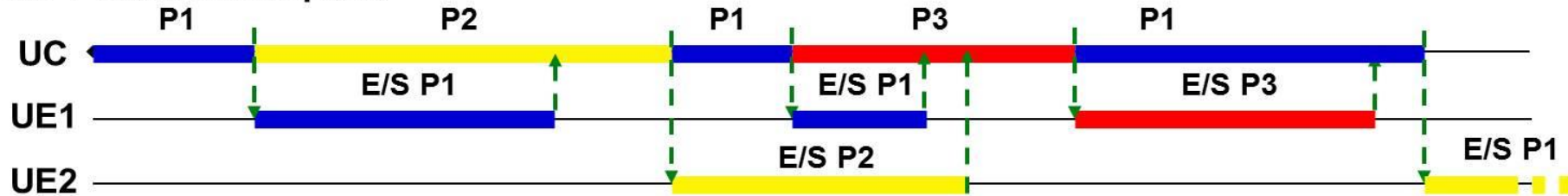


- Sans interruption
  - La tâche est interrompue seulement lors de requête externe
- Avec interruption
  - La tâche est interrompue lors de requête externe
  - La tâche est interrompue lorsqu'une tâche plus prioritaire est susceptible de s'exécuter
- En temps partagé
  - Chaque tâche a une tranche de temps
  - Interrompue si son temps est écoulé
  - Interrompue si requête externe

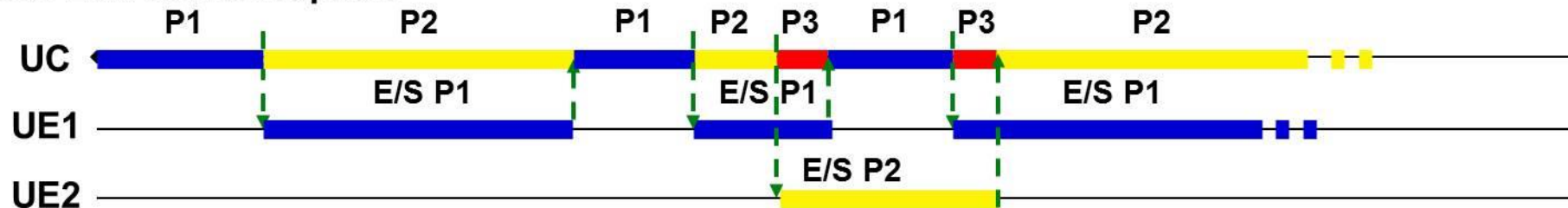
# Mode de multiprogrammation

\*\*\* P1 plus prioritaire que P2 qui est plus prioritaire que P3

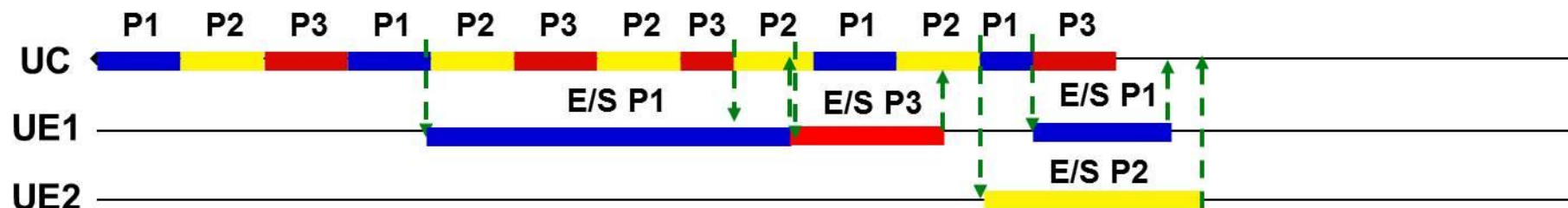
## Cas 1 Sans interruption



## Cas 2 Avec interruption



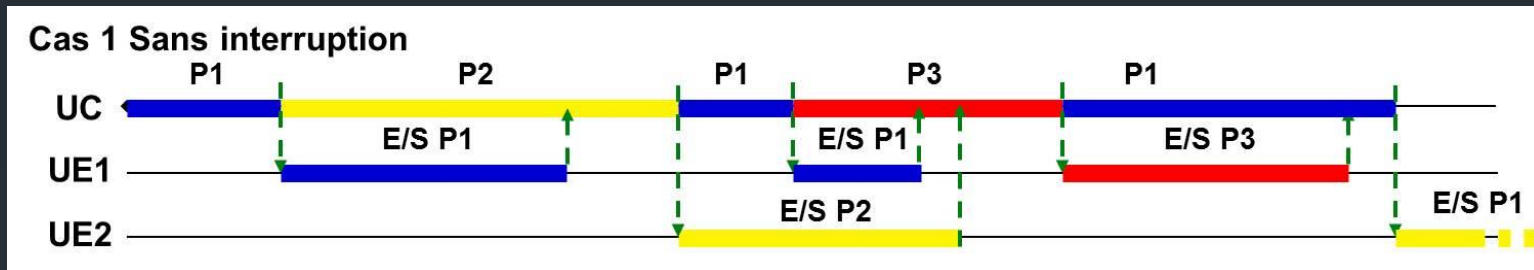
## Cas 3 Temps partagé





# Mode de multiprogrammation : Sans interruption

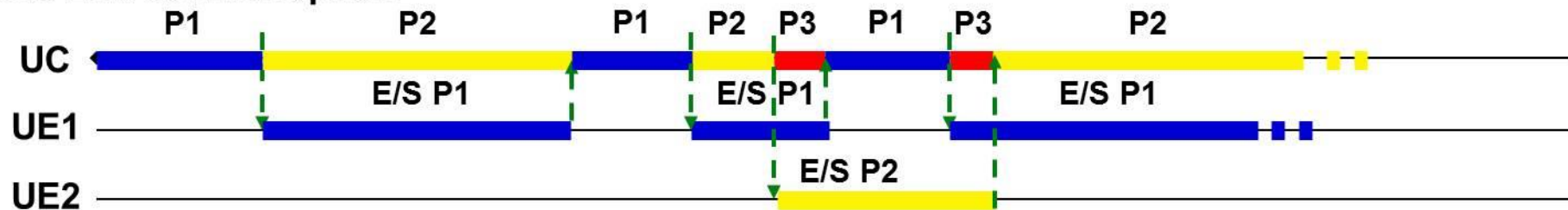
- La tâche s'exécute et dispose de la ressource jusqu'à la fin de son exécution ou jusqu'à ce qu'elle fasse une demande d'entrée/sortie.



# Mode de multiprogrammation : Avec interruption

- La tâche s'exécute et dispose de la ressource jusqu'à ce qu'elle fasse une demande d'entrée/sortie. Lorsque l'entrée/sortie est terminée, c'est le programme le plus prioritaire qui va s'exécuter.

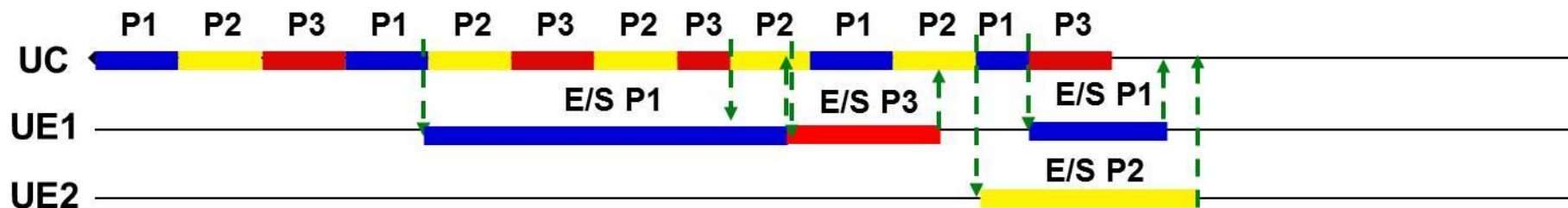
Cas 2 Avec interruption



# Mode de multiprogrammation : Temps partagé

- On ajoute aux interruptions d'entrées/sorties le interruption d'horloge..

## Cas 3 Temps partagé



## 4. Gestion de l'unité centrale



- **Rôle du répartiteur :**
  - sauvegarder contexte de la tâche suspendue;
  - rechercher tâche à lancer dans la queue d'exploitation.



- **Rôle du superviseur :**
  - introduire de nouveaux processus;
  - attribuer la priorité;
  - gérer les ressources.