



# Interblocages

## Master 2ITIC - UFR S.A.T

Prof. Ousmane THIARE

`othiare@ugb.edu.sn`  
`[www.ousmanethiare.com]`

16 avril 2020

# Chapitre 5 : Interblocages

Rappel

Définition d'un interblocage

Conditions nécessaires pour l'interblocage

Graphe d'allocation des ressources

Réduction du graphe d'allocation des ressources

Traitement des interblocages

La détection et la reprise

M  
tr  
in  
Pr  
in



- 1 Rappel
- 2 Définition d'un interblocage
- 3 Conditions nécessaires pour l'interblocage
- 4 Graphe d'allocation des ressources
- 5 Réduction du graphe d'allocation des ressources
- 6 Traitement des interblocages
- 7 La détection et la reprise
- 8 Méthodes de traitement des interblocages
- 9 Prévention des interblocages
- 10 Evitement des interblocages
- 11 Détection et guérison des interblocages

## Rappel

Définition d'un interblocage

Conditions nécessaires pour l'interblocage

Graphe d'allocation des ressources

Réduction du graphe d'allocation des ressources

Traitement des interblocages

La détection et la reprise

Méthodes de traitement des interblocages

Prévention des interblocages

# Chapitre 5 : Interblocages

# Interblocages

## Les processus et les ressources

### Rappel

Définition d'un interblocage

Conditions nécessaires pour l'interblocage

Graphe d'allocation des ressources

Réduction du graphe d'allocation des ressources

Traitement des interblocages

La détection et la reprise



L'exécution d'un processus nécessite un ensemble de ressources (mémoire principale, disques, fichiers, périphériques, etc.) qui lui sont attribuées par le système d'exploitation. L'utilisation d'une ressource passe par les étapes suivantes :

- **Demande de la ressource** : Si l'on ne peut pas satisfaire la demande il faut attendre. La demande sera mise dans une table d'attente des ressources.

# Interblocages

## Les processus et les ressources

### Rappel

Définition d'un interblocage

Conditions nécessaires pour l'interblocage

Graphe d'allocation des ressources

Réduction du graphe d'allocation des ressources

Traitement des interblocages

La détection et la reprise



L'exécution d'un processus nécessite un ensemble de ressources (mémoire principale, disques, fichiers, périphériques, etc.) qui lui sont attribuées par le système d'exploitation. L'utilisation d'une ressource passe par les étapes suivantes :

- **Demande de la ressource** : Si l'on ne peut pas satisfaire la demande il faut attendre. La demande sera mise dans une table d'attente des ressources.
- **Utilisation de la ressource** : Le processus peut utiliser la ressource.

# Interblocages

## Les processus et les ressources

### Rappel

Définition d'un interblocage

Conditions nécessaires pour l'interblocage

Graphe d'allocation des ressources

Réduction du graphe d'allocation des ressources

Traitement des interblocages

La détection et la reprise



L'exécution d'un processus nécessite un ensemble de ressources (mémoire principale, disques, fichiers, périphériques, etc.) qui lui sont attribuées par le système d'exploitation. L'utilisation d'une ressource passe par les étapes suivantes :

- **Demande de la ressource** : Si l'on ne peut pas satisfaire la demande il faut attendre. La demande sera mise dans une table d'attente des ressources.
- **Utilisation de la ressource** : Le processus peut utiliser la ressource.
- **Libération de la ressource** : Le processus libère la ressource demandée et allouée.

# Interblocages

## Les processus et les ressources

### Rappel

Définition d'un interblocage

Conditions nécessaires pour l'interblocage

Grappe d'allocation des ressources

Réduction du graphe d'allocation des ressources

Traitement des interblocages

La détection et la reprise



Lorsqu'un processus demande un accès exclusif à une ressource déjà allouée à un autre processus, le système d'exploitation décide de le mettre en attente jusqu'à ce que la ressource demandée devienne disponible ou lui retourner un message indiquant que la ressource n'est pas disponible : réessayer plus tard.

Les ressources peuvent être de plusieurs types :

- **Reutilisables ou disponibles** Existent-elles après son utilisation ?
- Reutilisable
  - Toutes les ressources physiques.
  - Quelques unes logiques (fichiers, mutex, verrous, etc.).

# Interblocages

## Les processus et les ressources

### Rappel

Définition d'un interblocage

Conditions nécessaires pour l'interblocage

Graphe d'allocation des ressources

Réduction du graphe d'allocation des ressources

Traitement des interblocages

La détection et la reprise



### ■ Disponibles

- Un processus engendre une ressource et un autre l'utilise.
- Ressources associées à la communication et à la synchronisation comme les messages, les signaux, les sémaphores, etc.
- **D'usage partagé** : Est-ce que la ressource peut être utilisée par plusieurs processus en même temps ? Les ressources partagées n'affectent pas les interblocages.
- **Avec un ou multiples exemplaires** : Existence-ils de multiples exemplaires d'une même ressource ?
- **Préemptible ou non préemptible** : Est-ce qu'on a le droit de retirer une ressource quand elle est utilisée par un processus ?



# Interblocages

## Les processus et les ressources

### Rappel

Définition d'un interblocage

Conditions nécessaires pour l'interblocage

Graphes d'allocation des ressources

Réduction du graphe d'allocation des ressources

Traitement des interblocages

La détection et la reprise

M  
tr  
in  
Pr  
in



La différence principale entre ressources préemptibles et non préemptibles est que les premières peuvent être retirées sans risque au processus qui les détient, tandis que les deuxièmes ne peuvent être retirées sans provoquer des problèmes. Comme exemples de ressources préemptibles nous avons le processeur (où le changement de contexte correspond à l'expropriation et l'état du processeur est copié à la **Table de Contrôle de Processus (BCP)**) et la mémoire virtuelle (le remplacement est une expropriation et le contenu de la page doit être copié au swap). Les imprimantes et les scanners sont des exemples de ressources non préemptibles. Pour étudier le problème des interblocages, nous allons considérer uniquement les ressources non préemptibles.

# Interblocages

## Définition d'un interblocage

### Rappel

### Définition d'un interblocage

### Conditions nécessaires pour l'interblocage

### Graphes d'allocation des ressources

### Réduction du graphe d'allocation des ressources

### Traitement des interblocages

### La détection et la reprise

M  
tr  
in  
Pr  
in



Des problèmes peuvent survenir, lorsque les processus obtiennent des accès exclusifs aux ressources. Par exemple, un processus  $P_1$  détient une ressource  $R_1$  et attend une autre ressource  $R_2$  qui est utilisée par un autre processus  $P_2$  ; le processus  $P_2$  détient la ressource  $R_2$  et attend la ressource  $R_1$ . On a une situation d'**interblocage** (deadlock en anglais) car  $P_1$  attend  $P_2$  et  $P_2$  attend  $P_1$ . Les deux processus vont attendre indéfiniment comme montré sur la figure ci-dessous.

# Interblocages

## Définition d'un interblocage

Rappel

Définition d'un interblocage

Conditions nécessaires pour l'interblocage

Graphe d'allocation des ressources

Réduction du graphe d'allocation des ressources

Traitement des interblocages

La détection et la reprise

M  
tr  
in  
Pr  
in

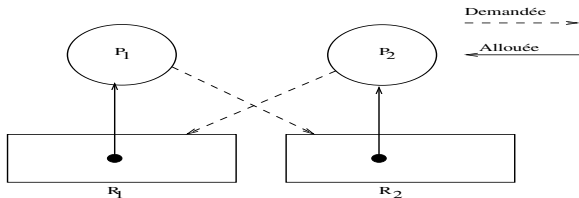


FIGURE: Situation d'interblocage de deux processus

# Interblocages

## Définition d'un interblocage

Rappel

Définition d'un interblocage

Conditions nécessaires pour l'interblocage

Graphe d'allocation des ressources

Réduction du graphe d'allocation des ressources

Traitement des interblocages

La détection et la reprise



En général, un ensemble de processus est en interblocage si chaque processus attend la libération d'une ressource qui est allouée à un autre processus de l'ensemble. Comme tous les processus sont en attente, aucun ne pourra s'exécuter et donc libérer les ressources demandées par les autres. Ils attendront tous indéfiniment.

# Interblocages

## Définition d'un interblocage

Rappel

Définition d'un interblocage

Conditions nécessaires pour l'interblocage

Graphe d'allocation des ressources

Réduction du graphe d'allocation des ressources

Traitement des interblocages

La détection et la reprise

Mi  
tr:  
in  
Pr  
in



**Exemple 1.** Interblocages.

**Accès aux périphériques.** Supposons que deux processus A et B veulent imprimer, en utilisant la même imprimante, un fichier stocké sur une bande magnétique. La taille de ce fichier est supérieure à la capacité du disque. Chaque processus a besoin d'un accès exclusif au dérouleur et à l'imprimante simultanément. On a une situation d'interblocage si :

- Le processus A utilise l'imprimante et demande l'accès au dérouleur.
- Le processus B détient le dérouleur de bande et demande l'imprimante.

# Interblocages

## Définition d'un interblocage

Rappel

Définition d'un interblocage

Conditions nécessaires pour l'interblocage

Grappe d'allocation des ressources

Réduction du graphe d'allocation des ressources

Traitement des interblocages

La détection et la reprise



**Accès à une base de données.** Supposons deux processus A et B qui demandent des accès exclusifs aux enregistrements d'une base de données. On arrive à une situation d'interblocage si :

- Le processus A a verrouillé l'enregistrement  $R_1$  et demande l'accès à l'enregistrement  $R_2$ .
- Le processus B a verrouillé l'enregistrement  $R_2$  et demande l'accès à l'enregistrement  $R_1$ .

# Interblocages

## Définition d'un interblocage

Rappel

Définition d'un interblocage

Conditions nécessaires pour l'interblocage

Graphe d'allocation des ressources

Réduction du graphe d'allocation des ressources

Traitement des interblocages

La détection et la reprise

M  
tr  
in  
Pr  
in



**Circulation routière.** Considérons deux routes à double sens qui se croisent, où la circulation est impossible. Un problème d'interblocage y est présent.

La résolution des interblocages constitue un point théorique très étudié. Mais, pour l'instant, il n'y a pas de solution complètement satisfaisante et la plupart des systèmes - notamment Unix - ne cherchent pas à traiter ce phénomène. Cependant, si on ne peut pas résoudre le problème dans sa généralité, on devra tenir compte de certaines techniques qui minimisent les risques.

# Interblocages

## Définition d'un interblocage

Rappel

Définition d'un interblocage

Conditions nécessaires pour l'interblocage

Graphe d'allocation des ressources

Réduction du graphe d'allocation des ressources

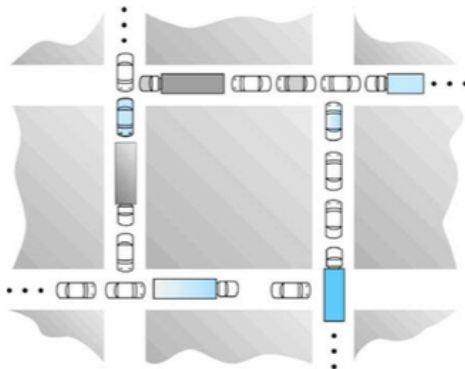
Traitement des interblocages

La détection et la reprise

M  
tr  
in  
Pr  
in



## Circulation routière.





# Interblocages

## Conditions nécessaires pour l'interblocage

Rappel

Définition d'un interblocage

Conditions nécessaires pour l'interblocage

Graphe d'allocation des ressources

Réduction du graphe d'allocation des ressources

Traitement des interblocages

La détection et la reprise



Pour qu'une situation d'interblocage ait lieu, les quatre conditions suivantes doivent être remplies (Conditions de Coffman) :

- **L'exclusion mutuelle.** A un instant précis, une ressource est allouée à un seul processus.

# Interblocages

## Conditions nécessaires pour l'interblocage

Rappel

Définition d'un interblocage

Conditions nécessaires pour l'interblocage

Graphe d'allocation des ressources

Réduction du graphe d'allocation des ressources

Traitement des interblocages

La détection et la reprise



Pour qu'une situation d'interblocage ait lieu, les quatre conditions suivantes doivent être remplies (Conditions de Coffman) :

- **L'exclusion mutuelle.** A un instant précis, une ressource est allouée à un seul processus.
- **La détention et l'attente.** Les processus qui détiennent des ressources peuvent en demander d'autres.

# Interblocages

## Conditions nécessaires pour l'interblocage

### Rappel

### Définition d'un interblocage

### Conditions nécessaires pour l'interblocage

### Graphes d'allocation des ressources

### Réduction du graphes d'allocation des ressources

### Traitement des interblocages

### La détection et la reprise



Pour qu'une situation d'interblocage ait lieu, les quatre conditions suivantes doivent être remplies (Conditions de Coffman) :

- **L'exclusion mutuelle.** A un instant précis, une ressource est allouée à un seul processus.
- **La détention et l'attente.** Les processus qui détiennent des ressources peuvent en demander d'autres.
- **Pas de préemption.** Les ressources allouées à un processus sont libérées uniquement par le processus.

# Interblocages

## Conditions nécessaires pour l'interblocage

### Rappel

### Définition d'un interblocage

### Conditions nécessaires pour l'interblocage

### Graphe d'allocation des ressources

### Réduction du graphe d'allocation des ressources

### Traitement des interblocages

### La détection et la reprise



Pour qu'une situation d'interblocage ait lieu, les quatre conditions suivantes doivent être remplies (Conditions de Coffman) :

- **L'exclusion mutuelle.** A un instant précis, une ressource est allouée à un seul processus.
- **La détention et l'attente.** Les processus qui détiennent des ressources peuvent en demander d'autres.
- **Pas de préemption.** Les ressources allouées à un processus sont libérées uniquement par le processus.
- **L'attente circulaire.** Il existe une chaîne de deux ou plus processus de telle manière que chaque processus dans la chaîne requiert une ressource allouée au processus suivant dans la chaîne.

# Interblocages

## Conditions nécessaires pour l'interblocage

Rappel

Définition d'un interblocage

Conditions nécessaires pour l'interblocage

Graphe d'allocation des ressources

Réduction du graphe d'allocation des ressources

Traitement des interblocages

La détection et la reprise



Par exemple, dans le problème de circulation, le trafic est impossible. On observe que les quatre conditions d'interblocage sont bien remplies :

- **Exclusion mutuelle** : Seulement une voiture occupe un endroit particulier de la route à un instant donné.
- **Détention et attente** : Aucun voiture ne peut faire marche arrière.
- **Pas de préemption** : On ne permet pas à une voiture de pousser une autre voiture en dehors de la route.
- **Attente circulaire** : Chaque coin de la rue contient des voitures dont le mouvement dépend des voitures qui bloquent la prochaine intersection.

# Interblocages

## Graphe d'allocation des ressources

### Rappel

### Définition d'un interblocage

### Conditions nécessaires pour l'interblocage

### Graphe d'allocation des ressources

### Réduction du graphe d'allocation des ressources

### Traitement des interblocages

### La détection et la reprise

M  
tr  
in  
Pr  
in



Le **graphe d'allocation des ressources** est un graphe biparti composé de deux types de nœuds et d'un ensemble d'arcs :

- **Les processus** qui sont représentés par des cercles.
- **Les ressources** qui sont représentées par des rectangles. Chaque rectangle contient autant de points qu'il y a d'exemplaires de la ressource représentée.
- Un arc orienté d'une ressource vers un processus signifie que la ressource est allouée au processus.
- Un arc orienté d'un processus vers une ressource signifie que le processus est bloqué en attente de la ressource.

Ce graphe indique pour chaque processus les ressources qu'il détient ainsi que celles qu'il demande.

# Interblocages

## Graphe d'allocation des ressources

Rappel

Définition d'un interblocage

Conditions nécessaires pour l'interblocage

Graphe d'allocation des ressources

Réduction du graphe d'allocation des ressources

Traitement des interblocages

La détection et la reprise



### Exemple 2.

Soient trois processus A, B et C qui utilisent trois ressources R, S et T comme illustré sur le tableau ci-dessous :

A	B	C
Demande R	Demande S	Demande T
Demande S	Demande T	Demande R
Libère R	Libère S	Libère T
Libère S	Libère T	Libère R

# Interblocages

## Graphe d'allocation des ressources

### Rappel

### Définition d'un interblocage

### Conditions nécessaires pour l'interblocage

### Graphe d'allocation des ressources

### Réduction du graphe d'allocation des ressources

### Traitement des interblocages

### La détection et la reprise



Si les processus sont exécutés de façon séquentielle : A suivi de B suivi C, il n'y pas d'interblocage. Supposons maintenant que l'exécution des processus est gérée par un ordonnanceur du type circulaire. Si les instructions sont exécutées dans l'ordre :

1. A détient R
2. B détient S
3. C détient T
4. A demande S
5. B demande T
6. C demande R



# Interblocages

## Graphe d'allocation des ressources

### Rappel

### Définition d'un interblocage

### Conditions nécessaires pour l'interblocage

### Graphe d'allocation des ressources

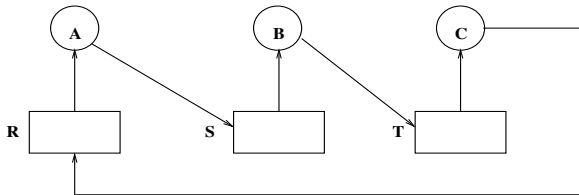
### Réduction du graphe d'allocation des ressources

### Traitement des interblocages

### La détection et la reprise



On atteint une situation d'interblocage, fait qui est montré sur la figure ci-après.



**FIGURE:** Situation d'interblocage de trois processus

# Interblocages

## Réduction du graphe d'allocation des ressources

### Rappel

### Définition d'un interblocage

### Conditions nécessaires pour l'interblocage

### Grappe d'allocation des ressources

### Réduction du graphe d'allocation des ressources

### Traitement des interblocages

### La détection et la reprise

M  
tr  
in  
Pr  
in



Un graphe réduit peut-être utilisé pour déterminer s'il existe ou non un interblocage. Pour la **réduction d'un graphe d'allocation des ressources**, les flèches associées à chaque processus et à chaque ressource doivent être vérifiées.

- Si une ressource possède seulement des flèches qui sortent (il n'y a pas des requêtes), on les efface.
- Si un processus possède seulement des flèches qui pointent vers lui, on les efface.
- Si une ressource a des flèches qui sortent, mais pour chaque flèche de requête il y a une ressource disponible dans le bloc de ressources où la flèche pointe, il faut les effacer.

# Interblocages

## Réduction du graphe d'allocation des ressources

Rappel

Définition d'un interblocage

Conditions nécessaires pour l'interblocage

Graphe d'allocation des ressources

Réduction du graphe d'allocation des ressources

Traitement des interblocages

La détection et la reprise



**Exemple 3.** Considérez quatre processus  $P_1, \dots, P_n$  qui utilisent des ressources du type  $R_1, R_2$  et  $R_3$ . Le tableau ci-dessous montre l'allocation courante et le nombre maximum d'unités de ressources nécessaires pour l'exécution des processus. Le nombre de ressources disponibles est  $A=[0,0,0]$ . Le graphe d'allocation des ressources pour l'état courant est montré sur la figure ci-après.

Processus	$R_1$	$R_2$	$R_3$	$R_1$	$R_2$	$R_3$
$P_1$	3	0	0	0	0	0
$P_2$	1	1	0	1	0	0
$P_3$	0	2	0	1	0	1
$P_4$	1	0	1	0	2	0

# Interblocages

## Réduction du graphe d'allocation des ressources

### Rappel

### Définition d'un interblocage

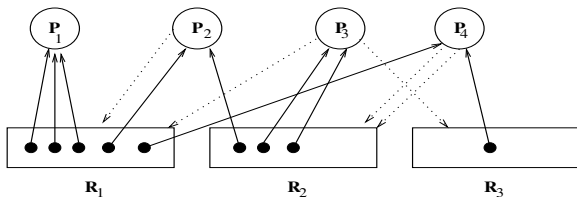
### Conditions nécessaires pour l'interblocage

### Graphe d'allocation des ressources

### Réduction du graphe d'allocation des ressources

### Traitement des interblocages

### La détection et la reprise



$P_1$  possède seulement des flèches qui pointent vers lui, alors on les efface. Les requêtes de  $R_1$  effectuées par  $P_2$  peuvent donc être allouées, et  $P_2$  aura seulement des flèches qui pointent vers lui. On les efface aussi. On efface la flèche de  $R_1$  vers  $P_3$  (qui a changé de direction) car  $R_1$  n'a que des flèches sortant. Le graphe réduit final est :

# Interblocages

## Réduction du graphe d'allocation des ressources

### Rappel

### Définition d'un interblocage

### Conditions nécessaires pour l'interblocage

### Graphe d'allocation des ressources

### Réduction du graphe d'allocation des ressources

### Traitement des interblocages

### La détection et la reprise

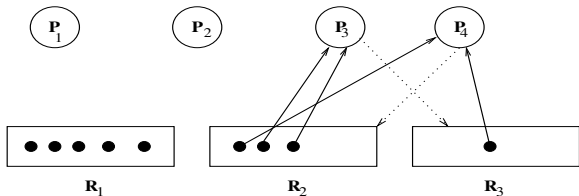


FIGURE: Graphe réduit d'allocation

# Interblocages

## Traitement des interblocages

### Rappel

Définition d'un interblocage

Conditions nécessaires pour l'interblocage

Graphes d'allocation des ressources

Réduction du graphe d'allocation des ressources

Traitement des interblocages

La détection et la reprise



Comme nous l'avons mentionné, les situations d'interblocage peuvent se produire dans un système. La question qui se pose est donc : doit-il prendre en compte ce problème ou l'ignorer ?

**Ignorer complètement les problèmes.** On peut faire l'autruche et ignorer les possibilités d'interblocages. Cette « stratégie » est celle de la plupart des systèmes d'exploitation courants car le prix à payer pour les éviter est élevé.

**Les détecter et y remédier.** On tente de traiter les interblocages, en détectant les processus interbloqués et en les éliminant.

# Interblocages

## Traitement des interblocages

### Rappel

### Définition d'un interblocage

### Conditions nécessaires pour l'interblocage

### Graphe d'allocation des ressources

### Réduction du graphe d'allocation des ressources

### Traitement des interblocages

### La détection et la reprise

M  
tr  
in  
Pr  
in



**Les éviter.** En allouant dynamiquement les ressources avec précaution. Le système d'exploitation peut suspendre le processus qui demande une allocation de ressource s'il constate que cette allocation peut conduire à un interblocage. Il lui attribuera la ressource lorsqu'il n'y aura plus de risque.

**Les prévenir.** En empêchant l'apparition de l'une des quatre conditions de leur existence.

# Interblocages

## Traitement des interblocages

### Rappel

### Définition d'un interblocage

### Conditions nécessaires pour l'interblocage

### Graphe d'allocation des ressources

### Réduction du graphe d'allocation des ressources

### Traitement des interblocages

### La détection et la reprise

M  
tr  
in  
Pr  
in



Le système d'exploitation peut suspendre le processus qui demande une allocation de ressource s'il constate que cette allocation peut conduire à un interblocage. Il lui attribuera la ressource lorsqu'il n'y aura plus de risque. **Les prévenir.** En empêchant l'apparition de l'une des quatre conditions de leur existence.



# Interblocages

## La détection et la reprise

### Rappel

### Définition d'un interblocage

### Conditions nécessaires pour l'interblocage

### Graphe d'allocation des ressources

### Réduction du graphe d'allocation des ressources

### Traitement des interblocages

### La détection et la reprise



Dans ce cas, le système ne cherche pas à empêcher les interblocages. Il tente de les détecter et d'y remédier. Pour détecter les interblocages, il construit dynamiquement le graphe d'allocation des ressources du système qui indique les attributions et les demandes de ressources. Dans le cas des ressources à exemplaire unique, il existe un interblocage si le graphe contient au moins un cycle. Dans le cas des ressources à exemplaires multiples, il existe un interblocage si le graphe contient au moins un cycle terminal (aucun arc ne permet de le quitter).

# Interblocages

## La détection et la reprise

### Rappel

#### Définition d'un interblocage

#### Conditions nécessaires pour l'interblocage

#### Grappe d'allocation des ressources

#### Réduction du graphe d'allocation des ressources

#### Traitement des interblocages

#### La détection et la reprise



Le système vérifie s'il y a des interblocages :

- A chaque modification du graphe suite à une demande d'une ressource (coûteuse en termes de temps processeur).
- Périodiquement ou lorsque l'utilisation du processeur est inférieure à un certain seuil (la détection peut être tardive).

# Interblocages

## Méthodes de traitement des interblocages

### Rappel

Définition d'un interblocage

Conditions nécessaires pour l'interblocage

Graphes d'allocation des ressources

Réduction du graphe d'allocation des ressources

Traitement des interblocages

La détection et la reprise

M  
tr  
in  
Pr  
in



Il existe essentiellement trois méthodes pour traiter le problème de l'interblocage : Prévention, Evitement, Détection-Guérison.

- **Prévention** : On élimine complètement le risque d'interblocage en faisant en sorte que l'une des ses quatre conditions d'apparition ne soit pas vérifiée.
- **Evitement** : On évite le risque d'interblocage en veillant à que le système évolue uniquement entre états sains.
- **Détection-Guérison** : On attend que l'interblocage arrive, on le détecte puis on applique une méthode de guérison.

# Interblocages

## Prévention des interblocages

### Rappel

### Définition d'un interblocage

### Conditions nécessaires pour l'interblocage

### Grappe d'allocation des ressources

### Réduction du graphe d'allocation des ressources

### Traitement des interblocages

### La détection et la reprise



- 1 Exclusion mutuelle** : Un processus ne doit jamais attendre une ressource partageable (exemple : il faut autoriser autant de processus que possible pour la lecture d'un fichier). Cependant, il n'est pas possible de prévenir les interblocages en niant la condition de l'exclusion mutuelle : certaines ressources sont non partageables (exemple : il n'est pas possible de partager un fichier entre plusieurs rédacteurs).

# Interblocages

## Prévention des interblocages

### Rappel

### Définition d'un interblocage

### Conditions nécessaires pour l'interblocage

### Graphe d'allocation des ressources

### Réduction du graphe d'allocation des ressources

### Traitement des interblocages

### La détection et la reprise

M  
tr  
in  
Pr  
in



- 2 Détection et attente :** Pour s'assurer que la condition d'occupation et d'attente ne se produit jamais dans le système, on doit garantir qu'à chaque fois qu'un processus qui requiert une ressource, il n'en détient aucune autre. Autrement dit, un processus ne doit demander des ressources supplémentaires qu'après avoir libéré les ressources qu'il occupe déjà.

# Interblocages

## Prévention des interblocages

### Rappel

### Définition d'un interblocage

### Conditions nécessaires pour l'interblocage

### Graphes d'allocation des ressources

### Réduction du graphe d'allocation des ressources

### Traitement des interblocages

### La détection et la reprise



## Exemple

Par exemple, imaginons un processus qui copie des données d'une unité de bandes vers un fichier du disque, trie le fichier puis imprime le résultat. Si l'on doit demander toutes les ressources au début du processus, celui-ci doit dès le départ requérir l'unité de bande, le fichier du disque et l'imprimante. Ainsi, il gardera l'imprimante pendant toute l'exécution, même s'il n'en a besoin qu'à la fin. L'autre façon de faire, serait d'affecter au processus uniquement l'unité de bande et le fichier sur disque, au début de son exécution. Il copie de l'unité de bande vers le disque et les libère ensuite tous les deux. Le processus doit à nouveau demander le fichier sur disque et l'imprimante. Après avoir copié le fichier sur l'imprimante, il libère ces deux ressources et se termine.

# Interblocages

## Prévention des interblocages

Rappel

Définition d'un interblocage

Conditions nécessaires pour l'interblocage

Graphe d'allocation des ressources

Réduction du graphe d'allocation des ressources

Traitement des interblocages

La détection et la reprise

M  
tr  
in  
Pr  
in



- 3 Pas de préemption :** Pour garantir que cette condition ne soit pas vérifiée, on peut utiliser le protocole suivant : Si un processus détenant certaines ressources en demande une autre qui ne peut pas lui être immédiatement allouée, toutes les ressources actuellement allouées à ce processus sont réquisitionnées. C'est à dire que ses ressources sont implicitement libérées.

# Interblocages

## Prévention des interblocages

### Rappel

### Définition d'un interblocage

### Conditions nécessaires pour l'interblocage

### Grphe d'allocation des ressources

### Réduction du graphe d'allocation des ressources

### Traitement des interblocages

### La détection et la reprise



Les ressources réquisitionnées sont ajoutées à la liste des ressources pour lesquelles le processus attend. Le processus démarrera seulement quand il pourra regagner ses anciennes ressources, ainsi que les nouvelles qu'il requiert.

Ce protocole présente malheureusement au moins deux inconvénients :

- Lenteur dans l'utilisation des ressources puisqu'on peut allouer plusieurs ressources et ne pas les utiliser pendant longtemps.
- La famine est possible puisqu'un processus peut être retardé indéfiniment parce que l'une des ressources qu'il demande est occupée.



# Interblocages

## Prévention des interblocages

### Rappel

Définition d'un interblocage

Conditions nécessaires pour l'interblocage

Graphe d'allocation des ressources

Réduction du graphe d'allocation des ressources

Traitement des interblocages

La détection et la reprise

M  
tr  
in  
Pr  
in



4 **Attente circulaire** : On peut garantir que la condition de l'attente circulaire ne se vérifie jamais, en imposant un ordre total sur les types de ressources et on forçant chaque processus à demander les ressources dans un ordre croissant d'énumération. Par exemple, on numérote les types de ressources du système, comme suit :

- $O(\text{Unité de bandes})=1$
- $O(\text{Unité de disque})=2$
- $O(\text{Imprimante})=3$
- $O(\text{Lecteur CD-ROM})=4$
- $O(\text{Lecteur Disquette})=5$
- etc.

# Interblocages

## Prévention des interblocages

### Rappel

Définition d'un interblocage

Conditions nécessaires pour l'interblocage

Graphe d'allocation des ressources

Réduction du graphe d'allocation des ressources

Traitement des interblocages

La détection et la reprise



Pour prévenir l'interblocage, on peut imaginer le protocole suivant : Un processus ne peut acquérir une ressource d'ordre  $i$  que s'il a déjà obtenu toutes les ressources nécessaires d'ordre  $j$  ( $j < i$ ). Par exemple, un processus désirant utiliser l'unité de bandes et l'imprimante, doit d'abord demander l'unité de bande ensuite l'imprimante. On peut démontrer que de cette façon, il n'y a aucun risque d'interblocage.

Cependant cet algorithme est inefficace, il peut conduire à une monopolisation inutile d'une ressource (un processus peut détenir une ressource alors qu'il n'en a pas besoin).

# Interblocages

## Evitement des interblocages

### Rappel

### Définition d'un interblocage

### Conditions nécessaires pour l'interblocage

### Graphe d'allocation des ressources

### Réduction du graphe d'allocation des ressources

### Traitement des interblocages

### La détection et la reprise



## Principe :

Si on dispose d'un système d'allocation de ressources avec une seule instance pour chaque type de ressource, on peut utiliser une variante du graphe d'allocation de ressources pour la prévention des interblocages.

En plus des arcs de requêtes et d'affectation, on introduit un nouveau type d'arc appelé arc de réclamation. Un arc de réclamation  $P_i \rightarrow R_j$  indique que le processus  $P_i$  peut demander la ressource  $R_j$  dans le futur (il est représenté en pointillé).

Quand le processus  $P_i$  demande réellement la ressource  $R_j$ , l'arc de réclamation  $P_i \rightarrow R_j$  est transformé en arc de requête. De même, lorsqu'une ressource est libérée par  $P_i$ , l'arc d'affectation  $R_j \rightarrow P_i$  est reconverti en arc de réclamation  $P_i \rightarrow R_j$ .

# Interblocages

## Evitement des interblocages

### Rappel

#### Définition d'un interblocage

#### Conditions nécessaires pour l'interblocage

#### Graphe d'allocation des ressources

#### Réduction du graphe d'allocation des ressources

#### Traitement des interblocages

#### La détection et la reprise

M  
tr  
in  
Pr  
in



### Principe :

On peut éviter que l'interblocage arrive en empêchant  $P_i$  qui demande la ressource  $R_j$ , n'obtienne pas cette ressource si la transformation de l'arc de requête  $P_i \rightarrow R_j$  en arc d'affectation provoque un circuit dans le graphe.

S'il n'existe pas de circuit, l'allocation de la ressource laissera le système dans un état **sain**. Si l'on trouve un circuit, l'allocation de la ressource laissera le système dans un état **malsain**. Le processus  $P_i$  devra donc attendre pour que sa requête soit satisfaite.

# Interblocages

## Evitement des interblocages

### Rappel

### Définition d'un interblocage

### Conditions nécessaires pour l'interblocage

### Graphes d'allocation des ressources

### Réduction du graphe d'allocation des ressources

### Traitement des interblocages

### La détection et la reprise



## Etat sain :

On vient de voir que le principe des méthodes d'évitement de l'interblocage est de garder le système toujours dans un état sain. Nous donnons maintenant une définition formelle de ce qu'est un état sain :

## Définition d'un état sain :

Un système est dans un état sain s'il existe une séquence saine. Une séquence de processus  $\langle P_1, P_2, \dots, P_N \rangle$  est une séquence saine pour l'état d'allocation courant si, pour chaque  $P_i$ , les requêtes de ressources de  $P_i$  peuvent être satisfaites par les ressources couramment disponibles, plus les ressources détenues par tous les  $P_j$ , avec  $j < i$ .

# Interblocages

## Evitement des interblocages

### Rappel

#### Définition d'un interblocage

#### Conditions nécessaires pour l'interblocage

#### Graphes d'allocation des ressources

#### Réduction du graphe d'allocation des ressources

#### Traitement des interblocages

#### La détection et la reprise

M  
tr  
in  
Pr  
in



### Etat sain :

#### Définition d'un état sain :

En effet, dans cette situation, si les ressources demandées par  $P_i$  ne sont pas immédiatement disponibles,  $P_i$  peut attendre jusqu'à la terminaison de tous les  $P_j$ . Une fois qu'ils ont fini,  $P_i$  peut obtenir les ressources nécessaires, achever sa tâche et rendre les ressources allouées. Quand  $P_i$  termine,  $P_{i+1}$  peut obtenir les ressources manquantes, et ainsi de suite. Si une telle séquence n'existe pas, on dit que le système est dans état malsain, c'est à dire qu'il y a un risque d'apparition d'interblocage.

# Interblocages

## Evitement des interblocages

Rappel

Définition d'un interblocage

Conditions nécessaires pour l'interblocage

Graphe d'allocation des ressources

Réduction du graphe d'allocation des ressources

Traitement des interblocages

La détection et la reprise



### Etat sain :

**Exemple** : un système possède 12 unités de bandes magnétiques et 3 processus P0, P1 et P2. On suppose que les besoins des trois processus en ressources sont : 10 pour P0, 4 pour P1, 9 pour P2. D'autre part, on suppose qu'à l'instant  $t_0$  l'état d'allocation des ressources par les processus est le suivant : 5 pour P0, 2 pour P1 et 2 pour P2. Le nombre d'unités de bandes libres est donc égal à 3.

Processus	Besoins maximaux	Besoins actuellement satisfaits	Besoins restant à satisfaire
P0	10	5	5
P1	4	2	2
P2	9	2	7

Nombre de ressources (unités de bandes) disponibles :  
03.

# Interblocages

## Evitement des interblocages

Rappel

Définition d'un interblocage

Conditions nécessaires pour l'interblocage

Graphes d'allocation des ressources

Réduction du graphes d'allocation des ressources

Traitement des interblocages

La détection et la reprise



**Etat sain :**

**Exemple :**

A l'instant  $t_0$ , le système est dans un état sain. En effet, la séquence  $\langle P_1, P_0, P_2 \rangle$  est saine, puisque l'on peut allouer immédiatement au processus  $P_1$  toutes ses ressources manquantes et les rendre ensuite. Dans ce cas le système disposerait de 5 unités de bandes, le processus  $P_0$  peut donc obtenir toutes ses ressources manquantes et les rendre ensuite. Le système disposerait alors de 10 unités de bandes et enfin le processus  $P_2$  peut obtenir ses unités de bandes et les rendre, le système disposerait donc de 12 unités de bandes.

Il est possible de passer d'un état sain à un état malsain. Supposons qu'à l'instant  $t_1$  le processus  $P_2$  demande et qu'on lui accorde 1 unité de bandes de plus. L'état du système serait alors le suivant :



# Interblocages

## Evitement des interblocages

Rappel

Définition d'un interblocage

Conditions nécessaires pour l'interblocage

Grappe d'allocation des ressources

Réduction du graphe d'allocation des ressources

Traitement des interblocages

La détection et la reprise

M  
tr  
in  
Pr  
in



**Etat sain :**

**Exemple :**

Processus	Besoins maximaux	Besoins actuellement satisfaits	Besoins restant à satisfaire
P0	10	5	5
P1	4	2	2
P2	9	3	6

Nombre de ressources (unités de bandes) disponibles :  
02.

# Interblocages

## Evitement des interblocages

### Rappel

Définition d'un interblocage

Conditions nécessaires pour l'interblocage

Graphe d'allocation des ressources

Réduction du graphe d'allocation des ressources

Traitement des interblocages

La détection et la reprise

Mi  
tr:  
in  
Pr  
in



### Etat sain :

**Exemple :** Le système n'est plus dans un état sain. En effet, on ne peut allouer toutes ses unités de bandes qu'au processus P1. Quand il les rendra, le système ne disposera que de 04 unités de bandes. Comme le processus P0 peut demander 5 unités et le processus P2 7 unités, le système n'aura pas suffisamment de ressources pour les satisfaire. Ou pourra alors avoir un cas d'interblocage, puisque P0 et P2 seront retardés indéfiniment. Pour éviter cette situation, il ne fallait pas accorder à P1 la ressource qu'il a demandée ; il fallait le faire attendre.

# Interblocages

## Evitement des interblocages

### Rappel

### Définition d'un interblocage

### Conditions nécessaires pour l'interblocage

### Graphe d'allocation des ressources

### Réduction du graphe d'allocation des ressources

### Traitement des interblocages

### La détection et la reprise

M  
tr  
in  
Pr  
in



## Algorithme du Banquier :

L'algorithme du banquier (Habermann et Dijkstra) est un algorithme d'évitement des interblocages qui s'applique dans le cas général où chaque type de ressources possède plusieurs instances. Le nom de l'algorithme a été choisi parce que cet algorithme pourrait s'appliquer dans un système bancaire pour s'assurer que la banque ne prête jamais son argent disponible de telle sorte qu'elle ne puisse plus satisfaire tous ses clients.

### Rappel

Définition d'un interblocage

Conditions nécessaires pour l'interblocage

Graphe d'allocation des ressources

Réduction du graphe d'allocation des ressources

Traitement des interblocages

La détection et la reprise

M  
tr  
in  
Pr  
in



### Algorithme du Banquier :

Quand un nouveau processus entre dans le système, il doit déclarer le nombre maximal d'instances de chaque type de ressources dont il aura besoin. Ce nombre ne doit pas excéder le nombre total de ressources du système. Au cours de son exécution, quand un processus demande un ensemble de ressources, l'algorithme vérifie si cela gardera toujours le système dans un état sain. Dans l'affirmative la demande est accordée, dans la négative la demande est retardée.

# Interblocages

## Evitement des interblocages

### Rappel

#### Définition d'un interblocage

#### Conditions nécessaires pour l'interblocage

#### Graphes d'allocation des ressources

#### Réduction du graphe d'allocation des ressources

#### Traitement des interblocages

#### La détection et la reprise



### Algorithme du Banquier :

Soient  $m$  le nombre de types de ressources du système, et  $n$  le nombre de processus. Pour fonctionner, l'algorithme maintient plusieurs structures de données :

- **Available** : C'est un Vecteur de longueur  $m$  indiquant le nombre de ressources disponibles de chaque type. Ainsi, si  $Available[j]=k$ , cela veut dire que le type de ressources  $R_j$  possède  $k$  instances disponibles.
- **Max** : C'est une matrice  $n \times m$  définissant la demande maximale de chaque processus. Ainsi, Si  $Max[i, j]=k$ , cela veut dire que le processus  $P_i$  peut demander au plus  $k$  instances du type de ressources  $R_j$ .

# Interblocages

## Evitement des interblocages

### Algorithme du Banquier :

- **Allocation** : C'est une matrice  $n \times m$  définissant le nombre de ressources de chaque type de ressources actuellement alloué à chaque processus. Ainsi si  $Allocation[i, j]=k$ , cela veut dire que l'on a alloué au processus  $P_i$   $k$  instances du type de ressources  $R_j$ .
- **Need** : C'est une matrice  $n \times m$  indiquant les ressources restant à satisfaire à chaque processus. Ainsi, si  $Need[i, j]=k$ , cela veut dire que le processus  $P_i$  peut avoir besoin de  $k$  instances au plus du type de ressources  $R_j$  pour achever sa tâche.
- **Request** : C'est une matrice  $n \times m$  indiquant les ressources supplémentaires que les processus viennent de demander. Ainsi, si  $Request[i, j]=k$ , cela veut dire que le processus  $P_i$  vient de demander  $k$  instances supplémentaires du type de ressources  $R_j$ .

Rappel

Définition d'un interblocage

Conditions nécessaires pour l'interblocage

Graphes d'allocation des ressources

Réduction du graphes d'allocation des ressources

Traitement des interblocages

La détection et la reprise



Rappel

Définition d'un interblocage

Conditions nécessaires pour l'interblocage

Graphe d'allocation des ressources

Réduction du graphe d'allocation des ressources

Traitement des interblocages

La détection et la reprise



### Algorithme du Banquier :

De ce qui précède, on peut remarquer que :

- 1 Ces structures de données peuvent varier dans le temps, en taille et en valeur.
- 2 La matrice Need peut être calculée à partir des matrices Max et Allocation :

$$Need[i, j] = Max[i, j] - Allocation[i, j]$$

# Interblocages

## Evitement des interblocages

### Rappel

Définition d'un interblocage

Conditions nécessaires pour l'interblocage

Grappe d'allocation des ressources

Réduction du graphe d'allocation des ressources

Traitement des interblocages

La détection et la reprise



## Algorithme du Banquier :

**Notations :** Pour des raisons pratiques, on utilise les notations suivantes :

- $Allocation_i$  : Vecteur désignant les ressources actuellement allouées au processus  $P_i$ .
- $Need_i$  : Vecteur désignant les ressources supplémentaires que le processus  $P_i$  peut encore demander.
- $Request_i$  : Vecteur désignant les ressources supplémentaires que le processus  $P_i$  vient de demander.
- $X \leq Y$  :  $X$  et  $Y$  sont des vecteurs si  $X[i] \leq Y[i]$  pour chaque  $i$  allant de 1 à  $n$ .



# Interblocages

## Evitement des interblocages

### Rappel

Définition d'un interblocage

Conditions nécessaires pour l'interblocage

Graphe d'allocation des ressources

Réduction du graphe d'allocation des ressources

Traitement des interblocages

La détection et la reprise



### Algorithme du Banquier :

L'algorithme du Banquier peut être scindé en deux parties complémentaires : Un algorithme de requête de ressources et un algorithme de vérification si l'état du système est sain.

Nous donnons maintenant le code de chacune des deux parties.

# Interblocages

## Evitement des interblocages

### Algorithme du Banquier :

**Algorithme de requête de ressources** Cet algorithme est appelé à chaque fois qu'un processus fait une demande de ressources

Etape 1 : Si  $Request_i \leq Need_i$   
Alors Aller à l'étape 2  
Sinon erreur : le processus a excédé ses besoins maximaux

Finsi

Etape 2 : Si  $Request_i \leq Available$   
Alors Aller à l'étape 3  
Sinon Attendre : les ressources ne sont pas disponibles.

Finsi

Etape 3 : Sauvegarder l'état du système (les matrices Available, Allocation et Need).  
Allouer les ressources demandées par le processus  $P_i$  en modifiant l'état du système de la manière suivante :

$Available := Available - Request_i$   
 $Allocation_i := Allocation_i + Request_i$   
 $Need_i := Need_i - Request_i$

Si  $Verification\_Etat\_Sain = Vrai$   
Alors L'allocation est validée  
Sinon L'allocation est annulée ; Restaurer l'ancien Etat du système

Finsi

Rappel

Définition d'un interblocage

Conditions nécessaires pour l'interblocage

Graphe d'allocation des ressources

Réduction du graphe d'allocation des ressources

Traitement des interblocages

La détection et la reprise



# Interblocages

## Evitement des interblocages

Rappel

Définition d'un interblocage

Conditions nécessaires pour l'interblocage

Graphes d'allocation des ressources

Réduction du graphe d'allocation des ressources

Traitement des interblocages

La détection et la reprise



## Algorithme du Banquier :

L'algorithme suivant est une fonction qui renvoie la valeur Vrai si le système est dans un état sain, Faux sinon.

Algorithme Verification\_Etat\_Sain

Début

Work : Tableau[m] de Entier ;  
Finish : Tableau[n] de Logique ;

Etape 1 : Work := Available  
Finish := Faux ;

Etape 2 : Trouver i tel que : Finish[i]=faux et Need<sub>i</sub> ≤ Work  
Si un tel i n'existe pas aller à l'étape 4.

Etape 3 : Work := Work + Allocation<sub>i</sub>  
Finish[i] := Vrai  
Aller à l'étape 2

Etape 4 : Si Finish = Vrai (pour tout i)  
Alors Verification\_Etat\_Sain := Vrai  
Sinon Verification\_Etat\_Sain := Faux

Finsi

Fin.

# Interblocages

## Evitement des interblocages

### Rappel

### Définition d'un interblocage

### Conditions nécessaires pour l'interblocage

### Grappe d'allocation des ressources

### Réduction du graphe d'allocation des ressources

### Traitement des interblocages

### La détection et la reprise

Mi  
tr:  
in  
Pr  
in



## Algorithme du Banquier :

**Remarque** : Cet algorithme peut demander  $m \times n^2$  opérations pour décider si un état est sain.

**Exemple** : Un système possède 5 processus (P0, P1, P2, P3, P4) et 3 types de ressources (A, B, C). Le type de ressources A possède 10 instances, le type de ressources B possède 5 instances et le type de ressources C possède 7 instances. A l'instant T0, l'état des ressources du système est décrit par les matrices Allocation, Max et Available suivantes :

# Interblocages

## Evitement des interblocages

Rappel

Définition d'un interblocage

Conditions nécessaires pour l'interblocage

Graphe d'allocation des ressources

Réduction du graphe d'allocation des ressources

Traitement des interblocages

La détection et la reprise



## Algorithme du Banquier : Exemple :

Matrice : Allocation

	A	B	C
P0	0	1	0
P1	2	0	0
P2	3	0	2
P3	2	1	1
P4	0	0	2

Matrice : Max

	A	B	C
P0	7	5	3
P1	3	2	2
P2	9	0	2
P3	2	2	2
P4	4	3	3

Matrice : Available

A	B	C
3	3	2

# Interblocages

## Evitement des interblocages

Rappel

Définition d'un interblocage

Conditions nécessaires pour l'interblocage

Graphe d'allocation des ressources

Réduction du graphe d'allocation des ressources

Traitement des interblocages

La détection et la reprise



## Algorithme du Banquier :

### Exemple :

Le contenu de la matrice Need peut être déduit par calcul,  $Need = Max - Allocation$

*Matrice : Need*

	A	B	C
P0	7	4	3
P1	1	2	2
P2	6	0	0
P3	0	1	1
P4	4	3	1

On peut vérifier que le système est dans un état sain. En effet, la séquence  $\langle P1, P3, P4, P2, P0 \rangle$  est saine.

# Interblocages

## Evitement des interblocages

### Rappel

### Définition d'un interblocage

### Conditions nécessaires pour l'interblocage

### Graphes d'allocation des ressources

### Réduction du graphe d'allocation des ressources

### Traitement des interblocages

### La détection et la reprise

Mi  
tr:  
in  
Pr  
in'



## Algorithme du Banquier :

### Exemple :

Supposons qu'à l'instant T1 le processus P1 demande une instance supplémentaire du type de ressources A et deux instances du type de ressources C. Nous avons alors :  $Request_1 = (1, 0, 2)$ .

Afin de décider si la requête peut être immédiatement accordée, on doit d'abord vérifier que

$Request_1 \leq Available$ , ce qui est vrai. On enregistre l'état du système (le contenu des matrices), puis on supposera que la requête a été satisfaite et on arrive à l'état suivant :

# Interblocages

## Evitement des interblocages

Rappel

Définition d'un interblocage

Conditions nécessaires pour l'interblocage

Graphe d'allocation des ressources

Réduction du graphe d'allocation des ressources

Traitement des interblocages

La détection et la reprise



## Algorithme du Banquier : Exemple :

Matrice : Allocation

	A	B	C
P0	0	1	0
P1	3	0	2
P2	3	0	2
P3	2	1	1
P4	0	0	2

Matrice : Need

	A	B	C
P0	7	4	3
P1	0	2	0
P2	6	0	0
P3	0	1	1
P4	4	3	1

Matrice : Available

A	B	C
2	3	0



# Interblocages

## Evitement des interblocages

### Rappel

Définition d'un interblocage

Conditions nécessaires pour l'interblocage

Graphes d'allocation des ressources

Réduction du graphe d'allocation des ressources

Traitement des interblocages

La détection et la reprise

Mi  
tr:  
in  
Pr  
in



## Algorithme du Banquier :

### Exemple :

On doit alors déterminer si le nouvel état est sain en appliquant l'algorithme de vérification de l'état sain, ce qui est vrai (la séquence  $\langle P1, P3, P4, P2, P0 \rangle$  est saine). On peut donc accorder la demande de ressource faite par P1.

A l'instant T2, une requête (3, 3, 0) arrive du processus P4. Cette requête est immédiatement rejetée parce que les ressources ne sont pas disponibles.

A l'instant T3, une requête (0, 2, 0) arrive du processus P0. Cette requête ne sera pas accordée parce que l'état obtenu est malsain.

### Rappel

Définition d'un interblocage

Conditions nécessaires pour l'interblocage

Graphe d'allocation des ressources

Réduction du graphe d'allocation des ressources

Traitement des interblocages

La détection et la reprise



## Critique de l'algorithme du Banquier :

Bien qu'il a l'avantage d'éviter les interblocages, l'algorithme du banquier a néanmoins quelques inconvénients :

- Coûteux : L'algorithme est en effet très coûteux en temps d'exécution et en mémoire pour le système. Puisqu'il faut maintenir plusieurs matrices, et déclencher à chaque demande de 2 ressources, l'algorithme de vérification de l'état sain qui demande  $m \times n^2$  opérations. (m est le nombre de types de ressources et n est le nombre de processus).

### Rappel

Définition d'un interblocage

Conditions nécessaires pour l'interblocage

Graphes d'allocation des ressources

Réduction du graphe d'allocation des ressources

Traitement des interblocages

La détection et la reprise



## Critique de l'algorithme du Banquier :

Bien qu'il a l'avantage d'éviter les interblocages, l'algorithme du banquier a néanmoins quelques inconvénients :

- Théorique : L'algorithme exige que chaque processus déclare à l'avance les ressources qu'il doit utiliser, en type et en nombre. Cette contrainte est difficile à réaliser dans la pratique.
- Pessimiste : L'algorithme peut retarder une demande de ressources dès qu'il y a risque d'interblocage (mais en réalité l'interblocage peut ne pas se produire).

# Interblocages

## Détection et guérison des interblocages

### Rappel

### Définition d'un interblocage

### Conditions nécessaires pour l'interblocage

### Graphe d'allocation des ressources

### Réduction du graphe d'allocation des ressources

### Traitement des interblocages

### La détection et la reprise

M  
tr  
in  
Pr  
in



Si un système n'emploie pas d'algorithme pour prévenir les interblocages, il peut se produire une situation d'interblocage. Dans ce cas le système doit fournir :

- Un algorithme qui examine l'état du système pour déterminer s'il s'est produit un interblocage.
- Un algorithme pour guérir (corriger) l'interblocage.

# Interblocages

## Détection de l'interblocage

### Rappel

### Définition d'un interblocage

### Conditions nécessaires pour l'interblocage

### Grappe d'allocation des ressources

### Réduction du graphe d'allocation des ressources

### Traitement des interblocages

### La détection et la reprise



Pour détecter un interblocage dans un système disposant de plus d'une instance pour chaque type de ressources, on peut utiliser l'algorithme de détection suivant, qui s'inspire de l'algorithme du banquier. . Cet algorithme est lancé périodiquement. Il utilise les structures de données suivantes :

- Request : C'est une matrice  $n \times m$  indiquant les ressources supplémentaires que les processus viennent de demander. Ainsi, si  $Request[i, j]=k$ , cela veut dire que le processus  $P_i$  vient de demander  $k$  instances supplémentaires du type de ressources  $R_j$ .

# Interblocages

## Détection de l'interblocage

### Rappel

### Définition d'un interblocage

### Conditions nécessaires pour l'interblocage

### Graphe d'allocation des ressources

### Réduction du graphe d'allocation des ressources

### Traitement des interblocages

### La détection et la reprise

M<sub>i</sub>  
tr:  
in  
Pr  
in



- Request : C'est une matrice  $n \times m$  indiquant les ressources supplémentaires que les processus viennent de demander. Ainsi, si  $Request[i, j]=k$ , cela veut dire que le processus  $P_i$  vient de demander  $k$  instances supplémentaires du type de ressources  $R_j$ .

# Interblocages

## Détection de l'interblocage

### Rappel

### Définition d'un interblocage

### Conditions nécessaires pour l'interblocage

### Graphe d'allocation des ressources

### Réduction du graphe d'allocation des ressources

### Traitement des interblocages

### La détection et la reprise

M:  
tr:  
in:  
Pr:  
in:



### Algorithme Détection d'interblocage

Début

Work : Tableau[m] de Entier ;  
Finish : Tableau[n] de Logique ;

Etape 1 : Work :=Available  
Pour i=1 jusqu'à N  
Faire  
    Si  $Allocation_i < \infty$   
        Alors Finish[i] :=Faux  
        Sinon Finish[i]:=Vrai  
    Finsi  
Fait ;

Etape 2 : Trouver un indice i tel que : Finish[i]=Faux et  $Request_i \leq Work$   
Si un tel i n'existe pas aller à l'étape 4.

Etape 3 : Work :=Work + Allocation<sub>i</sub>  
Finish[i] :=Vrai  
Aller à l'étape 2

Etape 4 : Si Finish[i]=Faux (pour un certain i)  
    Alors Le système est dans un état d'interblocage  
    Sinon Le système n'est pas dans un état d'interblocage

Finsi

Fin.

# Interblocages

## Détection de l'interblocage

### Rappel

### Définition d'un interblocage

### Conditions nécessaires pour l'interblocage

### Graphes d'allocation des ressources

### Réduction du graphe d'allocation des ressources

### Traitement des interblocages

### La détection et la reprise



**Exemple :** L'état d'allocation des ressources d'un système est donné par le contenu des trois matrices suivantes : Available, Allocation et Request.

Matrice : Allocation

	A	B	C	D
P0	1	0	1	0
P1	2	0	0	1
P2	0	1	2	0

Matrice : Request

	A	B	C	D
P0	2	0	0	1
P1	1	0	1	0
P2	2	1	0	0

Matrice : Available

A	B	C	D
5	2	3	1



# Interblocages

## Détection de l'interblocage

### Rappel

### Définition d'un interblocage

### Conditions nécessaires pour l'interblocage

### Graphe d'allocation des ressources

### Réduction du graphe d'allocation des ressources

### Traitement des interblocages

### La détection et la reprise



## Exemple :

En appliquant l'algorithme de détection, on trouvera que les contenus successifs des matrices Work et Finish sont :

Itération 1 :

Work			
5	2	3	1

Finish		
Faux	Faux	Faux

Indice choisi  $i=0$

Itération 2 :

Work			
4	2	2	1

Finish		
Vrai	Faux	Faux

Indice choisi  $i=1$

Itération 3 :

Work			
2	2	2	0

Finish		
Vrai	Vrai	Faux

Indice choisi  $i=2$

Itération 4 :

Work			
2	1	0	0

Finish		
Vrai	Vrai	Vrai

Indice choisi  $i=3$

# Interblocages

## Détection de l'interblocage

Rappel

Définition d'un interblocage

Conditions nécessaires pour l'interblocage

Graphes d'allocation des ressources

Réduction du graphe d'allocation des ressources

Traitement des interblocages

La détection et la reprise



### Exemple :

Le vecteur Finish a la valeur Vrai, donc le système n'est pas en situation d'interblocage.

Supposons maintenant que le processus P2 fait une demande supplémentaire d'une instance de la ressource D. Les données du problème deviennent donc :

Matrice : Allocation

	A	B	C	D
P0	1	0	1	0
P1	2	0	0	1
P2	0	1	2	0

Matrice : Request

	A	B	C	D
P0	2	0	0	1
P1	1	0	1	0
P2	2	1	0	1

Matrice : Available

A	B	C	D
5	2	3	1

Une exécution de l'algorithme de détection fera apparaître les contenus suivants :

# Interblocages

## Détection de l'interblocage

### Exemple :

Rappel

Définition d'un interblocage

Conditions nécessaires pour l'interblocage

Graphe d'allocation des ressources

Réduction du graphe d'allocation des ressources

Traitement des interblocages

La détection et la reprise

M  
tr  
in  
Pr  
in



Itération 1 :

Work			
5	2	3	1

Finish		
Faux	Faux	Faux

Indice choisi  $i=0$

Itération 2 :

Work			
4	2	2	1

Finish		
Vrai	Faux	Faux

Indice choisi  $i=1$

Itération 3 :

Work			
2	2	2	0

Finish		
Vrai	Vrai	Faux

Indice choisi  $i=2$

Le déroulement de l'algorithme s'arrête avec le vecteur Finish contenant une valeur fausse, le système se trouve donc en situation d'interblocage.

# Interblocages

## Détection de l'interblocage

### Rappel

### Définition d'un interblocage

### Conditions nécessaires pour l'interblocage

### Graphes d'allocation des ressources

### Réduction du graphes d'allocation des ressources

### Traitement des interblocages

### La détection et la reprise



## **Critique de l'algorithme de détection :**

L'algorithme de détection des interblocages est très coûteux s'il est exécuté après chaque demande de ressources. En effet, il sera aussi coûteux que l'algorithme d'évitement du Banquier. L'idée donc c'est de le lancer périodiquement, mais comment choisir cette période ?.

# Interblocages

## Guérison de l'interblocage

### Rappel

### Définition d'un interblocage

### Conditions nécessaires pour l'interblocage

### Graphes d'allocation des ressources

### Réduction du graphe d'allocation des ressources

### Traitement des interblocages

### La détection et la reprise



Il existe plusieurs solutions pour corriger un interblocage si le système détecte qu'il en existe un.

- Correction manuelle : Le système alerte l'opérateur qu'il s'est produit un interblocage, et l'invite à le traiter manuellement (en relançant le système par exemple).
- Terminaison de processus : On peut éliminer un interblocage en arrêtant un ou plusieurs processus. On peut choisir d'arrêter tous les processus, ou bien de les arrêter un à un jusqu'à éliminer l'interblocage.
- Réquisition de ressources : Pour éliminer l'interblocage, en procédant à la réquisition d'une ou plusieurs ressources, en les enlevant à un processus et en les donnant à un autre jusqu'à ce que l'interblocage soit éliminé.