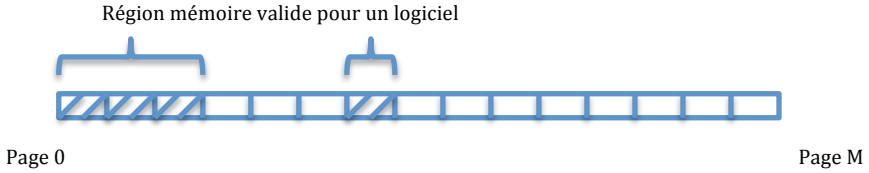
Prise de note Système d'Exploitation
Gestion mémoire Processeur

Quelles sont les différences de mémoire pour le noyau et pour les processus ?

- Noyau : à chaque programme pour le noyau il n'y a pas de bug, car les programmeurs ont la main sur tout le système
  - ➔ Pas de mécanisme pour que le système se protège lui-même
  - ➔ Gestion mémoire plus simple
- Processus : Le système doit gérer la priorité entre les processus. Lorsqu'un processus demande de la mémoire, le système ne lui donne pas de suite (cf allocation différée)

La taille d'une adresse mémoire va de 0 à 2^n



Parfois il n'y a qu'une partie d'une page allouée. Du coup si on va dans l'autre partie, on a un segmentationFault
Les régions mémoires sont des intervalles distincts d'adresse.

Au fur et à mesure que le processus s'exécute il a besoin de nouvelle région mémoire ou de les liberer. Le système doit mettre à jour les régions mémoires.
Si 2 régions mémoires sont contigües et ont les mêmes droits d'accès, le système les fusionnent. Si elles sont contigües mais sans les mêmes droits, le système en crée 2 nouvelles en remplacement.

Si il y a suppression à la fin d'une région, le système raccourcit la région. Sinon le système supprime entièrement et recrée une région.
Pour structurer les régions mémoires, on utilise un arbre binaire de recherche (AVL)
On dit que si le nombre de région mémoire est inferieur à 32 on utilise une liste, sinon un AVL.

Problème rencontré :
1. On accède à une région qui lui appartient mais qui a été supprimé (swappé)
2. On accède à une région qui lui appartient mais le système ne l'a pas alloué (allocation différée) et le fait
3. On accède à une région qui ne lui appartient pas (bug)

Allocation différée :
L'intérêt : A chaque demande de mémoire il accepte la demande, instruit le processus de l'accord, mais ne l'alloue vraiment que lorsque le processus y accède. Cela peut générer des pages fault.

Les allocations contigües sont plus pratiques car le cache charge automatiquement les adresses en bloc continues. Mais il est possible de la faire en non-contigüe.

Le noyau doit gérer la RAM pour lui-même et pour les processus.
Il utilise le buddy pour les gros besoins (besoin de frame) ou le slab, l'allocation dynamique destiné au noyau. Donc pour quelque octet on utilise le slab, mais le slab utilise le buddy, pour avoir des cadres physiques, n'ayant aucune mémoire.