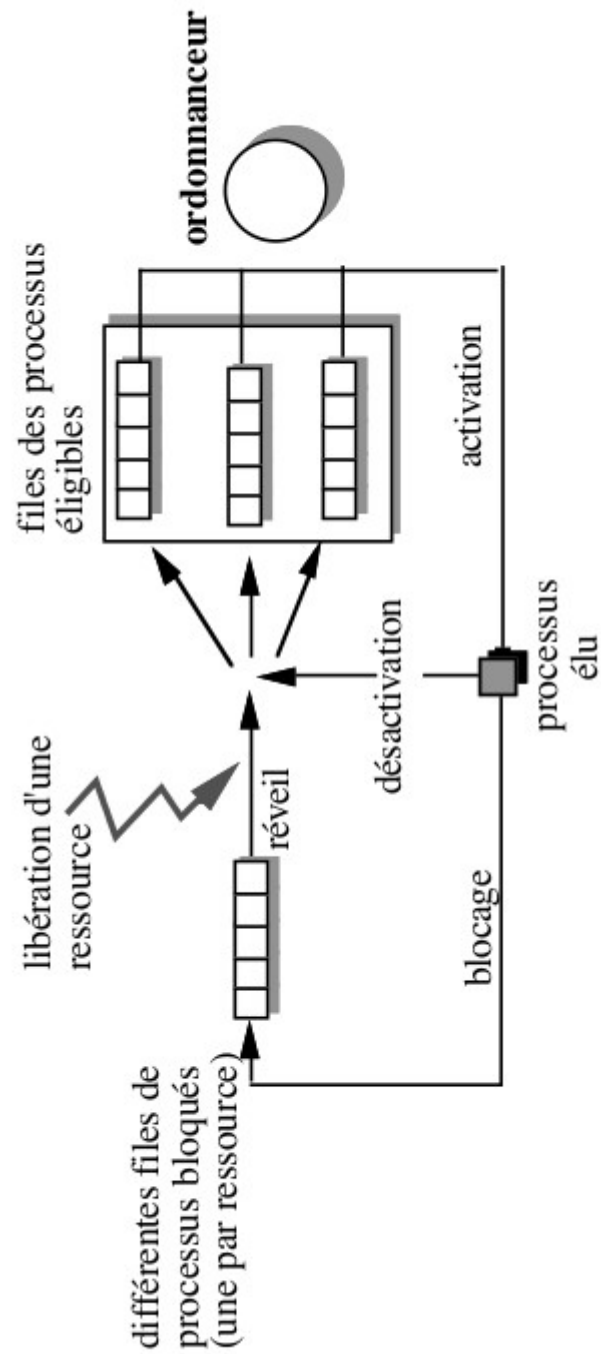


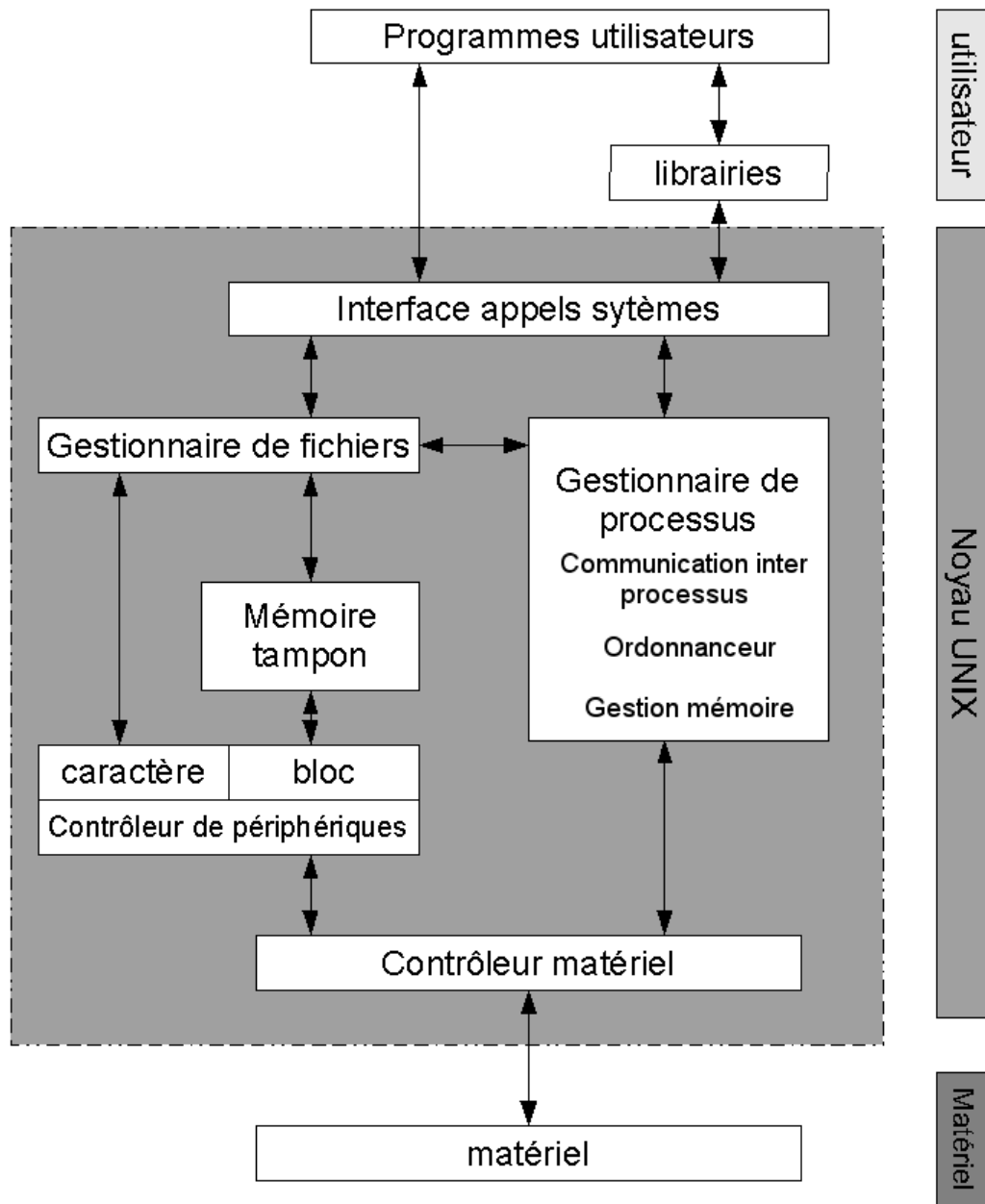
Avant-Projet

1.

a)



b)



c)

- i. Les registres de l'unité centrale sont des zones mémoire internes au processeur. Ces zones mémoires sont les plus proches du processeur. Elles sont par conséquent accessibles plus rapidement que tous les autres types de mémoire.

On rencontre divers registres au sein du processeur :

- Le compteur ordinal : indique l'emplacement de la prochaine instruction à exécuter.
- Le registre d'instruction : c'est le registre de destination dans lequel le CPU transfère l'instruction suivante à partir de la mémoire .
- L'accumulateur : stocke les résultats d'opérations arithmétiques et logiques.
- Le pointeur de pile (Stack Pointer) : indique la position du prochain emplacement disponible dans la pile mémoire.
- Les registres d'adresses et de données qui permettent la communication avec les bus correspondants.
- Le registre d'état, Program Status Word (PSW): il sert à stocker le contexte du processuer, il décrit son état.

Le pointeur de pile et le compteur ordinal sont destinés à stocker des adresses tandis que les autres servent à stocker des données.

La mémoire vive (RAM) est la mémoire dans laquelle le processeur place les données lors de leur traitement. Ce type de mémoire est accessible rapidement et est volatile, ce qui signifie que les données contenues dans cette mémoire sont perdues lorsque le système s'arrête.

- ii. Lorsque le système change de tâche, le changement de contexte est le fait de mémoriser l'état du processus courant (le contenu des registres du processeur, le pointeur de pile, etc.) puis de charger le processus suivant dans l'ordonnanceur. Le changement de contexte est une opération très coûteuse en terme de ressources, c'est pourquoi il existe une autre alternative : les threads.
- iii. Les interruptions systèmes sont représentées sous forme d'événements.

On distingue les événements d'ordre matériels des événements d'ordre logiciel.

Ces événements peuvent concerner le processus en question ou tout autre processus dans le système. L'événement est envoyé au processus correspondant par l'intermédiaire d'un signal dont bon nombre ont été vus en TP (SIGALRM, SIGCHLD, etc.). Ce dernier se chargera d'interrompre le processus correspondant. C'est à ce moment que le changement de contexte intervient afin de pouvoir reprendre le processus lors de la fin de l'interruption.

Les appels système sont des fonctions fournies par le noyau étant utilisées par les processus. Parmi les principaux appels systèmes, on distingue ceux concernant le système de fichiers (read(), open(), close(), write(), lseek(), stat(), link(), dup(), etc.), ceux concernant le contrôle des processus (execve(), fork(), wait(), getpid(), etc.), ainsi que ceux concernant la communication inter-processus (pipe(), semop(), etc.)

- iv. Le mode maître et le mode utilisateur sont deux états de fonctionnement possibles du processeur permettant l'isolation des fautes et de la mémoire entre les différents processus. En mode maître, seule la mémoire physique est accessible, alors qu'en mode utilisateur c'est la mémoire virtuelle qui est accessible.
- v. L'espace mémoire d'un processus lui est attribué par le gestionnaire de mémoire (MMU : Memory Management Unit). En effet, ce dernier répertorie les emplacements libres de la mémoire et alloue la mémoire nécessaire aux nouveaux processus. Un processus se voit attribué trois régions en mémoire ; une région pour le code exécutable (.text) ; une région pour la pile ; ainsi qu'une région pour les données (.data). De plus, le gestionnaire mémoire peut également récupérer la mémoire des processus qui s'achèvent. Cette récupération peut nécessiter une ré-allocation des processus en cours pour optimiser l'emploi de la mémoire. La zone mémoire attribuée à un processus peut donc changer au cours de son exécution. La mémoire est paginée, ce qui signifie qu'elle et les processus sont divisés en blocs de même taille appelés pages.
- vi. Pour permettre la communication entre tous les processus et non seulement entre processus père et processus fils, il y a la mémoire partagée. Celle-ci désigne un large bloc de mémoire vive (RAM) accédé par plusieurs processus.

L'exclusion mutuelle est une primitive de synchronisation de processus permettant d'éviter que des ressources partagées d'un système ne soient utilisées en même temps. On parle de mutex. Par exemple, les sémaphores sont des mutex avec un compteur.

- d) Allocation de mémoire pour un processus : utilisation des fonctions C : malloc() et free() Cette dernière permettant de restituer la mémoire alloué.

Création de mémoire partagée : shmget(); shmat(), pour l'affecter au processus, puis synchronisation de l'accès avec les sémaphores (créées avec semget()).

Utilisation d'un mutex : pthread_mutex_lock() bloque le mutex, pthread_mutex_unlock() le débloque.