

# Systèmes d'exploitation, 2ème année

Fichiers et processus

Yves STADLER

Université Paul Verlaine - Metz

5 octobre 2011

# Agenda

## Plan du module

- Système de gestion des fichiers ;
- Gestion des processus (Parallélisme, threads, ordonnancement) ;
- Synchronisation et concurrence ;
- Gestion de la mémoire ;
- Communications.

# Présentation générale d'un système d'exploitation

## Définition

Extension logicielle du matériel dans le but d'offrir un service suffisant aux utilisateurs.

## Objectif

- Masquer la complexité à l'utilisateur ;
- Faciliter l'accès aux ressources.

## Types de systèmes d'exploitations

- mode d'exploitation différé ;
- mode interactif d'exploitation, temps partagé ;
- mode temps réel, système embarqués ;
- mono-tâche, multi-tâches.

# Présentation générale d'un système d'exploitation

## Organisation d'un système d'exploitation

- Chaque système est basé sur un noyau (*kernel*) ;
- La noyau comprend deux parties : indépendante du matériel, dépendante du matériel ;

## Partie dépendante

- gestion des interruptions ;
- gestion mémoire ;
- gestion des entrées sorties (E/S ; IO).

# Présentation générale d'un système d'exploitation

## Partie indépendante

- ordonnanceur-distributeur ;
- gestion des processus ;
- pagination, va-et-vient ;
- sous-système de fichier ;
- gestion des entrées sorties (partie "haute").

# Présentation générale d'un système d'exploitation

## Architecture UNIX

- Portabilité ;
- Langage C ;
- Processus arborescents.

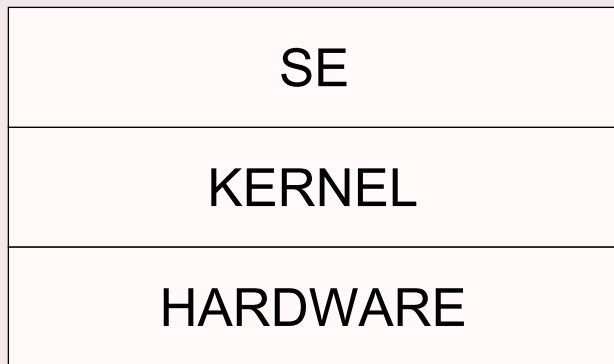
## Unix

- mono-tâche ou multi-tâches
- temps partagé

# Présentation du noyau

## Modèle en couche

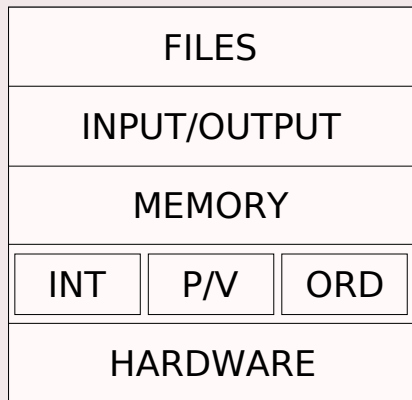
Le système peut se présenter comme un ensemble de couches.



# Présentation du noyau

## Modèle en couche

- Le noyau ou kernel constitue une interface entre le matériel et les programmes
- Il est lui-même divisé en couches
- Il est irremplaçable par l'utilisateur
- Il gère les processus, les périphériques, la mémoire





# Présentation du noyau

## Allocation de ressources

- Rôle central du *kernel* dans l'exécution des travaux :
  - représentation et gestion des processus ;
  - gestion des interruptions ;
  - gestion des entrées, sorties.

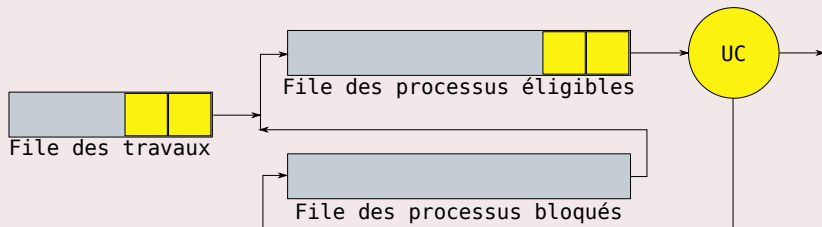


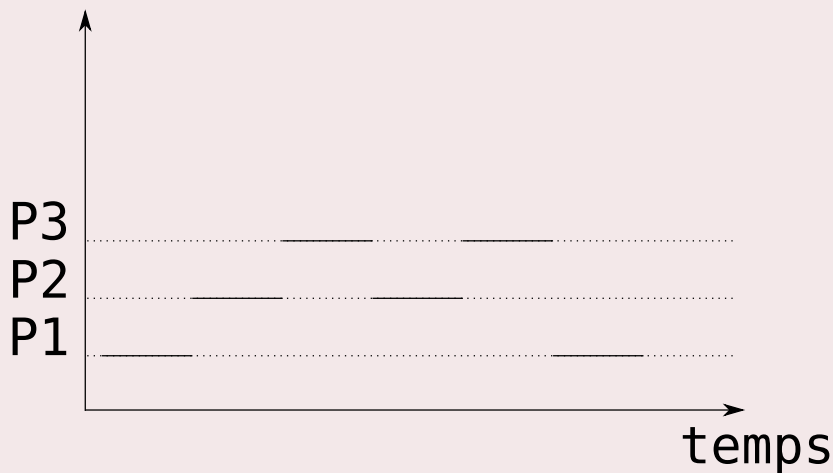
FIGURE: Exemple de modèle d'allocation de l'UC

# Présentation du noyau



FIGURE: Allocation de l'UC pour les systèmes mono-tâche

# Présentation du noyau

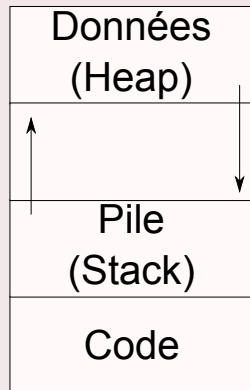


**FIGURE:** Allocation de l'UC pour les systèmes multi-tâches

# Rappel sur les processus

## L'espace de travail

- L'espace de travail est l'ensemble des données en mémoire nécessaires à l'exécution du processus.
- Le code : en langage ASM, liste des instruction pour le processeurs (Pointeur : Compteur Ordinal)
- Les données (le tas / heap) Ensemble des variables (globales/statiques/dynamiques) remplissage vers le bas
- La pile (stack) variables des appels de fonctions (inputs, valeur avant appel (co)...) remplissage vers le haut.



# Rappel sur les processus

## La zone u

- Données privées du processus. Seule la zone u du processus en cours est manipulable. (struct user <sys/user.h>). Son adresse se trouve dans le mot état.
  - pointeur sur la structure de processus de la table des processus.
  - uid réel et effectif
  - Compteurs des temps (users et system) consommés
  - Masque de signaux
  - Terminal de contrôle du processus si celui-ci existe.
  - Dernière erreur rencontrée pendant un appel système.
  - Valeur de retour du dernier appel système.

# Rappel sur les processus

## La zone u

- E/S (les structures associées aux entrées-sorties)
- "." et "/" (le répertoire courant et la racine courante (c.f. `chroot()`)
- La table des descripteurs
- Limites de la taille des fichiers de la mémoire utilisable
- `umask` (masque de création de fichiers)

# Rappel sur les processus

## Contexte

- son état
- son mot d'état : en particulier (La valeur des registres actifs; Le compteur ordinal )
- les valeurs des variables globales statiques ou dynamiques
- son entrée dans la table des processus
- sa zone u
- Les piles user et system
- les zones de code et de données.

## Contexte

Lorsqu'un nouveau processus va être exécuté, il y a commutation du mot d'état et changement de contexte. Ces changements sont dictés par l'ordonnanceur.

# Mécanisme de contrôle

## Mécanismes

- Les interruptions ;
- les appels systèmes ;
- les signaux horloge ;
- les primitives de synchronisation.

## Les interruptions

- Événement suffisamment important pour nécessiter une interruption du système.
- Rapide, immédiat.
- généré aléatoirement par un périphérique ou par l'UC ;
- peut-être générée par le système en interne, on parle alors de détournement.



# Mécanisme de contrôle

## Rôle des interruptions

- Imposer les changement d'état de l'UC ;
- Commutation de contexte, générée par une cause extérieur à l'instruction en cours.

## Rôle des interruptions

- L'UC doit être en mode interruptible ;
- L'interruption doit être prioritaire aux autres interruptions ;

# Mécanisme de contrôle

## Appel au superviseur - Définition

- Instruction qui a pour effet de provoquer une commutation de contexte du processeur.

## Appel au superviseur - Rôle

- Permettre l'appel depuis un programme d'une procédure du système nécessitant des droits étendus.
- Masquage interruption, ...

# Mécanisme de contrôle

## Signaux d'horloge

- Composant physique du système ;
- Essentiel car il rythme le système ;
- Génère des interruptions horloges.

## Signaux d'horloge

- Affectation du temps dans le mot horloge
- Pendant l'activation du processus, le mot horloge est décrémenté de 1 à chaque signal ;
- Quand le mot vaut 0, un signal d'interruption est généré et le superviseur prend une décision.

# Mécanisme de contrôle

## Primitives de synchronisation

- Nécessaires à l'asynchronisme ;
- Nécessaire à la protection mutuelle ;
- Plus que de simples appels de procédures.

## Exemple de l'exclusion mutuelle

- Deux processus A et B veulent mettre à jour le compte en banque d'Alice
- $C_{Alice} = C_{Alice} + Montant_a$  et  $C_{Alice} = C_{Alice} + Montant_b$
- Quel est le résultat final ?

## Suite

- Processus et ordonnancement.