

# Chapitre VI: Gestion des entrées/sorties logique

## 1 Introduction

Une E/S logique est une instruction à l'intérieur d'un programme qui fait appel à une procédure d'E/S (Exemple: instruction de lecture `Scanf` du langage C, instruction d'écriture ou affichage `Printf` du C, ouvrir un fichier .etc). C'est avec une E/S logique qu'un processus demande au système d'exploitation d'effectuer une opération d'E/S physique. L'E/S logique est traduite en un appel au superviseur SVC. Le superviseur (système) doit **allouer** un périphérique pour cet appel, le **libérer** et **gérer** plusieurs demandes simultanées de périphériques. Ceci le rôle d'un module du SE appelé **gestionnaire de périphérique**.

## 2 Fonctions du gestionnaire de périphériques

Le gestionnaire doit réaliser les fonctions de base suivantes :

- Définir une politique d'allocation d'un périphérique à un processus en faisant un ordonnancement (scheduling), à quel processus, à quelle date et pour combien de temps.
- Sauvegarder l'état de chaque périphérique (contrôleur et canaux d'E/S) dans des structures de données appelées bloc de contrôle (Unit control block UCB)
- Définir une politique de dé allocation (restitution) du périphérique.

## 3 Types de périphériques

Selon le rôle du périphérique on distingue deux types.

### 3.1 Unité périphérique d'E/S

Son rôle est de transférer des données entre système informatique et l'utilisateur.

**Exemple** : Clavier, Ecran, souris, .etc

### 3.2 Unité de stockage

Sert à stocker de l'information. Dans ce chapitre, nous allons prendre une étude de cas du disque magnétique (disque dur).

## 4 Structure physique d'un Disque Dur DD

Les données sont stockées sous forme de fichier physique dans le disque dur. Un fichier physique correspond à l'aspect physique des données. Il exprime l'espace alloué aux données (enregistrements<sup>1</sup>) sur le support Physique.

Le DD est constitué d'un ensemble plateaux ou disques empilés verticalement l'un au dessus de l'autre suivant un même axe. Chaque plateau a deux faces. Un bras portant une tête de lecture/écriture par face (un plateau = 2 têtes).

- La tête se déplace horizontalement : avance vers l'axe ou recule de l'axe.
- La pile tourne autour de l'axe à une vitesse de 4000 Tour/m à 15000 Tour/m (Atuellement 7200 T/m). Le disque a un seul sens rotation.

Les **pistes** sont des cercles concentriques : au nombre, elles mêmes sont divisés en un certain nombre de **secteurs** (exp. 4 à 64 secteurs par pistes).

Chaque piste est découpée en **secteurs** : 4 à 64 secteurs par pistes.

Le secteur est la plus petite unité **physique** de lecture/écriture. Sa taille varie entre 32 à 4096 O (généralement 512o).

L'unité de transfert entre la mémoire centrale et le Disque est le **bloc** ou **Cluster**. Chaque bloc est composé de quelques secteurs.

Un fichier sur disque peut occuper plusieurs blocs

Un **cylindre** est constitué par toutes les pistes superposées verticalement (une par face).

➔ Le matériel associé à un disque comprend un dispositif de lecture/écriture qui exécute les activités mécaniques : mouvement du bras et rotation des plateaux.

<sup>1</sup> Le type abstrait de donnée du fichier est un ensemble d'enregistrements

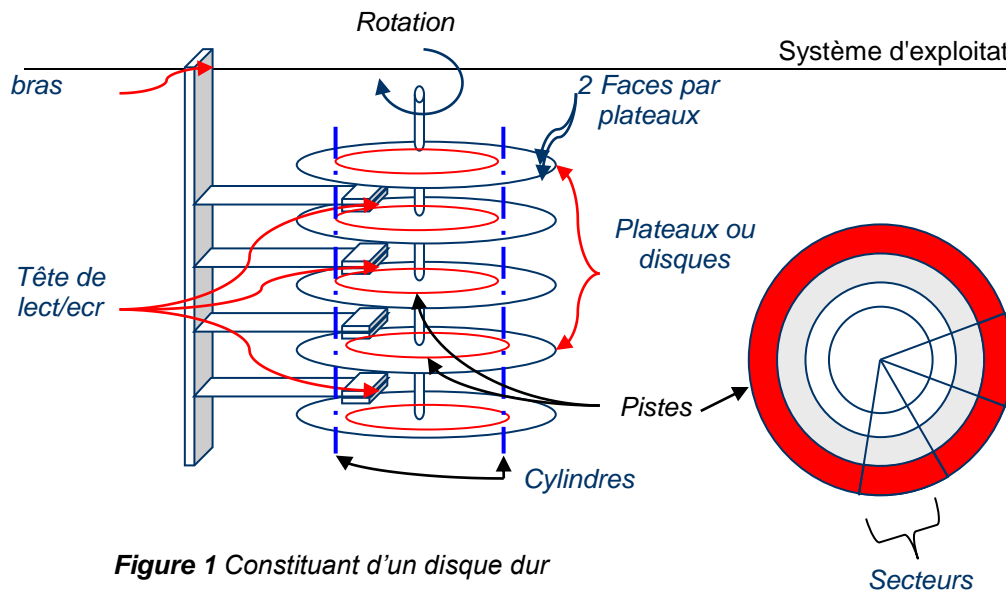


Figure 1 Constituant d'un disque dur

#### 4.1 Opération de lecture/écriture sur disque dur

A fin d'effectuer une opération de lecture ou d'écriture d'une donnée sur des secteurs disque, ces derniers doivent posséder des adresses. Une adresse d'un secteur est constituée de trois champs : **<N°Face ou N° tête , N° piste ou N° cylindre dans la face, et N° secteur dans la piste>**.

La réalisation d'une requête d'E/S comprend plusieurs étapes :

- Si le mécanisme de lect/écr est occupé par une autre requête, faire attendre les requêtes dans une file du contrôleur, pendant un **temps d'attente**.

- Le bras doit se déplacer vers la piste du numéro N° piste (N° cylindre). Le temps de déplacement est appelé **temps de recherche** (Seek Time). Ce temps dépend de la position actuelle du bras (N°piste='a') et le N° piste (N°piste='b') vers laquelle le bras doit se déplacer. On peut l'exprimer sous forme d'une relation en fonction du nombre de pistes séparant les deux pistes 'a' et 'b' et le temps 'd' que le bras met pour parcourir une seule piste :

$$\text{SeekTime} = |a-b| * d + c. \text{ 'c' est une constante de stabilisation de la tête.}$$

- La rotation du disque pour trouver le bon secteur dont le numéro est spécifié dans l'adresse de la requête. Ce temps de rotation dépend aussi de la position du secteur. Ce temps de rotation est appelé **temps de latence**.

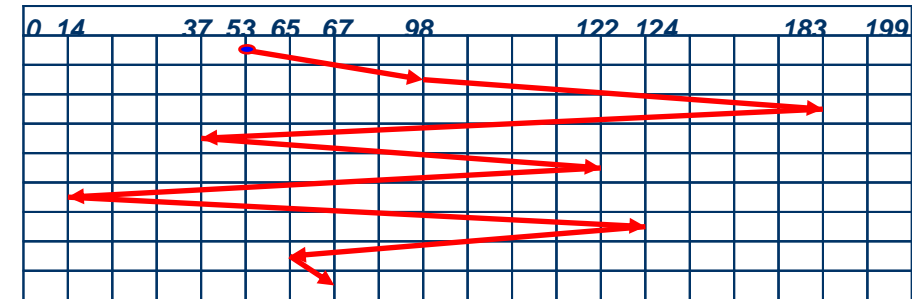
- Transférer le bloc (lecture/écriture) de la Ms à la MC, où on peut faire appel au DMAC. Le temps consommé s'appelle **temps de transfert**, Il dépend de la

capacité du bus auquel le disque dur est connecté et de la capacité de transfert du contrôleur du disque.

Alors le temps d'exécution d'une opération d'E/S est le suivant :

$$\text{Temps d'une opération d'E/S} = \text{Temps d'attente} + \text{Temps de recherche}$$

+ temps de latence + temps de transfert



Le temps d'attente dans la file → stratégie d'ordonnancement des requêtes.

$$\text{Nombre de pistes parcourues} = |53-183| + |183-37| + |37-122| + |122-14| + |14-124| + |124-65| + |65-67| = \mathbf{640 \text{ pistes}}$$

Figure 2 Exemple d'application de la politique FIFO, la courbe montre le déplacement du bras du DD entre les pistes.

#### 4.2 Politiques de scheduling (d'ordonnancement) d'E/S sur DD

Le rôle d'une politique (stratégie) d'ordonnancement consiste à choisir une requête de demande d'accès au disque parmi les requêtes existantes dans la file d'attente.

Plusieurs stratégies existent et qui se basent sur :

- 1) L'optimisation du **Temps de recherche**. Elles visent à optimiser les **déplacements du bras** d'une piste à une autre. Par conséquent elles adoptent une file contenant les numéros de la piste ciblée par la requête.
- 2) L'optimisation du **temps de latence**. Elles visent à optimiser la **rotation du disque**. Par conséquent elles adoptent une file contenant les numéros de secteurs ciblée par la requête dans une piste donnée.

### 4.2.1 Politiques pour optimiser le temps de recherche

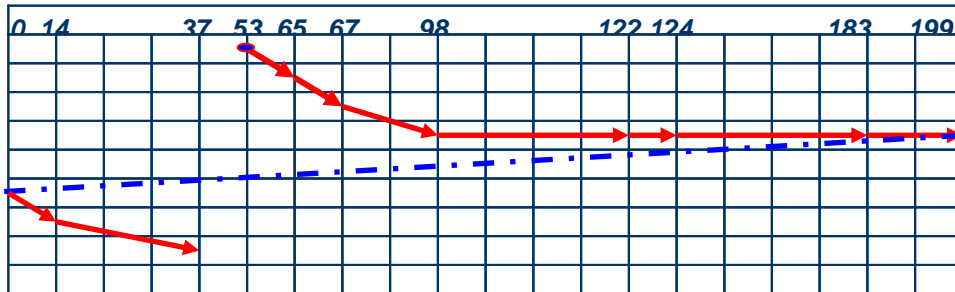
Pour toutes les politiques d'ordonnancement, nous prenons la liste suivante des requêtes avec les numéro de pistes = 98, 183, 37, 122, 14, 124, 65, 67 ; la tête du bras est initialement placée sur la piste N°53

#### 4.2.1.1 Politique FiFO

La première requête qui arrive est celle qui sera la première servie. Satisfaire la première puis déplacer la tête à la piste suivante selon la demande

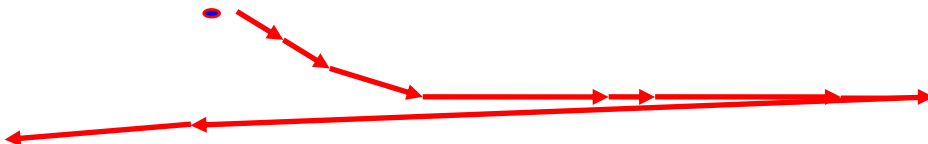
#### 4.2.1.2 Politique SSTF : Shortest Seek Time First

Son principe est de servir la requête dont la piste est la plus proche à la position



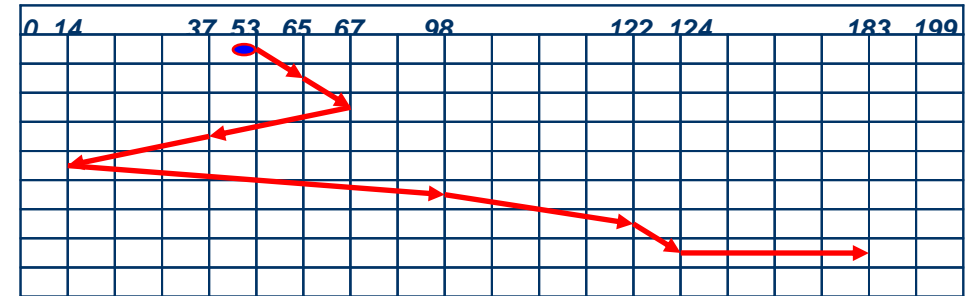
Nombre de pistes parcourues = 382 pistes

**Figure 5** Exemple d'application de la politique **Circular Scan**, la courbe montre le déplacement du bras entre pistes



Nombre de pistes parcourues = 331 pistes

**Figure 4** Exemple d'application de la politique **Scan (ascenseur)**, la courbe montre le déplacement du bras entre pistes



Nombre de pistes parcourues = 236 pistes

**Figure 3** Exemple d'application de la politique **SSTF**, la courbe montre le déplacement du bras entre pistes

courante du bras.

#### 4.2.1.3 Politique : Scan (Ascenseur)

Cette politique limite les changements de direction du bras en parcourant tout l'intervalle dans un sens (tant qu'il y a des requêtes à servir dans ce sens) puis dans l'autre sens.

#### 4.2.1.4 Politique : C-Scan (Circular Scan)

C'est une politique de l'ascenseur à sens unique ("circular scan") ne parcourt les requêtes que dans un sens et en fin d'intervalle revient à la première piste du sens de parcours sans s'arrêter sur les requêtes rencontrées sur son retour. Cette politique est utilisée lorsque les files d'attente sont fortement chargées.

#### 4.2.1.5 Politique : Look-Scan

Le principe du Scan mais ne pas déplacer le bras jusqu'à la fin (ou début) inutilement. S'arrêter à la dernière requête dans le sens de balayage.

#### 4.2.1.6 Politique :C-Look-Scan

C'est le principe du C-Scan combiné avec celui du Look-Scan.

#### 4.2.2 Politique pour optimiser le temps de Latence

Une fois on a décidé de la piste recevant la tête de lect/écr, il se peut que plusieurs requêtes demandent la même piste mais avec des **secteurs différents**. Alors, on peut s'intéresser aussi à l'ordonnancement de ce sous ensemble de requêtes suivant les secteurs demandés dont l'objectif est de **minimiser les rotations** entre les secteurs au sein d'une seule piste. On distingue deux stratégies :

- 1) Stratégie utilisant une file unique pour toutes les requêtes
- 2) Stratégie utilisant une file par secteur

##### 4.2.2.1 Stratégie utilisant une file unique pour toutes les requêtes

Les requêtes sont servies dans l'ordre FIFO

##### 4.2.2.2 Stratégie implémentant une file par secteur

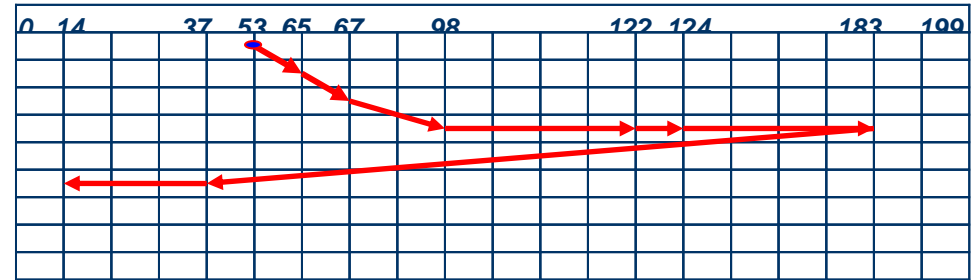
L'algorithme SLTF : **Shortest Latency Time First**

Cette stratégie satisfait d'abord la requête ayant le plus court temps de rotation par rapport à la position courante de la tête de lect/écr, comme le fait la méthode SSTF.

## 5 Conclusion

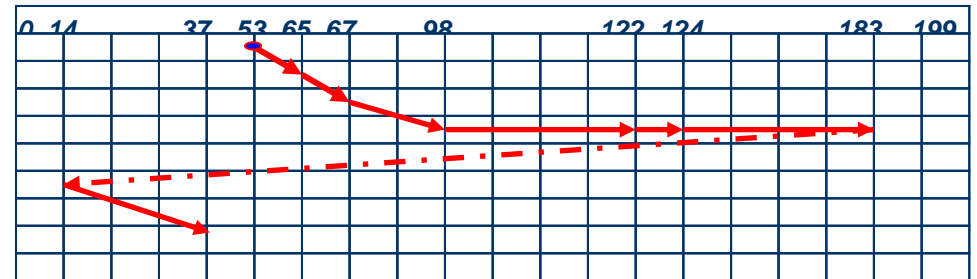
Les E/S logiques sont déclenchées suite à une instruction dans un programme. Cette E/S est accomplie par un périphérique géré par un gestionnaire d'E/S. Ce dernier utilise des politiques d'allocation et de dé-allocation des périphériques aux processus, dont l'objectif est d'améliorer les performances du SE. Le périphérique disque dur abrite les données demandées par les processus. De ce fait, deux classes de stratégies améliorent l'accès à ces données.

La première classe améliore le temps de recherche de la piste par le déplacement minimal du bras. Elle regroupe 6 stratégies, le FIFO, SSF (Shortest Seek First), Scan (toutes les requêtes sont suivies jusqu'à une extrémité puis on revient en sens inverse vers l'extrémité opposée), C-Scan (Scan avec reprise à l'extrémité opposée), Look-Scan (où on s'arrête à la dernière requête) et C-Look (Look avec reprise à partir de la requête de l'extrémité opposée).



Nombre de pistes parcourues = **316 pistes**

**Figure 7** Exemple d'application de la politique **Look Scan**, la courbe montre le déplacement du bras entre pistes



Nombre de pistes parcourues = **352 pistes**

**Figure 6** Exemple d'application de la politique **C-Look Scan**, la courbe montre le déplacement du bras entre pistes

La deuxième classe de stratégie optimise le temps de rotation du disque pour trouver le secteur demandé.