

# Stockage et manipulation de données, des fichiers aux bases de données

## I. Le stockage de masse

### A. Généralités sur le stockage

**Définition 1** La persistance [OSTEP 35] (ou non-volatilité) d'une mémoire signifie que les informations stocké dans la mémoire ne disparaissent pas lors d'une mise hors-tension.

**Définition 2** Le stockage de masse [NSIP] (ou stockage persistant) est une mémoire non-volatile (NVM) qui permet de stocker de grandes quantités d'informations.

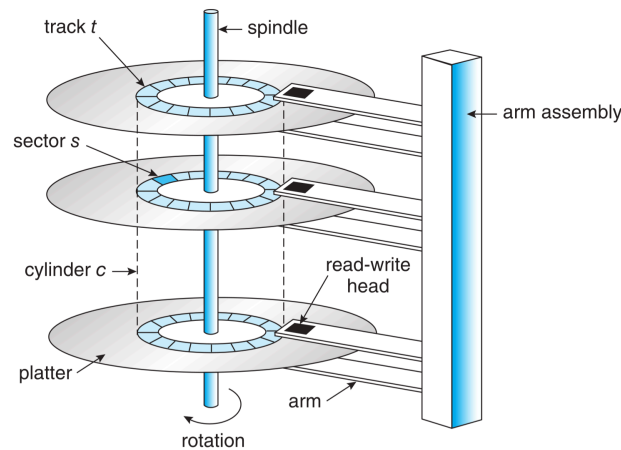
**Remarque 3** [OSC 11] Il existe plusieurs technologies de stockage de masse qui font des compromis différents entre prix, débit, latence et capacité.

**Définition 4** Un disque dur [OSC 11.1.1] (ou HDD pour Hard Disk Drive) est composé de plateaux magnétiques qui tournent et d'une tête de lecture et écriture.

**Définition 5** [OSC 11.1.1] Les données d'un disque dur sont stockés sur les plateaux le long de **pistes** et chaque piste est découpé en **secteurs**.

**Remarque 6** Ordonnancement des requêtes [OSC 11.2] Il existe plusieurs stratégies pour ordonnancer les requêtes sur un disque dur.

**Définition 7** Un SSD [OSC 11.1.2] (pour Solid State Drive) utilise de la mémoire flash non volatile fabriquée à partir de transistor.



**Remarque 8** [OSC 11.1.2] Comparé à un HDD, un SSD dispose de plusieurs avantages :

- une résistance aux chocs et vibrations (aucune pièces mobiles)
- une vitesse de lecture et écriture bien supérieure (2 à 100 fois)
- un temps d'accès plus court (10 à 100 fois)

Cependant, la mémoire flash possède aussi des désavantages :

- un coût de fabrication élevé (2 à 3 fois)
- une durée de vie limité en le nombre de réécriture par cellule de la puce mémoire

### B. Interagir avec un périphérique de stockage

**Définition 9** [OSTEP 36.1]

Les périphériques d'entrée-sortie d'un ordinateur sont connecté sur des bus mémoires :

- Les HDD et SSD lent sont connectés au bus E/S périphérique à l'aide de connecteurs SATA (Serial AT Attachment) ou USB (Universal Serial Bus)
- Les SSD rapides utilise le bus E/S général à l'aide d'une interface PCIe (Peripheral Component Interconnect Express).

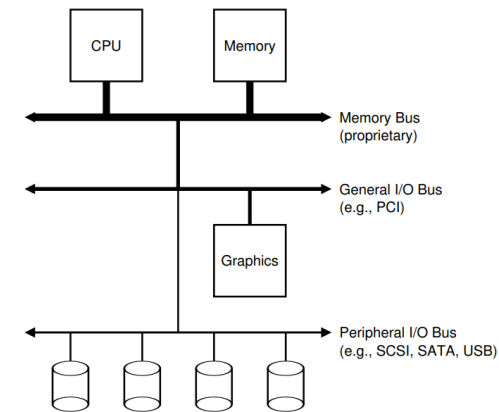


Figure 36.1: Prototypical System Architecture

**Définition 10** Interface. [OSTEP 36.2] Un périphérique de stockage interagit avec le système d'exploitation à l'aide d'une interface composé de 3 registres :

- **Etat** : décrit l'état du périphérique (notamment s'il est occupé),
- **Commande** : indique au périphérique l'action à faire,
- **Données** : permet le transfert de données entre le périphérique et l'OS

**Définition 11** Un pilote [OSTEP 36.7] permet d'encapsuler et abstraire le périphérique. Cela évite au système d'exploitation de connaître les détails d'implémentation du périphérique.

## II. Gestion des fichiers

### A. L'abstraction des fichiers et répertoires

**Définition 12** [OSTEP 39.1] Un **fichier** est une séquence d'octets que l'on peut lire ou modifier. Chaque fichier possède un nom bas niveau, souvent un nombre, appelé **nombre inode**.

**Définition 13** [OSTEP 39.1] Un **répertoire** est un fichier qui contient une liste de couple (nom utilisateur, nombre inode). Cela permet donc de ranger d'autres fichiers et répertoires dedans.

**Exemple 14** [OSTEP 39.1] Par exemple si un répertoire contient le fichier `foo` de nombre inode 10, alors la liste du répertoire contient le couple `(foo,10)`.

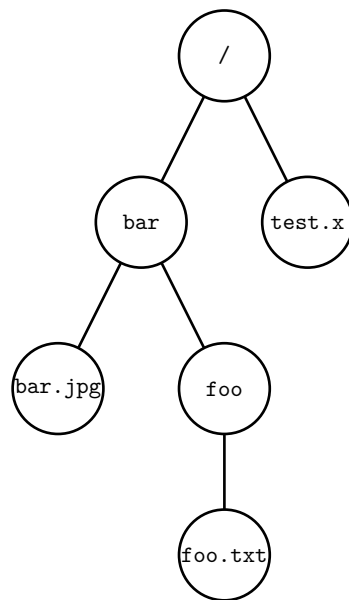
**Remarque 15** [OSTEP 39.1] En plaçant des répertoires dans d'autres répertoires, on obtient une hiérarchie de répertoires. Cette hiérarchie commence par un **répertoire racine** (pour un système Unix il s'agit de `/`) et descend dans des sous-répertoires.

**Définition 16** [OSTEP 39.1] Un **chemin d'accès** d'un fichier permet de décrire la position du fichier dans la hiérarchie de répertoire. Il peut être absolu en partant de la racine `/bar/foo/foo.txt` ou relatif `./foo/foo.txt` (en partant de `bar` par exemple).

### B. Le système de fichier

**Définition 17** [OSTEP Intro 2.4] Un **système de fichier** est le logiciel chargé de stocker et gérer de manière efficace et robuste les différents fichiers sur le stockage de masse.

**Exemple 18** [OSTEP 40] Le système de fichier *vsfs* (very simple file system) est un système de fichier simple permettant de présenter les concepts utiles à la conception d'un système de fichier.



**Définition 19** [OSTEP 40.2] Le système de fichier divise en général un disque en **blocs**. Ces blocs sont de taille fixe dans la majorité des systèmes de fichier, par exemple 4ko pour *Ext4*.

**Remarque 20** [TOR 2.3.1] Les systèmes de fichiers gèrent aussi les **permissions** de lecture, d'écriture et d'exécution des fichiers et répertoires.

**Définition 21** Lien physique [TOR 2.3.1] Comme les noms des fichiers ne sont pas stockés dans les *inodes*, mais dans le répertoire qui le contient, on peut associer plusieurs noms au même fichier. On peut créer ces liens physiques avec la commande `ln`.

**Définition 22** Un lien symbolique [TOR 2.3.1] est un fichier spécial dont le contenu est simplement *un chemin* vers un autre fichier. On peut créer un lien symbolique en passant l'option `-s` à la commande `ln`.

### C. Le système d'exploitation

**Remarque 23** On se concentre ici sur les systèmes d'exploitation Unix. Des appels systèmes équivalents existent pour d'autres OS.

**Définition 24** [OSTEP 39.3] Pour ouvrir et créer des fichiers, on peut utiliser l'appel système `open` et les fermer avec l'appel `close`.

**Définition 25** [OSTEP 39.3] L'appel système `open` renvoie un **descripteur de fichier**, un entier privé au processus en cours. Cela permet d'abstraire le fichier pour le processus qui passe par des appels `read` et `write` pour lire et modifier le fichier.

**Remarque 26** [OSTEP 39.4] En plus des fichiers du disque, le système d'exploitation met à disposition des fichiers spéciaux comme l'entrée standard `stdin`, la sortie standard `stdout`, la sortie d'erreur `stderr` ou des tubes (pipe).

**Remarque 27** [OSTEP 39.11] Il existe des appels similaires pour les répertoires : `mkdir`, `opendir`, `closedir`, `readdir`.

**Remarque 28** Dans de nombreux langages, ces appels systèmes possèdent des surcouches facilitant leur utilisation. Par exemple en C, la bibliothèque standard propose `fopen`, `fread` et `fwrite`.

### III. La base de données

**Définition 29** Une Base de Données [GARD I.1] est un ensemble de données modélisant les objets d'une partie du monde réel et servant de support à une application informatique.

On doit pouvoir retrouver le sous-ensemble de ces données qui correspond à un certain critère, peu importe le critère.

**Remarque 30** Ces bases de données peuvent être amenées à traiter des données personnelles, et font donc l'objet de réglementations telle que le RGPD.

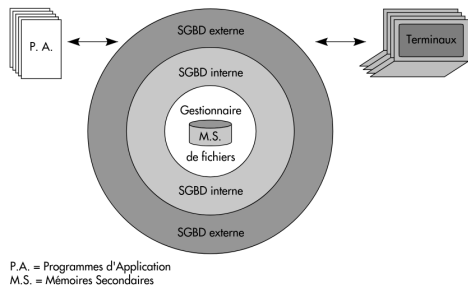
#### A. Les SGBD

**Définition 31** Un SGBD [GARD I.1] (Système de Gestion de Base de Données) est un ensemble de logiciels systèmes permettant aux utilisateurs d'insérer, de modifier et de rechercher efficacement des données spécifiques dans une grande masse d'informations partagée par de multiples utilisateurs.

**Remarque 32** [GARD I.1]

Un SGBD peut se découper en 3 couches :

- L'écriture des données sur disques, gérés par un système de fichier.
- La gestion des données stockées dans les fichiers, c'est le SGBD interne.
- la mise en forme et l'interface d'accès pour les utilisateurs, c'est le SGBD externe.



**Remarque 33** [GARD II.1] L'accès et la manipulation des données stockés dans un SGBD se fait à l'aide d'un langage unifié de description et de manipulation de données, comme le SQL.

**Définition 34** Niveaux d'abstraction [GARD II.2.2] Un SGBD à pour objectif d'abstraire les données stockés sur disque. On distingue 3 niveaux de description de données :

- Le niveau **conceptuel**, qui correspond à la structure canonique comprise par les utilisateur. Par exemple, en comprenant les types composé Film et Spectateur, et l'association Regarde qui relie les deux types à une certaine date.
- Le niveau **interne**, qui correspond à la représentation sur disque des données
- Le niveau **externe**, qui correspond à la manière dont les données sont vues dans les programmes d'application. C'est par exemple ici que se trouve les tables SQL.

#### B. Le Modèle Relationnel

**Définition 35** Le modèle relationnel est un modèle de représentation des bases de données, basés sur des objets complexes (ici, un film), appelés entités ou enregistrements.

**Définition 36** Une Relation [TOR 11.2.1] est un ensemble de n-uplets, tous de même taille.

**Exemple 37** Représentation tabulaire pour les relations:

Film				
titre text	année int	durée int	genre text	pays text
La mélodie du bonheur	1965	174	musical	États-unis
Crocodile Dundee 2	1988	112	aventure	Australie
Le livre de la jungle	1967	78	animation	États-unis
Casino Royale	2006	144	espionnage	Royaume-uni
Léon	1994	110	drame	France
⋮				

**Définition 38** [GARD II.4] Un SGBD fourni des fonctions des opérations de description, de recherche, de mise à jour et de transformation des données.

**Définition 39** [GARD VI.4] Le modèle relationnel défini plusieurs opérations ensemblistes sur les relations : l'union, la différence, le produit cartésien, la projection, la restriction et la jointure.

**DEV** **Algorithme 40** L'opération de jointure permet de composer deux relation à l'aide d'un critère de jointure. Son calcul est une opération pouvant être coûteuse, mais aussi très optimisée.

<u>Stockage et manipulation de données, des fichiers aux bases de données</u>	
<u>I. Le stockage de masse</u>	
<u>A. Généralités sur le stockage</u>	
1	Def La persistance [OSTEP 35]
2	Def Le stockage de masse
3	Rem [OSC 11]
4	Def Un disque dur [OSC 11.1.1]
5	Def [OSC 11.1.1]
6	Rem Ordonnancement des requêtes [OSC 11.2]
7	Def Un SSD [OSC 11.1.2]
<u>II. Gestion des fichiers</u>	
<u>A. L'abstraction des fichiers et répertoires</u>	
12	Def Un fichier [OSTEP 39.1]
13	Def Un répertoire [OSTEP 39.1]
14	Ex [OSTEP 39.1]
15	Rem [OSTEP 39.1]
16	Def Un chemin d'accès [OSTEP 39.1]
<u>B. Le système de fichier</u>	
17	Def Un système de fichier [OSTEP Intro 2.4]
18	Ex [OSTEP 40]
<u>III. La base de données</u>	
29	Def Une Base de Données [GARD I.1]
30	Rem
<u>A. Les SGBD</u>	
31	Def Un SGBD [GARD I.1]
32	Rem [GARD I.1]
33	Rem [GARD II.1]
34	Def Niveaux d'abstraction [GARD II.2.2]
8	Rem [OSC 11.1.2]
<u>B. Interagir avec un périphérique de stockage</u>	
9	Def [OSTEP 36.1]
10	Def Interface. [OSTEP 36.2]
11	Def Un pilote [OSTEP 36.7]
19	Def [OSTEP 40.2]
20	Rem [TOR 2.3.1]
21	Def Lien physique [TOR 2.3.1]
22	Def Un lien symbolique [TOR 2.3.1]
<u>C. Le système d'exploitation</u>	
23	Rem
24	Def [OSTEP 39.3]
25	Def [OSTEP 39.3]
26	Rem [OSTEP 39.4]
27	Rem [OSTEP 39.11]
28	Rem
<u>B. Le Modèle Relationnel</u>	
35	Def Le modèle relationnel
36	Def Une Relation [TOR 11.2.1]
37	Ex
38	Def [GARD II.4]
39	Def [GARD VI.4]
40	Algo

## Remarque

- La partie III base de données se concentre vraiment sur la montée en abstraction par rapport aux fichiers, et comment les données sont stockées et manipuler. On ne détail pas le SQL par exemple, on se content de mentionner son existence.
- La tortue n'est utilisée que pour la définition du modèle relationnel. La partie dédié dans [GARD] est très longue et pas très précise, alors que la tortue donne une définition simple avec un exemple tout de suite.

## Bibliographie

[OSTEP] R. H. Arpaci-Dusseau & A. C. Arpaci-Dusseau, *Operating Systems, three easy pieces*.

[NSIP] T. Balabonski & S. Conchon & J. Filliâtre & K. Nguyen, *Numériques et Sciences Informatiques 1er*.

[OSC] A. Silberschatz & P. B. Galvin & G. Gagne, *Silberschatz's Operating System Concepts, Global Edition*.

[TOR] T. Balabonski & S. Conchon & J. Filliâtre & K. Nguyen & L. Sartre, *MP2I MPI, Informatique Cours et exercices corrigés*.

[GARD] G. Gardarin, *Bases de données*.