DEUG

Université de Marne La Vallée

1993-1994

Première approche de MS-DOS

Petit mémento
à l'usage de l'utilisateur de MS-DOS

Université de Marne La Vallée
2, rue de la Butte Verte
93166 NOISY LE GRAND CEDEX

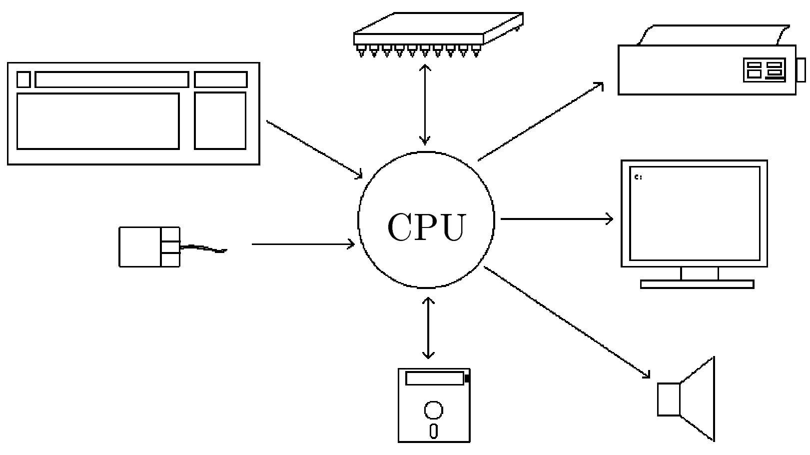
I. L'ordinateur

Un ordinateur est une machine de traitement de données.

Ces données peuvent être de natures variées et l'utilisateur doit pouvoir les communiquer à l'ordinateur et lui donner les ordres qui correspondent aux traitements qu'il désire faire sur ces données. Enfin, l'ordinateur doit être capable de lui retourner les résultats qu'il a obtenus.

La structure même de l'ordinateur découle de cette idée simple. On passe en revue globalement les composantes d'un ordinateur :

- Le **clavier** représente un moyen de communication avec la machine : dans le sens homme vers machine. La **souris** est un autre moyen de communication plus simple mais très performant. Ce sont des **périphériques d'entrée**.
- L'**écran** est un des dispositifs permettant la communication dans le sens machine vers homme. C'est par cet appareil que la machine vous "parle". On dit que c'est un **périphérique de sortie** au même titre que l'imprimante, la table traçante, le haut-parleur,...
- Les **unités de stockage** – disque dur, bande magnétique, disquette,... – représentent l'équivalent d'une bibliothèque. Un nombre souvent impressionnant de données y est stocké. Mettre hors tension l'ordinateur n'affecte pas le contenu de ces unités de stockage que l'on appelle aussi **mémoires de masse**.
- La **mémoire** de l'ordinateur, appelée **mémoire centrale** – RAM –, correspond plus à votre mémoire, celle dans laquelle votre cerveau extrait les informations dont il a besoin à un instant donné. Cependant une petite différence réside dans le fait que lorsque l'on éteint l'ordinateur, on perd à jamais le contenu de la mémoire centrale!
- Et pour **contrôler** tous ça, le **processeur**. A l'écoute des ordres qu'on lui donne via le clavier, il cherche ses instructions et les données à traiter dans la mémoire, exécute les instructions sur les données et peut par exemple afficher des résultats sur l'écran, ou simplement stocker les résultats en mémoire centrale ou en mémoire de masse.
- Pour que tout ces éléments communiquent les uns avec les autres – par exemple lorsque le processeur va chercher une information dans la mémoire centrale –, ils sont reliés par des fils électriques que l'on appelle des **bus** dans lesquels circulent les informations. Une **horloge interne** synchronise le tout.



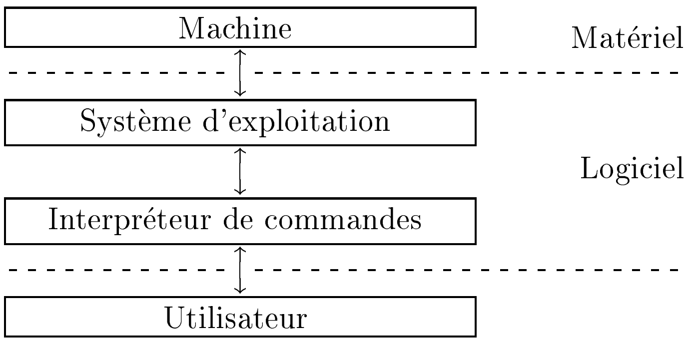
II. Un système d'exploitation

Il "cache" la vérité à propos de la structure matérielle de la machine, en tentant de rendre chaque entité qui compose un ordinateur comme quelque chose de facile à utiliser. Par exemple, lorsque l'utilisateur souhaite accéder à un fichier sur disquette, le système d'exploitation se charge de faire tourner la disquette à la bonne vitesse, d'y localiser les données, de positionner les têtes de lecture, d'amener les données en mémoire centrale et finalement d'arrêter le moteur du lecteur de disquettes.

MS-DOS se charge de toutes ces opérations, sans que l'utilisateur n'ait à savoir ce qu'il s'est réellement passé; ainsi les utilisateurs ne connaissant rien à la véritable structure de la machine peuvent quant même l'exploiter de façon convenable.

III. L'interpréteur de commandes

L'interpréteur de commandes de MS-DOS est un programme (`COMMAND.COM`) qui est chargé d'analyser les commandes tapées par l'utilisateur*, de les "comprendre", puis de les exécuter. Notez que l'interpréteur de commandes est une "couche logicielle" permettant de rendre plus facile l'accès au système d'exploitation.



* sans distinction entre minuscules et majuscules.

IV. Le dialogue avec l'ordinateur

On parle de dialogue avec l'ordinateur, mais quelle langue parle-t-il ? En fait, en tout et pour tout il ne comprend qu'un seul langage, que l'on appelle le **langage machine**. Ce langage est trop "primaire" pour être utilisable, d'où l'idée d'avoir recours à un autre langage qu'on traduira en langage machine.

Toute communication de "haut niveau" repose sur ce principe. "Programmer" par exemple dans un langage de haut niveau tel que le Pascal ou le C, c'est donner des instructions à la machine. Ces instructions seront alors analysées puis traduites en langage machine – par un **compilateur** – et enfin exécutées.

De la même manière, l'interpréteur de commandes se charge de comprendre des instructions dans un langage "assez naturel", et il donne ensuite les ordres à l'ordinateur en langage machine via le système d'exploitation.

Tout le problème, c'est de se comprendre !!!

V. L'arborescence de fichiers

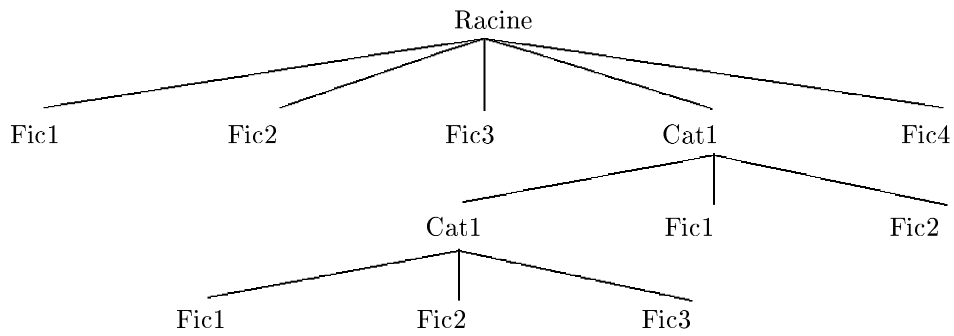
Sur les unités de sauvegarde – disque dur, disquette,..–, les données sont stockées sous forme d'entités bien séparées les unes des autres : ce sont les **fichiers**. Le premier exemple que l'on peut vous donner comme fichier, est le programme `command.com` que nous avons vu tout à l'heure. Mais, vous trouverez bien d'autres "types" de fichiers, qui contiendront par exemple vos données, votre c.v., vos cours..., et même nécessairement le texte de ce mémento...

Sous MS-DOS, les noms des fichiers ont **8 caractères** au maximum, plus une **extension de 3 caractères au plus**. Certains caractères ne sont pas admis tels que le "." ou encore le ":" par exemple.

On a l'habitude d'associer à chaque "type" de fichier, une extension qui lui est propre. C'est ainsi que l'on reconnait facilement un programme écrit dans le langage Pascal car le nom du fichier a l'extension "`pas`". Mais ce n'est pas l'unique exemple et on trouve ci-dessous une liste non exhaustive d'extensions que l'on rencontre habituellement :

- `.ASM` : Programme source assembleur
- `.BAK` : Sauvegarde
- `.BAS` : Programme source BASIC
- `.BAT` : Fichier BATCH
- `.COM` : Fichier de commande éxécutable
- `.EXE` : Fichier de commande éxécutable
- `.OBJ` : Fichier binaire objet
- `.SYS` : Fichier de configuration du système d'exploitation
- `.PAS` : Programme source PASCAL
- `.C` : Programme source C
- `.DOC` : Fichier de documentation
- `.TXT` : Fichier de données textuelles

Le système d'exploitation présente les fichiers sous forme d'**arborescence**. L'utilisateur peut alors évoluer "intuitivement" au sein de cette arborescence. Cette arborescence peut être créée, modifiée voire effacée par l'utilisateur. On présente ci-dessous un exemple d'arborescence :



VI. Références

Une référence est un nom permettant de situer un fichier dans une arborescence. Un chemin d'accès est une suite de noms de répertoires séparés par des \.

référence = unité:chemin–d'accès\nom–de–fichier

L'unité est optionnelle. Si elle est absente, on travaille sur l'arborescence de l'unité courante.

Ex: `b:\DOS\UTILS\FICHE.DAT` fait référence au fichier `FICHE.DAT` qui se trouve sur le lecteur `B`, répertoire `DOS`, sous-répertoire `UTILS`.

- Toute référence commençant par \ est une référence **absolue**, c'est-à-dire qu'elle part du sommet de l'arborescence (la racine).
- Toute autre référence est **relative**, elle commence au catalogue de travail.

Le répertoire **.** est le répertoire courant.

Par exemple, dans `\toto\titi\.\tutu`, `.` désigne le répertoire `\toto\titi`

Le répertoire **..** désigne le répertoire père du répertoire courant.

Par exemple, la référence `\toto\titi\..\.\titi\tutu` est équivalente à `\toto\titi\tutu`.

VII. Commandes principales de MS-DOS

VII.1. Positionnement sur un volume

Il suffit de taper '<x>:' où <x> est un des volumes accessibles (A,B,C...)

VII.2. Changement du catalogue de travail

C'est la commande "`cd` *référence*". Bien sûr la référence peut être soit absolue, soit relative.

VII.3. Lister une référence

C'est la commande "`dir` [/*p*][/*w*][/*s*][<*référence*>]".

Si la référence est un fichier alors des informations diverses sur le fichier seront données.

Si la référence est un répertoire alors des informations diverses sur les fichiers contenus dans ce répertoire seront données.

Si aucune référence n'est donnée, le répertoire courant est pris par défaut.

VII.4. Copier un fichier

C'est la commande "`copy` *ref.source ref.destination*". Attention la référence source doit être la référence d'un fichier, pas d'un répertoire.

VII.5. Effacer un fichier

C'est la commande "`del` *référence*", où *référence* doit être celle d'un fichier et non d'un répertoire.

VII.6. Renommer d'un fichier

La commande "`rename` *anciennne_référence nouvelle_référence*". Elle ne permet pas de déplacer un fichier, mais seulement de changer son nom.

VII.7. Créer un répertoire

C'est la commande "`mkdir` *référence*".

VII.8. Effacer un répertoire

C'est la commande "`rmdir` *référence*".

VII.9. Préparer une nouvelle disquette

C'est la commande "`format` *unité de disque*". Attention, cette commande efface définitivement toutes les données précédemment contenues sur la disquette (A utiliser avec précaution !!).