

Le mode “ invite de commande ”

Pourquoi encore étudier les commandes du DOS ?.....	2
Les fichiers et répertoires.....	2
Noms des fichiers et des répertoires.....	2
Les attributs des fichiers.....	3
L’invite.....	3
La ligne de commande.....	3
L’interpréteur de commandes.....	4
Commandes internes.....	5
Commandes externes.....	5
Fichiers “.bat”.....	5
Les commandes du DOS.....	6
Commandes “ répertoires ”.....	6
Commandes “ Fichiers ”.....	8
Affichage, édition, tris, comparaisons de fichiers.....	9
Commandes “ disques ”.....	11
Le Path.....	12
Comment ajouter un répertoire supplémentaire dans le Path ?.....	13
Index.....	14

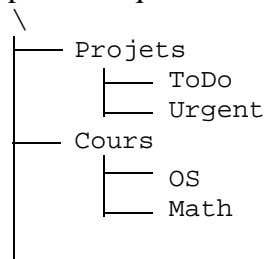
Pourquoi encore étudier les commandes du DOS ?

- ❑ Certaines commandes ne fonctionnent qu'en « mode DOS » appelé maintenant « mode invite de commande »
- ❑ Il est toujours nécessaire de savoir se servir d'une interface texte encore employé par certains programmes :
 - FDISK, FORMAT, commandes anciennes mais parfois encore utiles pour la partition et le formatage des disques,
 - PING, IPCONFIG, ARP, NET et d'autres commandes de ce genre pour la configuration d'un réseau
 - ou dans d'autres environnements tels que Linux en "mode terminal"
- ❑ Ces commandes reprennent les principaux *appels système* qui concernent les fichiers. C'est donc une bonne approche pour comprendre le rôle du système d'exploitation.
- ❑ En cas de gros problèmes elles permettent parfois de se dépanner en lançant l'outil de restauration du système à partir d'un démarrage sans échec avec l'invite de commande.
- ❑ Ces commandes sont essentielles pour faire des fichiers de commandes ".bat" ou ".cmd" bien utiles pour automatiser les tâches répétitives.

Les fichiers et répertoires

Les données enregistrées sur le disque s'y trouvent sous forme de documents appelés "fichiers". Initialement, le mot fichier signifie "collection de fiches". Pour un non initié, le terme prête à confusion car en informatique le mot fichier est utilisé pour désigner UN SEUL document. "Fichier" et "Document" sont ici des synonymes.

Les documents sont regroupés dans des répertoires (*directories*) aussi appelés dossiers. Ces derniers sont organisés selon une structure hiérarchique arborescente. Chaque répertoire peut contenir d'autres répertoires qui à leur tour peuvent contenir d'autres répertoires encore.



Noms des fichiers et des répertoires

Les fichiers et les répertoires ont des noms qui à l'époque du DOS avaient le format 8.3 Huit caractères suivis d'un point et de maximum trois caractères d'extension.

ex. : Chap1.txt

Ces noms ne pouvaient pas contenir de caractères accentués ni de caractères spéciaux tels que \$, %, #, l'espace etc. seul le caractère blanc souligné '_' était admis.

Ces restrictions ont disparu depuis Windows 95. Les espaces sont même autorisés dans les noms qui peuvent désormais être longs mais ces espaces sont toujours mal interprétés par les commandes du DOS. Il faudra mettre ces noms qui contiennent des espaces entre guillemets pour que l'interpréteur de commandes ne considère pas comme des noms de fichiers distincts.

ex : COPY "Lettre au Père Noël.pdf" C:\Courier\archives

Un nom tel que LettrePereNoel.pdf sera plus facile à manipuler pour l'écriture des scripts ou pour la portabilité du document vers d'autres systèmes. Recourez de préférence aux

conventions recommandées en programmation pour nommer des variables: une majuscule pour chaque nouveau mot.

Le répertoire racine (root), celui qui est à base du disque n'a pas vraiment de nom. On le désigne par le caractère back-slash '\

Le **nom complet** d'un fichier est son nom précédé du chemin complet qui permet de le retrouver depuis le répertoire racine.

ex. : C:\Cours\OS\Chap1.txt

C: est le nom du disque, le premier '\ est le sigle du répertoire racine, **Cours** est le nom d'un répertoire et **OS** celui de l'un de ses sous-répertoires, les '\ autres que le premier sont des séparateurs, et enfin **Chap1.txt** est un nom de fichier.

Notons aussi que DOS n'est pas sensible à la casse. Ce qui signifie non pas qu'il est particulièrement robuste mais que pour DOS les majuscules et les minuscules, c'est du pareil au même.

Les attributs des fichiers

Les attributs sont des informations qui sont enregistrées avec le nom du fichier.

R	<i>read only</i>	Lecture seule. Protégé contre toute modification
H	<i>hidden</i>	Caché ou plus exactement " masqué " car il est tout de même possible de demander leur affichage. (DIR /a)
S	<i>system</i>	Attribut réservé à certains fichiers sensibles
A	<i>archive</i>	Attribut pour organiser les sauvegardes. Cet attribut se positionne dès qu'on modifie le fichier. Les commandes de sauvegarde peuvent l'effacer dès que la sauvegarde est accomplie.

Ces attributs sont visibles sous Windows dans les propriétés des fichiers accessibles par un clic droit sur le nom du fichier, choisir ensuite la rubrique " Propriété " dans le menu contextuel.

L'invite

C:\>_

L'**invite** ou *prompt* du DOS est un message d'attente qui signale que le système est **prêt à recevoir une commande**. L'invite indique la position du répertoire actif : le nom du disque par défaut (ici **C:**) suivi du chemin complet depuis le répertoire racine représenté par le caractère " *back-slash* " '\

Le caractère d'attente '>' marque la fin du nom du répertoire courant.

Le **curseur** qui clignote indique où arriveront les caractères tapés au clavier.

La ligne de commande

Une ligne de commande est en quelque sorte une phrase qui se tape à la suite de l'invite.

Elle est de la forme suivante :

Commande [/options] [arguments] ↵

La ligne de commande est analysée par l'interpréteur de commandes. **Les espaces y séparent les mots des commandes et des arguments**. Prenez l'habitude de taper ces espaces, certaines versions du DOS tolèrent l'absence de certains espaces, d'autres pas. Si vous prenez l'habitude de ne pas taper ces espaces, vous risquez, en changeant de machine, de ne pas

pouvoir faire exécuter une commande que pourtant vous avez l'habitude de voir fonctionner ailleurs.

- Une ligne de commande commence toujours par un **mot clé**, ce mot représente une action que le système d'exploitation est appelé à faire. Le grammairien Mr Grevisse appellerait cela un verbe.

- Viennent ensuite les **options** parfois aussi appelées paramètres ou commutateurs. Elles sont facultatives, elles sont toujours représentées par un " slash " suivi d'une lettre. La fonction de l'option est de modifier la manière dont la commande va se réaliser. S'il vous arrivait de devoir expliquer ce rôle à un linguiste, dites-lui que c'est un adverbe.

- Les **arguments** viennent en complément. Dans la présentation de la syntaxe des commandes ces arguments sont mis entre crochets pour indiquer que leurs présences ne sont pas toujours indispensables.

Exemples :

```
1) C:\>VER
Windows 98 [Version 4.10.222]
C:\>
2) C:\>CLS
C:\>
3) C:\>CLS /?
Efface l'écran
C:\>
```

```
4) C:\>Edit Chap1.txt
5) C:\>Copy Chap1.txt Chap1.old
   1 fichier(s) copié(s)
C:\>
6) C:\> Time /t
20:26
C:\>
```

- 1) La commande **VER** sert à demander la version du système d'exploitation ne nécessite aucun argument.
- 2) De même la commande **CLS** (*clear screen* – effacer l'écran) n'a besoin d'aucun argument.
- 3) Nous avons ici utilisé la commande **CLS /?** pour demander l'aide à son propos. Cette option **/?** est disponible avec toutes les commandes du DOS. L'explication de la commande **CLS** tient en une seule ligne, mais avec d'autres commandes cette explication fait parfois plusieurs pages.
- 4) La commande **Edit Chap1.txt** lance le programme Edit qui est livré avec le DOS. L'argument de la commande est le nom du fichier texte à éditer.
- 5) La commande **COPY** est utilisée ici avec deux arguments. L'un indique le fichier source, le second indique le nom que doit prendre la copie.
- 6) La commande **TIME** est utilisée dans cet exemple avec l'option **/t** qui pour cette commande signifie qu'on ne veut pas modifier l'heure mais simplement l'afficher. Attention, une même lettre d'option (ici **/t**) peut avoir une signification différente d'une commande à l'autre.

L'interpréteur de commandes

L'interpréteur de commandes est un programme qui se nomme **COMMAND.COM** si vous travaillez avec une version DOS ou avec Windows 9x. Les versions ultérieures de Windows renferment un second interpréteur **CMD.EXE** dont les commandes ont été revisitées. Certaines commandes sont apparues ou ont disparues d'une version à l'autre. Ainsi la commande **HELP** a disparu dans Windows 98 mais est à nouveau disponible avec les versions ultérieures.

L'interpréteur de commandes **CMD.EXE** accepte les mêmes commandes que l'interpréteur **COMMAND.COM** mais autorise une plus grande variété d'options pour chacune d'elles. Ces potentialités nouvelles sont appelées les **extensions de commandes**. Grâce à elles les manipulations de variables sont plus aisées et l'écriture de fichiers de commande s'en trouve

un peu moins rébarbative. On dispose d'un langage "DOS" plus ressemblant aux autres langages évolués. Il suffit pour s'en convaincre de demander l'aide sur la commande SET (tapez SET /?); suivant que la commande est interprétée par COMMAND.COM ou par CMD.EXE l'aide fait 9 ou 180 lignes.

Comment lancer l'un ou l'autre de ces interpréteurs ?

Avec Windows 9x, Command.com est le seul interpréteur.

- Démarrer > Programme > Commandes MS-DOS
- ou Démarrer > Exécuter > Command

Avec Windows 2000 ou Windows XP

- Démarrer > Programmes > Accessoires > Invite de commande
- ou Démarrer > Exécuter > Cmd (ce qui revient au même)
- ou Démarrer > Exécuter > Command

A moins de devoir écrire des fichiers de commandes devant être compatibles avec Windows 98, il n'y a aucune raison d'utiliser l'interpréteur COMMAND.COM

Commandes internes

Il s'agit d'une trentaine de mots clés intégrés à l'interpréteur de commandes. Ce sont les commandes les plus courantes telles que CD, CLS, COPY, DATE, DEL, DIR, ECHO, EXIT, MD, PATH, PROMPT, REN, RD, TIME, TYPE, VER ou VOL.

Un ordinateur qui a démarré à partir d'une disquette de démarrage contenant l'interpréteur COMMAND.COM est capable de comprendre toutes ces commandes.

Commandes externes

Ce sont les commandes que l'interpréteur ne sait pas exécuter à lui seul. Les commandes telles que COMP, DEBUG, EDIT, FDISK, FORMAT, LABEL, MEM, MODE, MORE, TREE ou XCOPY sont des commandes externes. Les routines qui exécutent ces commandes ne font pas partie du fichier COMMAND.COM; si l'ordinateur a démarré à partir d'une disquette de démarrage, il faudra que celle-ci contienne le fichier FORMAT.COM ou EDIT.COM pour que ces commandes externes soient disponibles. Ce sont des programmes livrés avec le DOS. Les programmes tels que DEBUG ou EDIT sont parfois appelés "utilitaires".

Fichiers ".bat"

Les commandes du DOS peuvent être enregistrées dans un fichier texte auquel on donne l'extension ".bat" ou ".cmd". Ces fichiers sont alors en quelque sorte des programmes interprétés, des "scripts".

Les commandes du DOS

Ce chapitre a pour objectif de classifier les commandes pour en donner une vue d'ensemble facile à appréhender. Nous n'allons pas reprendre ici l'intégralité des commandes du DOS. Tous les détails sur chaque commande sont disponibles en tapant le nom de la commande suivi de l'option " point d'interrogation " `/?` , la seule option qui est reprise par toutes les commandes.

La commande `HELP` qui existait pourtant dans les anciennes versions DOS a disparu dans les versions Windows 98 et Millénium pour réapparaître dans les versions ultérieures. Elle donne la liste de toutes les commandes du DOS avec une ligne d'explication pour chacune.

Commandes " répertoires "

DIR " Directory " Affiche la liste des fichiers et sous répertoires

La commande `DIR` possède de nombreuses options. Tapez `DIR /?` pour les connaître. Les options `/a` , `/p` et `/s` sont particulièrement utiles.

`/a` est utile pour afficher les fichiers cachés.

`/p` l'affichage s'interrompt dès que l'écran est plein.
Appuyez sur une touche pour continuer ...

`/s` provoque aussi l'affichage dans les sous-répertoires.

Voici comment lancer la recherche du fichier `PERDU.TXT` dans tous les sous-répertoires du disque C.

```
C:\>DIR /s Perdu.txt
```

La commande `DIR` est la commande idéale pour s'exercer avec les caractères génériques. L'étoile représente une suite quelconque de caractères, le point d'interrogation ne se substitue qu'à un seul caractère.

Exemples :

<code>DIR</code>	Affiche tous les noms de fichiers et de sous-répertoires présents dans le répertoire courant
<code>DIR \Prj</code>	Affiche tous les noms de fichiers ou sous-répertoires présents dans le répertoire <code>\Prj</code>
<code>DIR *.txt</code>	Affiche tous les noms de fichiers ou de sous-répertoires dont l'extension est " txt " dans le répertoire courant
<code>DIR \Prj*.txt</code>	Affiche tous les noms de fichiers ou sous-répertoires dont l'extension est " txt " dans le répertoire <code>\Prj</code>
<code>DIR *.*</code>	Affiche tous les noms de fichiers ou sous-répertoires qui n'ont pas d'extension.
<code>DIR ABC??.*</code>	Affiche tous les noms commençant par ABC suivi de deux autres lettres quelle que soit l'extension
<code>DIR ABC*.*</code>	Affiche tous les noms commençant par ABC suivi d'un nombre quelconque de caractères et quelle que soit l'extension.
<code>DIR /s /p *.exe</code>	Recherche de tous les exécutables y compris dans les sous-répertoires, affichage pages par pages.
<code>DIR /a</code>	Affiche tous les noms de fichier y compris ceux des fichiers cachés.
<code>DIR /ah</code>	Affiche uniquement les noms des fichiers cachés

TREE Affiche l'arborescence des sous-répertoires

(La commande `TREE` n'est pas disponible avec Windows 98 ou Millénium)

CD “Change Directory” = Changer de répertoire

```
C:\>CD Windows  
C:\Windows>
```

⇒ Le répertoire \Windows devient le répertoire par défaut

Cas particuliers :

CD .. = allez au répertoire parent ; les deux points “..” équivalent à

CD \ = allez au répertoire racine

MD ou **MKDIR** “Make directory” = Créer un répertoire

Créer un sous-répertoire appelé Fred à la racine du disque C

```
C:\>MD Fred  
C:\>
```

Créer un sous-répertoire "Projets" dans le répertoire actif de la disquette



```
C:\>MD A:Projets  
C:\>
```

Créer un répertoire appelé “Nouveau Dossier”, il faut mettre ce nom entre guillemets !

```
C:\>MD "Nouveau Dossier"  
C:\>
```

Attention ! Les espaces sont pris par DOS pour des séparateurs.

Sans ces guillemets voici ce que cela donnerait en DOS “pur” :

```
C:\>MD Nouveau Dossier  
Trop de paramètres - Dossier  
C:\>
```

La commande sans guillemets serait acceptée par les versions plus récentes de Windows mais la commande créerait deux répertoires “Nouveau” et “Dossier”

RD ou **RMDIR** “Remove directory” - Supprimer un répertoire

Cette commande ne fonctionne que si le répertoire est vide.

Le répertoire racine “\” ne saurait être supprimé.

NB. Une commande externe DELTREE est fournie avec certaines versions du DOS pour supprimer d’un coup tous les fichiers et sous-répertoire en une seule commande.

Commandes “ Fichiers ”

COPY

COPY *source destination*

Copie d'un fichier source vers une autre destination ou sous un autre nom

Source est un nom de fichier.

Destination est soit un nom de fichier (le nom de la copie) avec ou sans chemin complet, soit un nom de répertoire ou d'unité où doit aboutir la copie qui aura le même nom que le fichier source.

Exemples :

```
C:\Test>COPY \Windows\Readme.txt A:\
```

Source : le fichier Readme.txt du répertoire C:\Windows

Destination : A:\Readme.txt

```
C:\Test>COPY A:\Readme.txt
```

Source : le fichier Readme.txt en provenance de A:\

Destination non précisée donc dans le répertoire actif ⇒ C:\Test\Readme.txt

MOVE

MOVE *Source Destination*

Déplace un fichier.

Cette commande qui n'existait pas en DOS “pur” peut aussi servir à renommer un fichier ou un répertoire.

REN ou RENAME

REN *Nom1 Nom2*

Commande pour renommer un fichier, un répertoire ou un groupe de fichiers ou de répertoires.

Exemples :

```
C:\Test>Ren Readme.txt Lisezmoi.txt
```

```
C:\Test>Ren *.txt *.old
```

Change toutes les extensions des fichiers “.txt” en “.old”

DEL “ Delete ” ou ERASE

Suppression de fichiers

Exemples :

DEL Readme.txt	Supprime le fichier Readme.txt
DEL Temp*.*	Supprime tous les fichiers du répertoire Temp
DEL *.bak	Supprime tous les fichiers d'extension “.bak”
DEL /S Fred*.txt	Supprime tous les fichiers .txt du répertoire Fred et de ses sous-répertoires. NB. Il serait prudent de visualiser au préalable tous les noms des fichiers qui seraient supprimés par la commande fort semblable : DIR /S Fred*.txt

ATTRIB

Affiche ou modifie des attributs de fichier.

Exemples :

ATTRIB \Temp	Affiche les attributs et les noms de tous les fichiers du répertoire Temp (= ATTRIB \Temp*.*)
ATTRIB	Affiche les attributs et les noms de tous les fichiers du répertoire courant. (= ATTRIB *.*)
ATTRIB +R Toto.txt	Met le fichier " Toto.txt " en lecture seule
ATTRIB +h Secret.txt	Masque le nom du fichier " Secret.txt "
ATTRIB -h -s Truc.sys	Enlève les attributs système et caché du fichier Truc.sys

XCOPY Copies de fichiers ou de répertoires

XCOPY *Source Destination*

Cette extension de la commande COPY permet aussi bien la copie de fichiers que de répertoires. Elle dispose de très nombreuses options.

- Les copies de répertoires nécessitent les options /S , /T et /E

/S Copie les répertoires et sous-répertoires à l'exception des répertoires vides.

/E Copie les répertoires et sous-répertoires, y compris les répertoires vides.

/T Crée la structure de répertoires mais ne copie pas les fichiers. N'inclut pas les répertoires ou sous-répertoires vides.

/T /E Crée la structure de répertoires sans recopier les fichiers en incluant les sous répertoires vides.

- Quelques options utiles pour les sauvegardes :

/A Copie uniquement les fichiers ayant l'attribut archive, ne modifie pas l'attribut.

/M Copie uniquement les fichiers ayant l'attribut archive, désactive l'attribut archive.

/D:j-m-a Copie les fichiers modifiés à partir de la date spécifiée. Si aucune date n'est donnée, copie uniquement les fichiers dont l'heure source est plus récente que l'heure de destination.

/U Copie seulement les fichiers dont une version antérieure existe déjà dans la destination.

Affichage, édition, tris, comparaisons de fichiers

TYPE

TYPE *NomFichier*

Affiche à l'écran tout le contenu d'un fichier texte.

MORE

MORE *NomFichier*

Affiche le fichier écran par écran. L'affichage s'interrompt dès qu'un écran est complet, il faut presser la touche espace pour passer à l'écran suivant.

FIND

FIND "*chaîne*" *NomFichier*

Recherche une chaîne de caractères dans un ou des fichiers.

Exemples :

<code>find "ABCD" *.txt</code>	Recherche tous les fichiers .txt qui contiennent la chaîne de caractère ABCD
<code>find /i "ABCD" *.txt</code>	Idem en ignorant la distinction majuscules / minuscules

FC *File Compare*

FC Fichier_1 Fichier_2

Compare les deux fichiers et affiche leurs différences

Commandes “ disques ”

Les disques sont désignés par une lettre suivie de deux points. A: et B: sont réservés aux disquettes, C: et suivants désignent le disque dur ou des lecteurs réseaux.

(Un lecteur réseau est un dossier partagé sur le réseau auquel on a attribué une lettre pour s'y référer comme s'il s'agissait d'un disque)

Changement de disque actif

Tapez la lettre qui représente le lecteur suivi de deux points.

Le résultat apparaît dans l'invite.

```
C:\Projets\>A:
A:\>C:
C:\Projets\>
```

LABEL

Permet de donner un nom (une étiquette) à un disque ou une disquette. Ce nom ne peut comporter que 11 caractères maximum.

```
C:\Projets\>LABEL A: Octet
```

VOL

Affiche le nom du disque et son numéro de série

```
C:\>vol A:
Le volume dans le lecteur A s'appelle OCTET
Le numéro de série du volume est 3C1A-0926
```

FORMAT

La commande Format sert à préparer un disque ou une disquette pour que le système d'exploitation puisse l'utiliser ensuite. Concrètement, cela veut dire que le support est divisé en pistes et en secteurs. Format y organise ensuite le secteur d'amorçage, le répertoire racine, la FAT et une copie de la FAT.

```
C:\>format a:
Insérez un nouveau disque pour le lecteur A:
et appuyez sur ENTRÉE lorsque vous êtes prêt...
Le type du système de fichiers est FAT.
Vérification de 1,44 Mo
Initialisation de table d'allocation des fichiers (FAT) en
cours...
Nom du volume (11 caractères, Entrée pour ne rien mettre) ?
TEST_DOS
Formatage terminé.

1.457.664 octets d'espace disque au total.
1.457.664 octets disponibles sur le disque.

512 octets dans chaque unité d'allocation.
2.847 unités d'allocation disponibles sur le disque.

12 bits dans chaque entrée FAT.

Le numéro de série du volume est 2823-5867

Formater un autre disque (O/N) ? n

C:\>
```

SYS

La commande SYS n'est plus reconnue par les versions récentes de Windows. Sous DOS et Windows 9x, elle servait à créer des disques ou disquettes système. Une disquette système est une disquette qui contient l'interpréteur de commandes et deux fichiers systèmes cachés.

Exemple réalisé sous Windows 98 :

```
C:\>sys a:
Système transféré
C:\>dir a: /a
  Le volume dans le lecteur A est OCTET
  Le numéro de série du volume est 2823-5867
  Répertoire de A:\
IO          SYS          222 390   05/05/99   22:22 IO.SYS
DRVSPACE   BIN           69 127   05/05/99   22:22 DRVSPACE.BIN
MSDOS      SYS              6  14/11/03   5:02 MSDOS.SYS
COMMAND    COM           95 874   05/05/99   22:22 COMMAND.COM
           4 fichier(s)          387 397 octets
           0 répertoire(s)       1 068 544 octets libres
C:\>
```

SUBST

SUBST assigne un nom de volume à un répertoire.

```
C:\>subst E: C:\Documents and settings\Fred\Mes Documents
C:\>
```

Du coup la commande **E:↵** équivaut à

```
CD C:\Documents and settings\Fred\Mes Documents
```

Le Path

Le PATH ou "chemin de recherche" définit la liste des répertoires où l'interpréteur de commandes doit chercher les fichiers exécutables.

Voici par exemple ce que l'on pourrait obtenir en tapant la commande PATH

```
C:\>path
PATH=C:\WINDOWS\system32;C:\WINDOWS;C:\DJGPP\BIN
C:\>
```

La commande PATH employée sans paramètre rend une liste des chemins vers différents répertoires. Des points-virgules jouent le rôle de séparateurs entre les répertoires de la liste.

L'interpréteur de commandes, quand il recherche un fichier exécutable, commence par consulter le répertoire courant ; si le fichier recherché ne s'y trouve pas, la recherche se poursuit en scrutant dans l'ordre les répertoires successifs indiqués par le PATH.

Voici ce que serait la réponse si la commande invoquée ne correspond ni à une commande interne ni à un fichier exécutable :

```
C:\>xyz
'xyz' n'est pas reconnu en tant que commande interne
ou externe, un programme exécutable ou un fichier de commandes.
C:\>
```

Les fichiers exécutables ont pour extension **.com**, **.exe** ou **.bat** et c'est dans cet ordre que l'interpréteur les considère. Autrement dit, si le répertoire courant contient un fichier *xyz.com* et un fichier *xyz.bat*, la commande *xyz* provoquera l'exécution de *xyz.com* au lieu de celle de *xyz.bat* (moins prioritaire)

Comment ajouter un répertoire supplémentaire dans le Path ?

Fort de votre expérience en mode invite de commande, vous décidez d'installer quelques utilitaires dans un répertoire C:\Util que vous venez de créer. Le chemin de ce répertoire doit être ajouté au PATH.

```
C:\>PATH %PATH%;c:\Util
C:\>
```

Cette modification du PATH n'est cependant valable que pour la session DOS courante. Il faudrait insérer cette commande dans le fichier AUTOEXEC.BAT pour qu'elle soit exécutée à chaque démarrage.

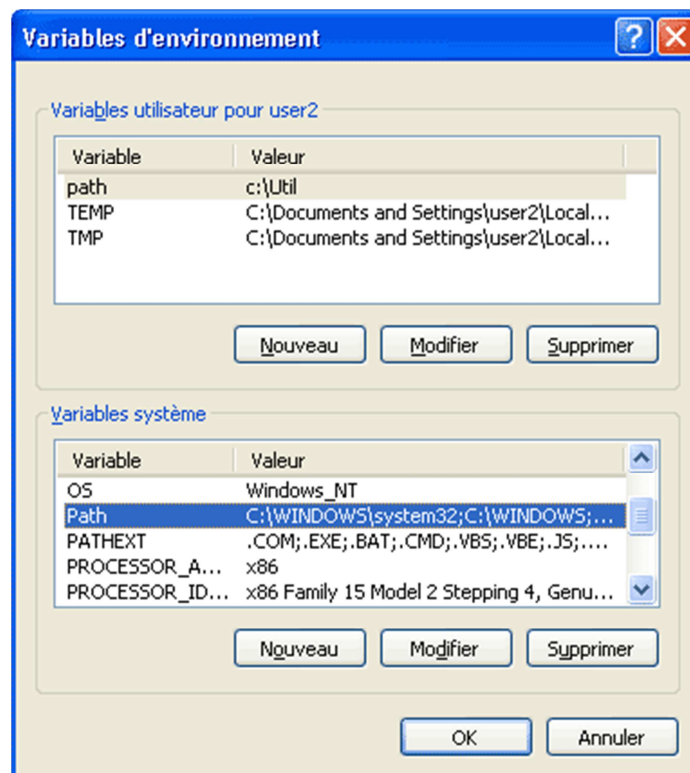
Remarque PATH est à la fois une commande et le nom d'une variable d'environnement. Elle aurait pu être modifiée comme suit :

```
C:\>SET PATH=%PATH%;c:\Util
C:\>
```

Sous Windows 2000 ou XP, puisque Autoexec.bat ne s'utilise plus, la procédure classique pour modifier la variable d'environnement PATH est la suivante :

Clic-droit sur poste de travail > Propriété (ou Panneau de configuration > Système) > Onglet avancé > bouton Variables d'environnement...

Modifiez la **variable système** PATH si en temps qu'administrateur vous y êtes autorisé. Si vous n'êtes pas administrateur, ou si cette modification ne concerne personne d'autre que vous, ajoutez ou modifiez la **variable utilisateur** PATH



Index

.		
.BAT	5	
C		
CD.....	7	
CLS	4	
Cmd.exe	4	
Command.com	4	
Commandes externes	5	
Commandes internes	5	
COPY	8	
D		
DEL	8	
DIR	6	
F		
FC	10	
FIND	9	
FORMAT	11	
I		
Interpréteur de commandes	4	
Invite	3	
L		
LABEL	11	
Ligne de commande	3	
M		
MD	7	
MKDIR	7	
MORE	9	
MOVE	8	
P		
Path	12	
Prompt.....	3	
R		
RD	7	
REN	8	
RENAME	8	
RMDIR	7	
S		
SUBST	12	
SYS	12	
T		
TREE	6	
TYPE	9	
V		
VER	4	
VOL	11	
X		
XCOPY	9	