

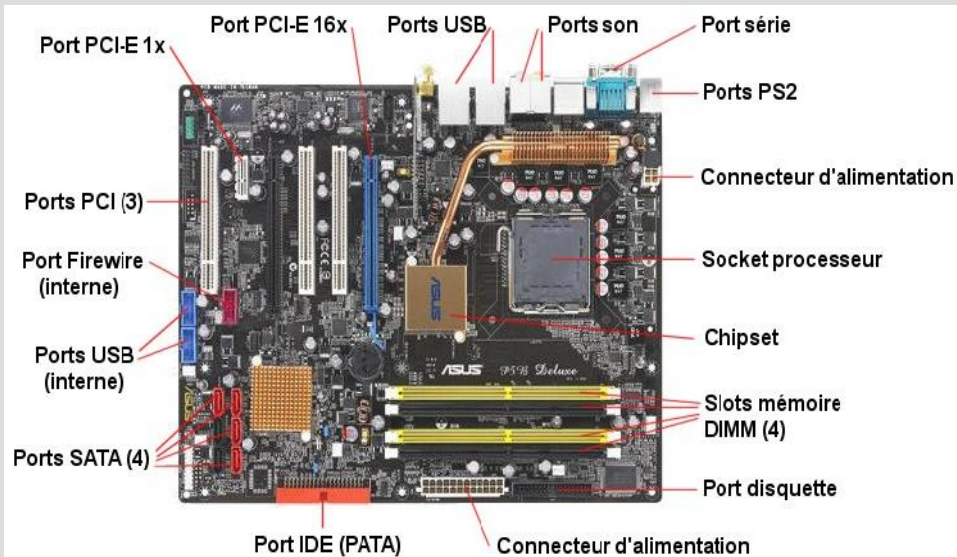
# **Systèmes d'exploitations Windows**

EHTP  
Année 2017/2018

## **Plan**

- I. Introduction**
- II. MS Dos**
- III. Variables d'environnement**
- IV. Les scripts batchs**
- V. Windows server**

## Carte mère d'un ordinateur(1)



## Carte mère d'un ordinateur(2)

- **La carte mère:**

La carte mère permet d'interconnecter tous les composants de l'ordinateur, elle en conditionne les possibilités d'extension.

- **Le socket:**

C'est lui qui recevra le processeur et servira à faire l'interface avec la carte mère.

- **Le chipset:**

Il gère les flux de données entre le processeur et les autres composants via le Front Side Bus (FSB).

- **Les slots mémoire:**

Destinée à recevoir les barrettes de mémoire.

- **Le processeur**

Le processeur est un circuit électronique complexe qui exécute chaque instruction,

## Carte mère d'un ordinateur(3)

### ▪ Les ports:

- ✓ Port PCI (Peripheral Component Interconnect) : permet de connecter des cartes d'extension sur la carte mère via un bus PCI (cartes réseau, son, graphique) débit théorique 8 Go/s en 32 bits
- ✓ Ports SATA (Serial Advanced Technology Attachment) pour connecter une mémoire de masse (typiquement un disque dur ou un lecteur de CD/DVD) à la carte mère : 6 Gbits/s pour SATA 3
- ✓ Ports série : transmettent des données en série (bit à bit), dont notamment le Universal Serial Bus (USB) : débit 4,8 Gbit/s pour USB 3

## Carte mère d'un ordinateur(4)

### ▪ Les différents types de mémoires:

#### ✓ La mémoire vive RAM (Random Access Memory)

- Mémoire dans laquelle on peut lire et écrire.
- Mémoire *volatile* (perd son contenu dès la coupure du courant).

#### ✓ La mémoire morte ROM (Read Only Memory)

- Mémoire dans laquelle on ne peut que lire.
- Mémoire permanente (conserve indéfiniment son contenu).
- Contient le BIOS,

# Introduction

## Systèmes d'exploitations

angl. « Operating System (OS) »

Qu'est-ce que c'est?

« Programme assurant la gestion de l'ordinateur et de ses périphériques »

[[www.dicofr.com](http://www.dicofr.com)]

A quoi ca sert?

- à simplifier la vie des utilisateurs et des programmeurs
- à gérer les ressources de la machine d'une manière efficace

## Abstraction

Cacher la complexité des machines pour l'utilisateur afin d'utiliser la machine sans savoir ce qui est derrière

Abstraction du terme « Machine » selon Coy:

- machine réelle = Unité centrale + périphériques
- machine abstraite = machine réelle + système d'exploitation
- machine utilisable = machine abstraite + application

## Exigences à un Système d'exploitation

### 澁 Généralités

- 澁 Satisfaire les utilisateurs et les programmeurs
- 澁 Gérer 2D, 3D, vidéo, audio, réseau, CD, DVD, clé USB, ...
- 澁 Plusieurs utilisateurs (itinérants) --> multi-utilisateurs
- 澁 être extensible

### 澁 De plus en plus gros et complexe :

- 澁 Efficace, évolutif, maintenable

## Exigences de l'utilisateur

灬 « Faut que ça marche ! »  
(comme j'en ai envie ...)

灬 « Ça imprime pas ... »

灬 = Machine utilisable (machine étendu)

## Exigences du programmeur

灬 Simplifier l'accès aux ressources de la machine :

灬 Mémoire, processeur, périphériques, fichiers, programmes,  
réseaux, communication interne

灬 Modèle de programmation simple et unifié

灬 Efficacité dans tous les cas

灬 = Machine étendue

## Quelques définitions

Processus

Traitement par lots

Systèmes Multi-tache

Systèmes Multi-utilisateurs

Systèmes Multi-processeurs

Systèmes temps réel

Systèmes distribués

## Définitions: Processus

Déf.:

Un processus est un programme lors de l'exécution

(aspect dynamique d'un programme)

## Définitions: Traitement par lots (Batch processing)

Un utilisateurs donne plusieurs commandes (« Jobs ») dans une queue d'exécution de programmes

Entièrement séquentielle

p.ex. pour faire plusieurs calculs pendant la nuit

p.ex. autoexec. bat

## Définitions: Systèmes Multi-tache (Multitasking)

Assurer l'exécution de **plusieurs programmes** en **même temps (c-à-d. plusieurs processus)**

Chaque processus a besoin du processeur

- situation concurrente
- solution: « scheduling »



## Définitions: Systèmes Multi-processeurs

système avec plusieurs processeurs

- parallèle
- vrai multi-tache
- doit assurer qu'il y a l'exécution d'autant de processus que processeurs en même temps

contrairement: système avec un seul processeur

- quasi-parallèle
- arrêter et reprendre les différents processus

Gestion avec le « scheduler » (ordonnancement des processus)

## Définitions: Systèmes Multi-utilisateurs (« time-sharing »)

permettre à **différentes personnes** de travailler avec **un ordinateur en même temps**

connexion par

- via le terminal de l'ordinateur lui-même
- à distance (telnet, ssh, ftp, ...)

donner l'impression à chaque utilisateur qu'il est seul

exige une gestion des droits

- de fichiers (pour éviter la destruction des fichiers etc.)
- de processus

## **Définitions: Multi-utilisateurs**

Login

Type:

- Administrateur (« root »)
- Groupes
- Utilisateurs

pour gérer les droits

## **Définitions: Systèmes Temps réels**

Sert pour le pilotage et le contrôle des déroulements externes (p.ex. centrale électrique)

doit garantir des temps de réactions données pour des signaux extérieur urgents

plusieurs systèmes d'exploitations n'y arrivent pas car l'interruption de certaines activités met le système dans un état instable

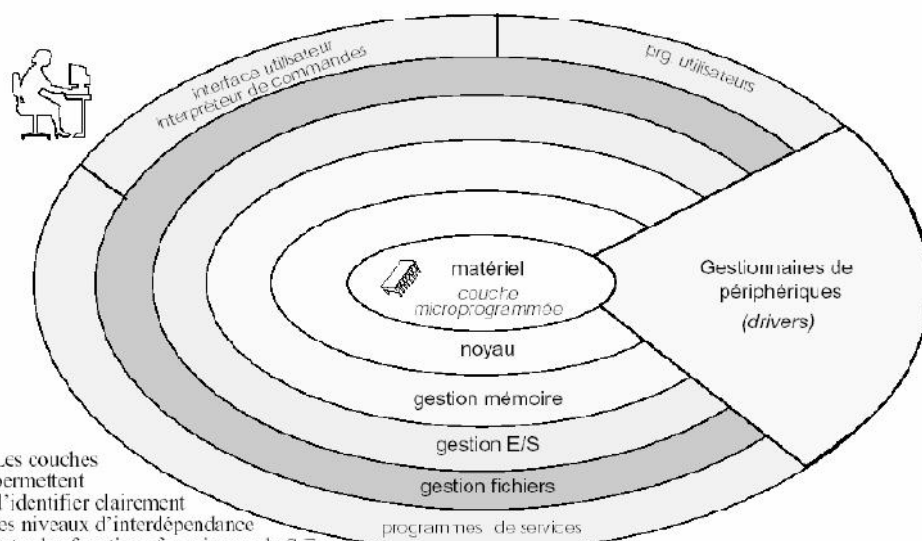
## Définitions: Systèmes distribués

doit permettre l'exécution d'un **seul programme** sur **plusieurs machines**

distribuer les processus et les remettre ensemble

pour gros calculs, p.ex. inversion de grandes matrices

## Composition



## Noyau (kernel)

Le noyau assure ces fonctionnalités :

- gestion des périphériques (au moyen de pilotes)
- gestion des files d'exécution (aussi nommée processus) :
  - attribution de la mémoire à chaque processus
  - ordonnancement des processus (répartition du temps d'exécution sur le ou les processeurs)
  - synchronisation et communication entre processus (services de synchronisation, d'échange de messages, mise en commun de segments de mémoire, etc.)
- gestion des fichiers (au moyen de systèmes de fichiers)
- gestion des protocoles réseau (TCP/IP, IPX, etc.)

## Noyau

Au lancement:

- *Boot*: le programme qui initialise l'ordinateur
- Permet aux premiers services d'accéder aux applications système : gestion de la mémoire, accès aux disques durs et accès aux périphériques

Mémoire centrale (vive): répartition

- *Réservée pour le système*
- *Réservée pour les applications*

Assure une indépendance entre les applications et les matériels

## Les bibliothèques dynamiques

= *Libraries*

Regroupent les opérations souvent utilisées, selon les fonctionnalités (E/S, fichier, ...)

Ces opérations sont disponibles pour être appelées et exécutées par d'autres programmes

## Le système de fichiers

Un système de fichiers est une structure de données permettant de stocker les informations et de les organiser dans des fichiers sur des mémoires secondaires (disque dur, disquette, CD-ROM, clé USB, etc.)

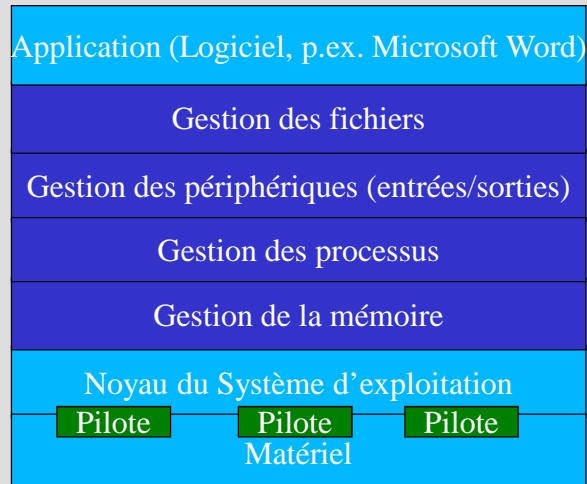
- E.g. une structure hiérarchique
- /u/nie/HTML/IFT6800/introduction.txt

Ce stockage de l'information est persistant

Une telle gestion des fichiers permet de traiter, de conserver des quantités importantes de données ainsi que de les partager entre plusieurs programmes informatiques

Il offre à l'utilisateur une vue abstraite sur ses données et permet de les localiser à partir d'un chemin d'accès

## SE: Modèle en couches



## Historique (avant les Systèmes d'Exploitations)

1945 - 55 : tubes et interrupteurs

⚠ Pas de système d'exploitation

1955 - 65 : transistors, cartes perforées

⚠ Traitement par lots

1965 - 80 : circuits intégrés, disques

⚠ Multiprogrammation, temps-partagé, entrées/sorties

⚠ Unix, version BSD, AT&T, interface POSIX

1980 -- : ordinateurs personnels (PC)

⚠ Interface graphique (concept crée vers 1960, Stanford)

⚠ Réseaux et systèmes distribués

--> Système d'exploitation nécessaire

## Systèmes d'exploitations

CP/M (depuis 1974), Digital Research

UNIX (depuis 1969-1979), premier par AT&T

MS-DOS (depuis 1981), Microsoft

MacOS (depuis 1984), Apple

Windows (depuis 1991), Microsoft

Linux (depuis 1992), OpenSource

## Systèmes d'exploitations

CP/M (depuis 1974), Digital Research

- Gestion de disque dur, mais pas d'arborescence
- Pas de graphisme
- Exemple:
  - CPU 8088, 2 MHz
  - 64 KO de RAM
  - 5 MO de disque dur
- cf. la loi de Murphy

## Systèmes d'exploitations

UNIX (depuis 1969-1979), AT&T

- a servi de modèle pour MS-DOS, Windows, ..
- Multi-tâche et Multi-utilisateurs
  - accès simultané aux fichiers, périphériques, mémoire, processeurs, ..
- Protection mémoire : aucun programme ne peut faire planter le système
- systèmes de fichiers hiérarchique
- GUI X-Windows

## Systèmes d'exploitations

MS-DOS (depuis 1981), Microsoft



## Systèmes d'exploitations

MacOS (depuis 1984), Apple

- premier GUI

## Systèmes d'exploitation Windows

Windows 3.11

- pas de multitâche, pas de multi-utilisateurs

Windows 95

- **multi-tâche**
- premier système 32 bit

Windows 98

- Internet intégré dans le GUI
- Plug & Play

parallèlement Windows NT

- système d'exploitation réseaux **multi-utilisateur**

Windows 2000, Windows XP, Windows 7, Windows 8,,,,,

- jumelage entre système d'exploitations réseaux et « stand-alone »

## **Systèmes d'exploitations**

### **Linux (depuis 1992), OpenSource**

- finlandais Linus Thorwald
- Licence GPL (General Public Licence) – OpenSource
- Multi-tâche et Multi-utilisateurs
- Distributions
  - Red Hat
  - Fedore
  - S.u.S.e
  - Debian
  - Mandrake..

### **Ms Dos MicroSoft Disk Operating System**

## MS-DOS?

**Pourquoi faut-il encore connaître le MS-DOS?**

**Car il sera votre seul recours si Windows ne se lance pas...**

37

## Utiliser le mode MS-DOS

### **Qu'est-ce que MS-DOS ?**

Pour pouvoir gérer les fichiers et répertoires présents sur le disque, l'ordinateur a besoin d'un logiciel appelé : **système d'exploitation**.

Le plus connu actuellement étant Windows.

Le **DOS** (Disk Operating System) est un système d'exploitation qui a pour charge de gérer les unités de disques.

38

## Utiliser le mode MS-DOS

L'objectif de **MS-DOS** est d'optimiser l'utilisation de l'ordinateur de manière à réduire les temps d'exécution des programmes.

C'est ce travail qu'effectue le **DOS** en gérant les lecteurs de disquettes, les disques durs, la mémoire et en y ajoutant quelques fonctions utiles à l'utilisateur.

Chaque commande doit être validée par la touche **[Entrée]**.

39

## Comment est organisé votre ordinateur?

A chaque disque est attribué une lettre:

**A:** désigne le lecteur de disquette

**C:** désigne le premier disque dur (**ou la partition primaire**)

**D: E: F:** etc. désignent les disques durs suivants et ensuite les lecteurs de CD (ou autres)

Chaque disque est divisé en répertoires (ou dossiers) dans lesquels sont rangés d'autres répertoires et/ou des **fichiers**

40

## Les fichiers

Ce sont eux qui sont porteurs d'informations...

Ils portent un nom et une extension: **NOM.EXT**

L'extension **.EXT** détermine le type de fichier :

**.TXT** désigne les fichiers textes

**.BAT** désigne les fichiers batchs

**.EXE** désigne les fichiers exécutable

41

## Les fichiers

**B\*.\*** Sélection de tous les fichiers dont le nom commence par B.

**C\*.EXT** Sélection de tous les fichiers dont le nom commence par C et dont on précise l'EXTension.

**\*.\*** Sélection de tous les fichiers sans exception.

42

## Les partitions

Par défaut quand vous installez votre système d'exploitation (Windows) votre disque dur contient une seule partition...

Les partitions une fois créées et formatées sont considérées par le système comme étant des disques durs indépendants:

Exemple:

- 1<sup>ère</sup> partition -> **disque C:**
- 2<sup>ème</sup> partition -> **disque D:**

43

## Les fichiers Batchs (.bat)

Les fichiers **batchs** sont des fichiers textes portant l'extension .bat; Ils peuvent donc être édité avec n'importe quel éditeur de fichier texte, Notepad sous Windows ou Edit sous DOS

Ils servent à exécuter une suite de commandes MS-DOS les unes après les autres de façon automatique.

Le plus connu des fichiers Batchs est : AUTOEXEC.BAT qui s'exécute automatiquement au démarrage de votre ordinateur

44

## Fonctionnement du DOS

Une fois sous **DOS**, vous arrivez sur un écran noir basique : aucune image, que du texte.

Le curseur est toujours placé sur la ligne de commande.

Comme son nom l'indique, cette ligne vous servira à exécuter des commandes comme par exemple créer un répertoire (commande **md**), copier un fichier (commande **copy**) voire formater le disque dur (commande **format**)...

45

## Fonctionnement du DOS

Au départ, la ligne indique sur quel disque dur et dans quel répertoire on se situe.

Par exemple, la ligne ci-dessous indique que l'on se situe sur le disque **C:** dans le répertoire **Windows**.

46

## Commandes principales

**HELP:** Afficher toutes les commandes

**DIR:** Afficher le contenu d'un répertoire

47

## DIR: Afficher le contenu d'un répertoire

**DIR \*.txt:** Afficher tous les fichiers avec l'extension TXT.

**DIR /P:** Afficher le contenu d'un répertoire écran par écran.

**DIR /W:** Afficher le contenu d'un répertoire en condensé.

**DIR X\*. \* /P:** Afficher tous les fichiers commençant par la lettre X, écran par écran.

48



## DIR: Afficher le contenu d'un répertoire

**DIR nom.doc:** Recherche du fichier « nom.doc» dans le répertoire.

**DIR nom.doc/s:** Recherche du fichier « nom.doc» sur tout le disk.

**DIR /O:** Afficher par ordre alphabétique.

49

## CD: Changer de répertoire

**CD nom:** Se placer dans le répertoire «nom»

**CD.. :** Se placer dans le répertoire parent

**CD\ :** se placer dans le répertoire racine

50

## MD

**MD nom:** Crée un répertoire "**nom**" dans le répertoire en cours.

51

## FDISK

**FDISK:** Sert à gérer les partitions sur le disque dur.

- (Attention, si vous supprimez / modifiez une partition, les données du disque sont perdues).

52

# FORMAT

**FORMAT:** Sert à formater une partition d'un disque dur.

(Attention, formater efface toutes les données du disque).

- **Format c:** formate le disque C: ,
- **Format d:** formate le disque D: etc...

53

## COPY Copier des fichiers, XCOPY Copier des fichiers et des répertoires

**COPY nom.ext A:** Copier le fichier «nom.ext» dans le répertoire de la disquette

**COPY \*.\*A:** Copier tous les fichiers du répertoire courant dans le répertoire principal de la disquette.

**XCOPY \*.TXT A:** Copier tous les fichiers avec extension TXT du répertoire courant dans le répertoire de la disquette

**XCOPY fichier.\*A:** Copier tous les fichiers nommés «fichier» (de n'importe quelle extension) du répertoire courant dans le répertoire de la disquette

**XCOPY \*.\*C:/s :** Copier tous les sous-répertoire et les fichiers du répertoire courant dans le répertoire principal de C: (le répertoire de destination peut bien sur être différent)

54

## REN ancien.ext nouveau.ext

**REN A B:** Renommer le fichier A en B.

Exemple :

**REN ancien.ext nouveau.ext** va remplacer le nom du fichier *ancien* par *nouveau*.

55

## Commande

**DEL:** Effacer un fichier

**UNDELETE:** Restorer un fichier effacé

**DELTREE:** Effacer un répertoire et son contenu

**MOVE:** Déplacer des fichiers

**COMP:** Comparer deux fichiers

**EDIT:** Editer un fichier texte ou batch

**TYPE:** Afficher un fichier texte (en lecture seule)

56

## Commande

**MORE:** Afficher un long fichier texte

**CLS:** Efface l'écran

**CONVERT C: /fs:ntfs:** Conversion du disque C en NTFS, au lieu de FAT32 (irréversible)

**DISKCOPY:** Copier un disque

**SCANREG:** Sauvegarder/Restorer la base de registre

57

## ATTRIB

**ATTRIB:** Affiche, définit ou supprime les attributs assignés aux fichiers ou aux répertoires.

**a** = archive,

**r** = lecture seule,

**s** = system,

**h** = caché.

58

## Exemple de modification des attributs de fichiers sous MS-DOS

**ATTRIB +r fichier:** définit l'attribut Lecture seule à «fichier»

**ATTRIB -r fichier:** supprime l'attribut Lecture seule à «fichier»

**ATTRIB +h fichier:** définit l'attribut fichier caché à «fichier»

**ATTRIB -h fichier:** supprime l'attribut fichier caché à «fichier»

59

## Exemple de modification des attributs de fichiers sous MS-DOS

**ATTRIB +a fichier:** définit l'attribut archive seule à «fichier»

**ATTRIB -a fichier:** supprime l'attribut archive seule à «fichier»

**ATTRIB +s fichier:** Définit l'attribut system caché à «fichier»

**ATTRIB -s fichier:** Supprime l'attribut system caché à «fichier»

60

## Autres commandes

**CHKDSK X /f**: Vérifie et corrige les erreurs sur le disque X.

Remplacer X par l'unité du disque (par exemple CHKDSK C: /f ou CHKDSK D: /f ...)

**IPCONFIG**: Afficher la configuration TCP/IP

61

## Manipulation de volumes (disques)

**CHLDSK** : Vérifie un disque et affiche un relevé d'état.

**CHKNTFS** : Affiche ou modifie la vérification du disque au démarrage.

**COMPACT** : Modifie ou affiche la compression des fichiers sur une partition NTFS.

**CONVERT** : Convertit des volumes FAT en volumes NTFS. Vous ne pouvez pas convertir le lecteur en cours d'utilisation.

**DISKCOMP** : Compare les contenus de deux disquettes.

**DISKCOPY** : Copie le contenu d'une disquette sur une autre.

**FORMAT** : Formate un disque pour utilisation avec Windows.

**LABEL**: Crée, modifie ou supprime le nom de volume d'un disque.

**RECOVER** : Récupère l'information lisible d'un disque défectueux.

**SUBST** : Affecte une lettre de lecteur à un chemin d'accès.

**VERIFY** : Indique à Windows s'il doit ou non vérifier que les fichiers sont écrits correctement sur un disque donné.

**VOL** : Affiche le nom et le numéro de série du volume.

## Manipulation de volumes (Système)

**ASSOC** : Affiche ou modifie les applications associées aux extensions de fichiers.

**AT** : Planifie l'exécution de commandes ou programmes sur un ordinateur.

**DOSKEY** : Modifie les lignes de commande, rappelle des commandes Windows, et permet de créer des macros.

**CHCP** : Modifie la page de code active ou affiche son numéro.

**CMD** : Lance une nouvelle instance de l'interpréteur de commandes de Windows.

**DATE** : Affiche ou modifie la date.

**FTYPE** : Affiche ou modifie les types de fichiers utilisés dans les associations d'extensions.

**GRAFTABL** : Permet à Windows d'afficher un jeu de caractères en mode graphique.

**MODE** : Configure un périphérique du système.

**PATH** : Affiche ou définit le chemin de recherche des fichiers exécutables.

**PROMPT** : Modifie l'invite de commande de Windows.

**SET** : Affiche, définit ou supprime des variables d'environnement Windows.

**TIME** : Affiche ou définit l'heure de l'horloge interne du système.

**VER** : Affiche le numéro de version de Windows.

## Variables d'environnement



## Les variables d'environnement (1/7)

Possibilité de définir des valeurs (chaînes de caractères) à mémoriser associées à des noms

Par exemples :

- Les chemins d'accès à vos programmes favoris
- Le chemin du répertoire d'accueil *HOMEPath*
- Le chemin du répertoire *temp* servant à stocker les fichiers temporaires

Les variables d'environnement sont utilisables avec l'interpréteur Windows

Accéder et définir les variables d'environnement :

- Par une commande de l'interpréteur : *set*  
Les modifications sont locales à l'interpréteur de commande  
Les modifications ne sont pas mémorisées à la fermeture de l'interpréteur
- En modifiant les propriétés du poste de travail :  
Les modifications sont mémorisées même si l'utilisateur se déconnecte

65

## Les variables d'environnement (2/7)

Accès et modification par le poste de travail :

- Dans un explorateur ou sur son bureau, clic droit sur l'icône *Poste de travail*, menu *Propriétés* puis onglet *Avancé* (Fig. 12)
- Clic sur le bouton *Variables d'environnement* (Fig. 12)
- Lire / créer / modifier / supprimer ses variables d'environnement (les variables système ne sont pas modifiables) (Fig. 13)

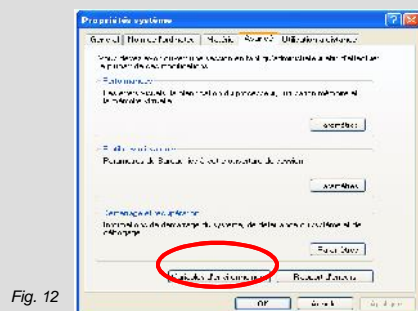


Fig. 12

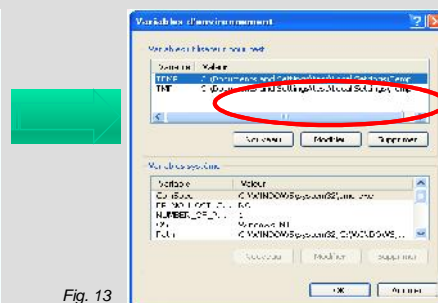


Fig. 13

66

## Les variables d'environnement (3/7)


- Accès et modification par la commande `set` :
  - Afficher toutes les variables actuelles : `set` sans argument
  - Afficher toutes les variables commençant par *début* : `set début`
  - Accéder à la valeur de la variable *toto* par un interpréteur : `%toto%`
  - Afficher la variable nommée *toto* : `echo %toto%`

[illegible]

67

## Les variables d'environnement (4/7)

- Accès et modification par la commande `set` (suite) :
  - Créer / modifier une variable : `set nom=variable`
  - Supprimer la variable nommée `toto` : `set toto=`
  - Saisir la valeur d'une variable nommée `nom` avec l'option `/p` :  
`set /p nom=`
  - Saisir la valeur avec un message d'invite :  
`set /p nom="Donnez une valeur : "`



The screenshot shows a Windows command prompt window with the title bar "C:\WINDOWS\system32\cmd.exe". The command history is as follows:

```

C:\Documents and Settings\test>set toto=valeur
C:\Documents and Settings\test>set toto
toto=valeur
C:\Documents and Settings\test>set /p toto="Donnez la valeur de toto : "
Donnez la valeur de toto : 120
C:\Documents and Settings\test>set toto
toto=120
C:\Documents and Settings\test>set toto=
C:\Documents and Settings\test>set toto
La variable d'environnement toto n'est pas définie.
C:\Documents and Settings\test>
C:\Documents and Settings\test>_

```

68

## Les variables d'environnement (5/7)

- Accès et modification par la commande `set` (suite) :
  - Faire des opérations arithmétiques (+ - / x) sur les entiers  
Utiliser l'option `/a` avec des variables ne contenant que des chiffres  
Syntaxe : `set /a nom_variable=expression_arithmétique`  
Exemple pour incrémenter un compteur : `set /a compteur=%compteur% + 1`

The first screenshot shows a command prompt window with the following commands and output:

```
C:\WINDOWS\system32\cmd.exe
C:\Documents and Settings\test>set compteur=0
compteur=0
C:\Documents and Settings\test>set /a compteur=compteur% + 1
1
C:\Documents and Settings\test>set /a compteur=%compteur% + 1
2
C:\Documents and Settings\test>set /a compteur=%compteur% + 1
3
C:\Documents and Settings\test>
```

The second screenshot shows a command prompt window with the following commands and output:

```
C:\WINDOWS\system32\cmd.exe
C:\Documents and Settings\test>set toto=0
toto=0
C:\Documents and Settings\test>set titi=1
titi=1
C:\Documents and Settings\test>set /a toto=titi / %toto%
Erreur de division par zéro.
C:\Documents and Settings\test>set /a toto=titi / %toto%
Erreur de division par zéro.
C:\Documents and Settings\test>set /a toto=titi / %toto%
Erreur de division par zéro.
C:\Documents and Settings\test>
```

69

## Les variables d'environnement (6/7)

Les variables à valeur dynamique

- Variable dont la valeur est donnée par le système

Quelques variables dynamiques :

Variable	Valeur
%DATE%	Date courante au format jj/mm/aaaa
%TIME%	Heure courante au format hh:mm:ss,ss
%CD%	Répertoire courant
%ERRORLEVEL%	Code de retour de la dernière commande
%RANDOM%	Entier aléatoire entre 0 et 32768

The screenshot shows a command prompt window with the following commands and output:

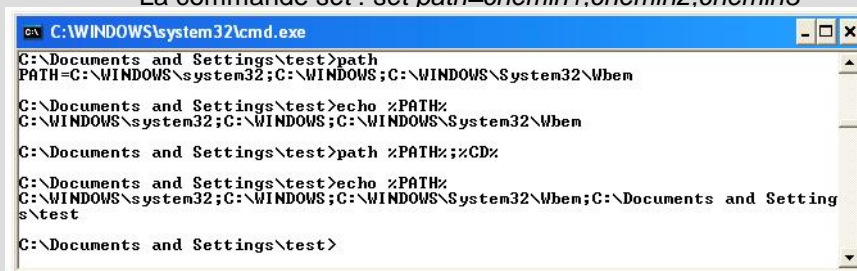
```
C:\WINDOWS\system32\cmd.exe
C:\Documents and Settings\test>echo %TIME%
17:32:31.04
C:\Documents and Settings\test>echo %DATE%
06/07/2010
C:\Documents and Settings\test>set toto=0
toto=0
C:\Documents and Settings\test>set /a toto=titi / %toto%
Erreur de division par zéro.
C:\Documents and Settings\test>echo %ERRORLEVEL%
9169
C:\Documents and Settings\test>
```

70

## Les variables d'environnement (7/7)

La variable d'environnement *PATH*

- Elle donne une liste de chemins de recherche (séparés par des point-virgules) de fichiers exécutables
- Deux manières d'afficher la variable :
  - La commande *path* sans argument
  - Afficher directement la variable : *echo %PATH%*
- Deux manières de modifier la variable :
  - La commande *path* : *path chemin1;chemin2; chemin3*
  - La commande *set* : *set path=chemin1;chemin2;chemin3*



```
C:\WINDOWS\system32\cmd.exe
C:\Documents and Settings\test>path
PATH=C:\WINDOWS\system32;C:\WINDOWS;C:\WINDOWS\System32\Wbem
C:\Documents and Settings\test>echo %PATH%
C:\WINDOWS\system32;C:\WINDOWS;C:\WINDOWS\System32\Wbem
C:\Documents and Settings\test>path %PATH%;%CD%
C:\Documents and Settings\test>echo %PATH%
C:\WINDOWS\system32;C:\WINDOWS;C:\WINDOWS\System32\Wbem;C:\Documents and Settings\test
C:\Documents and Settings\test>
```

71

## Les scripts batchs

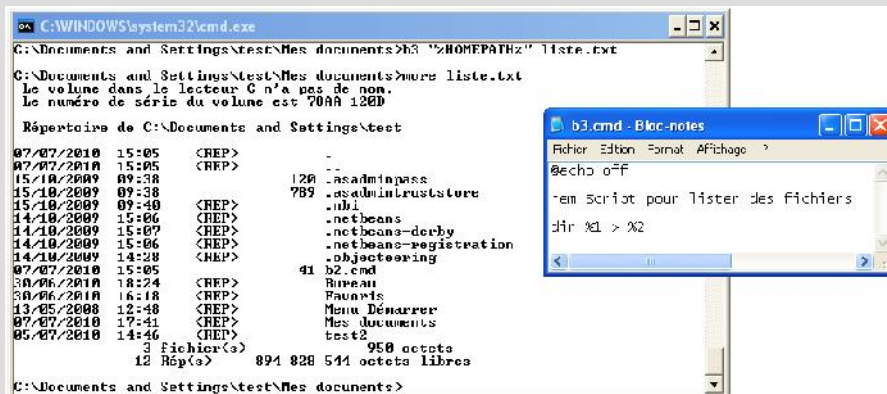


## Les scripts batch (3/8)

Donnez des arguments à un script batch :

- Possibilité de généraliser une tâche en spécifiant des paramètres au lancement du script
- Les valeurs des arguments du script sont stockées dans les variables %1 à %9 et utilisables dans les commandes de ce script
- Par exemple, pour lancer le fichier *toto.bat* avec 2 arguments : *toto arg1 arg2*

Ajoutez des commentaires avec la commande *rem*



```
C:\WINDOWS\system32\cmd.exe
C:\Documents and Settings\test\Mes documents>h3 "%HOMEPATH%" liste.txt
C:\Documents and Settings\test\Mes documents>more liste.txt
Le volume dans le lecteur C n'a pas de nom.
Le numéro de série du volume est 70AA 120D

Répertoire de C:\Documents and Settings\test

07/07/2010 15:05 <REP> -
07/07/2010 15:05 <REP> -
15/10/2009 09:38 120 .asadminpass
15/10/2009 09:38 785 .asadminstructure
15/10/2009 09:40 <REP> .nbi
14/10/2009 15:06 <REP> .netbeans
14/10/2009 15:07 <REP> .netbeans-derby
14/10/2009 15:06 <REP> .netbeans-registration
14/10/2009 14:28 <REP> .objecteering
07/07/2010 15:05 41 b2.cmd
30/06/2010 18:24 <REP> Bureau
30/06/2010 16:18 <REP> Paquets
13/05/2008 12:48 <REP> Menu Démarrer
07/07/2010 17:41 <REP> Mes documents
05/07/2010 14:46 <REP> test2
3 fichier(s) 950 octets
12 Rép(s) 894 828 544 octets libres

C:\Documents and Settings\test\Mes documents>
```

```
b3.cmd - Bloc-notes
Fichier Edition Format Affichage
@echo off
rem Script pour lister des fichiers
dir %1 > %2
```

75

## Les scripts batch (4/8)

Exécution conditionnelle avec la commande *if*

- Exécuter une commande uniquement si une condition est vraie :  
Par exemple, effectuer une copie d'un fichier de taille importante si la taille du disque de destination est assez grande  
Par exemple, exécuter une commande si la commande précédente s'est bien déroulée
- Syntaxe générale de l'instruction :  
*if condition commande1 [else commande2]*  
qui se lit : « si la *condition* est vraie alors lancer *commande1* sinon lancer *commande2* »
- La partie *else* est optionnelle
- *commande1* et *commande2* peuvent être mis entre parenthèses pour éviter toute ambiguïté
- *commande1* et *commande2* peuvent être composés de plusieurs commandes séparés par &

76

## Les scripts batch (5/8)

Les conditions de l'instruction *if* :

- Existence d'un fichier :  
*if exist fich commande*  
qui se lit : « si le fichier nommé *fich* existe alors lancer *commande* »  
On peut donner un chemin complet au fichier sinon l'interpréteur cherche à partir du répertoire courant
- Existence d'une variable d'environnement :  
*if defined var commande*  
qui se lit : « si la variable nommée *var* est définie alors lancer *commande* »  
Test l'existence de la variable est non sa valeur !

77

## Les scripts batch (6/8)

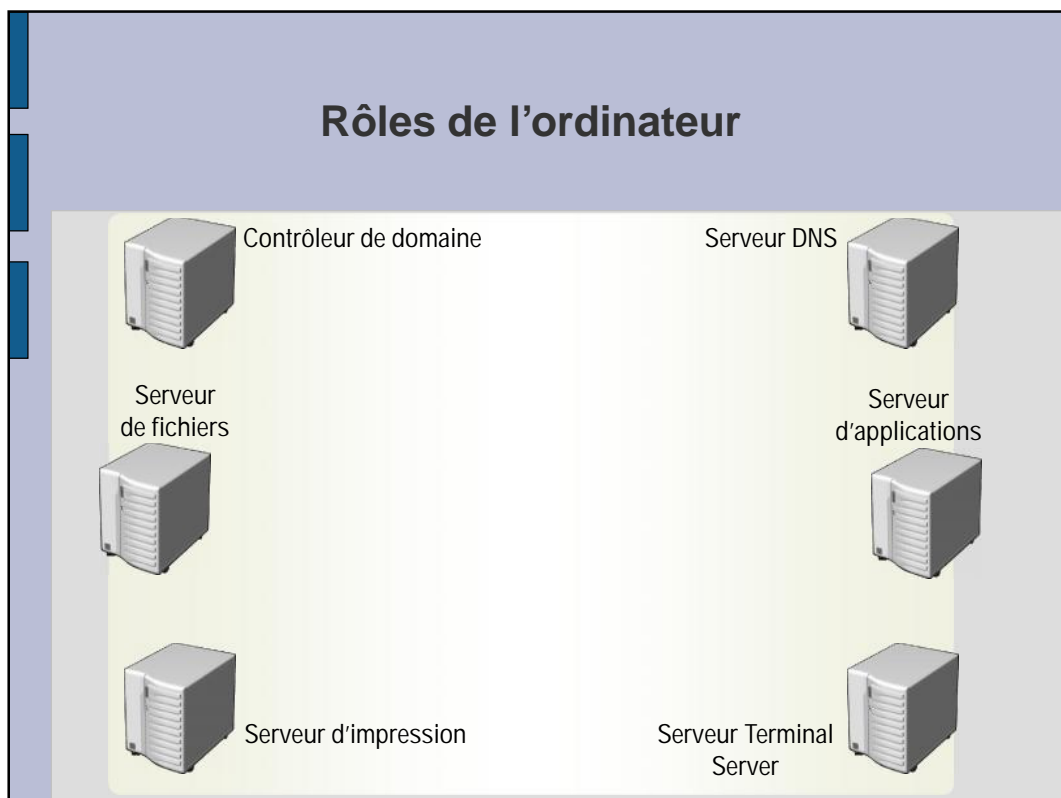
Les conditions de l'instruction *if* (suite) :

- Valeur de sortie de la dernière commande :  
*if errorlevel n commande*  
qui se lit : « si la valeur de retour de la commande précédente est supérieure ou égal à *n* alors lancer *commande* »  
Par convention *ERRORLEVEL* vaut 0 pour une exécution sans erreur de la dernière commande
- Comparaison de deux chaînes de caractères :  
*if "chaine1"=="chaine2" commande*  
qui se lit « si la valeur de *chaine1* égal la valeur de *chaine2* alors lancer *commande* »  
Pour ne pas tenir compte de la différence majuscule/minuscule utilisez l'option */i*

78





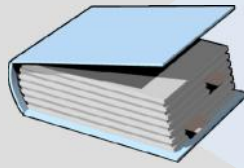


## Description d'un service d'annuaire

Il identifie les ressources

Il fournit un cadre cohérent pour :

- l'attribution de noms
- la description
- la localisation
- les accès
- l'administration
- la sécurité



### Avantages d'Active Directory

• Intégration DNS

• Évolutivité

• Administration centralisée

• Administration déléguée

## Architecture Windows server

Un annuaire est au cœur de l'architecture

Il contient tous les éléments et ressources présents dans le système informatique :

utilisateur  
ordinateur  
groupe  
domaine  
application  
imprimante ...

Il est stocké dans une base de données

## Architecture Windows

Tout élément dans l'annuaire est un objet possédant des attributs

(ex : un utilisateur à un nom, prénom, login ...)

Chaque domaine contient une partition de l'annuaire

Chaque partition peut contenir jusqu'à 10 millions d'objets

## Architecture Windows

La protection des objets de l'annuaire

- Tous les objets d'Active Directory sont protégés par une liste de contrôle d'accès (ACL) qui définissent qui est autorisé à voir l'objet et quelles actions peuvent être effectuées sur celui-ci.
- Les ACL s'appliquent à l'objet, ainsi qu'à ses attributs.

## Architecture Windows

La distinction entre PDC et BDC disparaît.

Tous les Contrôleurs d'un même domaine possèdent un réplica complet de l'annuaire du domaine accessible en écriture.

## CD/SCG

Contrôleur de domaine: Serveur qui authentifie des connexions au domaine et gère la stratégie de sécurité et la base de données de comptes principale pour un domaine.

Serveur de catalogue global : Serveur d'annuaire qui contient un réplica partiel d'Active Directory pour chaque domaine dans une forêt d'entreprise.

## Utilisateur/Groupe

Utilisateur : Objet Annuaire utilisé pour identifier un compte spécifique et unique dans Active Directory.

Groupe : Ensemble d'utilisateurs, d'ordinateurs, de contacts et d'autres groupes. Les groupes peuvent être utilisés comme des ensembles de sécurité et de distribution de messages électroniques.

## Structure logique

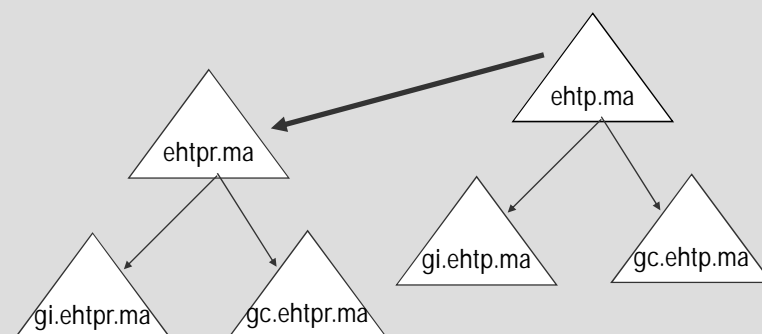
- Forêts
- Arborescences
- Domaines
- Unités d'organisation

## Forêt

Ensemble d'un ou plusieurs domaines Active Directory organisés comme homologues et interconnectés par des relations d'approbation transitives bidirectionnelles. Tous les domaines présents dans une forêt partagent un schéma, une configuration et un catalogue global communs.

## Forêts

**Forêt: ensemble d'arborescences**



## Arborescences

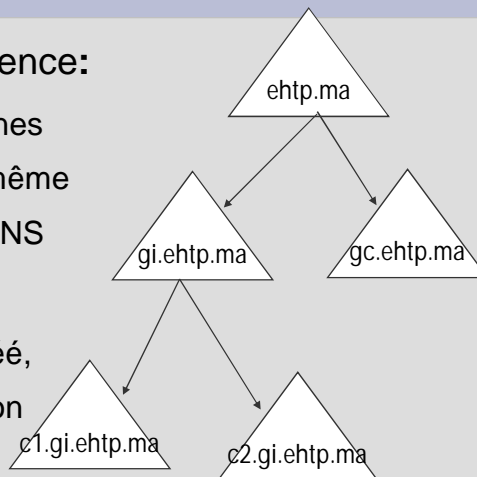
Arbre ou arborescence:

ensemble de domaines  
appartenant à une même  
hiérarchie de nom DNS

Domaine racine:

premier domaine créé,  
non renommable, non  
supprimable

Domaine enfant



## Domaine

**Limite de sécurité:** chaque domaine dispose de ses propres stratégies de sécurité.

**Ensemble d'ordinateurs en réseau** qui partagent une base de données de Gestionnaire de comptes de sécurité (SAM) et qui peuvent être administrés en tant que groupe.

**Mode d'un domaine:** mixte ou natif (dépend de l'OS des contrôleurs de domaine); s'il reste des contrôleur de domaine NT4.

## Unités d'organisation (UO)

Conteneur Active Directory utilisé à l'intérieur de domaines. Les UO sont des conteneurs logiques dans lesquels des utilisateurs, des groupes, des ordinateurs et d'autres UO sont placés. Elles ne peuvent contenir que des objets de leur domaine parent. Une UO est la plus petite unité à laquelle il soit possible d'appliquer une stratégie de groupe ou une délégation d'autorité.

### **Permet**

- De déléguer des pouvoirs
- De simplifier la sécurité
- D'appliquer une stratégie à des ordinateurs ou utilisateurs

## Architecture Windows server Unités Organisationnelles

### Elles peuvent contenir :

des utilisateurs  
des ordinateurs  
des groupes  
d'autres UO

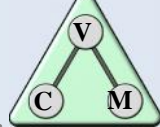
### Ils sont utilisés pour l'affectation :

des permissions sur les objets et ressources réseaux  
de la stratégie système



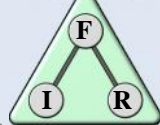
## Modèles hiérarchiques des unités d'organisation

### Fondée sur la fonction



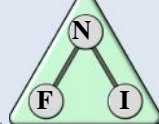
V – Ventes  
C – Consultants  
M – Marketing

### Fondée sur l'organisation



F – Fabrication  
I – Ingénierie  
R – Recherche

### Fondée sur l'emplacement



N – Norvège  
F – France  
I – Indonésie

### Exemples de modèles hybrides

Fonction  
Organisation

Emplacement  
Fonction

Organisation  
Emplacement

## Atelier pratique