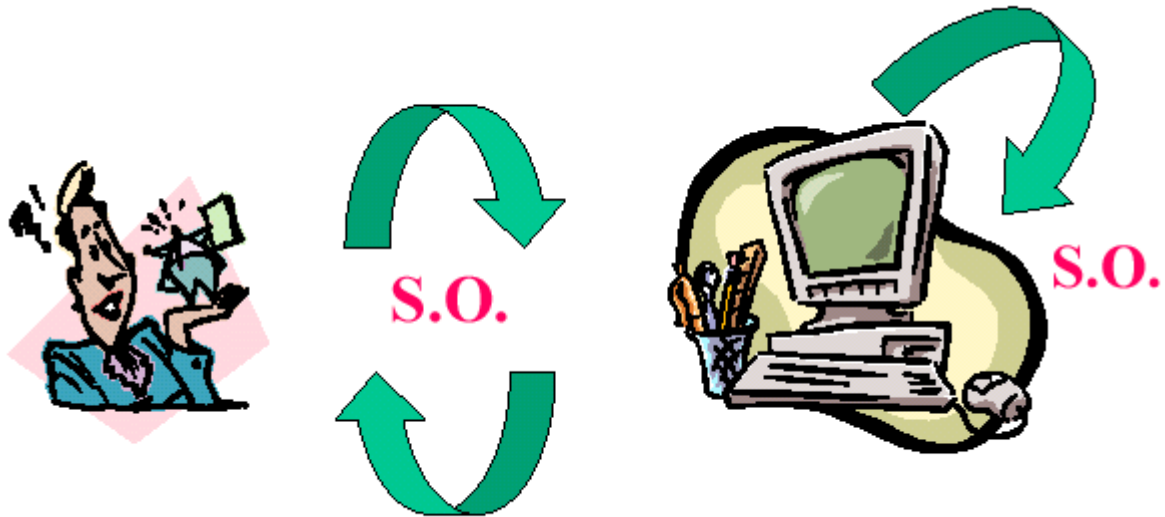
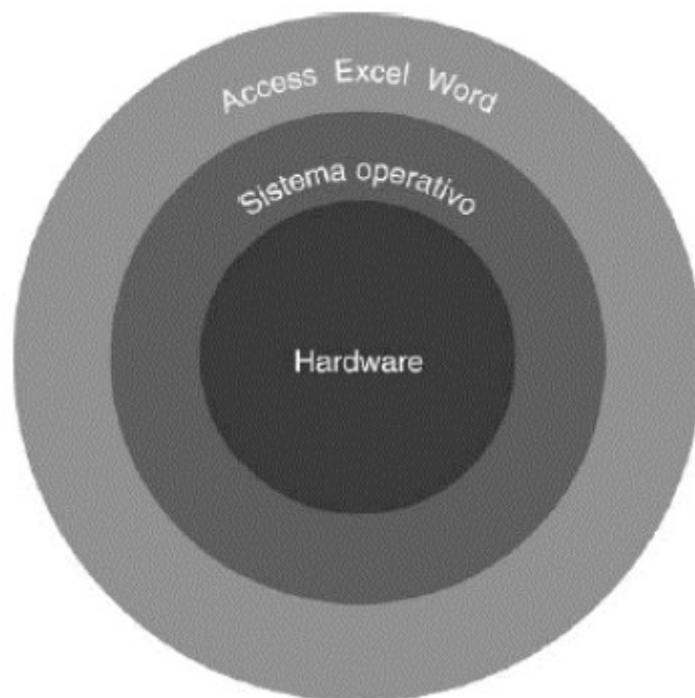


Un **Sistema Operativo** è un insieme complesso di programmi che, interagendo tra loro, devono svolgere una serie di funzioni per **gestire il comportamento** del computer e per agire come **intermediario** consentendo all'utente un utilizzo della macchina semplice ed economico.



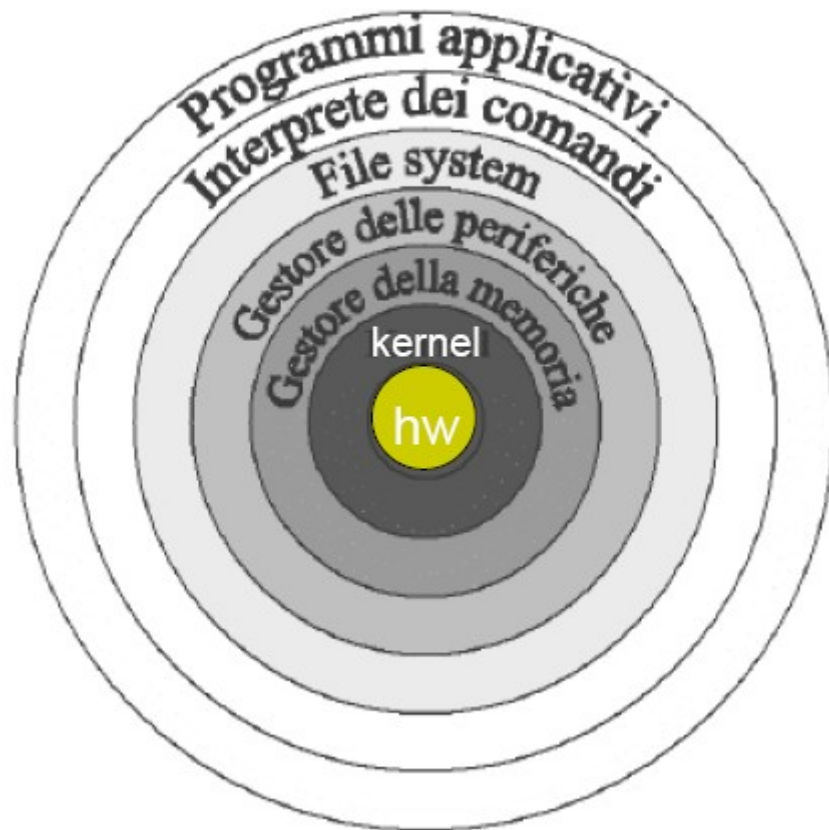
In un modello a strati il SO si pone come un guscio (shell) tra la macchina reale ([HW](#)) e le applicazioni¹ :



¹ Programmi usati per risolvere problemi in ambito specifico

Le funzioni di un SO si possono illustrare con una architettura a strati:

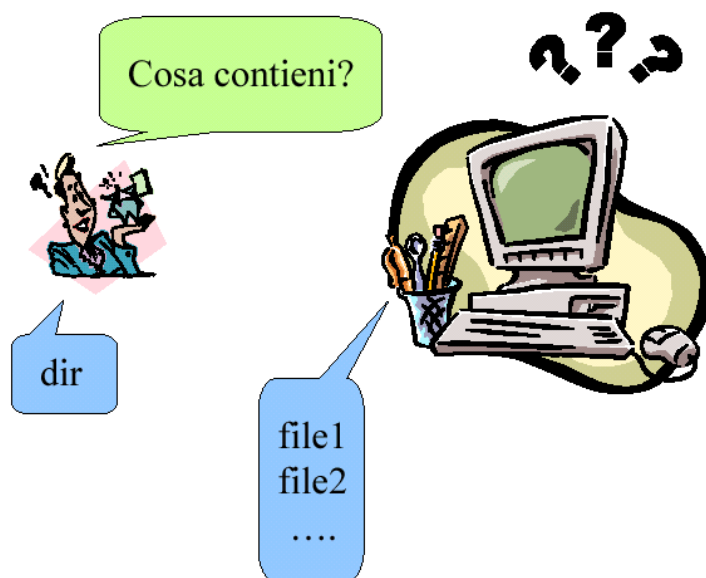
- La gestione della CPU (kernel)
- La gestione delle memoria di lavoro a semiconduttore (RAM)
- La gestione delle interfacce HW (tastiera, mouse, video, ...)
- La gestione delle memorie a supporto magnetico
- L'interpretazione dei comandi



I linguaggi dei comandi sono suddivisi, per grandi linee, in:

- **testuali**: i comandi vengono impartiti al sistema attraverso una serie di **parole del linguaggio** scritte (da tastiera) su una riga di comando. Possono essere complessi da utilizzare.
- **a finestre**: il file system viene mostrato in forma di **icone** e i comandi vengono impartiti al sistema attraverso **operazioni eseguite con il mouse** (click, trascinamento di icone). Sono più semplici da utilizzare ma meno espressivi dei linguaggi testuali.

Con **interfaccia utente** si intendono le funzioni di interpretazione ed esecuzione dei linguaggi dei comandi



Il Linguaggio Comandi

Esempio: MS- DOS

Un insieme di **comandi** (parole chiave del S. O.) vengono immessi da tastiera essenzialmente per gestire il file system.

Il S. O. organizza, infatti, i file² gestendo:

- la struttura delle cartelle (directory) e sottocartelle (subdirectory) organizzate ad albero
- la modalità di memorizzazione su memoria di massa e di recupero da memoria di massa

Esempio di comandi:

dir : mostra i file contenuti nella directory corrente

dir/w per visualizzare il contenuto della directory corrente in forma compatta (videata)

dir/p per visualizzare il contenuto della directory corrente a pagine con informazioni aggiuntive

cd : cambia la directory corrente

cd per **risalire** alla directory principale (root o radice di un albero che si dirama in sottodirectory)

cd.. per **risalire** alla "directory padre" (Cambia Directory)

ren : cambia il nome di un file

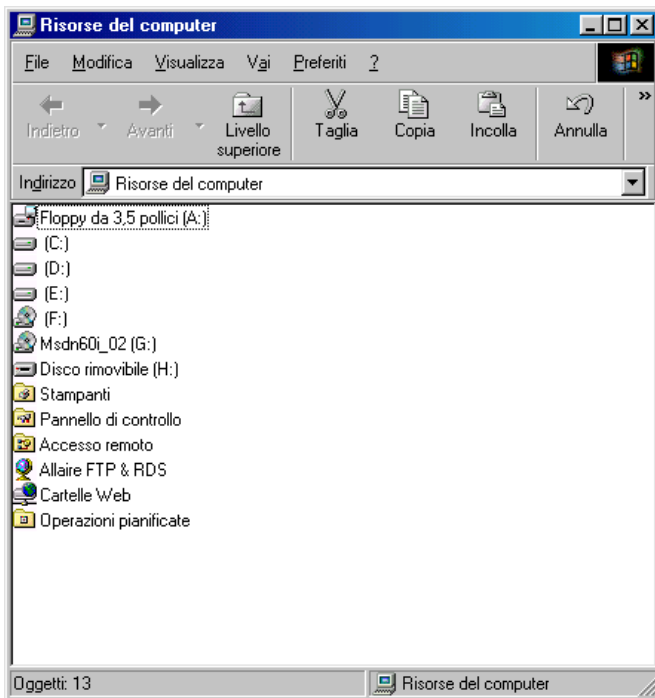
copy : copia il file in un'altra directory

del : cancella un file

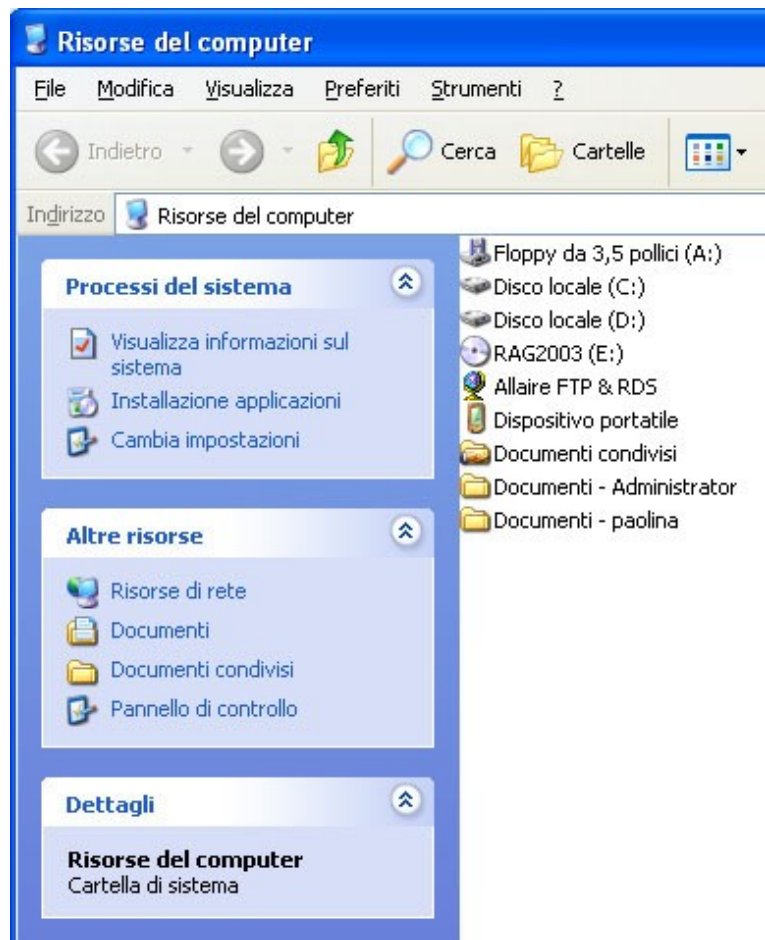
cls : cancella lo schermo

² Per **file** si intende un insieme di dati e/o istruzioni archiviati su *supporto magnetico* (memorie di massa)

Esempio di Graphical User Interface tipica di Windows 98



...o Windows XP



Attività:

Seguendo, in ambiente Windows XX, il percorso: Programmi → Accessori → Prompt dei comandi usa alcuni comandi del Sistema Operativo MS-DOS che si presenta con interfaccia a “riga di comando”

Esempio di interfaccia a “**riga di comando**”:

```
C:\>dir/p
Il volume nell'unità C non ha etichetta.
Numero di serie del volume: 00C2-1D75

Directory di C:\

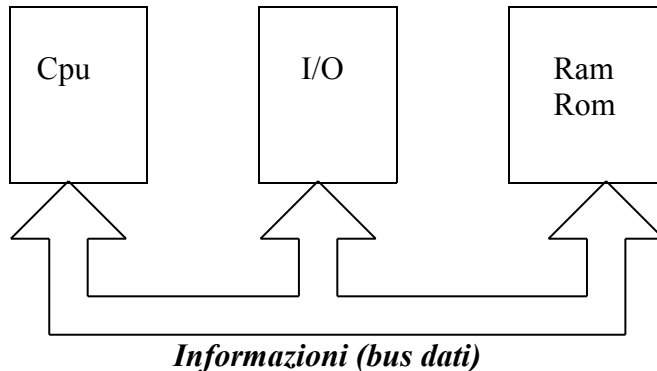
28/08/2003 07.57          151 liprefs.js
30/08/2003 19.41        <DIR>      lj631en
03/09/2003 22.38        <DIR>      zanic
28/08/2003 12.03        <DIR>      WINDOWS
28/08/2003 12.05        <DIR>      UALUEADD
28/08/2003 12.05        <DIR>      SUPPORT
28/08/2003 12.06        <DIR>      Documents and Settings
28/08/2003 12.14        <DIR>      Programmi
28/08/2003 12.16          0 CONFIG.SYS
28/08/2003 12.16          0 AUTOEXEC.BAT
28/08/2003 06.20        <DIR>      WUTemp
          3 File          151 byte
          8 Directory 15.489.564.672 byte disponibili

C:\>
```

Hardware: [architettura base](#) di un calcolatore

Nell'architettura di un **sistema a microprocessore** (modello logico-funzionale di Von Neumann) si distinguono tre grandi blocchi funzionali: CPU, I/O (periferici) e memorie che **condividono un unico canale per scambiarsi informazioni** (il bus dati).

Tale contemporaneità si indica col termine architettura “*parallela*”.



- **CPU** (*central processing unit*, o Microprocessore): con funzioni di elaborazione (operazioni aritmetico-logiche) e controllo
- **I/O** (input/output): dispositivi periferici³ per consentire lo scambio di informazioni tra l'elaboratore e l'esterno. Costituiscono l'interfaccia Hardware con l'operatore umano
- **Memoria di lavoro o centrale** (a semiconduttore): **RAM** (*Random Access Memory*) per memorizzare temporaneamente dati e programmi (volatile⁴, con veloci tempi di accesso da parte della CPU, costosa)
ROM (*Read Only Memory*) non modificabile, non volatile (*istruzioni incise su chip o firmware per gestire la fase di avvio all'accensione: esecuzione di test diagnostici sull'Hardware e bootstrap: lettura da disco con caricamento del Sistema Operativo d'avvio*)

Prende nome di **bus di sistema** il collegamento tra gli elementi funzionali illustrati; tale bus fornisce il supporto fisico per la trasmissione dei dati tra i vari elementi. Collega due unità funzionali alla volta: una trasmette e l'altra riceve. Il trasferimento avviene sotto il controllo della CPU. Su questo supporto (costituito da più linee) viaggiano dati, indirizzi e comandi. Tali linee si distinguono spesso logicamente⁵ in:

- bus dati (**data bus**)
- bus indirizzi (**address bus**)
- bus comandi (**command bus**)

Bus dati: bidirezionale. Serve per lo scambio di informazioni cioè trasmettere dati dalla memoria al registro dati o viceversa.

Bus indirizzi: unidirezionale. Serve per individuare il dispositivo periferico o la cella di memoria che può accedere al bus dati per operazioni di lettura o scrittura.

Bus comandi: ogni linea è unidirezionale. Serve per gestire le diverse modalità nello scambio informativo: lettura o scrittura con la memoria, comando di stampa verso una periferica etc...

Se la dimensione (numero di bit) del bus dati è uguale alla dimensione della parola si può trasferire in parallelo un intero dato. Altrimenti occorrono più trasferimenti. Esistono vari tipi di bus di sistema ad esempio ISA, EISA, PCI.

³ Anche la memoria di massa è vista come dispositivo periferico.

⁴ **Volatilità:** perdita del contenuto informativo in assenza di alimentazione

⁵ Fisicamente le stesse linee possono essere usate, in tempi diversi, per funzioni diverse (ad esempio prima per indirizzare il dispositivo che accede al canale condiviso e poi per trasmettere dati)