

Software

Elementi di Informatica

Hardware e Software

Hardware e Software sono "logicamente" equivalenti

"L'hardware è software pietrificato" (Karen Panetta Lenz)

- Tutto ciò che viene effettuato da un software, può essere inglobato nell'hardware e viceversa: il software di oggi potrebbe essere l'hardware di domani
- La scelta dipende da: costo, velocità, affidabilità e frequenza di aggiornamento delle funzioni

Software

- L'hardware da solo non è sufficiente per il funzionamento dell'elaboratore, è necessario *introdurre* il **software**...
- ovvero, un insieme di programmi che permettono di trasformare dei circuiti elettronici in un oggetto in grado di svolgere delle funzioni di natura diversa

Software

Hardware: l'insieme delle componenti fisiche che costituiscono un calcolatore

Software: si suddivide in due categorie:

- **Software di base:** insieme di programmi che permettono di gestire l'intero sistema (sistema operativo, compilatori, interpreti...)
- **Software applicativo:** programmi per specifiche applicazioni.

Il Sistema Operativo

Interfaccia fra l'utente e l'hardware.

Permette di...

1. Gestire efficientemente l'elaboratore ed i suoi dispositivi
2. Creare un ambiente per l'interazione dell'utente

Il compito principale di un sistema operativo:

Fornire un sistema "virtuale", più semplice da usare rispetto all'hardware che si ha effettivamente a disposizione

- Virtuale: una cosa che non esiste fisicamente ma soltanto logicamente

Il Sistema Operativo

L'interfaccia

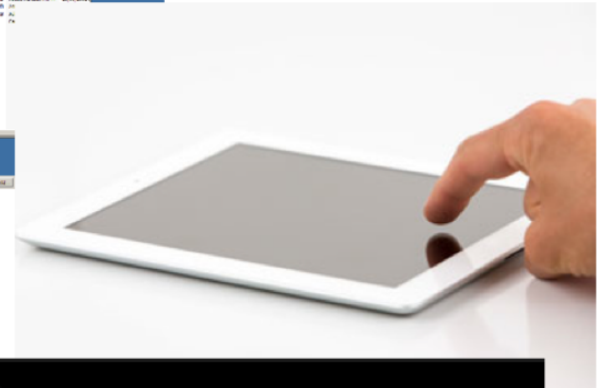
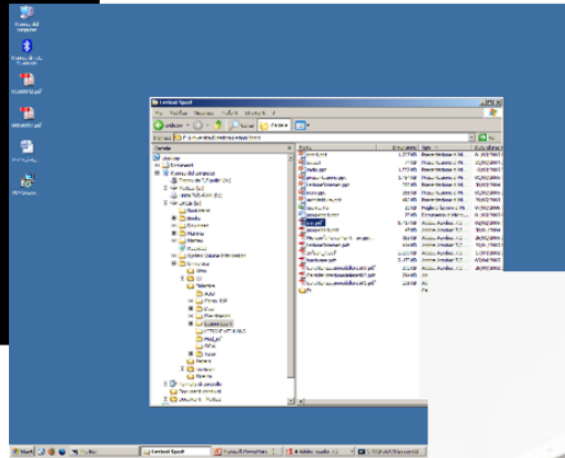
```
Microsoft Windows XP [Versione 5.1.2600]
(C) Copyright 1985-2001 Microsoft Corp.

C:\Documents and Settings\Matteo>cd ..
C:\Documents and Settings>cd ..

C:\>dir
Il volume nell'unità C è Matteo
Numero di serie del volume: 1014-9F79

Directory di C:\
10/01/2006  14:52           0 AUTOEXEC.BAT
10/01/2006  14:52           0 CONFIG.SYS
10/01/2006  14:59    <DIR>      Documents and Settings
23/03/2006  10:58    <DIR>      Programmi
25/01/2006  12:59    <DIR>      Temp
29/03/2006  09:41    <DIR>      WINDOWS
                2 File          0 byte
                4 Directory 9.332.330.496 byte disponibili

C:\>edit AUTOEXEC.BAT
C:\>_
```



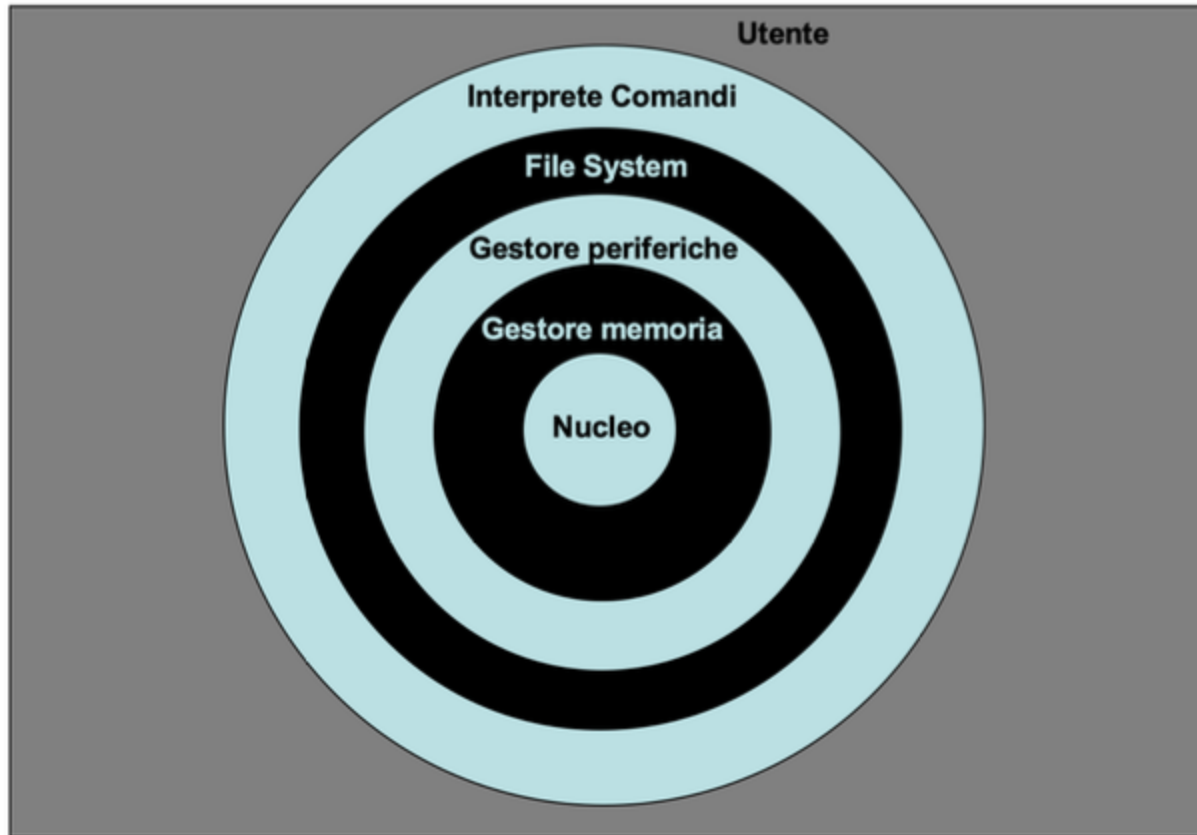
Architettura di un Sistema Operativo

Modello a strati gerarchici

Struttura organizzata su diversi livelli: gerarchia di macchine virtuali, con il compito di gestire specifiche risorse fornendo meccanismi logici di accesso che ne regolamentino l'uso e ne mascherino l'effettivo funzionamento

L'utente è ignaro di tutti i dettagli delle operazioni svolte dai livelli inferiori della gerarchia e conosce solo le operazioni del livello più alto

Architettura di un Sistema Operativo



Bootstrap

Il sistema operativo viene mandato in esecuzione al momento dell'accensione del calcolatore (bootstrap)



Modalità operative di un SO

- mono utente / multi utente
- mono programmato / multi programmato

Modalità operative di un SO

Sistema mono-utente e mono-programmato

- Un solo utente può eseguire un solo programma alla volta
- Il programma viene lanciato, eseguito e quindi terminato

Ma la CPU viene sfruttata al meglio? ...no, si spreca molto tempo!

La CPU è molto più veloce dei supporti di memoria secondaria e delle altre periferiche, e passa la maggior parte del suo tempo in attesa del completamento delle operazioni demandate a questi dispositivi.

Modalità operative di un SO

Sistema mono-utente e multi-programmato

- E' possibile eseguire più programmi *contemporaneamente*
- Le risorse disponibili devono essere suddivise su tutte le applicazioni *contemporaneamente* attive

COME È POSSIBILE CHE UN SINGOLO COMPUTER, CON UN SINGOLO PROCESSORE E UNA SOLA MEMORIA, SIA CAPACE DI GESTIRE PIÙ PROGRAMMI CONTEMPORANEAMENTE?

Modalità operative di un SO

Sistema mono-utente e multi-programmato

In realtà in un determinato istante solo un programma (processo) si trova in esecuzione

- Quando la CPU è nello stato di *idle* la si può sfruttare per eseguire (parte di) un altro processo
- Il sistema operativo si occupa dell'alternanza tra i processi in esecuzione

Programmi e Processi

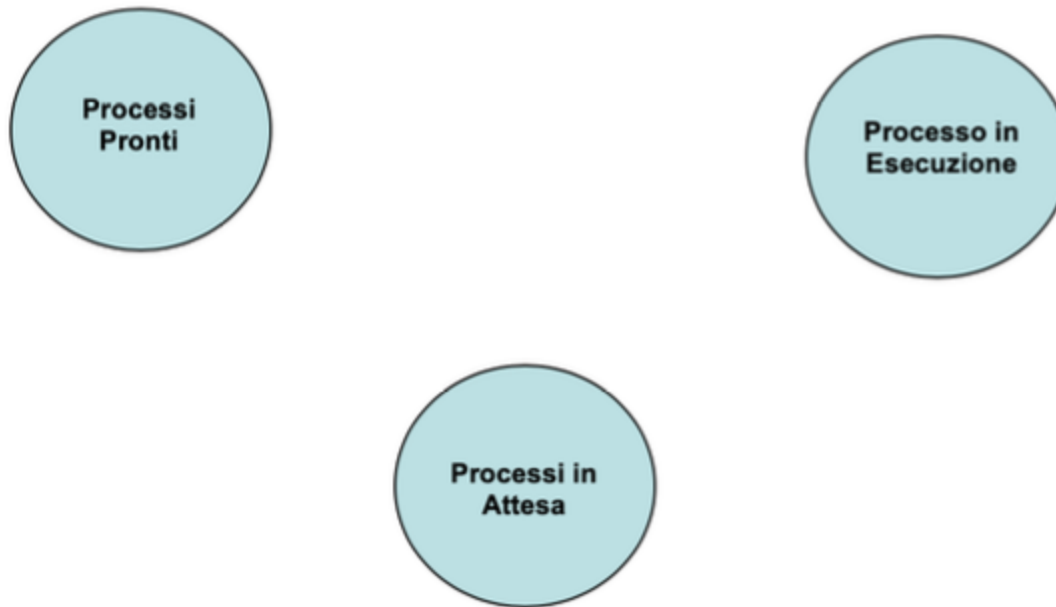
Programma: entità statica, memorizzato in genere su un dispositivo di memoria di massa

Processo: entità dinamica, programma in esecuzione. Contiene le istruzioni e i dati utilizzati dal programma (contesto del programma)

| A un programma possono corrispondere diversi processi.

| Un processo può a sua volta richiedere l'esecuzione di altri processi (si parla di processo padre e processi figli).

Stati di un processo



Stati di un processo

Più programmi sembrano essere eseguiti *contemporaneamente*

In realtà in esecuzione c'è sempre un solo processo ma, se l'alternanza è molto *veloce*, si ha l'impressione di apparente simultaneità

Un processo non si ferma mai in attesa di I/O o di una risorsa? Più utenti vogliono usare il computer? ... è necessario far sì che la risorsa più importante del computer - la CPU - sia distribuita tra i processi dello stesso utente e di utenti diversi.

Si parla di **scheduling** del processore

Il Nucleo

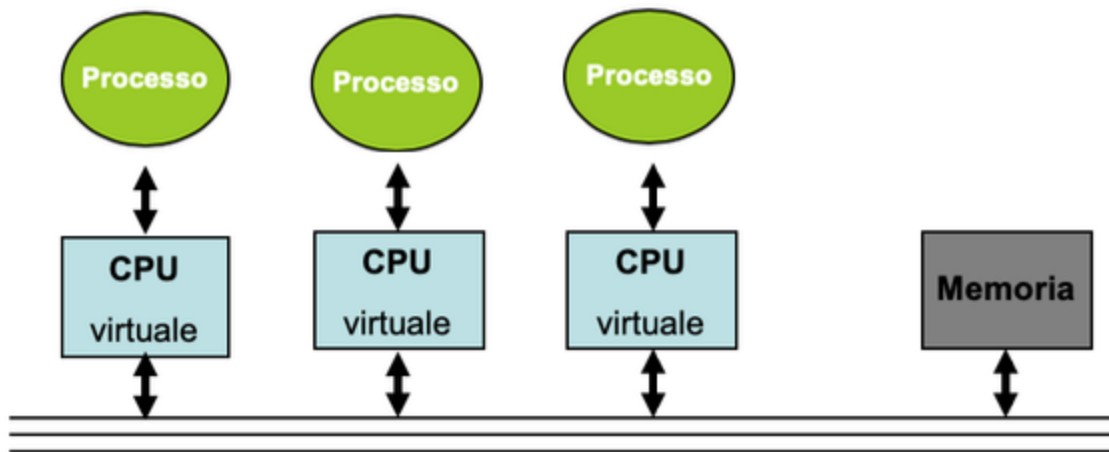
Compiti del nucleo:

- strato che dialoga direttamente con l'hardware (CPU)
- esecuzione dei programmi e risposta a eventi generati dalle periferiche

Requisito fondamentale: consentire a utenti/programmi diversi la condivisione delle risorse

- Offrire virtualmente ad ogni utente/programma tutta la macchina
- Tante CPU virtuali

Il Nucleo



Algoritmi di scheduling

- FIFO (First In First Out)
- SJF (Shortest Job First)
- Priorità

Round Robin

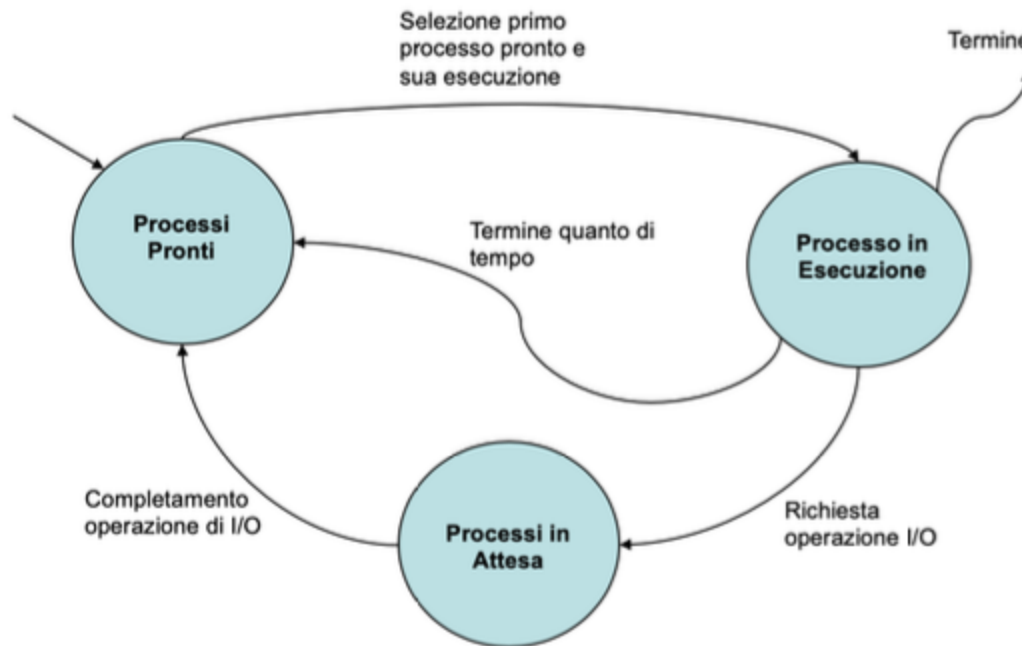
Ad ogni processo viene assegnato un quanto di tempo di CPU (time slice)

Terminato il quanto di tempo, il processo viene sospeso e rimesso nella coda dei processi pronti

La CPU viene assegnata ad un altro processo pronto

Un processo può usare meno del quanto che gli spetta se deve eseguire operazioni di I/O oppure ha terminato la sua computazione

Stati di un processo

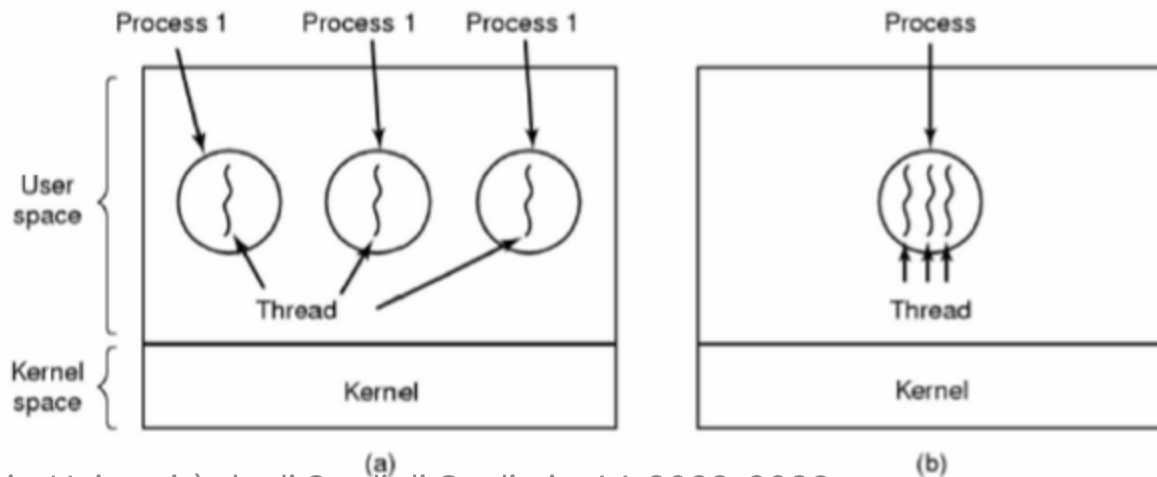


Thread

In un processo è possibile distinguere due componenti

1. Le risorse allocate al processo: files aperti, processi figli...
2. La sua esecuzione e tutti i dettagli relativi, stato della CPU...

È pensabile e vantaggioso avere più esecuzioni per lo stesso processo, che ne condividono le risorse. Ciascuna di queste è denominata **thread**



Thread

Benefici del multithreading

- La creazione e la terminazione di un thread risultano essere molto più “leggera” (condivisione di risorse)
- Modularità e flessibilità delle applicazioni
- Parallelismo di esecuzione: uno stesso processo può essere elaborato in parallelo da più processori
- Aumento del livello di multiprogrammazione: migliore utilizzazione della/e CPU

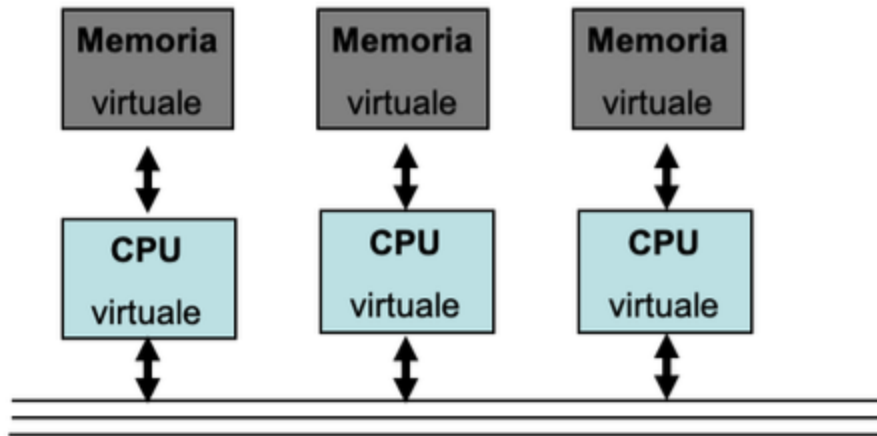
Gestore della memoria

- I programmi per essere eseguiti devono essere caricati nella memoria centrale
- Il SO coordina le operazioni per la gestione dei processi e per la conseguente allocazione della memoria

Il gestore della memoria si preoccupa di condividere la RAM tra i vari processi in esecuzione in modo che :

- ogni processo abbia il suo spazio privato distinto dagli altri (e inaccessibile agli altri)
- ogni processo abbia abbastanza memoria per eseguire il proprio algoritmo

Gestore della memoria



Gestore della memoria

Gestione statica

Ricopiare interamente lo spazio di indirizzamento di un programma da memoria secondaria a RAM all'inizio dell'esecuzione

PROBLEMA: non possono girare programmi con spazio di indirizzamento più grande della RAM disponibile!

Gestore della memoria

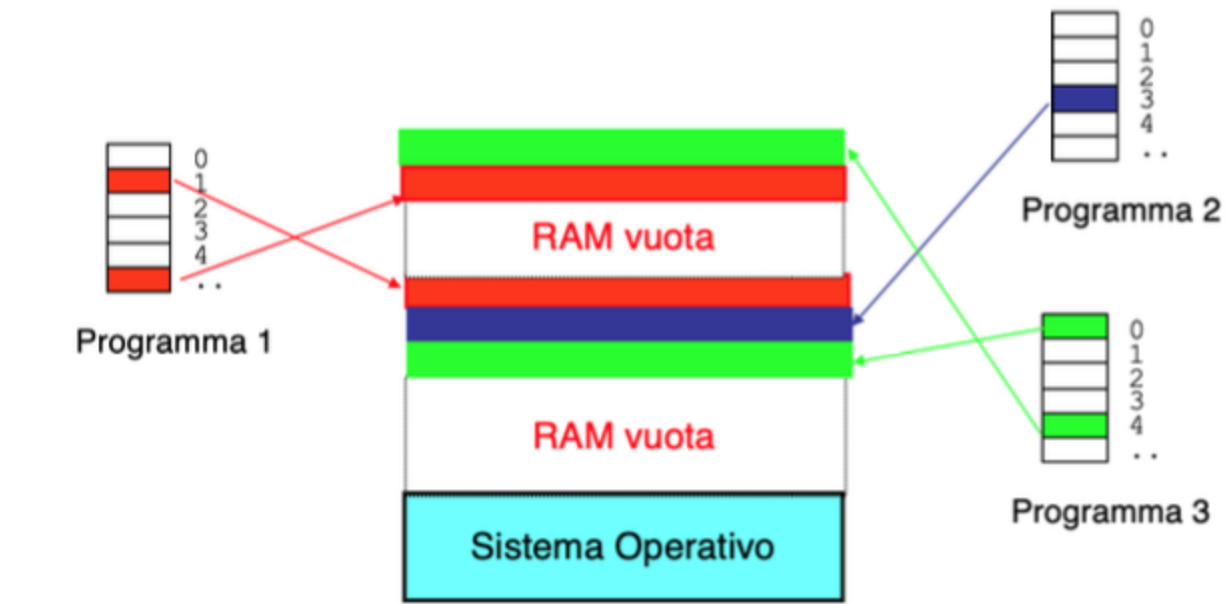
Gestione dinamica

In ogni istante il SO carica in RAM solo i pezzi che servono per l'esecuzione corrente

Paginazione: lo spazio di indirizzamento di ogni processo è diviso in *fette* (pagine) tutte della stessa ampiezza. In ogni istante solo le pagine necessarie sono caricate in memoria

Gestore della memoria

Gestione dinamica



Memoria Virtuale

- La dimensione di memoria (RAM) può essere anche superiore a quella fisicamente disponibile (memoria virtuale).
- Viene sfruttato lo spazio fisico di un dispositivo di memoria secondario (swap/paging)

Per realizzare questo compito il Gestore della Memoria svolge le seguenti funzioni:

- carica in memoria centrale il codice che deve essere eseguito e lo scarica quando non serve più;
- offre meccanismi di protezione per impedire ad ogni processo di modificare zone di memoria che non gli appartengono.

Gestore delle periferiche

Una delle funzioni principali di un SO è di controllare tutte le periferiche connesse al PC

Il SO deve:

- comandare i dispositivi
- ascoltare gli interrupt
- gestire gli errori
-

Deve fornire un'interfaccia tra i dispositivi ed il resto del sistema che sia semplice e facile da usare.

Il SO deve gestire la virtualizzazione delle periferiche.

Gestore delle periferiche

Le unità I/O sono generalmente composte da componenti meccanici e da componenti elettronici.

Il componente elettronico: device controller (adapter) è di solito una scheda che viene inserita nel PC che permette al PC stesso di interfacciarsi con il dispositivo.

Il componente meccanico: è il dispositivo stesso.

Il SO non tratta mai direttamente con il dispositivo ma sempre con il controller.

File System

- Fornire modo omogeneo per identificare nel sistema le risorse fisiche
- Garantire protezione negli accessi
- Informazioni memorizzate non per blocchi ma per nomi

Il FS viene implementato tramite un'associazione tra la struttura logica di un file e di una directory e il disco.

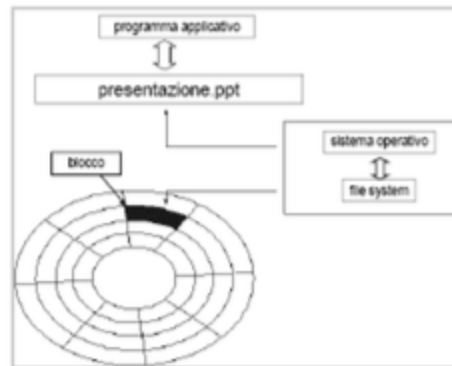
File System

File -> block 1001, 1002, ... (Physical)

name, owner, ... (Logic)

Un file è costituito da un record...

Nome	Attr.	Block 0	Block 1	..2	..3	..N
------	-------	---------	---------	-----	-----	-----



File System

FAT

La File Allocation Table (FAT) è una mappa di allocazione dei blocchi della memoria di massa.

Ogni elemento della mappa contiene uno fra i seguenti valori:

- l'identificatore di blocco vuoto;
- la posizione del prossimo blocco del file;
- l'identificatore di ultimo blocco.