

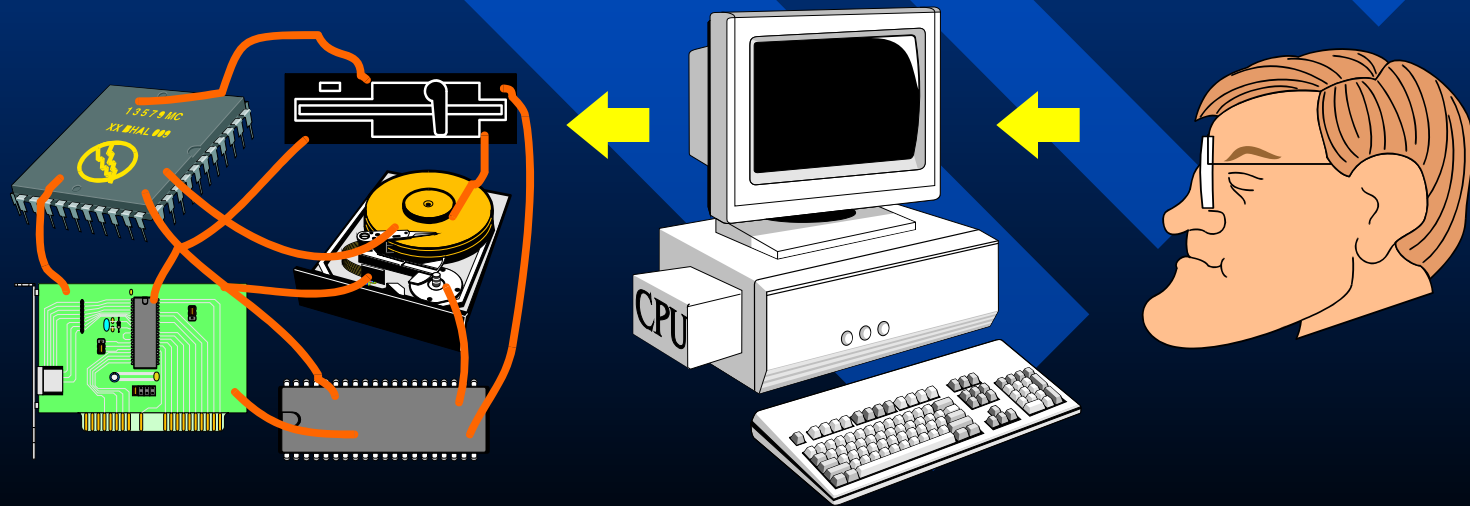
INTRODUZIONE AI SISTEMI OPERATIVI

Introduzione

- Il software può essere diviso in due grandi classi:
 - i *programmi di sistema* che gestiscono le operazioni del sistema di elaborazione
 - i *programmi applicativi* che risolvono i problemi dei loro utilizzatori;
- L'insieme dei *Programmi di Sistema* viene comunemente identificato con il nome di **Sistema Operativo (SO)**.

Scopo del Sistema Operativo

- Gestione delle risorse del sistema di elaborazione
- Rendere **AGEVOLE** l'interfaccia tra l'uomo e la macchina.



Attività svolte dal Sistema Operativo

- Gestione della memoria di massa (**file system**);
- Gestione della memoria **RAM**;
- Gestione dei **processi**;

Attività svolte dal Sistema Operativo

(Cont.)

- Gestione dell' interfaccia utente;
- Accesso **simultaneo** di più **utenti** alla stessa macchina;
- Esecuzione **simultaneamente** di più **processi** sulla stessa macchina.

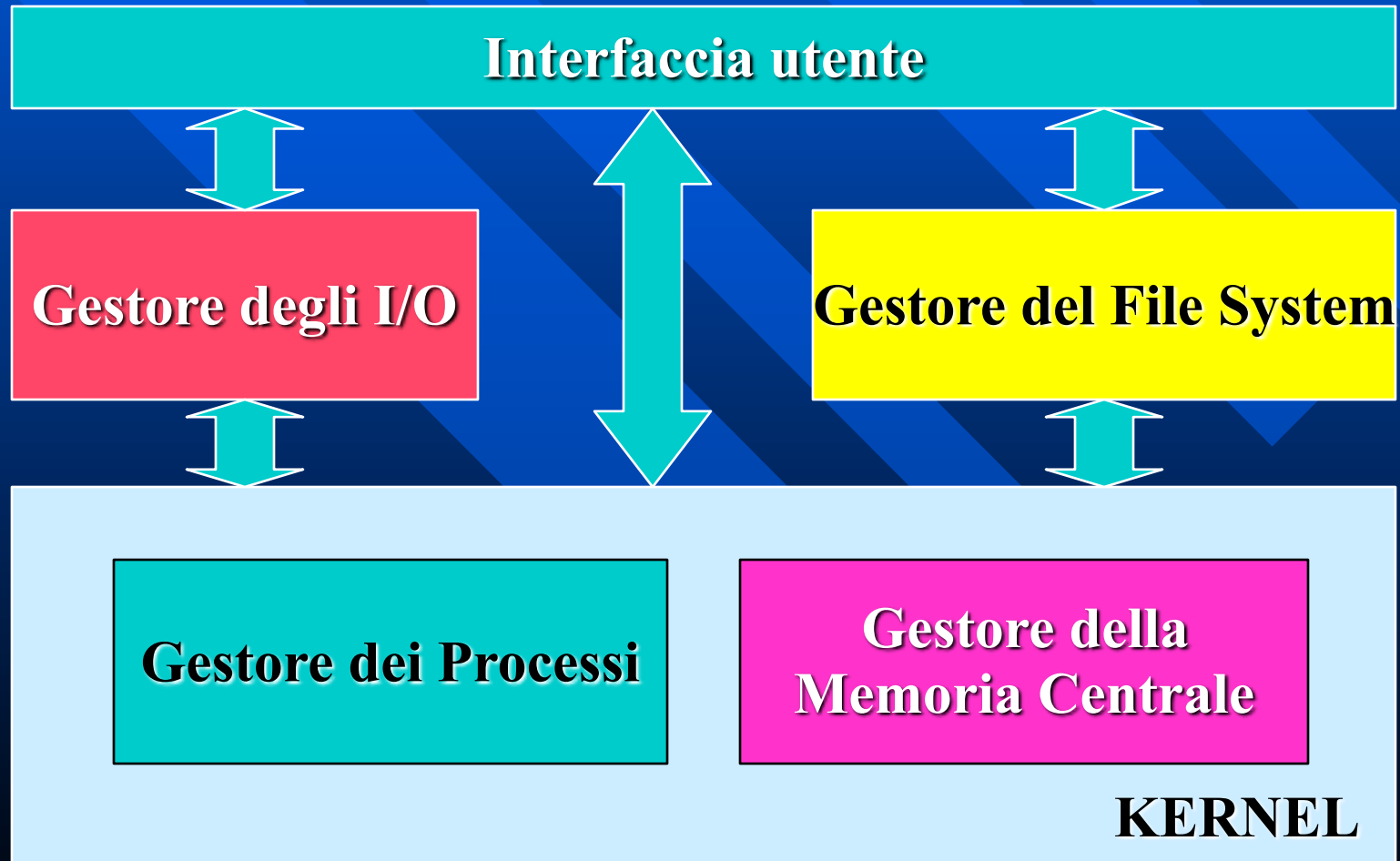
Aspetti importanti di un sistema operativo

- **Struttura:** come è organizzato il SO?
- **Condivisione:** quali risorse vengono condivise tra utenti e/o programmi? In che modo?
- **Efficienza:** come massimizzare l'utilizzo delle risorse disponibili?
- **Affidabilità:** come reagisce il SO a malfunzionamenti (HW/SW)?
- **Estendibilità:** è possibile aggiungere funzionalità al sistema?
- **Protezione e Sicurezza:** il SO deve impedire interferenze tra programmi/utenti. In che modo?
- **Conformità a standard:** portabilità, estendibilità, apertura

Struttura del Sistema Operativo

- I SO sono generalmente costituiti da un insieme di **moduli**, ciascuno dedicato a svolgere una *determinata funzione*;
- I vari moduli del SO **interagiscono** tra di loro secondo regole precise al fine di realizzare le funzionalità di base dalla macchina.

Struttura del Sistema Operativo (cont.)



Gestore dei Processi

Processi

Processo = programma in esecuzione

- il **programma** è un' *entità passiva* (un insieme di byte contenente le istruzioni che dovranno essere eseguite)
- il **processo** è un' *entità attiva*:
 - ❖ è l' *unità di lavoro/esecuzione* all'interno del sistema. Ogni attività all'interno del SO è rappresentata da un processo
 - ❖ è l' *istanza di un programma* in esecuzione

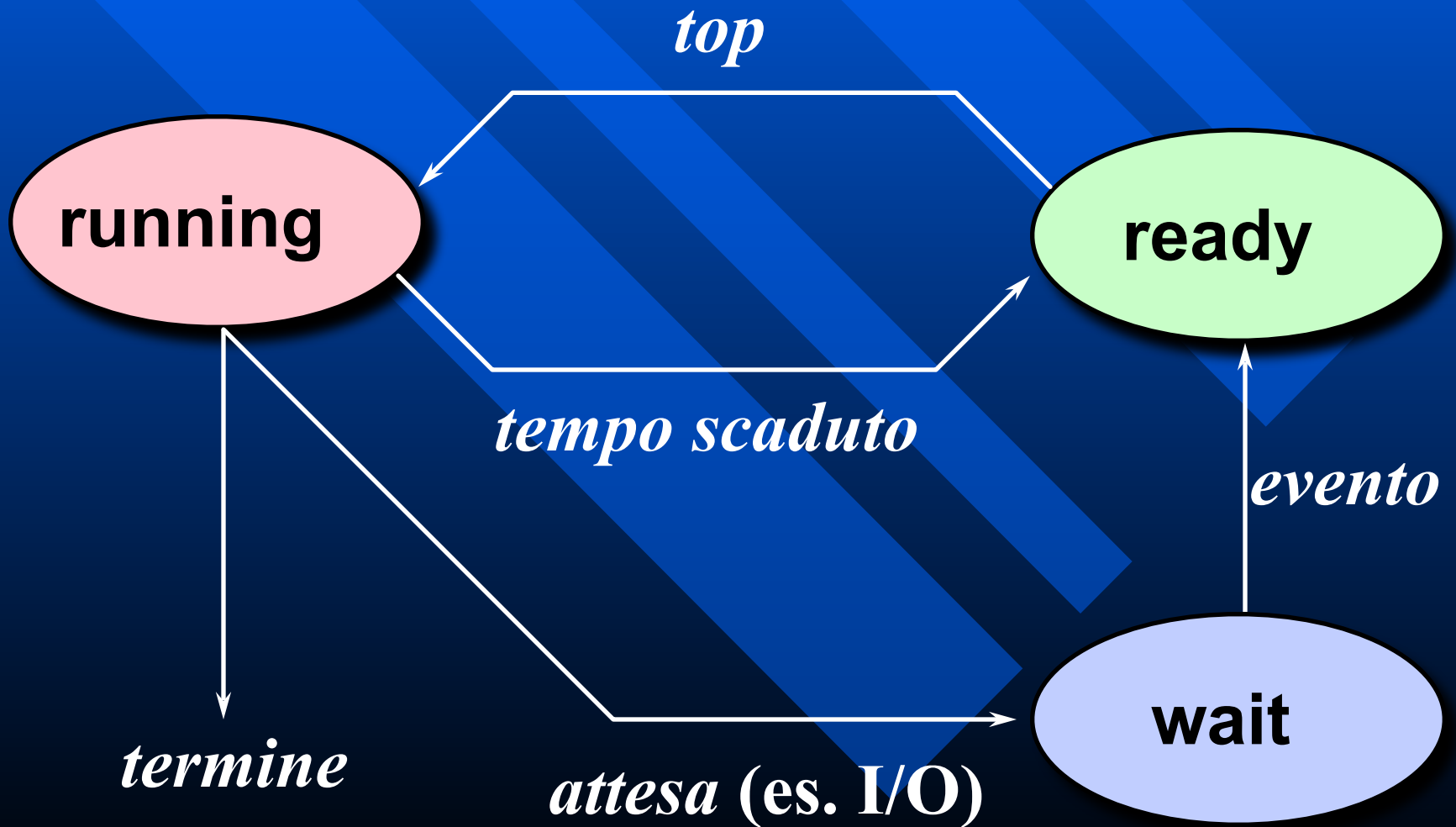
Gestore dei Processi

- E' il modulo che si occupa di **controllare** la *sincronizzazione*, *interruzione* e *riattivazione* dei programmi in esecuzione cui viene assegnato un processore;
- La gestione dei processi viene compiuta in vari modi, in funzione del tipo di utilizzo cui il sistema è rivolto.

Gestore dei Processi (Cont.)

- Il programma che si occupa della distribuzione del tempo di CPU tra i vari processi attivi, decidendone l'avvicendamento, è comunemente chiamato Scheduler.
- Nel caso di elaboratori multi-processore si occupa anche di gestire la cooperazione tra le varie CPU presenti nel sistema.

Schedulazione



Criteri per la scelta dello Scheduling

- Esistono vari algoritmi di scheduling che tengono conto di varie esigenze e che possono essere più indicati in alcuni contesti piuttosto che in altri.

Criteri per la scelta dello Scheduling

- La scelta dell'algoritmo da usare dipende da cinque principali criteri:
 - **Utilizzo del processore**: la CPU (ovvero il processore) deve essere attivo il più possibile, ovvero devono essere ridotti al minimo i possibili tempi morti.
 - **Produttività**: il numero di processi completati in una determinata quantità di tempo.
 - **Tempo di completamento**: il tempo che intercorre tra la sottomissione di un processo ed il completamento della sua esecuzione.
 - **Tempo d'attesa**: il tempo in cui un processo pronto per l'esecuzione rimane in attesa della CPU.
 - **Tempo di risposta**: il tempo che trascorre tra la sottomissione del processo e l'ottenimento della prima risposta.

Politiche di Scheduling

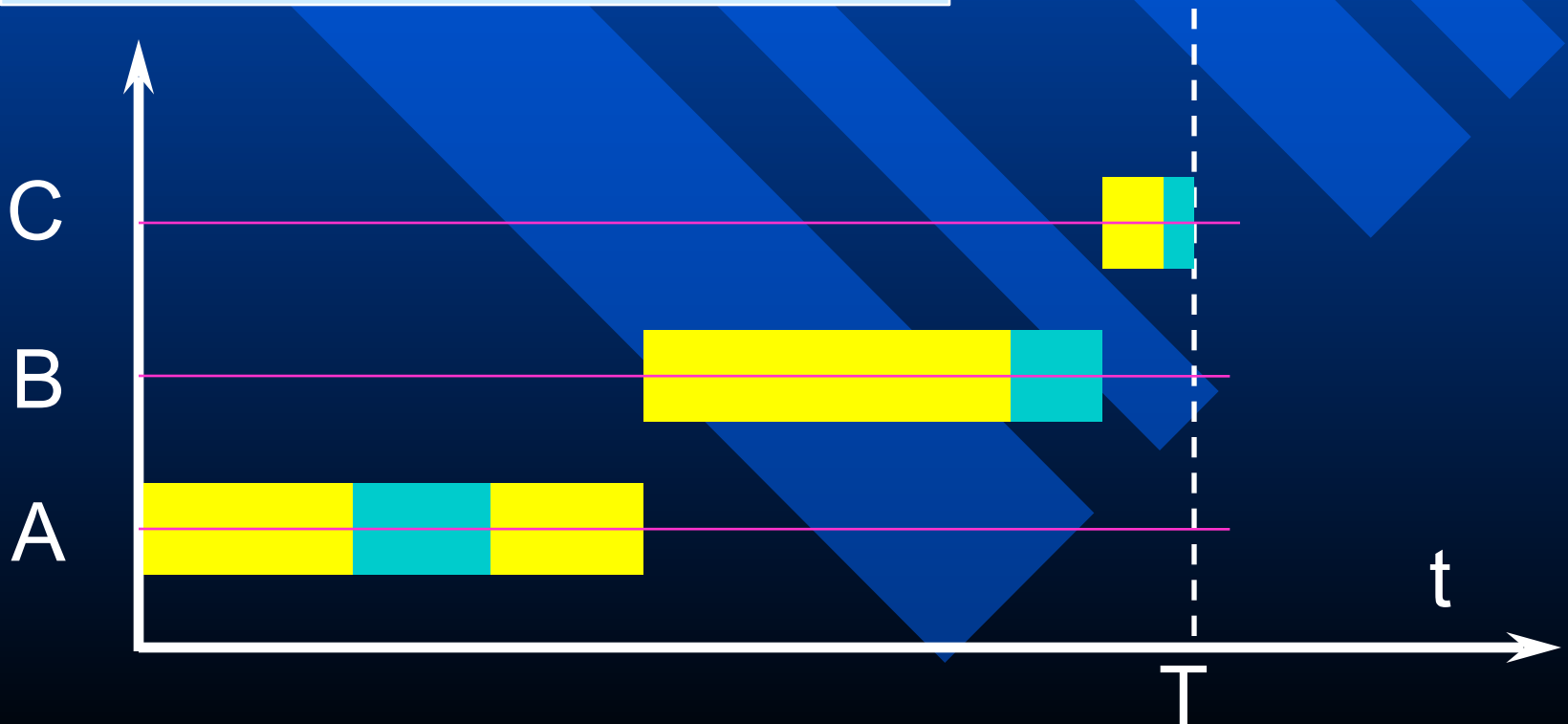
- Le politiche di schedulazione utilizzate dallo scheduler sono raggruppabili in due grandi categorie:
 - **Preemptive**: la CPU in uso da parte di un processo può essere tolta e passata a un altro in un qualsiasi momento;
 - **Non Preemptive**: una volta che un processo ha ottenuto l'uso della CPU non può essere interrotto fino a che lui stesso non la rilascia.

Sistemi Mono-Tasking

- I SO che gestiscono l'esecuzione di un solo programma per volta sono catalogati come *mono-tasking*;
- Non è possibile sospendere l'esecuzione di un programma per assegnare la CPU a un altro
- Sono storicamente i primi SO (es MS-DOS).

Sistema Mono-Tasking

- Tempo di utilizzo della CPU.
- Tempo di attesa di eventi esterni.

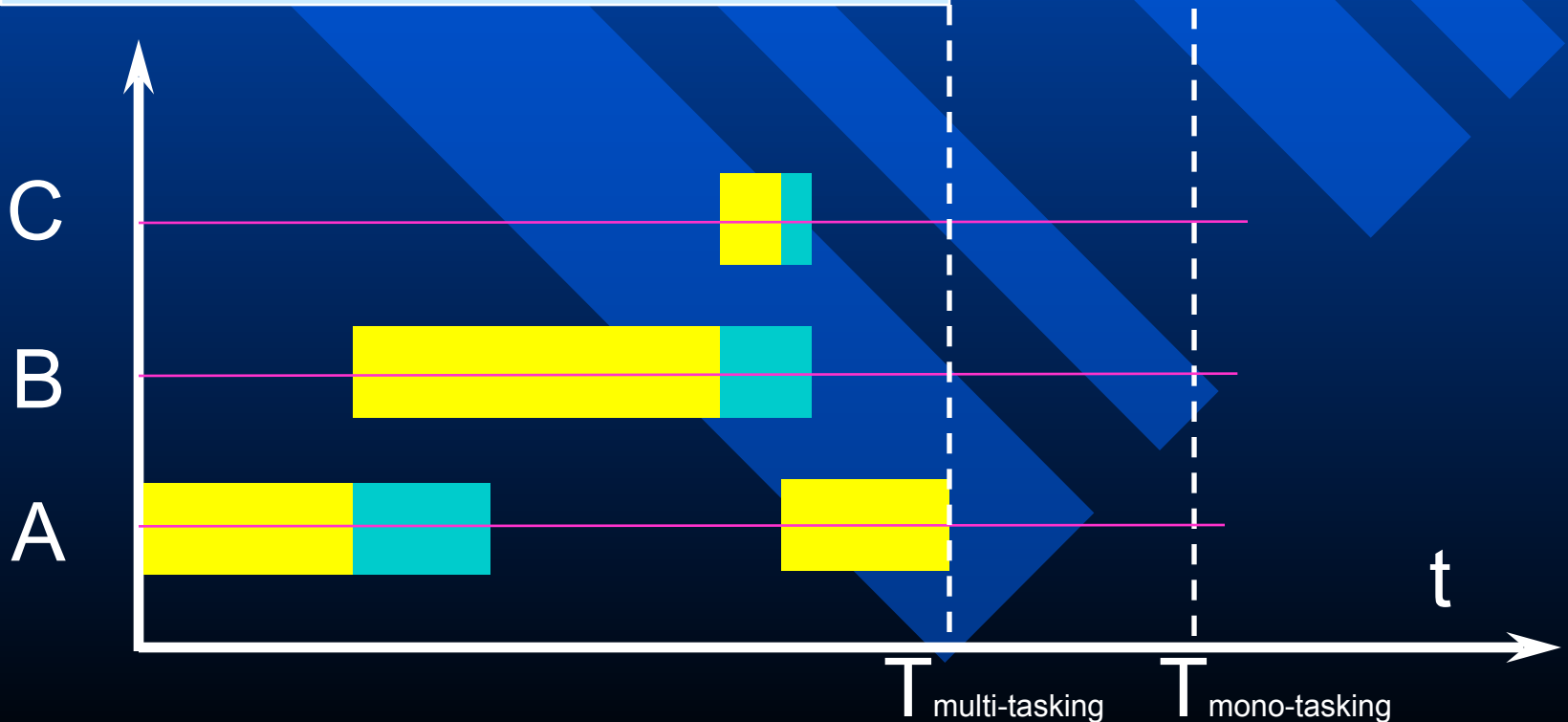


Sistemi Multi-Tasking

- I SO che permettono l'esecuzione contemporanea di più programmi sono definiti *multi-tasking* (Windows-NT , Linux);
- Un programma può essere interrotto e la CPU passata a un altro programma

Sistema Multi-Tasking

- Tempo di utilizzo della CPU.
- Tempo di attesa di eventi esterni.

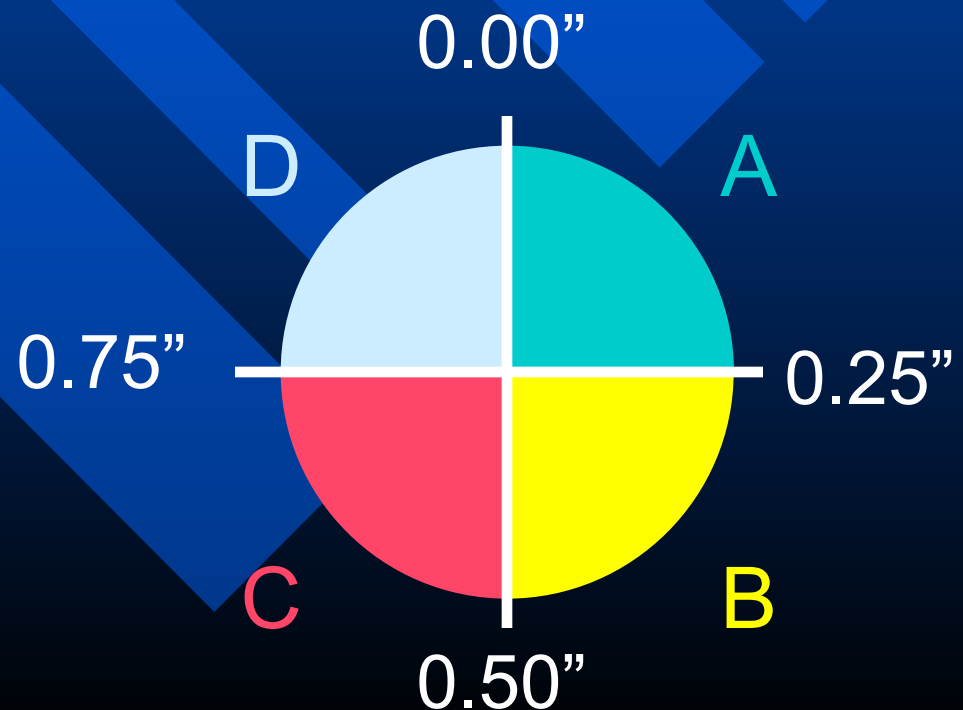


Sistemi Time-Sharing

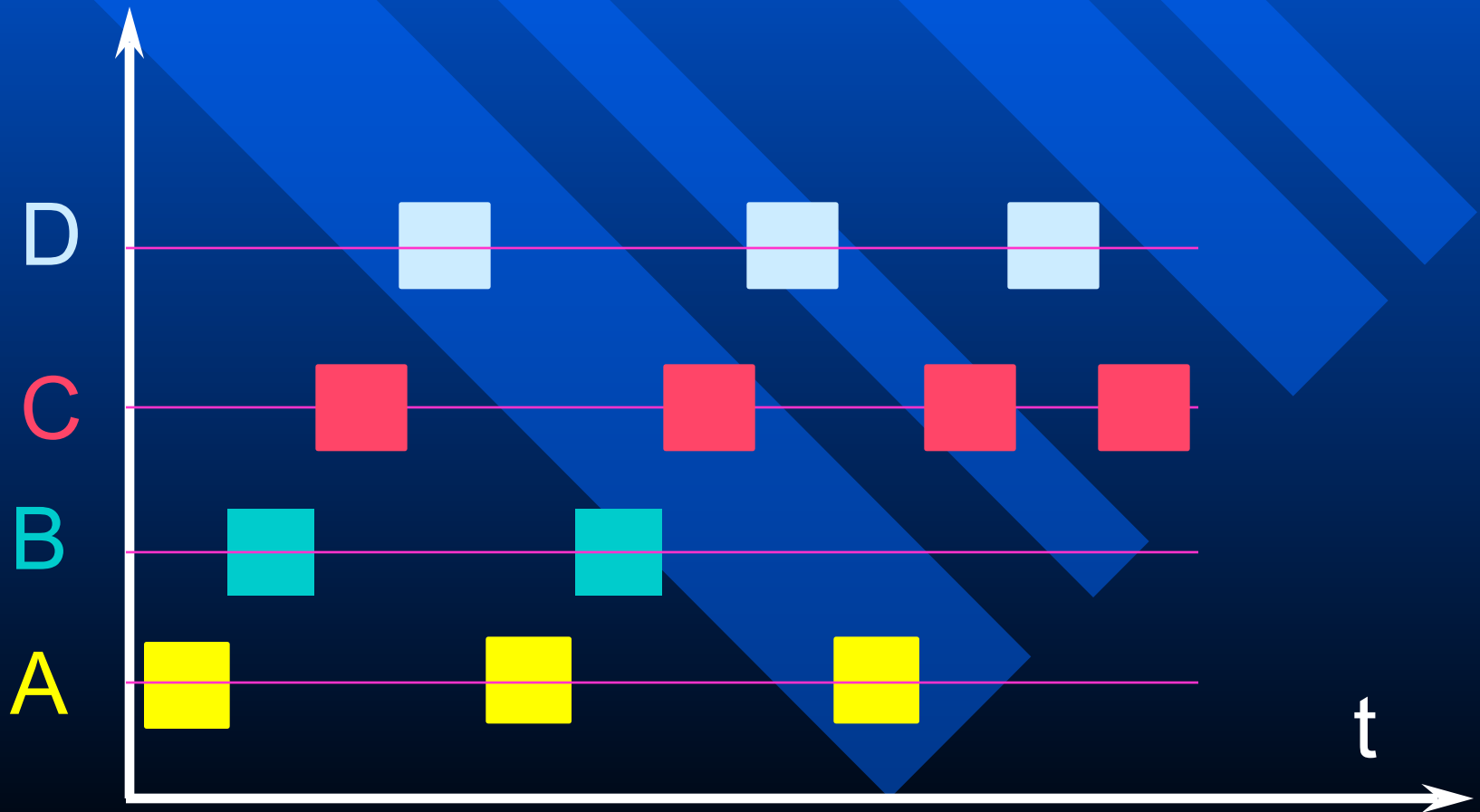
- Un'evoluzione dei sistemi multi-tasking sono i sistemi *time sharing*.
- Ogni programma in esecuzione viene eseguito ciclicamente per piccoli *quanti di tempo*.
- Se la velocità del processore è sufficientemente elevata si ha l'impressione di un'evoluzione parallela dei processi.

Time-sharing: schema

- Ipotesi: 1 MIPS, 4 processi, 0.25 s/utente
- Conseguenze:
 - 0.25 MIPS/utente
 - $T_{ELA} = 4 \times T_{CPU}$



Time-sharing: diagramma temporale



Gestore della Memoria

Gestore della Memoria

- L'organizzazione e la gestione della memoria centrale è uno degli aspetti più critici nel disegno di un SO;
- Il *gestore della memoria* è quel modulo del SO incaricato di assegnare la memoria ai vari task (per eseguire un task è necessario che il suo codice sia caricato in memoria);

Gestore della Memoria (Cont.)

- La **complessità** del gestore della memoria dipende dal **tipo** di SO;
- Nei sistemi **multi-tasking** più programmi **contemporaneamente** possono essere caricati in memoria.
- **Problema**: come allocare lo spazio in maniera **ottimale**

Allocazione Lineare

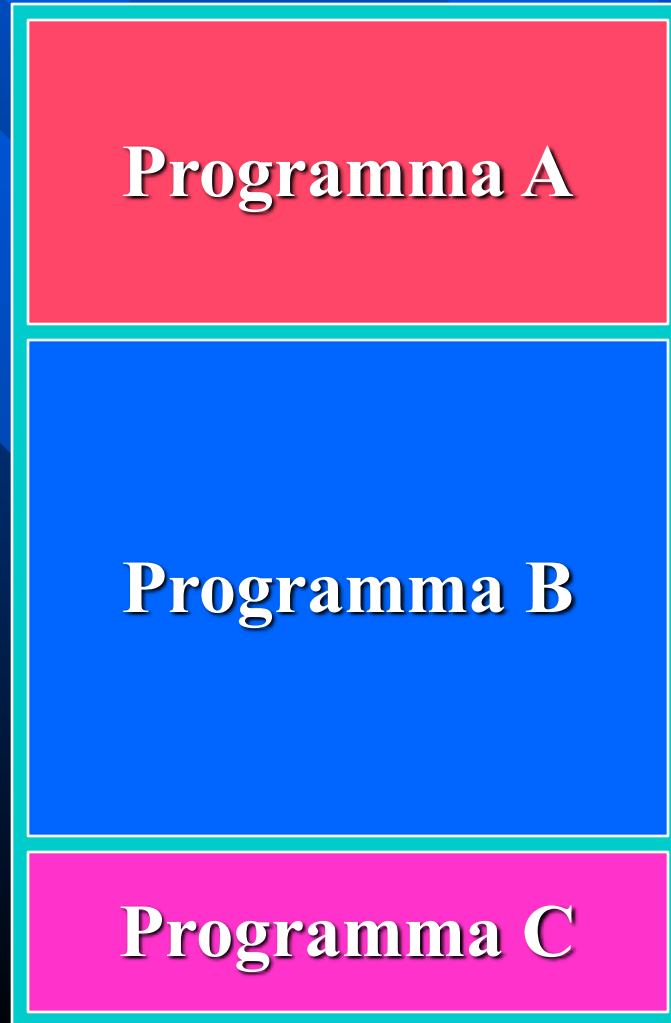
Memoria

0000x

Programma A

Programma B

Programma C



Allocazione Lineare

Memoria

0000x



PROBLEMA !!!!

FRAMMENTAZIONE

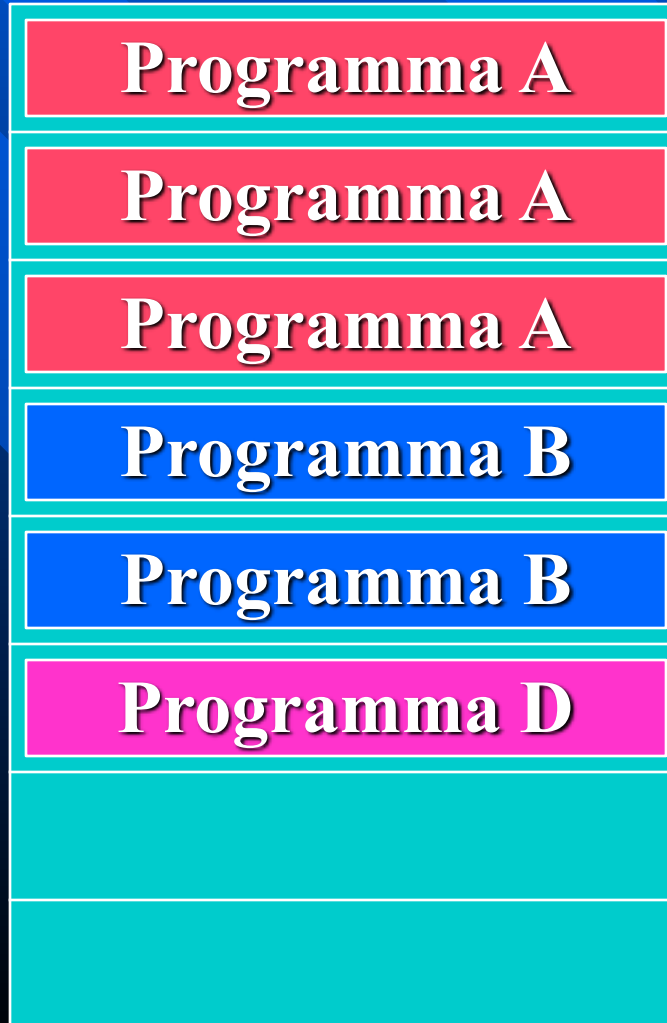


Programma C

Paginazione

Memoria

0000x



Paginazione

Memoria

0000x



Memoria Virtuale

- Spesso la memoria non è sufficiente per contenere completamente tutto il codice dei vari task;
- Si può *simulare* una memoria più grande tenendo nella memoria di sistema (RAM) solo le parti di codice e dei dati che servono in quel momento;
- Si usa il concetto di *memoria virtuale*.

Memoria Virtuale (Cont.)

- I dati dei programmi non in esecuzione possono essere tolti dalla memoria centrale e parcheggiati su disco nella cosiddetta *area di swap*;
- Il rapporto tra le dimensioni dell'area di swap e della RAM è di **3 : 1** (max);
- I moderni processori posseggono meccanismi hardware per facilitare la gestione della memoria virtuale.

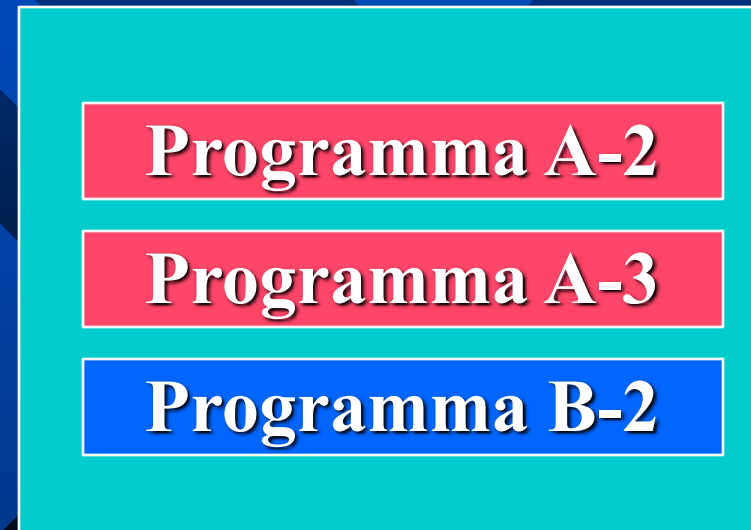
Memoria Virtuale

Memoria

0000x



Swap



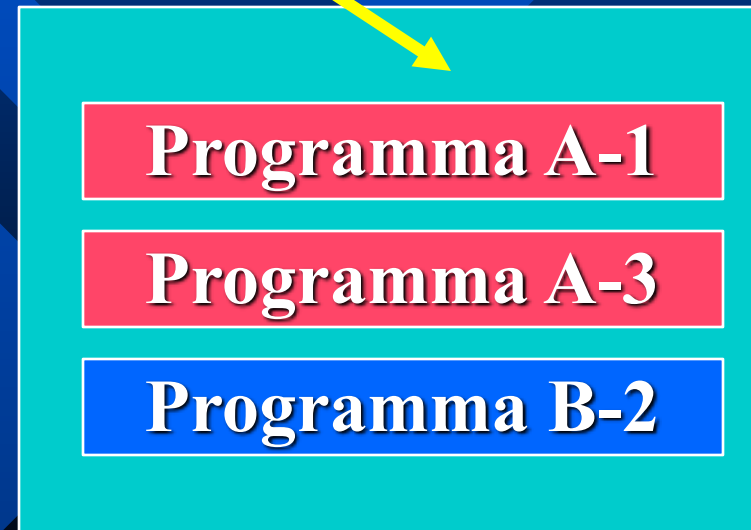
Memoria Virtuale

Memoria

0000x



Swap



Gestore del File System

Gestore del File System

- Il *gestore del file system* è quel modulo del sistema operativo incaricato di gestire le informazioni memorizzate sui dispositivi di memoria di massa;
- Il gestore del file system deve garantire la correttezza e la coerenza delle informazioni;

Gestore del File System (Cont.)

- Nei sistemi **multi-utente**, deve mettere a disposizione dei **meccanismi di protezione** in modo tale da consentire agli utenti di **proteggere** i propri dati dall'**accesso** da parte di altri utenti **non autorizzati**.

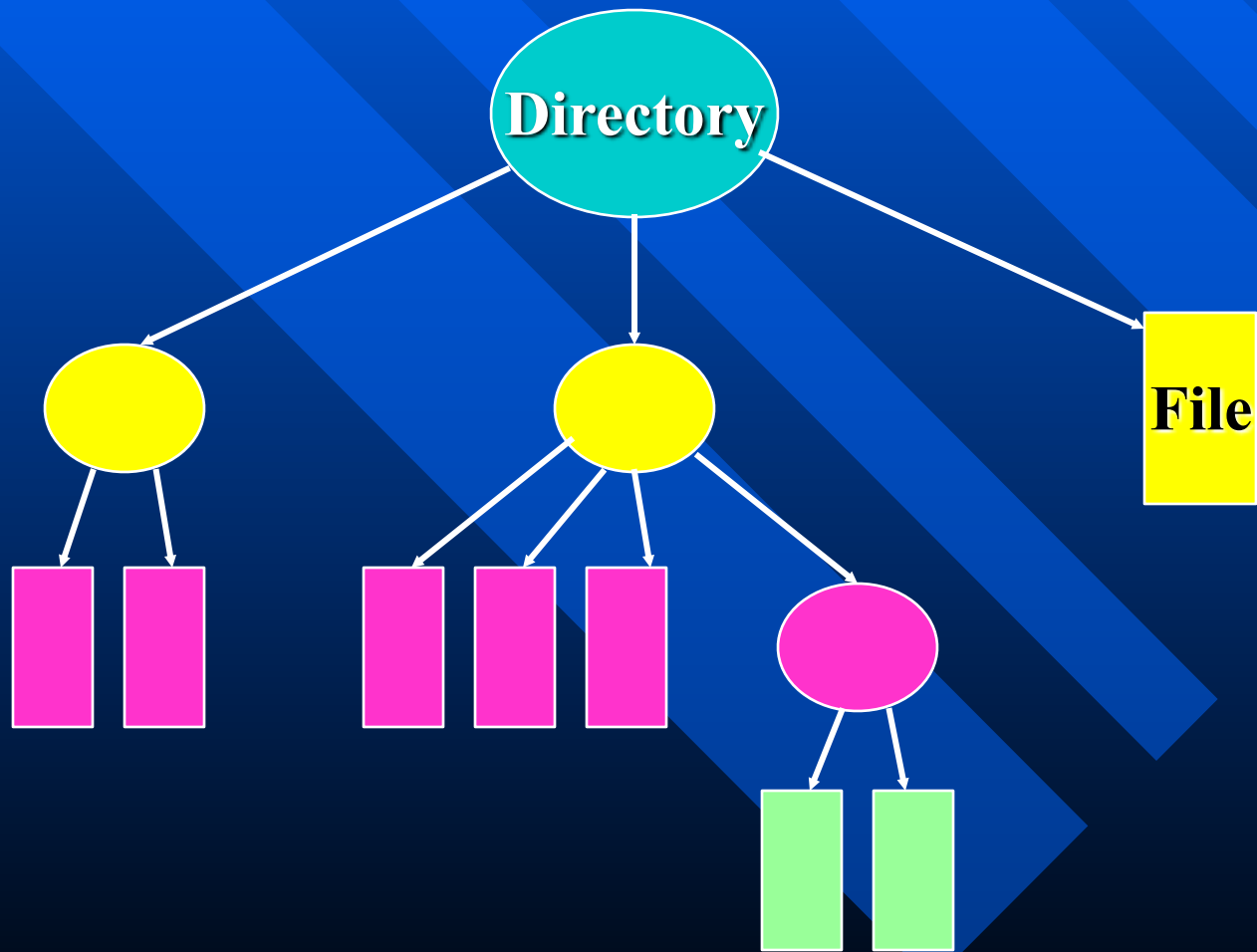
Gestore del File System (Cont.)

- Le *funzioni* tipiche che deve svolgere sono:
 - Fornire un meccanismo per *l'identificazione* dei Files;
 - Fornire opportuni *metodi* per *accedere* ai dati;
 - Rendere *trasparente* la *struttura fisica* del supporto di *memorizzazione*;
 - Implementare *meccanismi di protezione* dei *dati*.

Organizzazione

- Quasi tutti i sistemi operativi utilizzano un'organizzazione *gerarchica* del File System;
- L'elemento utilizzato per raggruppare più file insieme è la *directory*;
- L'insieme gerarchico delle directory e dei file può essere rappresentato attraverso un *grafo* delle directory.

Grafo delle Directory



Gestore dei Dispositivi di I/O

Gestore dei dispositivi di I/O

- Il *gestore dei dispositivi di I/O* è quel modulo del SO incaricato di **assegnare** i **dispositivi** ai **task** che ne fanno richiesta e di controllare i dispositivi stessi;
- Da esso dipende la **qualità** e il **tipo di periferiche** riconosciute dal sistema.

Device Driver

- Il controllo dei dispositivi di I/O avviene attraverso speciali programmi detti *Device Driver*;
- I device driver sono spesso realizzati dai produttori dei dispositivi stessi che ne conoscono le caratteristiche fisiche in maniera approfondita.

Device Driver (Cont.)

- Questi programmi **implementano** normalmente le seguenti funzioni:
 - Rendono **trasparenti** le caratteristiche fisiche tipiche di ogni dispositivo;
 - Gestiscono la **comunicazione** dei segnali verso i dispositivi;
 - Gestiscono i **conflitti**, nel caso in cui due o più task vogliono accedere contemporaneamente allo stesso dispositivo.

Interfaccia Utente

Interfaccia utente

- Tutti i Sistemi Operativi implementano dei meccanismi per rendere agevole l'utilizzo del sistema da parte degli utenti;
- L'insieme di questi meccanismi di accesso al computer prende il nome di *Interfaccia Utente*.

Interfaccia utente (Cont.)

- *Interfaccia testuale:*
 - Interprete dei comandi (shell)
 - Esempio MS-DOS e/o Linux
- *Interfaccia grafica* (a finestre):
 - L'output dei vari programmi viene visualizzato in maniera grafica all'interno di finestre
 - L'utilizzo di disegni rende più intuitivo l'uso del calcolatore;
 - Esempio WINDOWS

I Sistemi Presenti in Commercio

I Sistemi Operativi presenti in commerciali

- In commercio sono presenti una grande quantità di diversi Sistemi Operativi;
- In passato la tendenza delle case costruttrici di sistemi di elaborazione era di sviluppare sistemi operativi proprietari per le loro architetture;
- La tendenza attuale è quella di sistemi operativi eseguibili su diverse piattaforme.

MS-DOS

- CPU Intel 80x86 (16 bit)
- monotask
- monoutente
- file-system gerarchico
- memoria limitata (1 MB / 640 KB)
- nessuna protezione
- PC- / IBM- / DR-DOS

MS-Windows

- CPU Intel 80386/486/Pentium
- multitask imperfetto
(non ha la *preemption*)
- monoutente
- stesso file system del MS-DOS
- interfaccia grafica a finestre e menù
- sistema a 16 bit !!!

Windows-NT

- CPU Intel 80386/486/Pentium/Sparc/Alfa
- multitask
- monoutente
- NTFS (NT File System)
- microkernel, thread
- non solo per Intel 80x86
(DEC-AXP, MIPS-R4000, ...)
- sistema a 32 bit

UNIX

- nato negli anni '60 (AT&T Bell Labs)
- rimasto all'avanguardia perchè sviluppato nelle università (UCB)
- multitask
- multiutente
- ottima integrazione in rete
- portabilità dei programmi