

Il software

L'insieme dei programmi che un elaboratore può eseguire

Scopo della lezione

- Descrivere il **software di base** e le sue funzionalità principali
 - la gestione della memoria centrale per l'esecuzione di più programmi simultaneamente
 - il file system come organizzazione logica delle informazioni sul disco
 - la gestione degli utenti con privilegi diversi

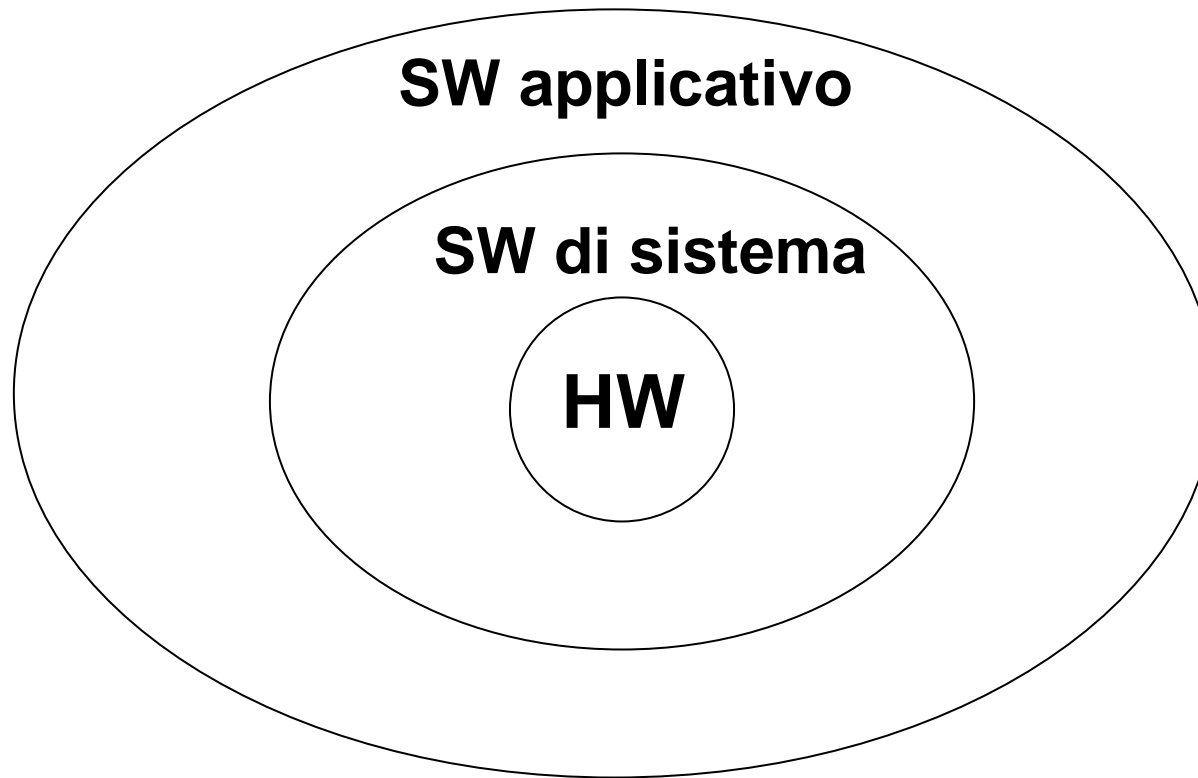
Il software

- Distinguiamo due principali categorie di prodotti software
 - **sw di sistema**, o di base
 - **sw applicativo**

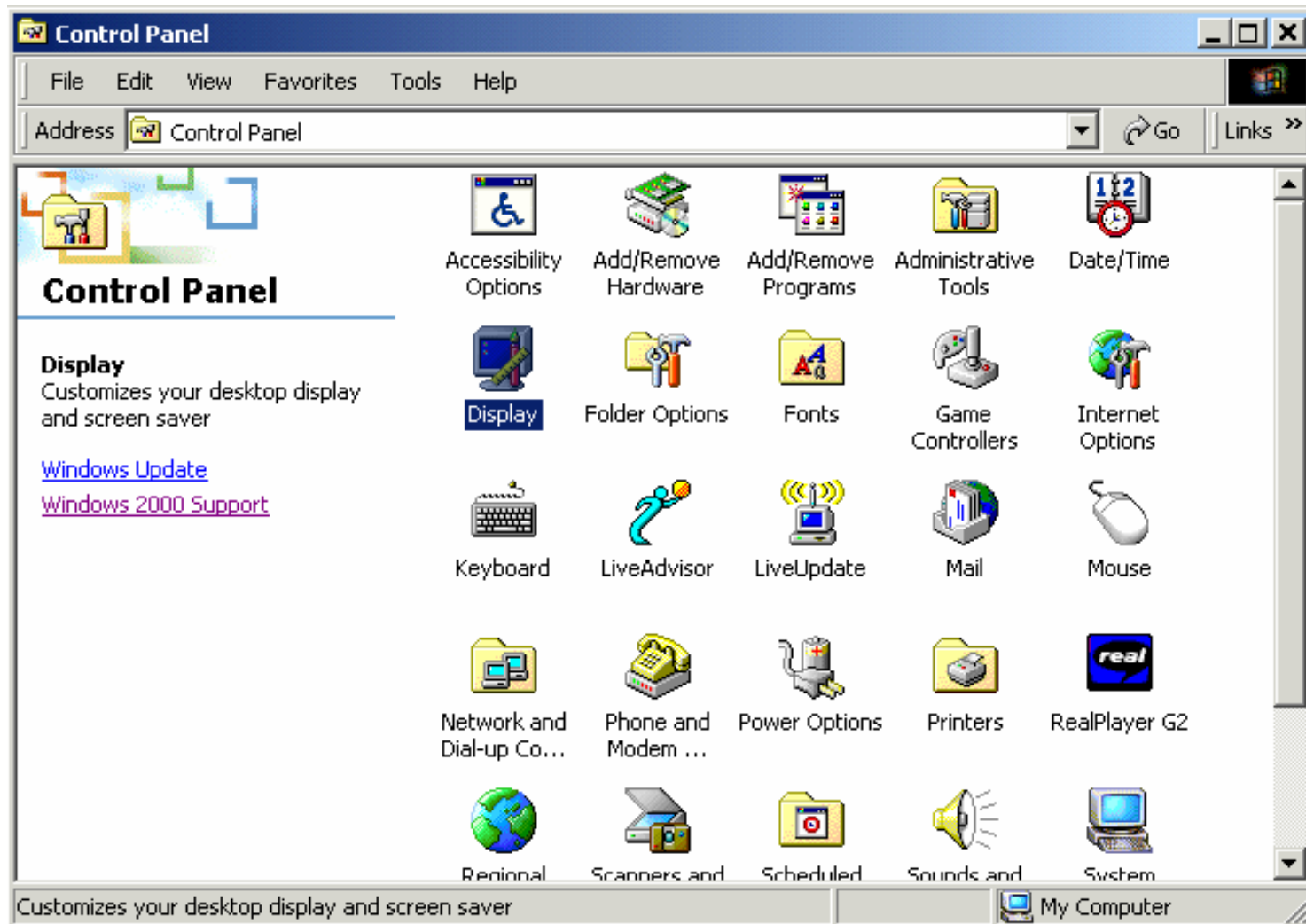
- Il **sw applicativo** è costituito dall'insieme di programmi che “fanno svolgere” al calcolatore una serie di funzionalità rivolte principalmente agli utilizzatori finali
 - video scrittura
 - foglio elettronico
 - contabilità, paghe e stipendi
 - visualizzazione grafica
 - posta elettronica
 - navigazione in rete
 - ecc.
- E' generato per una certa CPU e per un certo sistema operativo
 - la versione Wintel (Windows su Intel) di Excel non puo` essere eseguita su un Macintosh.
- La versione iniziale, in linguaggio ad alto livello, viene tradotta in un certo linguaggio macchina con un dato S.O.

- Il **sw di sistema** (sistema operativo – S.O.) è costituito da un insieme di programmi il cui scopo è
 - facilitare l'uso della macchina nascondendone alcuni dettagli;
 - garantire l'uso “onesto” della macchina da parte di più utenti allo stesso tempo;
 - fornire funzionalità per la stesura di programmi applicativi, facilitando quindi il lavoro dei programmatori.
- Appartengono al sw di sistema
 - il sistema operativo
 - i driver per i dispositivi di I/O
 - i compilatori
 - programmi di “utilità” per lo sviluppo del software e la gestione dei computer

L'architettura generale



Il sistema operativo



Il sistema operativo

Il sistema operativo viene fornito (memorizzato su un CD) insieme all'elaboratore: **ogni elaboratore può essere utilizzato solo se vi è stato preventivamente installato un sistema operativo.**

Dopo l'installazione una copia del sistema operativo risiede sul disco fisso.

- Un insieme di programmi che
 - gestiscono le risorse (cpu, memoria centrale, spazio su disco, periferiche) di un elaboratore permettendo:
 - esecuzione di più programmi simultaneamente;
 - accesso a più utenti simultaneamente;
 - protezione di dati e programmi da comportamenti scorretti degli utenti;
 - rendono più facile l'uso del calcolatore a programmatori e utenti finali poiché forniscono
 - funzionalità di alto livello che mascherano l'hw ai programmatori
 - interfaccia user friendly per gli utenti finali

- I sistemi operativi più diffusi in ambiente PC sono
 - Windows
 - Linux
 - MacOSX

1. Singolo utente e un solo programma in esecuzione:
 - MS-DOS
2. Singolo utente e vari programmi in esecuzione contemporaneamente
 - W95/98
3. Molti utenti, ciascuno esegue vari programmi diversi contemporaneamente
 - WNT, Unix, Linux, MacOSX



EVOLUZIONE

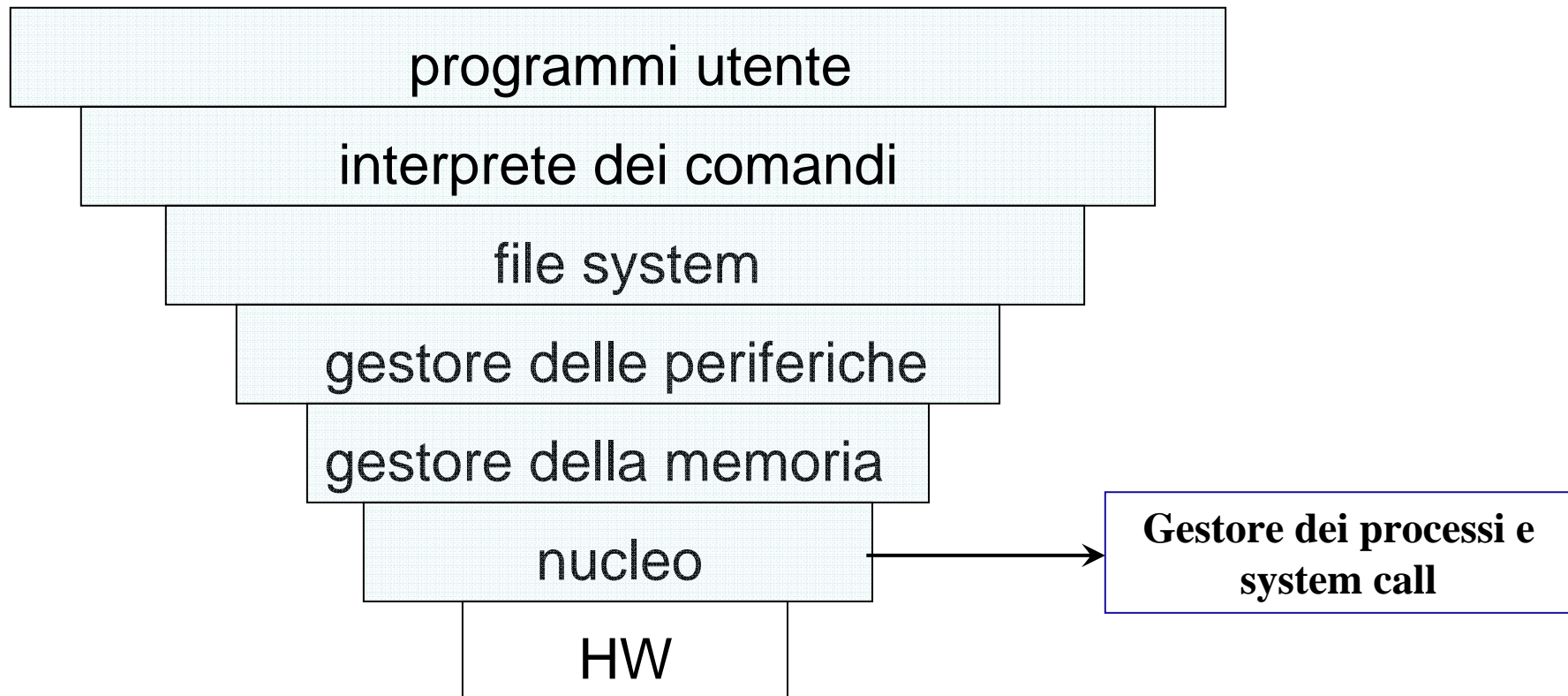
Versioni

- Contraddistinte dal nome e dalla **release**, specificata da due numeri **X.Y**
 - **X** rappresenta la versione principale del sistema operativo
 - rilasciata per correggere errori e aggiungere nuove funzionalità alla release precedente
 - **Y** il livello di aggiornamento
 - rilasciata per apportare modifiche minori alla release corrente
- Gli aggiornamenti di sistema operativo, o più in generale di sw, vengono chiamati
 - **patch** nel mondo Unix/Linux
 - **service pack** nel mondo Windows
 - **update** nel mondo MacOSX
- In genere sono distribuiti attraverso CD o attraverso la rete

- Anche il software applicativo è caratterizzato da un numero di versione
- Ogni versione di sw applicativo è compatibile solo con certe versioni di sistema operativo
- Prima di usare un sw applicativo è necessario verificare la versione di S.O. di cui si dispone

S.O. - Struttura logica e componenti

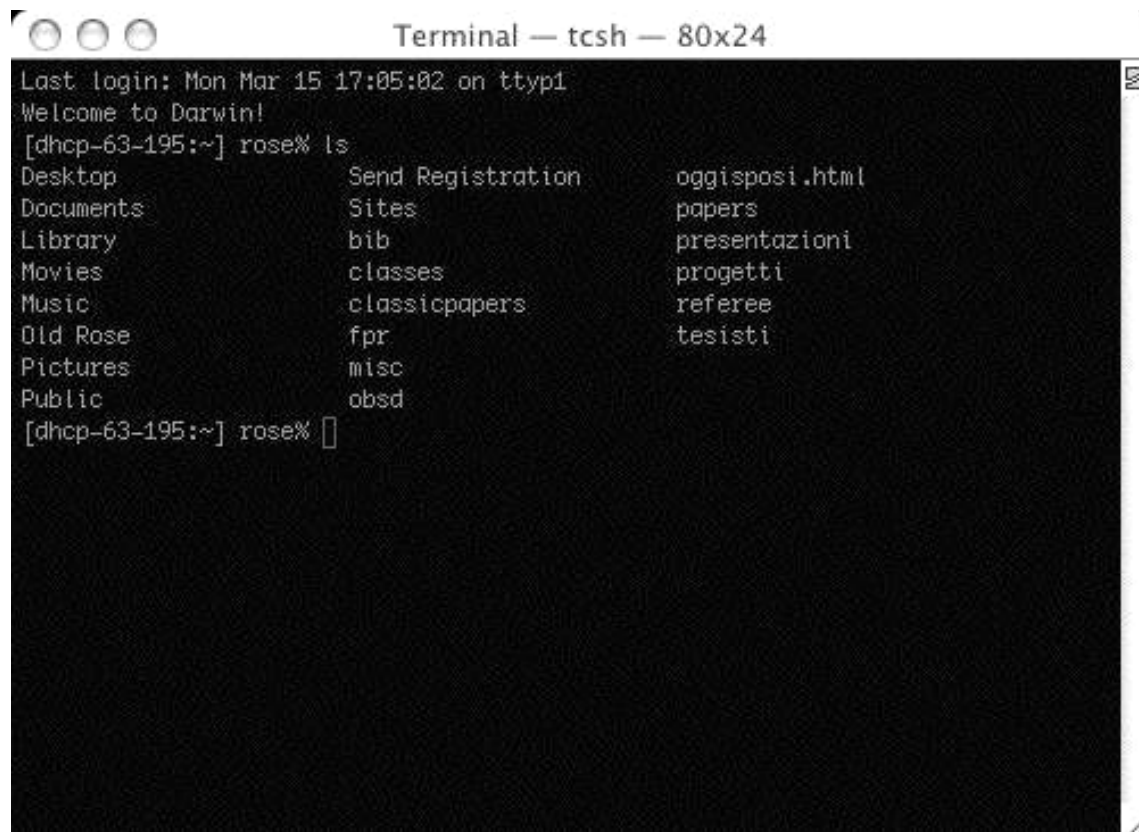
- Data la sua complessità, il sistema operativo è suddiviso logicamente in strati funzionali
 - ciascuno strato offre dei servizi agli strati superiori e nasconde quello che sta sotto



S.O. - Interprete dei comandi

E' l'interfaccia di alto livello verso l'utente mediante la quale si chiede l'esecuzione dei programmi o di comandi. Può essere:

- Interfaccia a caratteri sulla "linea di comando"
 - shell, tipica di ambienti Unix-like
 - una per ciascun "terminale" aperto
 - Xwindow



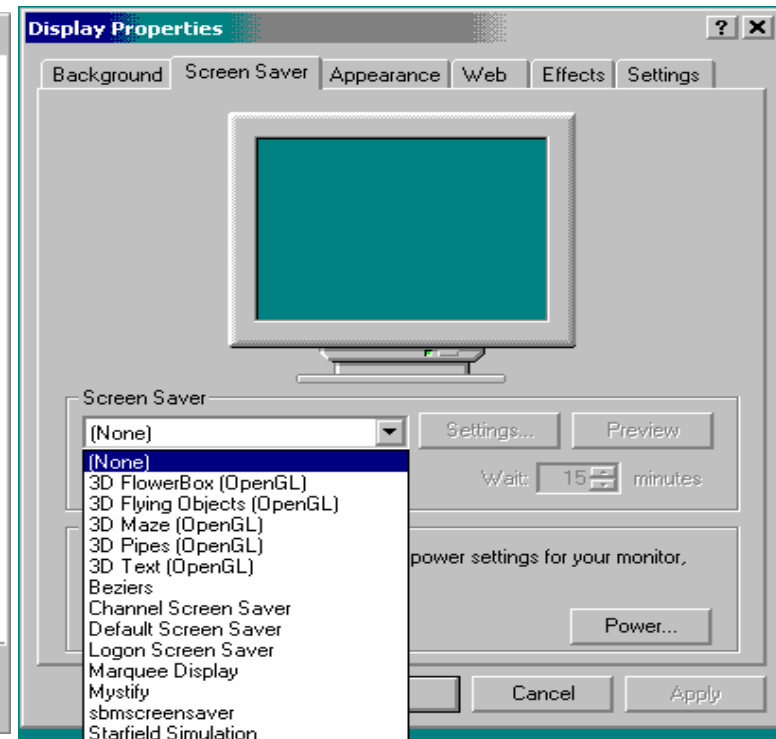
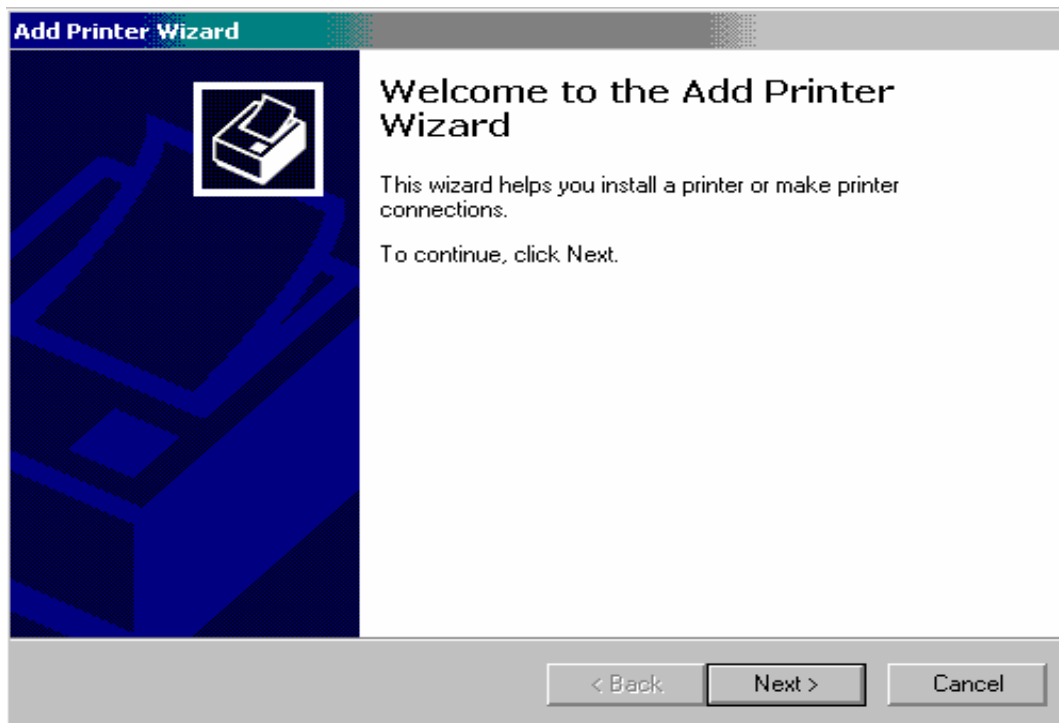
```
Terminal — tcsh — 80x24
Last login: Mon Mar 15 17:05:02 on ttys1
Welcome to Darwin!
[dhcp-63-195:~] rose% ls
Desktop          Send Registration  oggisposi.html
Documents        Sites              papers
Library          bib                presentazioni
Movies           classes            progetti
Music            classicpapers     referee
Old Rose         fpr                tesisti
Pictures         misc
Public           obsd
[dhcp-63-195:~] rose% [
```

- grafica con finestre e icone
 - Macintosh
 - KDE/Gnome, MacOSX, Windows



S.O. - Gestore delle periferiche

- Raccolta di driver per i vari tipi di periferica
 - lettura/scrittura
 - controllo degli accessi
- Strumenti per l'aggiunta/rimozione di nuovi driver
 - Add printer wizard, PrintCenter, ecc.
- Richiede la conoscenza delle specifiche hw dei dispositivi

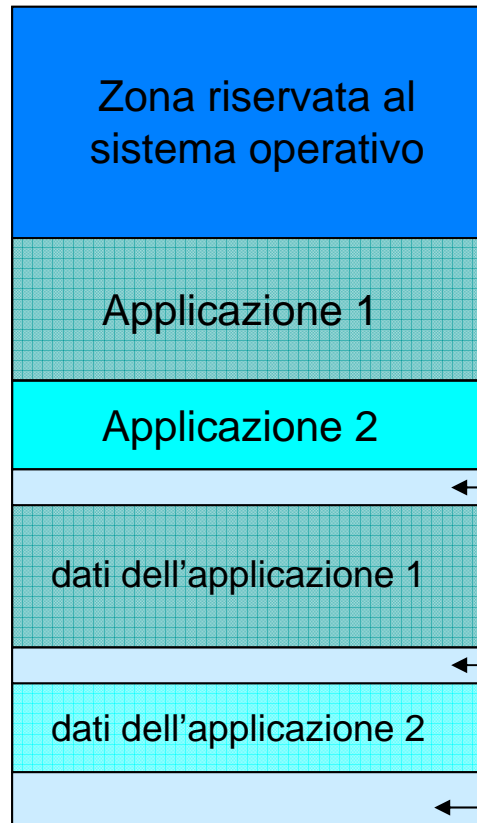


S.O.- Gestore della memoria

E' una raccolta di procedure che permettono la condivisione della memoria tra programmi eseguiti "simultaneamente". Permette di:

- assegnare la memoria (fisica e/o virtuale) ai programmi da eseguire
- spostare i programmi in memoria sul disco fisso per fare posto a quelli nuovi
- caricare i programmi da disco fisso (o da memorie di massa) in memoria
- assicurare che programmi diversi usino aree di memoria separate e impermeabili per evitare errori

Memoria centrale

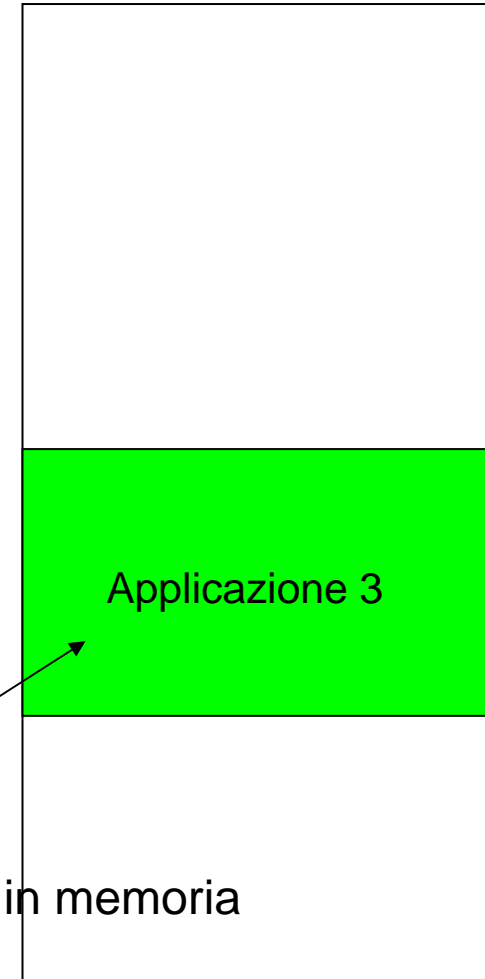


Parte di memoria rimasta libera



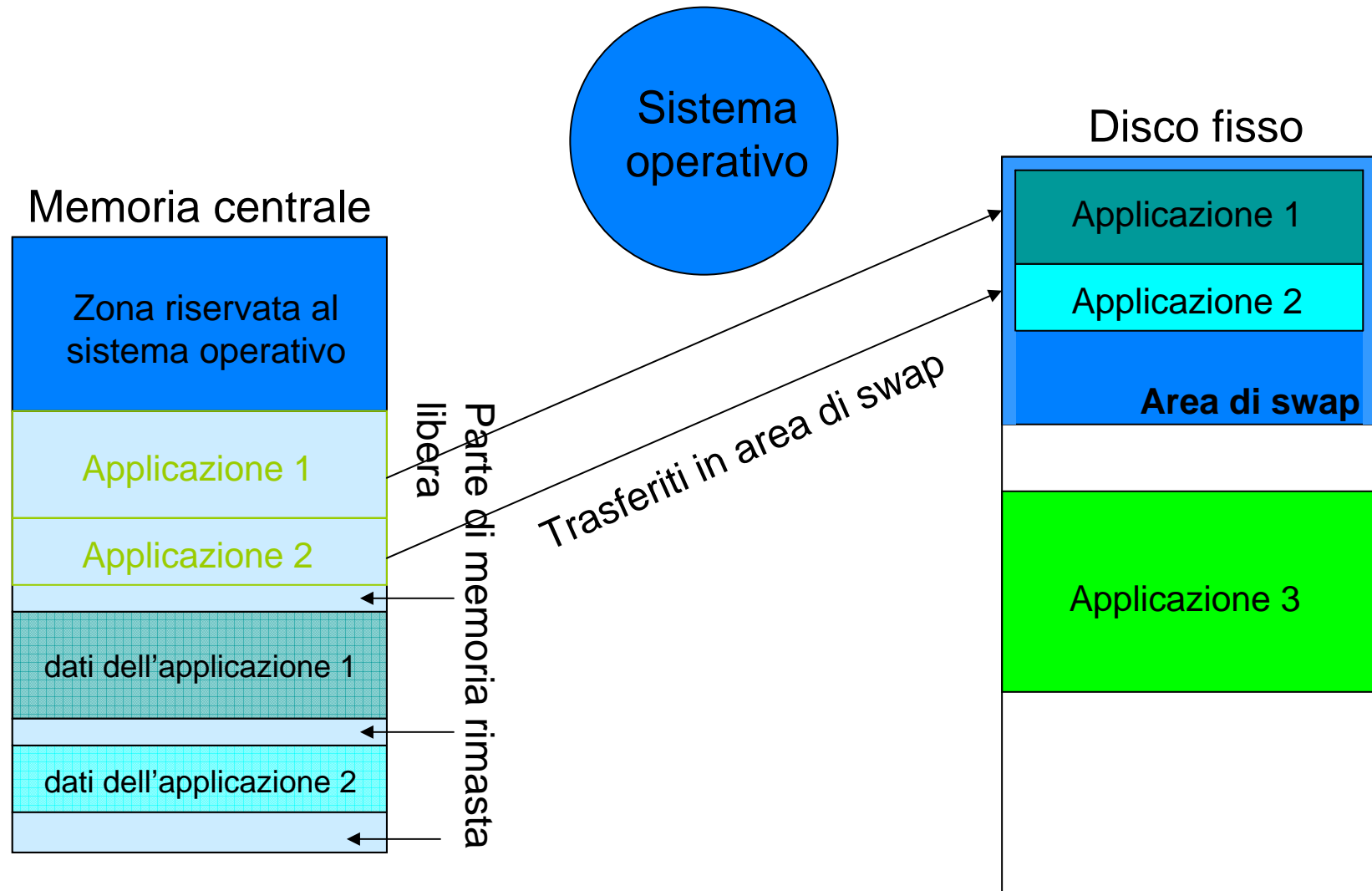
Non c'è spazio !
(nemmeno riordinando)

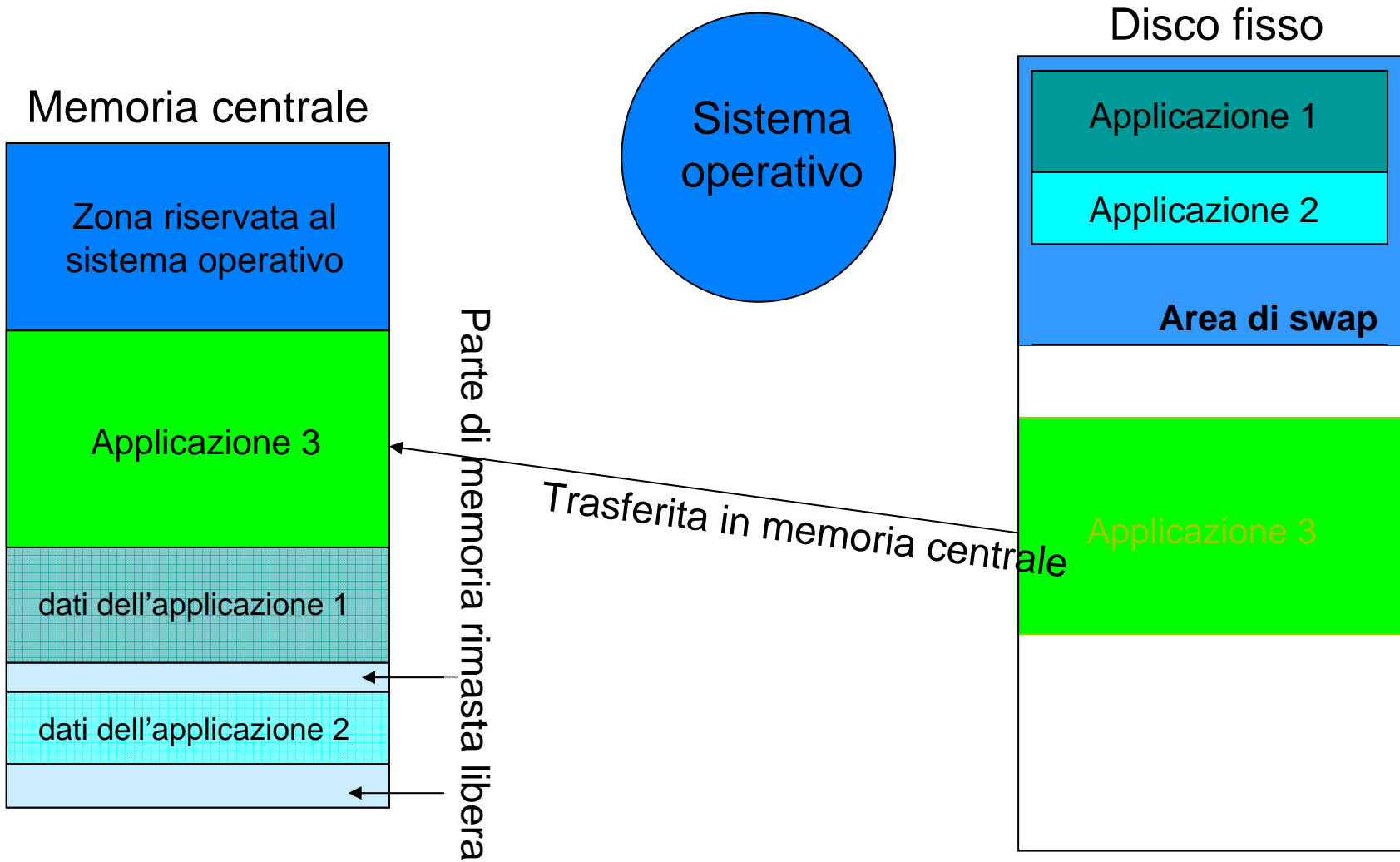
Disco fisso



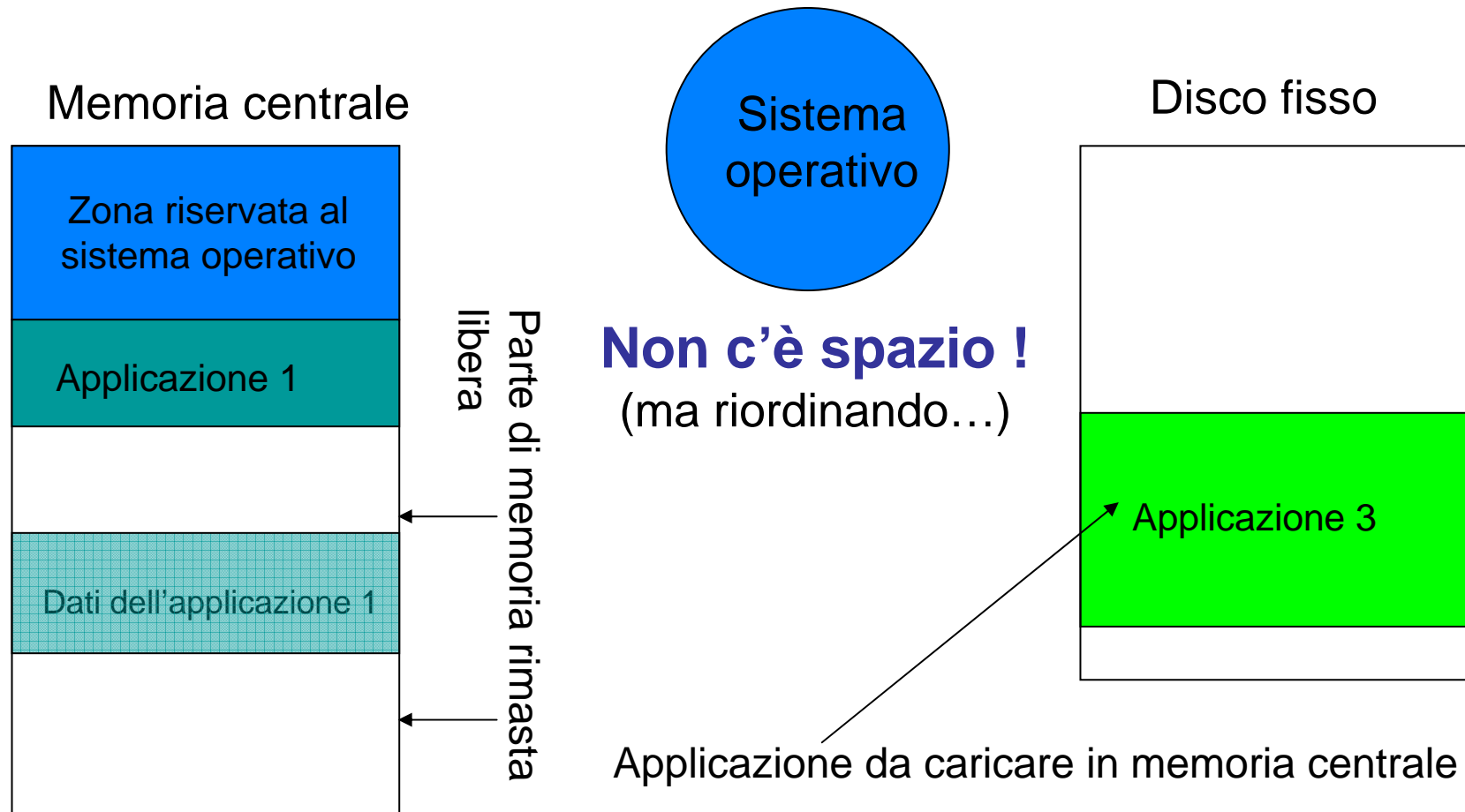
Applicazione da caricare in memoria centrale

- Quando lo spazio disponibile in RAM è inferiore a quello richiesto da un programma, il sistema operativo dovrebbe impedire l'apertura di nuovi programmi per mancanza di memoria
- Una soluzione meno drastica consiste nell'utilizzare una zona del disco fisso come "espansione" della memoria centrale -una **memoria virtuale**- e di usarla come deposito temporaneo di informazioni contenute nella memoria centrale ma attualmente non utilizzate
- La parte di disco fisso dedicata a questa funzione di memoria virtuale è detta **area di swap**
 - non è accessibile da parte dell'utente
- Il sistema operativo sceglie quali parti della memoria centrale scaricare temporaneamente nell'area di swap e quali parti dell'area di swap riportare in memoria centrale

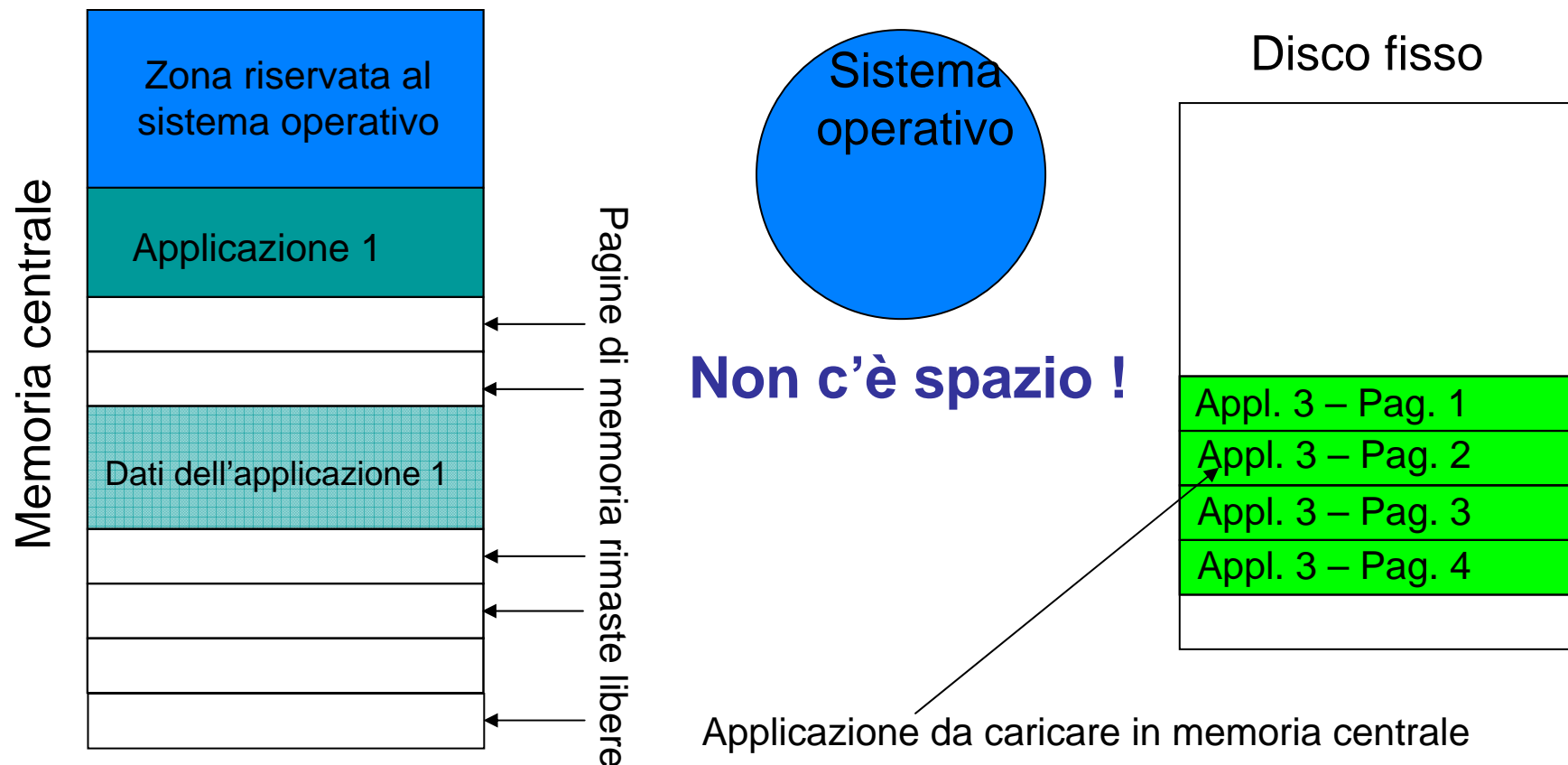


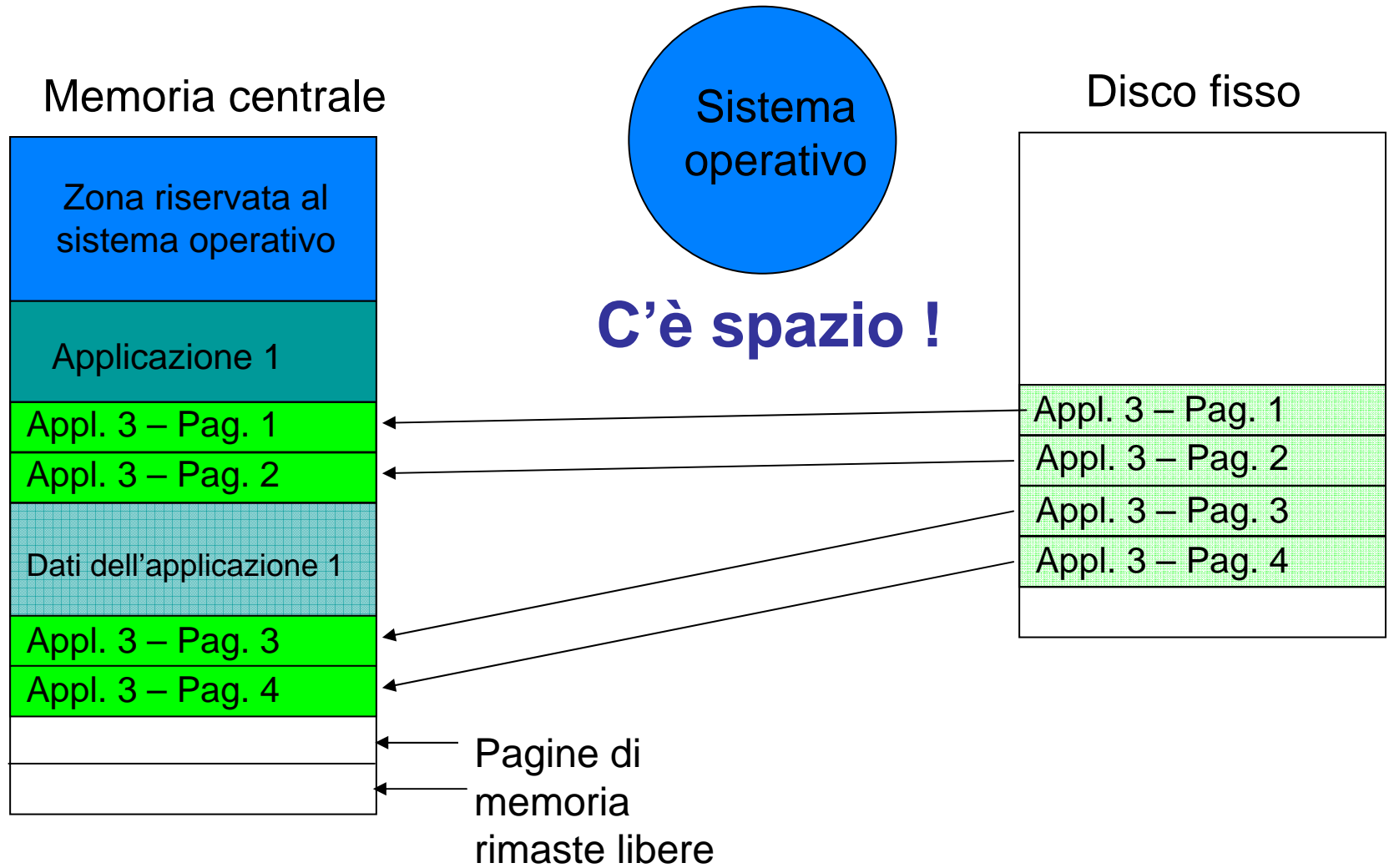


- I problemi non sono completamente risolti se si cerca spazio contiguo nella memoria centrale
- Ecco un caso tipico non risolto applicando semplicemente lo swapping:



- **La soluzione** consiste nel suddividere sia il contenuto dei file che la memoria fisica in **pagine**, permettendo il trasferimento di alcune pagine dalla e nella memoria centrale al posto dei file interi
 - Ciò non danneggia l'esecuzione dei programmi poichè solo piccole parti del codice e dei dati sono necessarie in memoria in ciascun istante di tempo
 - Quando una porzione non presente diventa necessaria la si può andare a prendere sul disco





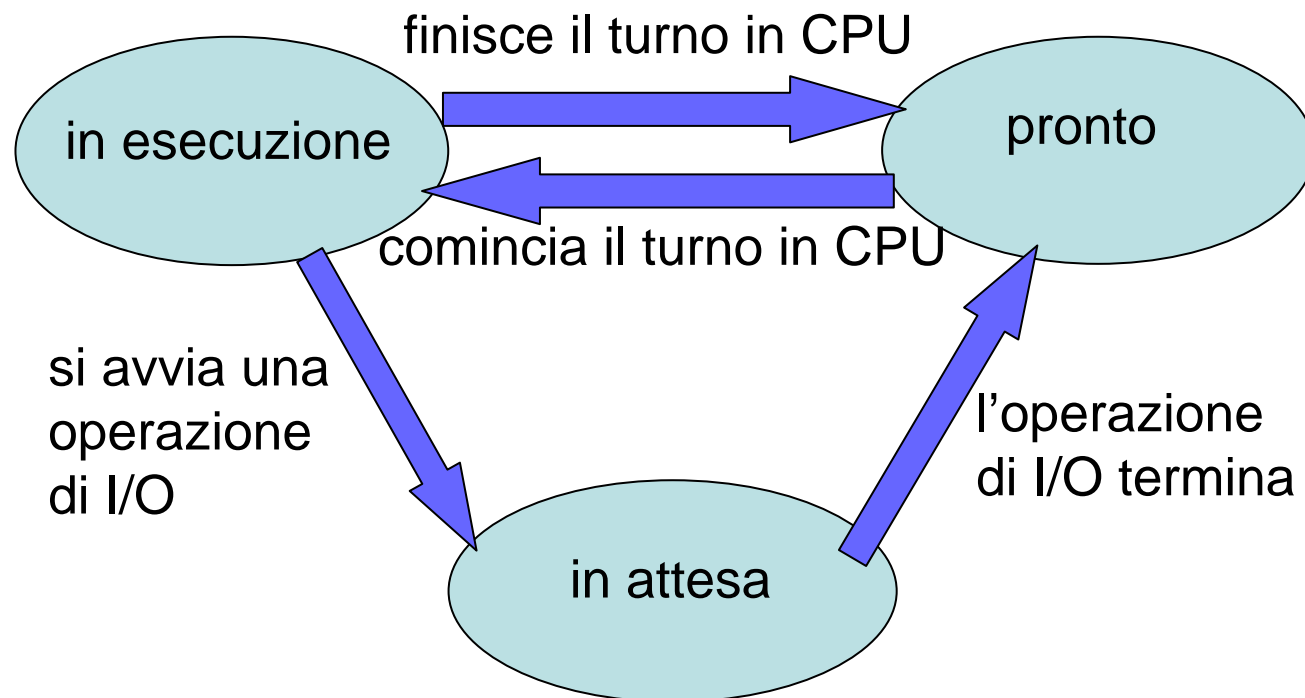
- La combinazione di paginazione e swapping permette un utilizzo ottimizzato della memoria centrale
- Il sistema operativo di volta in volta decide quali pagine scaricare dalla memoria centrale all'area di swap e quali portare dall'area di swap in memoria centrale
- L'utente può accorgersi di una *page fault* (cioè del fatto che la pagina richiesta non è presente nella memoria centrale) durante l'utilizzo di una applicazione, perché percepisce un ritardo tra un comando inviato al computer e la risposta
- Ciò è dovuto al fatto che l'algoritmo di gestione della memoria è abbastanza complesso

- Il sistema operativo deve
 - scegliere quali pagine trasferire in base all'uso che l'utente ne sta facendo
 - mantenere aggiornate le tabelle che tengono traccia della corrispondenza tra indirizzi logici (cioè la sequenza di pagine di un dato file) e indirizzi fisici (cioè la loro posizione nella memoria fisica)
 - se necessario, trovare delle pagine da spostare sul disco per fare posto a quelle da caricare
 - copiare il contenuto di quelle pagine se sono state modificate

S.O. - Nucleo

- Componente che si appoggia direttamente sull'hardware
- Implementa i meccanismi responsabili di reagire agli eventi esterni alla CPU, detti **interrupt**
- Offre funzionalità di alto livello per accedere agli altri componenti del sistema mediante **system call** o **API**
- Poichè la CPU è una sola, solo un programma può essere in esecuzione in ciascun istante di tempo
- Dato però che la CPU è molto più veloce delle periferiche e dei riflessi umani, si può dare all'utente l'impressione che più programmi, o **processi**, siano eseguiti simultaneamente su CPU (apparentemente) dedicate
 - eseguendone una frazione a turno per ciascuno
 - sovrapponendo l'esecuzione delle operazioni sulle periferiche di un programma con l'esecuzione in CPU di un altro

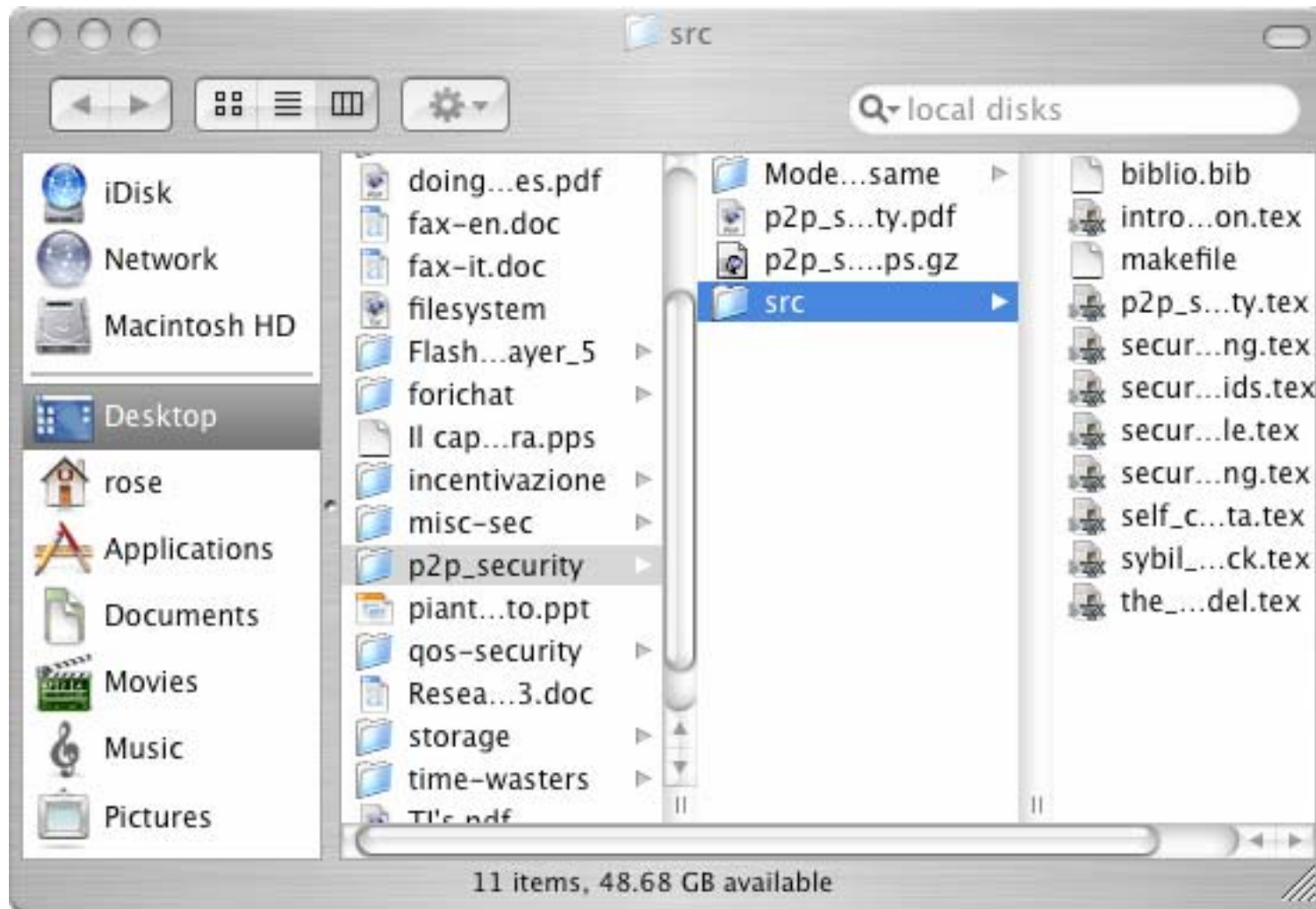
- Per poter condividere la CPU tra molteplici processi, è necessario distinguere tra i possibili **stati** di un processo
 - **in esecuzione**: il processo sta usando la CPU; un solo processo può essere in questo stato
 - **Pronto**: il processo potrebbe usare la CPU se fosse libera; più di un processo può essere in questo stato
 - **in attesa**: Il processo sta aspettando che termini un'operazione di ingresso/uscita; più di un processo può essere in questo stato



- Il sistema operativo deve tenere traccia dello stato di ogni processo
 - a quale istruzione si è interrotto
 - su quali dati stava operando
 - quali pagine sono presenti in memoria
 - quali sono i privilegi assegnati per l'esecuzione
 - quante volte ha usato la CPU e per quanto tempo ogni volta
 - quante operazioni di I/O ha svolto
- Lo **scheduler** dei processi decide quale processo mandare in esecuzione tra quelli nella lista dei processi “pronti” in base a vari criteri
 - uso della CPU fino a quel momento
 - uso dell'I/O fino a quel momento
 - priorità di esecuzione

- **Le system call o Application Programming Interface (API)** costituiscono il tramite fornito dal sistema operativo tra i programmi e l'hardware
- Filtrando gli accessi alle risorse mediante codice di sistema -le system call- il sistema operativo può garantire la condivisione corretta e onesta delle risorse
- Le system call nascondono i dettagli implementativi dei vari dispositivi
 - dipendono dal S.O. e dall'architettura dell'elaboratore

S.O. - File System



Il FILE SYSTEM è la componente del S.O. che provvede alla gestione delle informazioni residenti su memorie di massa, fornendo le seguenti funzioni

- memorizzazione
 - aggiornamento
 - cancellazione.
-
- In un sistema sono generalmente presenti tre tipi di memorie di massa
 - HD, floppy, CD/DVD
 - I vari sistemi operativi usano nomi diversi per le unità
 - in ambiente Windows, si usa una lettera seguita da due punti
 - C:\ D:\ E:\
 - in Unix si usa la barra / seguita da nomi standard dei device

FILE

- Le informazioni sono memorizzate su tali memorie in “contenitori” chiamati **file** o **documenti**
- Ogni file è caratterizzato dal nome, dalla posizione all’interno del file system e da altre informazioni
- Il contenuto di un file è definito dal suo proprietario, l’utente che lo ha creato
- Le modalità di accesso sono definite dall’utente
 - lettura, scrittura, esecuzione per sé e per gli altri utenti

– FORMATO DEI FILE

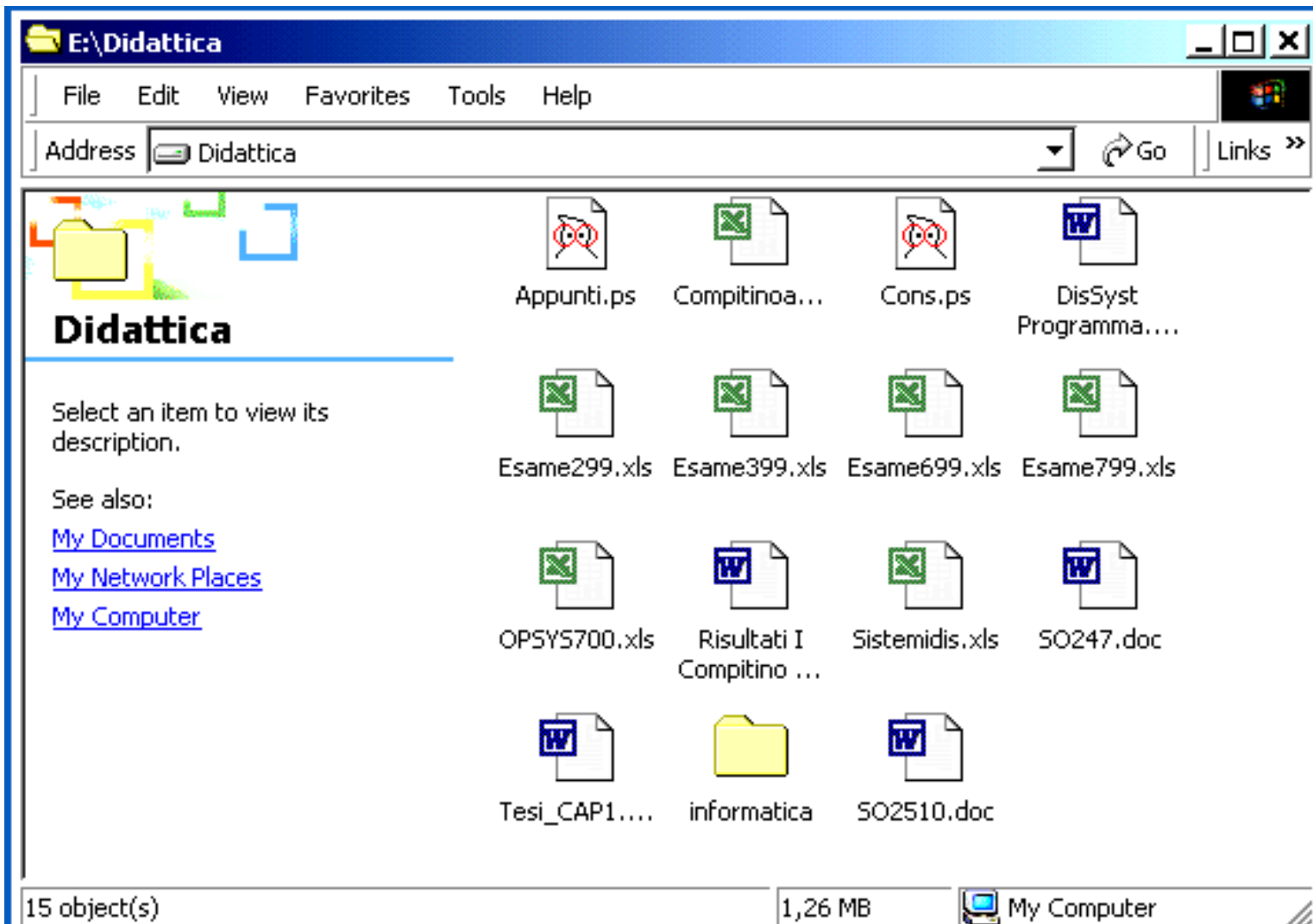
- I programmi manipolano informazioni che memorizzano su file in modo che possano essere riutilizzate in sessioni successive
- Il passaggio dai dati manipolati dal programma alla loro rappresentazione su file richiede una **codifica** in un **formato** che consenta di salvare sia i dati che le informazioni relative alla loro rappresentazione

- Le informazioni contenute in un file creato con una data applicazione vengono memorizzate secondo un particolare formato
- Ciascuna applicazione è in grado di riconoscere e operare su file il cui formato è noto all'applicazione stessa
- Per interpretare correttamente il contenuto di un file, un programma deve riconoscerne ed interpretarne il formato
- Al fine di rendere esplicito il formato di un file si utilizzano delle **convenzioni** nella composizione del nome dei file
- Il formato di un file viene reso esplicito aggiungendo un'opportuna **estensione** al suo nome seguendo il formato

<nome>.<estensione>
- L'estensione in genere è composta da tre o quattro caratteri
- L'uso delle estensioni è solo una **convenzione**
 - comodità per l'utente
 - il sistema operativo usa le informazioni sulla natura del file presenti all'inizio del file stesso

- Il formato delle informazioni contenute in un file dipende dalla natura delle informazioni
 - testo: sequenze di bit che rappresentano ciascuna un simbolo della scrittura
 - 8 bit per il codice ASCII, 16 bit per Unicode
 - binario: sequenze di 32 (64) bit interpretate come istruzioni, del linguaggio macchina o di formattazione, o numeri
- Le informazioni contenute in un file sono tutte di formato omogeneo
 - non si possono avere binario e testo insieme
 - la codifica deve essere omogenea
- Ogni byte contenuto in un file di testo è interpretabile come un carattere secondo una particolare codifica
- Sono visualizzabili con un generico **editor**
- Sono tipicamente associati all'estensione .txt, ma molti programmi che fanno uso di tali file prevedono un'estensione specifica, come
 - .java per il codice di un programma scritto in java
 - .c per il codice di un programma scritto in C
 - .html o .htm per pagine web

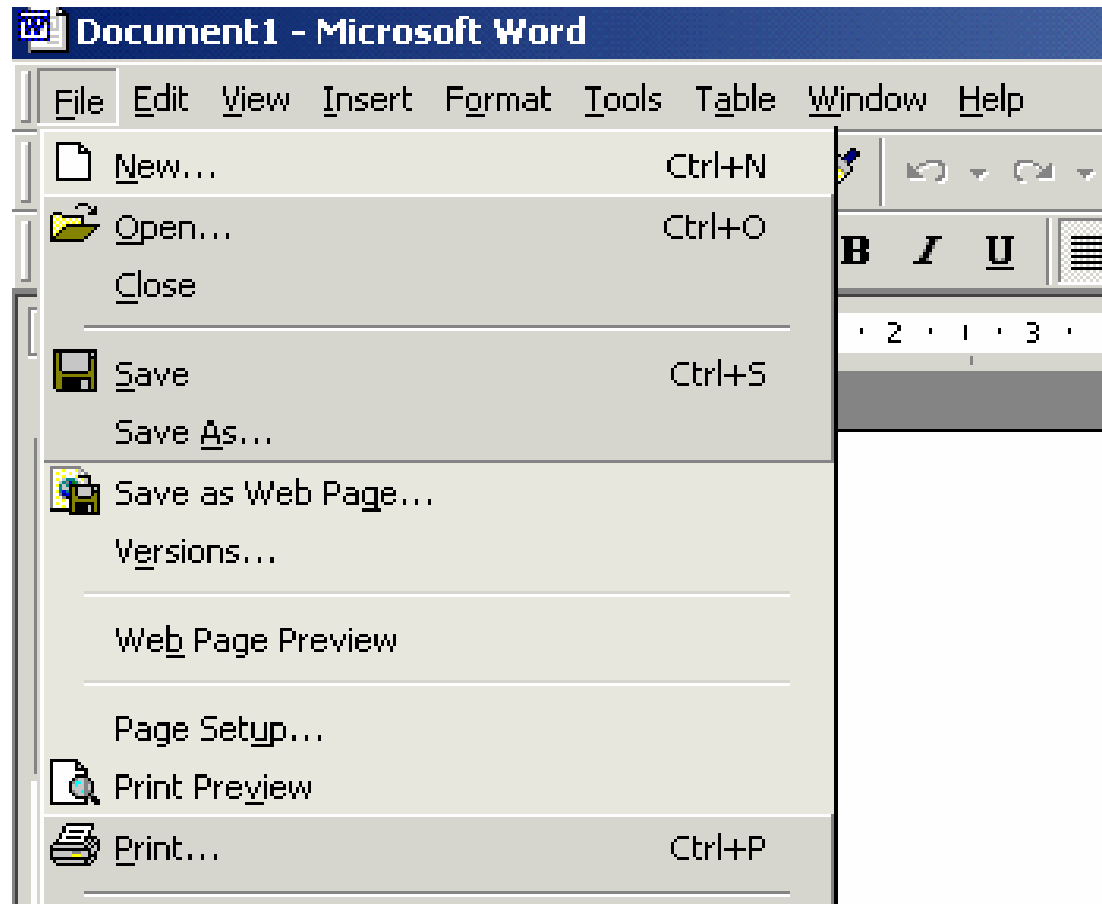
- I singoli byte di un file binario non codificano esclusivamente caratteri ma numeri o istruzioni del processore
- Sono utilizzabili solamente da programmi appositamente scritti per leggerli/scriverli
- Ognuno ha una propria estensione:
 - .exe, .dll per i file eseguibili in Windows
 - .out per i file eseguibili in Unix
 - .doc per i file di Word
 - .bmp per un immagine bitmap
- In ambiente Windows il formato dei file è visibile nell'estensione che compare nel loro nome (se visibile) e attraverso la relativa icona
- In ambiente MacOSX il formato dei file è visibile attraverso la relativa icona o dall'estensione che compare nel nome



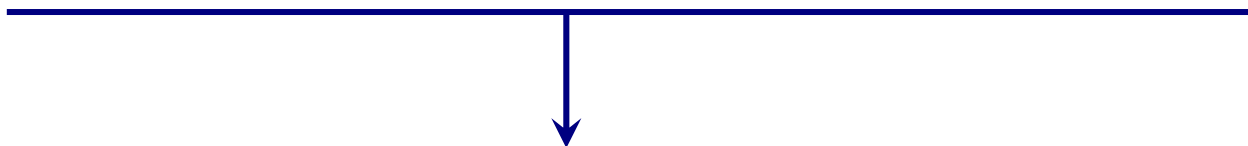
File System - Operazioni sui file

- I file possono essere
 - creati
 - rinominati
 - aggiornati
 - aperti/chiusi
 - cancellati
- Sui file è possibile operare direttamente dall'interno delle applicazioni o dall'interprete dei comandi
 - rinomina, cancellazione, apertura
- **Dall'interno di un'applicazione si può**
 - creare un nuovo file
 - modificare un file
 - stampare il contenuto di un file
 - salvare il contenuto del file con un nuovo nome o in un altro formato

File System - Operazioni sui file



- Per modificare un file è necessario
 - aprirlo
 - modificarne il contenuto mediante l'applicazione
 - Salvarlo



- Open
 - rende disponibile il contenuto del file
 - per poter operare su un qualunque file è sempre necessario aprirlo preventivamente
- Close
 - rende il contenuto del file non più accessibile
 - eseguita esplicitamente dall'utente o dall'applicazione alla chiusura di questa

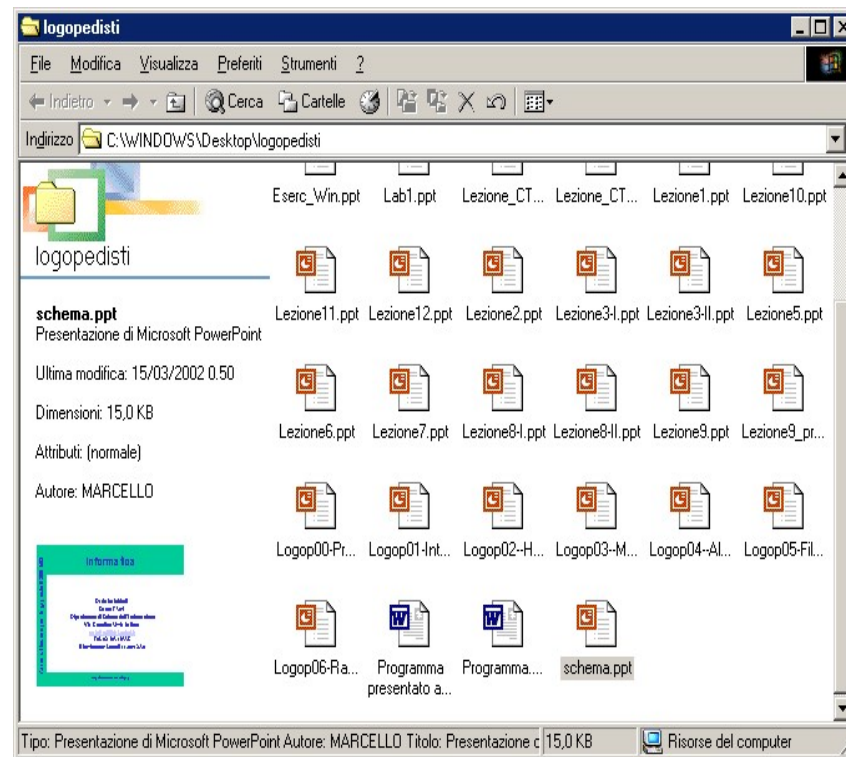
File System - Permessi sui file

- In alcuni tipi di file system esistono **i permessi di accesso** (specificati dall'utente che crea il file); essi **indicano “cosa” ogni altro utente (o gruppi di utenti) “può fare” sul file**
- **permessi di accesso** ai file:
 - **Read (R)** indica se possono essere letti i contenuti di un file o di una cartella
 - **Write (W)** indica se è possibile modificare il contenuto di un file o di una cartella
 - **Execute (X)** indica se è possibile eseguire un file o posizionarsi all'interno di una cartella
- E' possibile impostare i permessi di accesso separatamente per il possessore del file, per gli utenti appartenenti al suo stesso gruppo e per i rimanenti utenti

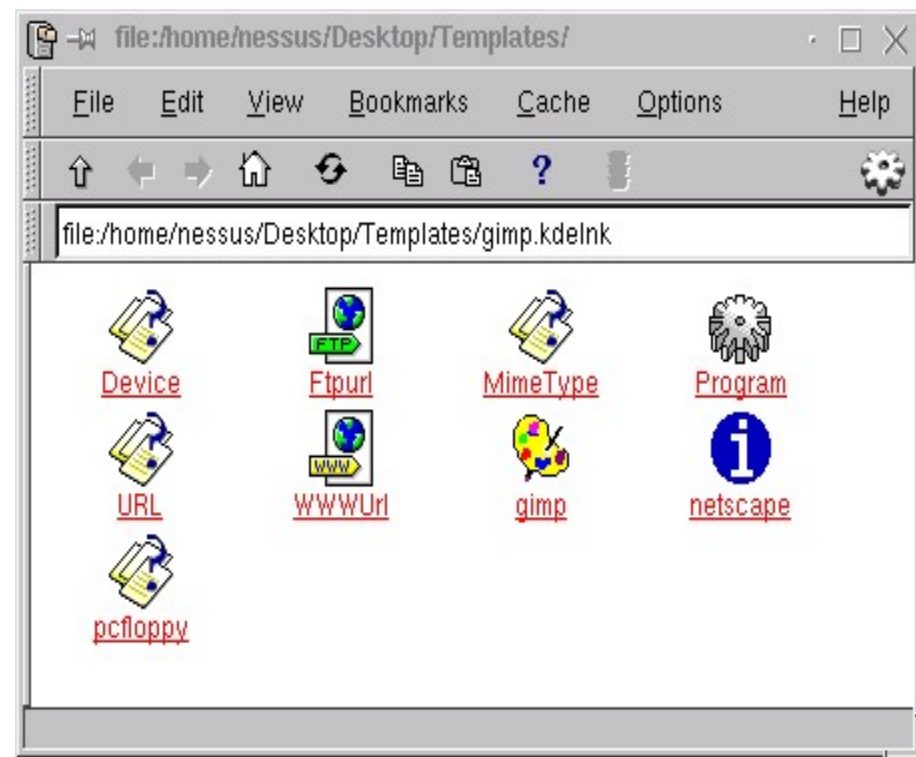
File System - Cartelle

- Oltre ai file dati esiste un altro tipo di file: le **cartelle** (**directory**, **folder**)

Cartella Windows



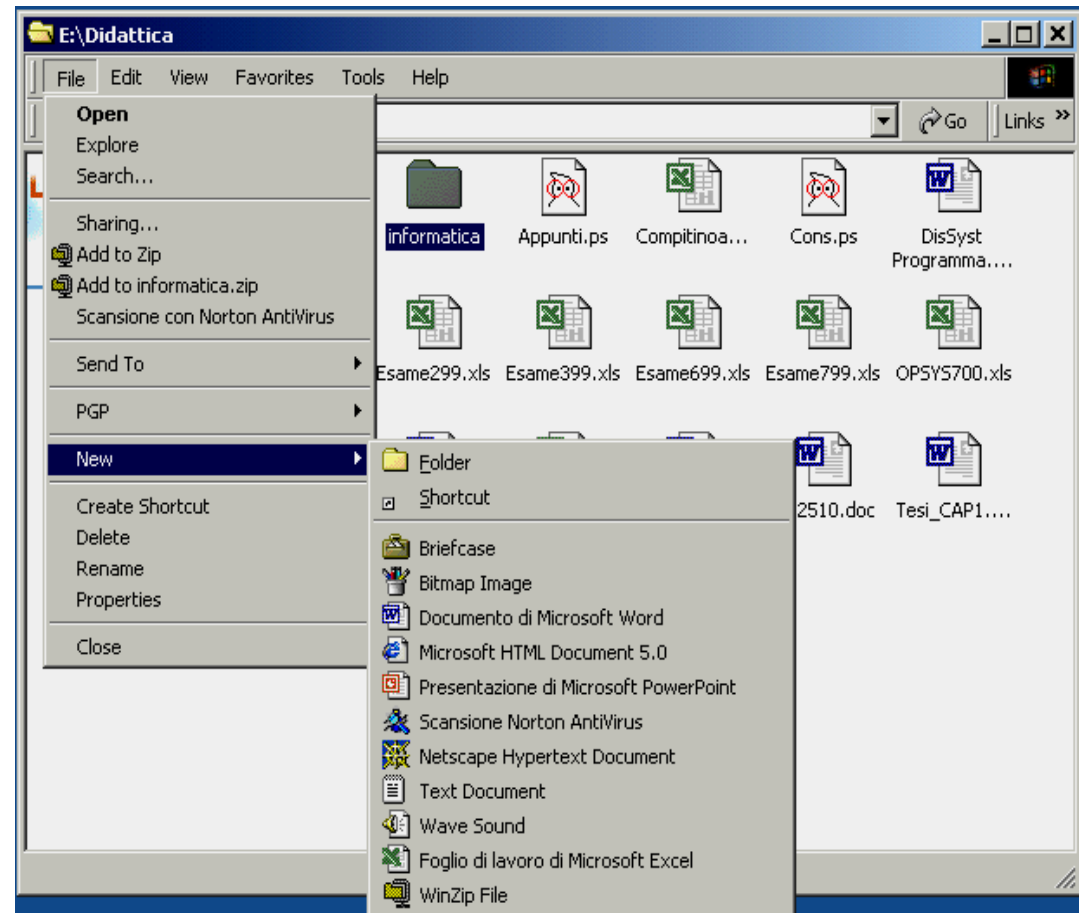
Cartella Linux



- Le cartelle sono dei contenitori logici di file **OVVERO** sono file contenenti le informazioni su altri file
- Permettono all'utente di organizzare da un punto di vista logico l'insieme dei file raggruppandoli secondo criteri a lui significativi
- Le cartelle possono essere gestite direttamente dagli utenti sfruttando gli strumenti messi a disposizione dal file system
- Alcune applicazioni generano automaticamente le proprie cartelle durante l'installazione
- Le cartelle possono a loro volta contenere altre cartelle, che possono contenere altre cartelle e così via
- In questo caso si parla di **gerarchia di cartelle**:
 - **ATTENZIONE!** è una struttura logica: la struttura fisica sul disco è diversa!

File System - Operazioni sulle cartelle

- **Creazione:** creare una cartella ed assegnarle un nome
- **Rinomina:** modificare il nome di una cartella
- **Apertura:** per accedere al contenuto di una cartella
- **Cancellazione:** rimuovere una cartella e tutto il suo contenuto
 - la cartella viene spostata nel cestino
 - il cestino viene svuotato

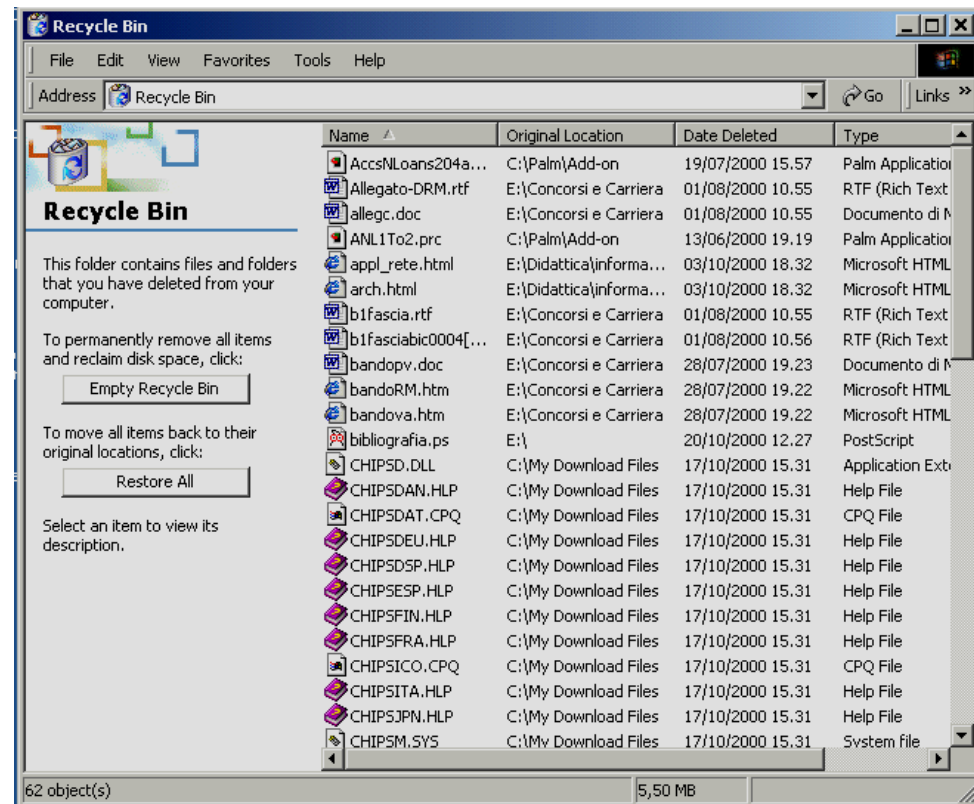


File System – Il cestino o recycle bin

Il cestino è una directory “speciale” che mantiene tutti i file (e le directory) eliminati dall’utente, in caso di suoi eventuali errori.

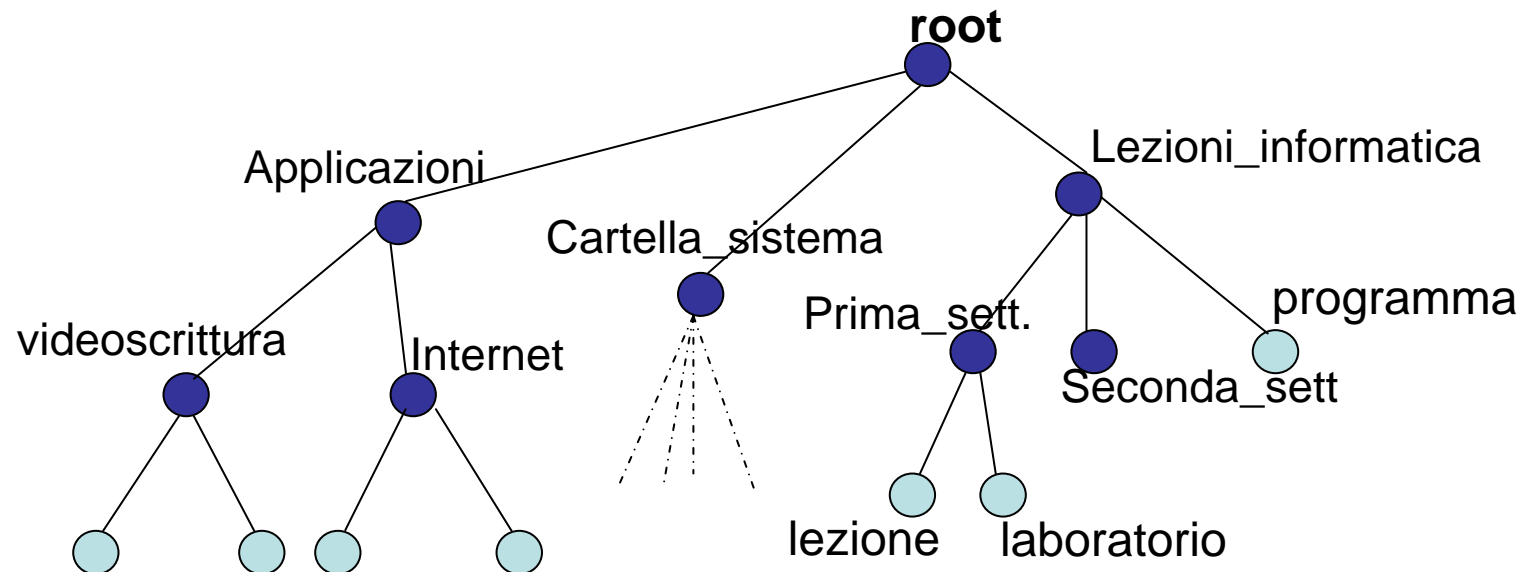
I file nel cestino possono sempre essere “ripristinati” → riportati alla loro posizione iniziale, prima della cancellazione.

MA... Se i file vengono eliminati dal cestino vengono cancellati definitivamente!!



File System - Struttura logica

- Il file system ha una struttura gerarchica ad albero nella quale
 - i file di dati sono le estremità, o **foglie**
 - esiste un'unica directory cui tutta la struttura fa capo, detta la radice (**root**), il livello più alto della gerarchia
 - in Windows, è l'unità disco su cui risiedono le cartelle



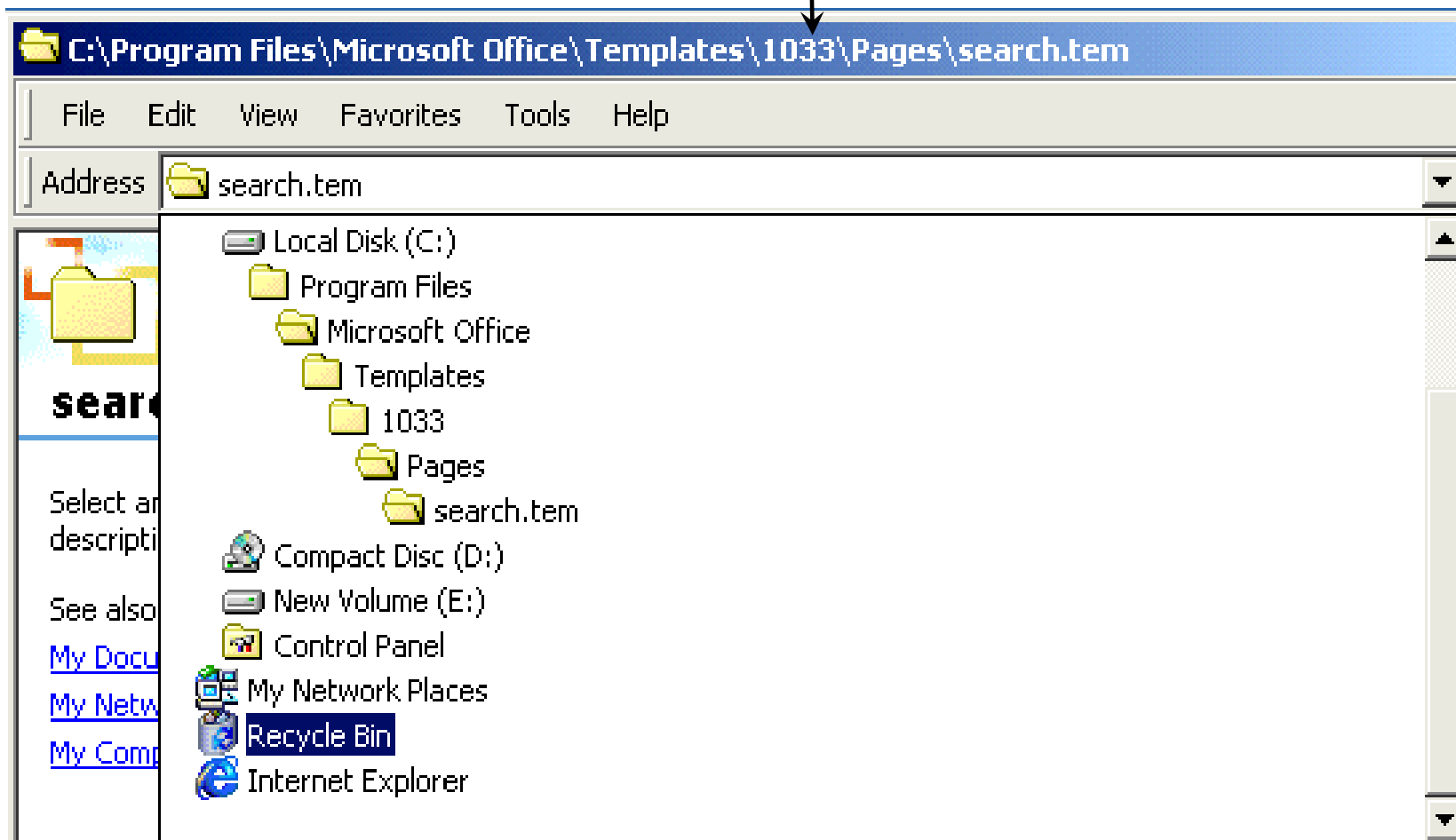
File System - Pathname

- Il **path** (o **pathname**) di un file è la sequenza dei nomi delle cartelle (separati da “\”) che si devono aprire per raggiungere il file
- insieme al nome, individua univocamente un file poiché ogni file del file system è individuato in base alla sua posizione nell’albero, cioè al cammino (o **path**) che si deve percorrere per raggiungerlo partendo dalla radice
 - Es. il programma del corso di informatica è individuato dal cammino
`root\Lezioni_informatica\programma`
- Ogni sistema operativo prevede una sintassi per la specifica dei cammini

Pathname assoluto e relativo

- Un nome può essere specificato per intero
 - **pathname assoluto**
- o relativamente alla cartella in cui si trova (detta .)
 - **pathname relativo**
 - `.\programma` o semplicemente `programma`
- la cartella padre, un livello sopra nella gerarchia, è indicata come `..` rispetto alla cartella corrente
 - trovandosi in `.\Prima_sett`
 - `..\Seconda_sett\` equivale a
`root\Lezioni_informatica\Seconda_sett`

pathname assoluto



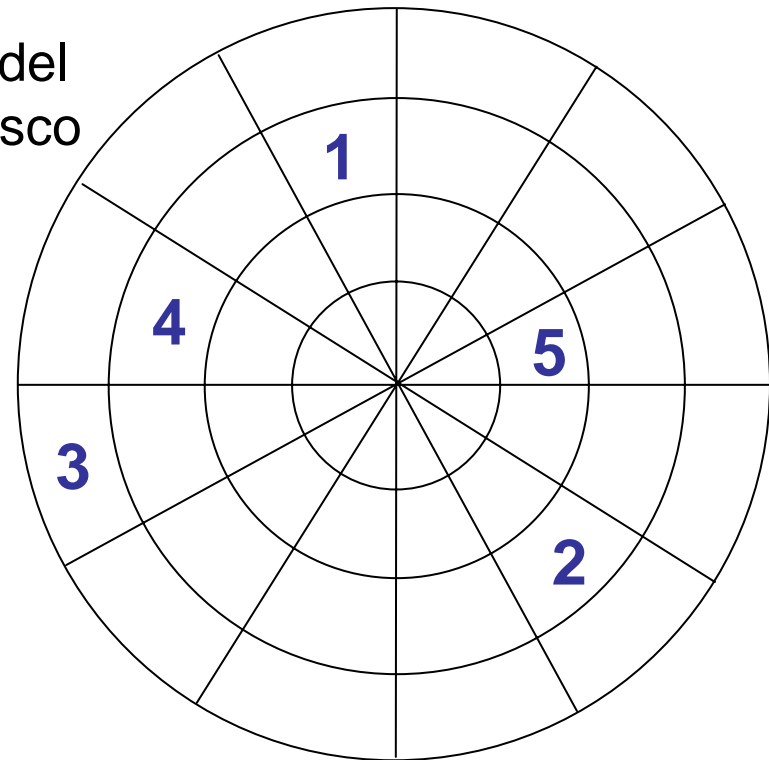
File System - Struttura fisica

- Gestita dal sistema operativo
 - trova spazio sufficiente (non necessariamente contiguo) per memorizzare i file
 - attiva il trasferimento dei dati
 - aggiorna le tabelle opportune per essere in grado di recuperare i file

blocchi di un file

1
2
3
4
5

blocchi del
file su disco
fisso



- La struttura fisica del file system può diventare troppo frammentata: quando succede si eseguono routine specifiche per compattare i blocchi liberi
- La gestione della struttura fisica del file system richiede un impegno computazionale non indifferente da parte del sistema operativo

File System - Visualizzazione

- Quando si accende il personal computer ci si trova davanti una **scrivania (desktop)** su cui compaiono cartelle e documenti
- Ogni sistema operativo visualizza in modo particolare i contenuti del file system:
 - **In Windows**
 - La radice della struttura gerarchica è il **desktop**
 - I nodi di un pathname sono separati da “\”
 - Ogni disco è individuato da una lettera
 - **In Linux**
 - La radice della struttura gerarchica è la directory **root**, indicata con “/”
 - I nodi di un pathname sono separati da “/”
 - Ogni disco è “nascosto” nel file system

S.O. - Utenti e privilegi

- Un sistema operativo che consente l'uso di un computer a più utenti (simultaneamente e non) distingue tra vari tipi di utenti
 - possibilità di eseguire quali operazioni
 - separazione tra utenti e protezione dei dati
- Solo un utente può fare qualsiasi operazione per poter gestire il sistema
 - administrator, root, superuser
- Gli utenti comuni hanno potere limitato per contenere i danni che potrebbero fare
 - rimuovere file di sistema, cancellare dati di altri utenti, cambiare l'ordine della coda di stampa, ecc.
- Gli utenti comuni possono accedere solo ai propri file e installare localmente nelle proprie cartelle sw applicativo al quale gli altri utenti non possono accedere

- L'amministratore del sistema può installare il sw applicativo in modo che tutti gli utenti lo possano usare e che il sw possa accedere alle risorse di cui ha bisogno
- L'amministratore di sistema può aggiungere nuovi utenti nel sistema o disabilitare quelli esistenti
- Gli utenti comuni non possono farlo perchè non hanno "privilegi" a sufficienza nel loro profilo