

I sistemi operativi

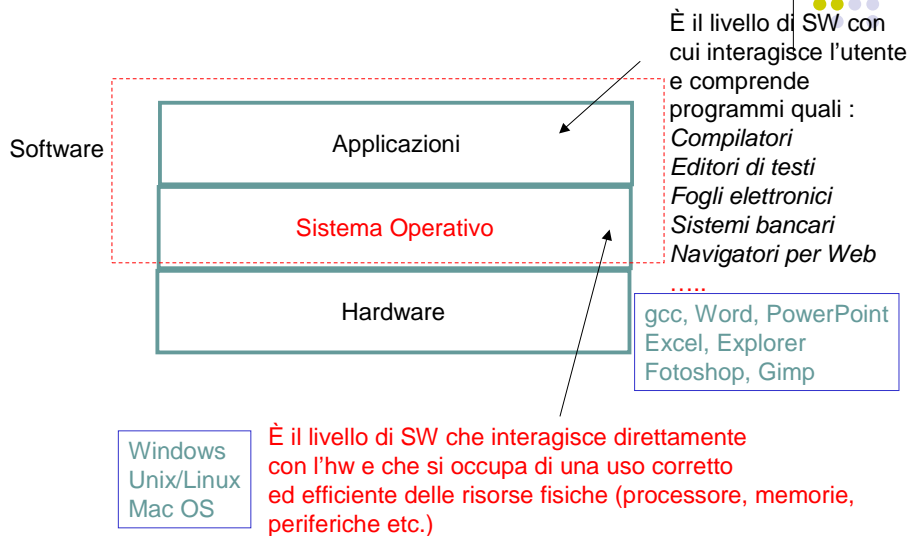
Università degli Studi di Salerno
Corso di Laurea in Scienze della
Comunicazione
Informatica generale (matr. Dispari)
Docente: [Angela Peduto](#)
A.A. 2005/2006



Il software

- Il termine software è utilizzato per indicare la parte immateriale del computer in opposizione a quella fisica definita hardware.
- Il software è diviso in due categorie:
 - **software di sistema**
 - **software applicativo**
- Si può dire che il software di sistema garantisce le funzioni base che permettono l'utilizzo del computer
- Il software applicativo viene utilizzato per risolvere specifici problemi, consiste nei programmi che svolgono il lavoro reale per gli utenti finali

Cos'è un sistema operativo ?



Angela Peduto - Informatica generale
A.A. 2005/06

3

Quali sono le funzioni di un SO ?

- Esegue applicazioni :
 - carica il programma binario prodotto della compilazione (e residente su disco) nella RAM,
 - cede il processore all'applicazione da eseguire
- Facilita l'accesso ai dispositivi di I/O
 - interagisce con le periferiche facendosi carico di tutti i dettagli fisici (es. modem, reti, video...)
 - mette a disposizione operazioni di lettura/scrittura ad alto livello che possono essere usate senza conoscere i dettagli tecnici della periferica

Angela Peduto - Informatica generale
A.A. 2005/06

4

Quali sono le funzioni di un SO ? (2)



- Archivia dati e programmi :
 - mette a disposizione dell'utente una visione astratta della memoria secondaria (il **file system** basato sulle astrazioni : *file/archivi e folder/cartelle*)
 - gestisce la realizzazione di queste astrazioni sul supporto fisico (disco) gestendo tutti i dettagli legati alla lettura/scrittura dei settori

Quali sono le funzioni di un SO ? (3)



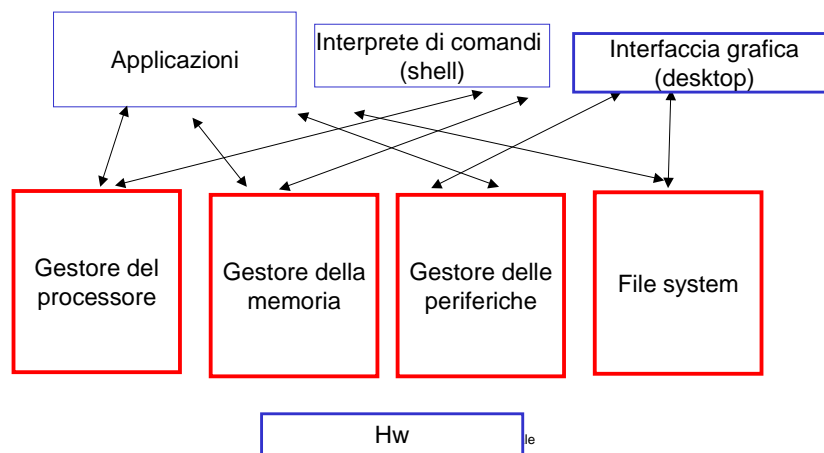
- Gestisce le risorse
 - ripartisce le risorse disponibili fra le varie applicazioni (processore, RAM, periferiche)
 - evita che ci siano malfunzionamenti dovuti all'uso contemporaneo di risorse
 - es: un word processor e un web browser che inviano contemporaneamente dati alla stampante provocano una stampa erronea
 - ottimizza le prestazioni scegliendo delle politiche che permettano di sfruttare al meglio tutte le parti del computer

Quali sono le funzioni di un SO ? (4)



- Gestisce malfunzionamenti del sistema
 - rileva e gestisce situazioni anomale
 - es: se il disco ha un settore difettoso, il SO può trasportare automaticamente le informazioni residenti su quel settore da un'altra parte
 - es: se un'applicazione cerca di effettuare una operazione non permessa (come leggere i dati di un'altra applicazione) può bloccare l'applicazione segnalando all'utente la situazione errorea
- ... i SO sono sistemi complessi che non sempre riescono a fare correttamente tutto quello che dovrebbero

Quali sono le parti di un SO ?



I processi



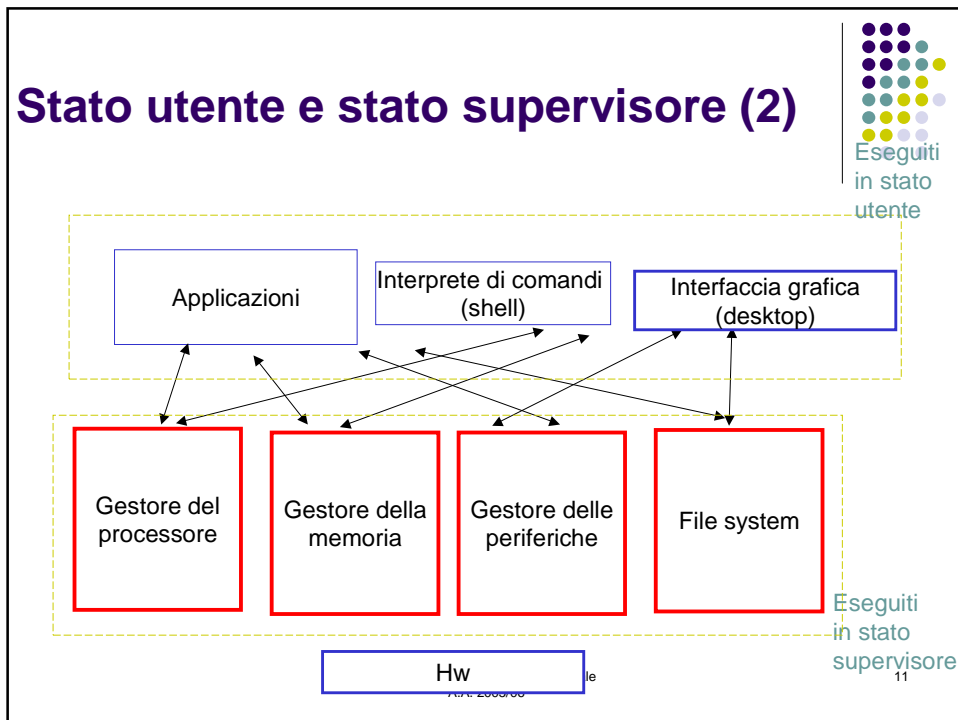
- Quando un programma utente è stato attivato il processore esegue una dopo l'altra le istruzioni assembler che lo compongono
- Un **programma** in esecuzione viene detto 'processo'
- Diversità tra programma e processo:
 - 1 programma genera 1 o più processi
- il processo rappresenta lo stato di avanzamento del programma istante per istante.
- **Processo utente**: generato dall'esecuzione di un applicativo utente.
- **Processo di sistema**: generato dal SW di sistema (Sistema Operativo, utilities, ecc.).

Stato utente e stato supervisore



- Stato utente :
 - modalità di funzionamento del computer che permette l'accesso solo a un numero limitato di risorse
 - es : un sottoinsieme delle istruzioni assembler (non si può accedere alle istruzioni che istruiscono le interfacce di I/O), una sola parte della RAM etc.
- Stato supervisore :
 - modalità che permette l'accesso a tutte le risorse

Stato utente e stato supervisore (2)



Stato utente e stato supervisore (3)

- I programmi che girano in stato utente *richiedono* servizi al SO tramite invocazione di sottoprogrammi speciali
 - *system call* o *chiamate di sistema*
- Il sistema operativo decide come e quando effettuare il servizio
- Il sistema operativo può interrompere un programma che gira in stato utente per eseguire altri programmi o per effettuare operazioni di 'gestione' della macchina

Esecuzione di un programma



- Quando un programma utente è stato attivato il processore esegue una dopo l'altra le istruzioni assembler che lo compongono
- Un programma in esecuzione viene detto 'processo'
- Problema : Come fa il Sistema Operativo a riprendere il controllo del processore dopo l'attivazione di un processo?
 - Terminazione (normale o erronea)
 - Interruzione

Terminazione di un processo



- Un processo termina :
 - Quando esegue una istruzione assembler di terminazione
 - Quando effettua una operazione illecita (es. cerca di accedere a memoria privata di altri processi)
 - Quando c'è un errore che non lo permette di proseguire (es. overflow, etc)
- In tutti questi casi il processore ricomincia automaticamente ad eseguire il sistema operativo ad un indirizzo prefissato

Interruzione di un processo



- Il sistema operativo può bloccare un processo in un qualsiasi istante della sua esecuzione per effettuare qualche operazione di gestione della macchina
- Questo avviene attraverso il meccanismo hardware delle *interruzioni*

Interruzione di un programma (2)



- Come funzionano le *interruzioni*
 - ogni periferica può 'richiedere attenzione' inviando un segnale di *interruzione* usando le linee di controllo del bus
 - alla fine dell'esecuzione di ogni istruzione assembler il processore controlla la presenza di una interruzione
 - se è presente il controllo passa automaticamente al sistema operativo

Classificazione dei S.O.



- Classificazione dei S.O. in base alla modalità di gestione dei processi:
 - **mono-tasking**
 - **multi-tasking:**
 - **multi-tasking 'semplice'**
 - **time-sharing**
 - **time-sharing con priorità**
 - **real-time**

Classificazione dei S.O. (2)



- **Mono-tasking:** esecuzione sequenziale dei processi
- **Multi-tasking:** Esecuzione "*contemporanea*" di più processi (molti programmi risiedono contemporaneamente nella RAM)
 - **Semplice:** mentre il processo i attende per un'op. di I/O viene mandato in esecuzione il processo j
 - **Time-sharing:** si usa la tecnica dei *quanti* di tempo vengono mandati in esecuzione ciclicamente
- **Real-time:** suddivisione del tempo in quanti, assegnazione di priorità statica ai processi (alle attività critiche valori alti di priorità).

Il gestore del processore



- Nei computer attuali, tutti i programmi che compongono il SO ed i programmi applicativi 'avviati' sono attivi contemporaneamente sul calcolatore (multiprogrammazione)
 - es : Windows 2000 + Word + cd player ...
- Il *gestore del processore* si preoccupa di far condividere il processore a tutti i programmi attivi in modo corretto
 - lo scopo è dare l'impressione che ognuno abbia il suo processore dedicato

Il gestore del processore (2)

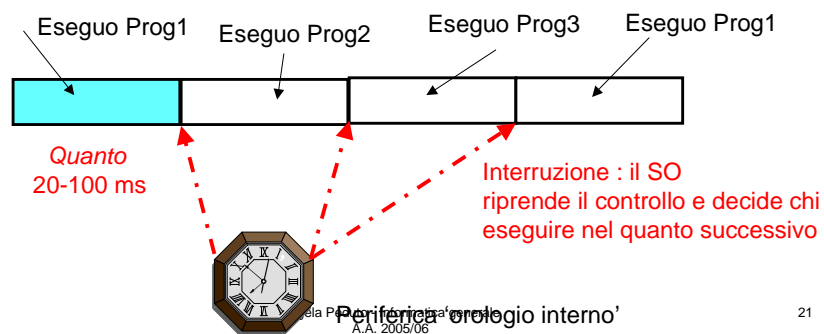


- Nei sistemi attuali sono in esecuzione più processi 'contemporaneamente'.
- La CPU esegue comunque una sola istruzione per volta.
- Il S.O. gestisce l'avvicendamento dei processi in esecuzione: assegna la CPU ai vari processi istante dopo istante (lo scopo è dare l'impressione che ognuno abbia il suo processore dedicato)

Il gestore del processore (3)



- Si usa la tecnica dei *quanti* di tempo
 - es: 3 programmi attivi Prog1, Prog2, Prog3
 - vengono mandati in esecuzione ciclicamente



21

Il gestore della memoria



- Tutti i programmi che compongono il SO ed i programmi applicativi attivi usano contemporaneamente la RAM
- Il *gestore della memoria* si preoccupa di fare condividere la RAM ai 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 e raccogliere i suoi dati

22

Il gestore della memoria (2)



- La gestione statica
 - ricopiare *interamente* lo spazio di indirizzamento di un programma da memoria secondaria a RAM all'inizio dell'esecuzione
 - abbiamo già visto un esempio :



Il gestore della memoria (3)



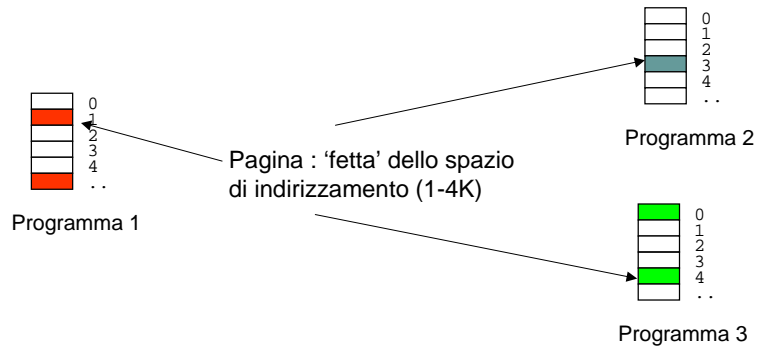
- Problemi della gestione statica
 - non posso girare programmi con spazio di indirizzamento grande della RAM !
 - Attualmente ogni programma usa almeno 4GB di spazio di indirizzamento, con RAM assai più piccole...
- Soluzione : gestione dinamica (a pagine)
 - ad ogni istante carico in RAM solo i pezzi che mi servono per l'esecuzione corrente

Il gestore della memoria (4)



- Paginazione

- lo spazio di indirizzamento di ogni processo è diviso in 'fette' (*pagine*) tutte della stessa ampiezza



Angela Peduto - Informatica generale
A.A. 2005/06

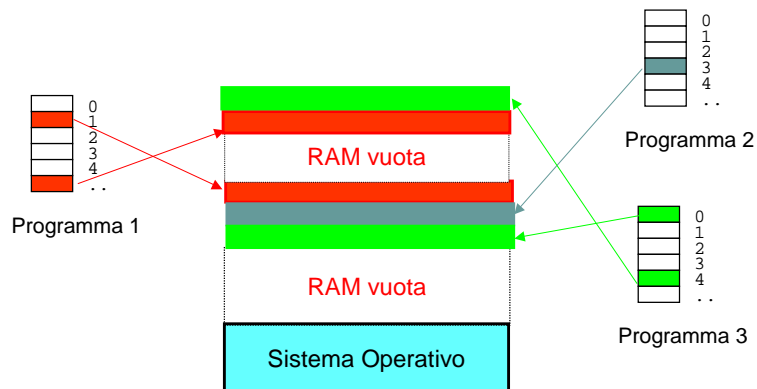
25

Il gestore della memoria (5)



- Paginazione

- ad ogni istante solo le pagine necessarie sono caricate in memoria (**località!**)



26

Gestione delle periferiche



Ogni periferica è gestita da due entità cooperanti :

- il controller :
un dispositivo hardware che colloquia direttamente con la parte elettrico/meccanica della periferica
- il driver :
un programma software che fa parte del sistema operativo e che conosce le caratteristiche della periferica ed è capace di colloquiare con il controller

Gestione delle periferiche (2)



- Solitamente il driver non è sviluppato da chi costruisce il SO (es. Microsoft) ma da chi costruisce il dispositivo
- Quando viene acquistata una nuova periferica il driver corrispondente deve essere inserito nel Sistema Operativo (*installazione*)
 - operazione rischiosa, perché se il nuovo driver è difettoso può rovinare il vostro sistema (il driver essendo nel SO ha accesso a tutta la macchina senza restrizioni!....)

Driver e controller (3)



- Quando si installa un nuovo driver bisogna fornirgli delle informazioni a basso livello per permettergli di colloquiare correttamente con il resto del sistema (*configurazione*)
 - es: stabilire quali linee del bus utilizzare per inviare una interruzione ...
- Periferiche plug&play : quelle per cui la configurazione può essere fatta automaticamente dal sistema
 - sono generalmente tutte così nei PC odierni

Il File system



File System



- È la parte del SO che
 - permette di memorizzare dati e programmi in modo persistente
 - permette di organizzare dati e programmi in modo da renderne agevole la localizzazione da parte dell'utente umano
 - può essere modificata per cancellare dati obsoleti, aggiornare l'organizzazione etc ..

File System (2)

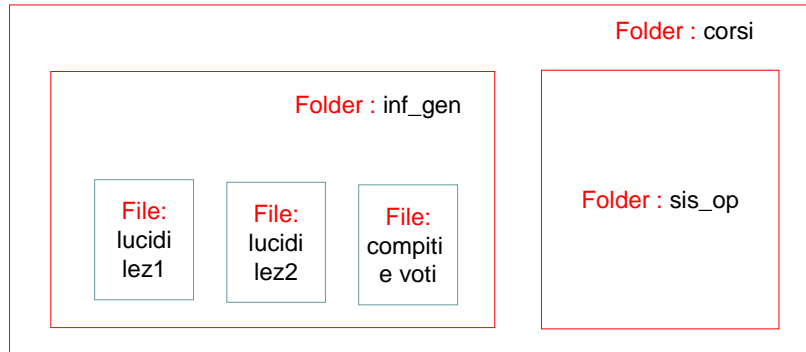


- Le astrazioni fornite sono :
 - il file (o archivio) : una sequenza di byte che contiene dati o programmi rappresentati con una opportuna codifica binaria
 - il folder/directory (o cartella) : astrazione che permette di collezionare insieme più file e/o folder



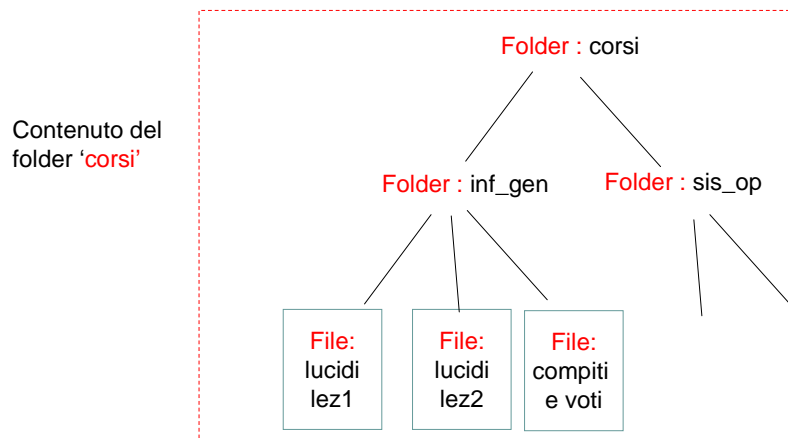
File System (3)

- Esempio : una possibile organizzazione dei file relativi ai diversi corsi che insegno



File System (4)

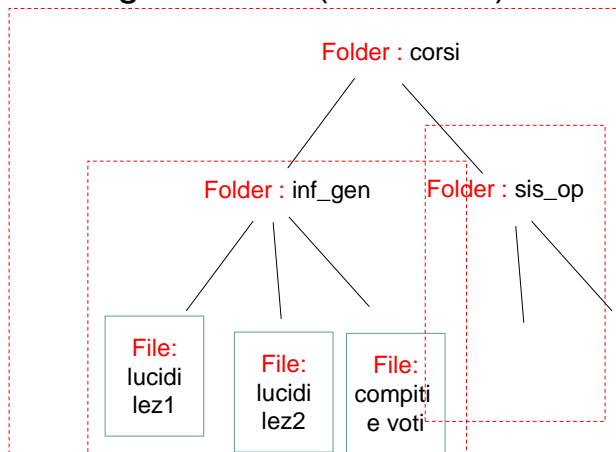
- La struttura è gerarchica (ad *albero*)



File System (5)

- La struttura è gerarchica (*ad albero*)

Contenuti dei
due folder
inf_gen e sis_op



Angela Peduto - Informatica generale
A.A. 2005/06

35

File

- Come si crea un file
 - utilizzando una applicazione
 - editor di testi per creare file di testo secondo diversi formati (doc,html,text) es. Word
 - applicazioni per gestire grandi quantità di dati persistenti (*basi di dati*) es. Access
 - fogli elettronici es. Excel
 - dando comandi diretti al SO
 - es. chiedendo di duplicare un file esistente (taglia e incolla ...) via shell o interfaccia grafica

Angela Peduto - Informatica generale
A.A. 2005/06

36

File (2)



- Nome di un file
 - ogni file ha un nome (es *lez1.doc*, *lez2.ppt* etc)
 - il nome deve essere unico nel folder che contiene il file
 - la parte dopo il punto (.) si chiama 'estensione'

lez2.ppt estensione

di solito l'estensione da informazione sul formato del file e/o sull'applicazione utilizzata per crearlo

File (3)



- Nome di un file (cont.)
 - alcuni SO (come Windows) non mostrano normalmente l'estensione
 - *lez1.doc* viene mostrato come '*lez1*'
 - tuttavia l'estensione è presente e serve al sistema per capire che tipo di file è e quale applicazione attivare quando desideriamo 'leggerlo'
 - es. un doppio click su un file *lez1* attiva automaticamente Word per visualizzarlo

Folder



- Anche i folder sono identificati univocamente da nomi
 - es *inf_gen*, *corsi*
 - non usano estensioni
 - contengono solo informazioni sui file/folder contenuti (il formato di queste dipende dal dal file system usato, e quindi dal SO)
 - non dobbiamo discriminare fra codifiche diverse

Folder (2)



- Per creare un folder basta invocare una opportuna funzione del SO
 - es. scegliere una opportuna opzione (*new folder*, *nuova cartella*) in un menu dell'interfaccia grafica
- Per cancellare file/folder basta invocare una opportuna funzione del SO
 - es. trascinarli con il mouse fino al cestino
- I file/folder possono essere spostati da un folder all'altro

Sicurezza



- Come evitare che i nostri file siano letti da altri utenti che hanno accesso al sistema?
 - Assegnare ad ogni utente : *Account* (nome unico identificativo) e *password* (segreta, abbastanza complessa!)
 - Ogni file appartiene ad un utente
 - Il proprietario può specificare se un file può essere letto o scritto anche da altri
 - Il file system al momento dell'accesso controlla che l'operazione sia effettivamente consentita

Sicurezza (2)



- Il *superutente* (root, administrator)
 - E' l'account dell'amministratore del sistema
 - Può accedere in maniera non ristretta dappertutto
 - I file del sistema operativo sono generalmente dell'utente root e non sono accessibili da nessun altro utente
 - ovviamente il superutente è oggetto di numerosi tentativi di violazione
 - termini come *attack*, *hacker* si riferiscono a questo

Sicurezza (3)



- I vecchi sistemi Windows (3.1,95,98) non avevano alcun meccanismo di protezione
 - tutto gira in stato kernel
 - da cui la facilità di danneggiare il sistema operativo o i file utente (es. con virus)
 - molto pericoloso in connessioni Internet non sporadiche
- Windows NT, 2000, XP sono protetti
 - è necessario però creare utenti non privilegiati
- Sistemi Unix/Linux protetti fino dagli anni 70