

Sistemi operativi

Un **Sistema Operativo** (abbreviato come SO o OS dall'inglese Operating System) è quella componente software di un computer responsabile della gestione e coordinamento di tutte le attività e la condivisione delle risorse hw/sw di un computer. Uno dei suoi scopi principali è quello di gestire l'interazione tra l'uomo e la macchina nascondendo i dettagli delle operazioni hardware.



Principali componenti di un sistema operativo

Un generico sistema operativo moderno si compone di:

- Il **kernel**
- Un **gestore di file system**
- Un **scheduler**
- Un **gestore di memoria**
- Un'interfaccia utente (**shell o GUI**)

File System

- Il **File System** si occupa del modo in cui i file sono immagazzinati e organizzati su un dispositivo di archiviazione, come un hard-disk o un CD-ROM. Esistono molti tipi di file system, creati per diversi sistemi operativi, per diverse unità di memorizzazione e per diversi usi. Si possono identificare due grandi classi di file system: **file system per unità locali**, destinate ad organizzare fisicamente i dati su un disco, e i **file system distribuiti**, nati per condividere i dati fra più computer collegati attraverso una rete, superando le differenze fra sistemi operativi e filesystem locali delle varie macchine.
- Esempi di Filesystem per unità locali:
 - Ext3, JFS (Linux)
 - FAT, FAT32, NTFS (Windows)
 - HFS+ (Mac OSX)
- Filesystem distribuiti
 - NFS (Network File System)
 - AFS (Andrew File System)

Primi sistemi operativi

Monitor: il primo tipo di sistema operativo mai implementato su un computer con un'interfaccia interattiva minima per impartire i comandi. Permette solo di scrivere in memoria il programma da eseguire e di lanciarlo. Spesso i suoi comandi sono semplici chiamate dirette a sottoprogrammi in linguaggio macchina. Si limita a inizializzare il sistema ed a rimanere poi in attesa di comandi, cedendo il controllo completo della macchina ai programmi che vengono lanciati.

- **Interprete**: è un interprete di comandi che funge anche da **interfaccia utente (shell)**. Questa shell primitiva di solito funge anche da interprete per un linguaggio di programmazione: a seconda delle scelte dei progettisti del software può essere un vero linguaggio oppure un più semplice **linguaggio di scripting** con cui creare **comandi batch** cioè interpretati ed eseguiti lasciando libero l'interprete di eseguire altri comandi (es: Commodore 64).
- **DOS (Disk Operating Systems)**: un computer diventa molto più utile ed efficace se dotato di una memoria di massa e di un software (**file system**) che permetta di organizzare e gestire i dati sui mezzi di memorizzazione secondo una struttura ben precisa. I sistemi operativi che risiedevano su disco e capaci di gestire un **file system** sono detti genericamente Disk Operating Systems, cioè DOS appunto. Il più famoso è il MS-DOS della Microsoft, oggi non più in uso ed evolutosi negli anni nelle varie versioni di Windows, fino alla attuale Windows 7.

Kernel

- Il **kernel** è il cuore di un sistema operativo. Si tratta di un software che ha il compito di fornire ai moduli che compongono il sistema operativo e ai programmi in esecuzione sul computer (chiamati **processi o task**) le funzioni fondamentali ed un accesso controllato all'hardware, sollevandoli dai dettagli della sua gestione.

Scheduler - multitasking

- Lo **scheduler** è il componente fondamentale dei sistemi operativi multitasking, cioè quelli in grado di eseguire più processi/task contemporaneamente. Se ad esempio viene chiesto al sistema di eseguire contemporaneamente due processi A e B, la CPU eseguirà per qualche istante il processo A, poi per qualche istante il processo B, poi tornerà ad eseguire il processo A e così via. In questo modo abbiamo la sensazione che i processi siano eseguiti allo stesso tempo in parallelo.
- Lo scheduler si occupa di gestire lo **stato dei processi** come, ad esempio, fare avanzare un processo interrompendone temporaneamente un altro, realizzando così un cambiamento di contesto (**context switch**).
- Al cambio di task il sistema operativo è costretto a salvare tutti o quasi i registri (memoria locale) della CPU e ricaricarli con quelli salvati dal task che subentra. Esistono vari algoritmi di scheduling che permettono di scegliere nella maniera più efficiente possibile quale task far proseguire.



Gestore della memoria

- Il **gestore di memoria** è la componente del sistema operativo che si occupa di gestire ed assegnare la memoria ai processi che ne fanno richiesta. La gestione della memoria è necessaria per tenere traccia di quanta memoria è impegnata e di quanta invece è disponibile per soddisfare nuove richieste: in mancanza di un sistema di gestione, si avrebbe prima o poi il caso di processi che ne sovrascrivono altri, con gli ovvi inconvenienti.



Interfaccia utente

- **Interfaccia utente**: si tratta di un programma che permette all'utente di interagire con il computer. Esistono sostanzialmente due famiglie di interfacce utente: **interfaccia a linea di comando** e interfacce grafiche (**GUI** = Graphic User Interface)
- L'interfaccia a linea di comando (**shell o terminale**) è l'ambiente di lavoro attraverso il quale è possibile impartire al computer comandi e richiedere l'esecuzione di programmi in modo puramente testuale (es: DOS)
- Con le GUI l'interazione con il computer non avviene più in modo testuale, ma operando su icone e finestre che permettono di fare operazioni complesse con un clic del mouse



Sistemi operativi real-time

- **Sistema operativo real-time**: è un particolare tipo di sistema operativo in grado di garantire una risposta entro un dato tempo limite (millisecondi o microsecondi) a qualunque evento esterno.
- Non hanno memoria virtuale
- Per gestire le periferiche, i sistemi real-time usano spesso, al posto degli interrupt, il così detto **polling** (verifica ciclica di tutte le unità di input/output), meno efficiente, ma deterministico
- I sistemi real-time sono normalmente utilizzati in ambito industriale dove è necessario ottenere una risposta dal sistema in un tempo massimo prefissato.



Memoria virtuale

- Dovendo ospitare in memoria centrale più programmi nello stesso tempo, i sistemi multitask hanno bisogno di più memoria rispetto a quelli monotask. Per far ciò si può utilizzare un'architettura a **memoria virtuale**, capace di simulare (in MS) uno spazio di memoria centrale maggiore di quello fisicamente presente
- In questo caso, il gestore della memoria si occupa anche di **mappare** (INDIRIZZARE) la memoria virtuale offerta ai programmi sulla memoria fisica e sui dischi rigidi del sistema, copiando da memoria a disco rigido e viceversa le parti di memoria necessarie di volta in volta ai programmi, senza che i programmi stessi o gli utenti debbano preoccuparsi di nulla.
- Questa operazione è normalmente gestita via hardware tramite una **MMU**, (Memory Management Unit), generalmente integrata all'interno della CPU.



Sistemi operativi multiutente

- **Sistema multiutente**: se un computer può far girare più programmi contemporaneamente, allora può anche accettare comandi da più utenti contemporaneamente.
- I problemi di sicurezza di questi sistemi si risolvono assegnando un **account** univoco per ogni utente, assegnando un proprietario ai file ed ai programmi con un sistema di permessi per l'accesso ad essi e prevedendo una gerarchia di utenti (cioè di account). Così facendo, il sistema rifiuterà tutti i comandi potenzialmente "pericolosi" e li accetterà soltanto se impartiti da un utente in cima alla gerarchia, cioè l'amministratore del sistema (generalmente **Administrator** nei sistemi Windows).