



***Architettura del Sistema Operativo:
gestore delle periferiche, file system,
interprete dei comandi***

1. Gestore delle periferiche

Le periferiche che un Sistema Operativo deve gestire sono costituite da quei componenti hardware connessi al computer attraverso delle porte (USB, seriali...): quali il mouse, la tastiera, la stampante, il masterizzatore ecc.

Il compito del Sistema Operativo è quello di "regolare" l'utilizzo di queste risorse, ovvero deve assegnarle ai processi che ne fanno richiesta durante la loro esecuzione, garantendo una visione astratta dei dispositivi in maniera tale che l'utente abbia l'impressione di essere l'unico utilizzatore della periferica.

Tale astrazione viene realizzata dai **driver**: programmi che "pilotano" la periferica e che fanno da interfaccia tra la periferica e i programmi che vi possono accedere richiamando il driver relativo.

Ad esempio Il sistema operativo non comunicherà direttamente con una stampante per la stampa di un documento, ma passerà al driver il documento ordinandogli di occuparsene.

I driver inoltre:

- gestiscono gli errori che possono verificarsi nell'interazione tra CPU e periferiche;
- consentono di gestire anche eventuali conflitti che possono presentarsi nell'accesso ad una periferica.

Infatti, può succedere che più richieste di stampa arrivino ad una stampante. In tal caso il driver riesce a virtualizzare il concetto di stampante facendo credere agli utenti di possederne più di una.

Il meccanismo che consente questa virtualizzazione è detto **spooling** (da SPOOL: *Simultaneous Peripheral Operation On Line*): in pratica il driver si assicura che venga stampato il primo documento arrivato e contestualmente accetta le eventuali altre richieste di stampa che memorizza su disco fisso in un'area ben precisa chiamata **area di spool**. Queste richieste saranno evase utilizzando il metodo **FIFO** secondo il quale il primo documento che arriva in coda (**First In**) è anche il primo che sarà stampato (**First Out**).

Ogni dispositivo di Input/Output possiede un driver per la sua gestione che garantisce l'affidabilità del sistema operativo e l'immediato aggiornamento di nuove tecnologie, essi sono in genere distribuiti dalla casa produttrice della periferica.

Il sistema operativo rende efficiente la comunicazione verso un determinato dispositivo periferico anche mediante l'utilizzo di un'area di memoria detta **buffer**, in cui si accodano i dati diretti verso la periferica o provenienti dalla periferica e destinati ai programmi.

Un altro componente fondamentale per il gestore delle periferiche è Il **controller**.

Il controller è un dispositivo elettronico dedicato a gestire e a far accedere al bus una o più unità periferiche; possiede una propria memoria interna e una serie di registri specializzati¹.

Nelle versioni più recenti dei sistemi operativi, la necessità di configurare "manualmente" ogni periferica tramite appositi driver viene sostituita da funzioni **Plug&Play (PnP)**. I Plug & Play, in sostanza, sono device autoinstallanti infatti la traduzione letterale dell'espressione è "collega e usa" e indica proprio la capacità dei dispositivi di poter essere usati senza dover procedere all'installazione o alla configurazione degli stessi.

Una volta associati al dispositivo al quale vanno collegati, i device "Plug & Play" provvedono autonomamente ad installare i driver necessari e sono subito all'uso. È quello che succede quando vengono collegati al computer un mouse, una chiavetta USB oppure un disco rigido esterno, solo per fare qualche esempio.

2. Gestore del file system

La gestione delle informazioni ha un ruolo importante nell'architettura di un Sistema Operativo in quanto le informazioni sono la risorsa fondamentale di ogni applicazione.

Il termine **file**, in informatica, indica un contenitore di informazioni/dati in formato digitale, tipicamente presenti su un supporto digitale di memorizzazione opportunamente formattato in un determinato file system. Le informazioni scritte/codificate al suo interno sono leggibili solo tramite uno specifico software in grado di effettuare l'operazione².

I file sono oggetti che esistono indipendentemente dalle applicazioni che li usano e sono collocati in unità periferiche, come strutture permanenti di memorizzazione delle informazioni.

La parte del Sistema Operativo che gestisce le informazioni è il **file system**: esso si occupa della gestione delle informazioni memorizzate su disco in particolare, identifica e cataloga i file presenti, gestisce le operazioni sui file, assegna i file ai processi che lo richiedono, stabilisce i meccanismi di protezione, gestisce l'allocazione delle aree disco.

In definitiva esso realizza l'indipendenza dai dispositivi fisici facendo corrispondere ad un modello logico, **File System Logico** (è il modo in cui l'organizzazione delle informazioni è presentata all'utente), una struttura fisica, **File System Fisico** (è il modo in cui le informazioni sono scritte fisicamente sulla memoria di massa in modo tale che il sistema possa ritrovarle per poterle leggere e/o modificare), nascosto all'utente.

Poiché le directory possono contenere sia file che altre directory, viene a definirsi un'organizzazione dei file e delle directory con struttura gerarchica ad albero, detto **directory tree** o **albero delle directory**, praticamente si tratta di un grafo in cui i nodi sono connessi tra loro da frecce che rappresentano la relazione di contenimento, in modo tale che dal nodo radice è possibile raggiungere qualsiasi altro nodo ognuno secondo un percorso univoco.

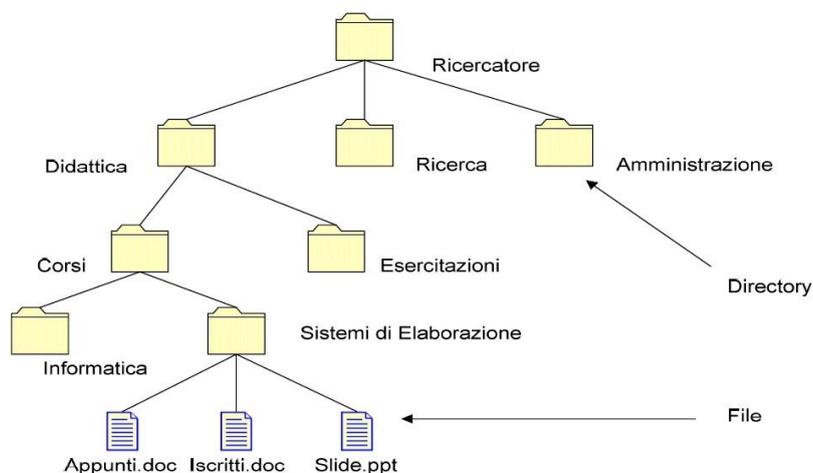


Figura 1: Esempio di albero

Un filesystem è pertanto il meccanismo che fa corrispondere l'albero delle directory (struttura logica) ai settori sul disco (struttura fisica).

Ogni sistema operativo supporta uno o più file system. Alcuni file system sono nativi per determinati sistemi, ad esempio FAT32 sotto MS-DOS o Windows9x; ext2, ext3, reiserFS sotto Linux/UNIX e HFS sotto Mac. Altri invece sono comuni a tutti, si pensi ai file system ISO(2) dei CD-ROM. Esistono anche file system basati su TCP/IP (come NFS), che si estendono su più macchine connesse in rete in maniera del tutto trasparente per l'utente.

2.1 Struttura logica

Come già detto in precedenza, il file system logico ha una struttura costituita da due tipi di oggetti, la **directory** ed il **file**, e la sua organizzazione è quella di un **albero** i cui **nodi** rappresentano gli oggetti del filesystem ed in particolare i file rappresentano le **foglie** dell'albero ovvero i nodi terminali.

Si definisce **path** (percorso), la sequenza delle cartelle a cui accedere per raggiungere un determinato nodo dell'albero.

Il path può essere di due tipi:

- **assoluto**: è un percorso che descrive la posizione di un file o di una cartella indipendentemente dalla directory di lavoro corrente partendo sempre dalla root directory (compresa);
- **relativo**: è un percorso che descrive la posizione di un file o di una cartella rispetto alla directory di lavoro corrente.

I processi possono invocare funzioni di sistema per operare sul file system quali: creazione e cancellazione di un file, modifica del nome del file e così via. Il file system decide le modalità con cui vengono effettuate le operazioni; per esempio se si cerca di creare un file che esiste già può far fallire l'operazione o sostituire il file con quello nuovo vuoto.

2.2 Struttura fisica

Il supporto più utilizzato per la memorizzazione dei file è il disco. La memorizzazione sul disco avviene per settori.

Fisicamente i file vengono memorizzati come sequenze di blocchi; un blocco è una unità di informazione che può essere trasferita con un'unica operazione di lettura o scrittura dalla memoria al disco e viceversa.

Per conoscere quali blocchi sono assegnati a ciascun file è necessario utilizzare delle informazioni aggiuntive.

Le tecniche principali utilizzate sono:

- **blocchi concatenati**: i blocchi del file sono collegati tra loro in una lista concatenata, ovvero in ogni blocco è presente un puntatore al blocco successivo;

l'accesso ai blocchi può avvenire solo in modo sequenziale scorrendo la catena di puntatori, inoltre si hanno molti accessi al disco;

- **tabella di puntatori (o FAT):** si tratta di una tabella formata da tanti elementi quanti sono i blocchi del disco; ogni elemento può contenere un puntatore al blocco successivo se il blocco appartiene a un file, o un'indicazione che il blocco è libero; il numero del primo blocco del file viene memorizzato nel descrittore del file e la catena di puntatori viene scorsa solo all'interno della tabella, che può essere copiata in memoria centrale per rendere più veloce l'accesso;
- **mappa dei blocchi:** per questa tecnica si utilizza una tabella per ogni file, contenente la sequenza dei puntatori ai blocchi del file; la tabella può essere memorizzata nel descrittore del file ed essere caricata in memoria centrale quando il file viene aperto.

Altra cosa fondamentale è che ci deve sempre essere un modo per individuare i blocchi liberi da assegnare a un nuovo file; per fare questo possono essere usate varie modalità:

- tutti i blocchi liberi sono concatenati in una lista dello spazio libero;
- con la **tabella di puntatori** i blocchi liberi possono essere collegati in una lista di puntatori all'interno della tabella; la ricerca dei blocchi liberi viene fatta scandendo la tabella e rende semplice la ricerca di blocchi liberi con requisiti di vicinanza;
- **mappa dei blocchi liberi:** si può avere un puntatore al primo blocco di un'area libera seguito dal numero di blocchi liberi contigui;
- **bit map:** si utilizza una sequenza di bit, uno per blocco, che indica se il blocco corrispondente è libero o occupato.

I record di un file possono essere:

- sbloccati quando ogni blocco contiene un record logico;

- bloccati quando ogni record contiene più record logici;
- multiblocco quando un record logico occupa più blocchi;

Il file system fisico si occupa della memorizzazione dei file usando diversi metodi di allocazione dei blocchi, con blocchi contigui o no.

Se vengono utilizzati blocchi contigui la dimensione del file deve essere dichiarata in fase di creazione; in particolare sul disco deve esserci un'area libera della dimensione richiesta inoltre è un problema stabilire la dimensione del file: se è sovradimensionato si spreca spazio, in caso contrario può essere necessario in seguito ampliare il file, eventualmente spostandolo in una zona di disco più ampia.

Per questo si preferiscono in genere metodi di allocazione in cui si utilizzano blocchi non contigui.

In ogni caso è possibile riscontrare il problema della frammentazione; in particolare è possibile avere frammentazione interna perché i blocchi possono non essere occupati interamente e frammentazione esterna quando si cancellano dei blocchi.

2.3 Tipi di file system

Esistono diversi tipi di file system, tutti con diverse strutture e proprietà logiche, come ad esempio velocità e dimensioni.

Il tipo di file system può variare in base al sistema operativo e alle esigenze di tale sistema operativo. I tre sistemi operativi per PC più comuni sono Microsoft Windows, Mac OS X e Linux. I sistemi operativi mobili includono Apple iOS e Google Android.

I principali file system includono quanto segue:

La tabella di allocazione dei file (FAT) è supportata dal sistema operativo Microsoft Windows. FAT è considerato semplice e affidabile ed è modellato su file system legacy. FAT è stato progettato nel 1977 per i floppy disk, ma è stato successivamente adattato per i dischi rigidi. Sebbene efficiente e compatibile con la maggior parte dei sistemi operativi attuali, FAT non può eguagliare le prestazioni e la scalabilità dei file system più moderni.

Il file system globale (GFS) è un file system per il sistema operativo Linux ed è un file system del disco condiviso. GFS offre accesso diretto allo storage a blocchi condiviso e può essere utilizzato come file system locale.

GFS2 è una versione aggiornata con funzionalità non incluse nel GFS originale, come un sistema di metadati aggiornato. Secondo i termini della GNU General Public License, entrambi i file system GFS e GFS2 sono disponibili come software libero.

Il file system gerarchico (HFS) è stato sviluppato per l'uso con i sistemi operativi Mac. HFS può anche essere indicato come Mac OS Standard ed è stato sostituito da Mac OS Extended. Introdotto originariamente nel 1985 per floppy e dischi rigidi, HFS ha sostituito il file system Macintosh originale. Può essere utilizzato anche su CD-ROM.

Il file system NT, noto anche come New Technology File System (NTFS), è il file system predefinito per i prodotti Windows dal sistema operativo Windows NT 3.1 in poi. I miglioramenti rispetto al precedente file system FAT includono un migliore supporto dei metadati, prestazioni e utilizzo dello spazio su disco. NTFS è supportato anche nel sistema operativo Linux tramite un driver NTFS gratuito e open source. I sistemi operativi Mac hanno il supporto di sola lettura per NTFS.

Universal Disk Format (UDF) è un file system indipendente dal fornitore utilizzato su supporti ottici e DVD. UDF sostituisce il file system ISO 9660 ed è il file system ufficiale per DVD video e audio scelto dal DVD Forum.

3. L'interprete dei comandi

L'interprete dei comandi (shell) è la parte più esterna del sistema operativo con la quale l'utente interagisce.

Attraverso l'interprete, l'utente richiede un servizio al sistema operativo:

- l'utente inserisce un comando in accordo ad una certa sintassi
- l'interprete riconosce il comando ed attiva una serie di funzioni dei livelli interni
- l'esecuzione di tali funzioni porta al compimento del comando richiesto

L'interfaccia di una shell può essere:

- grafica (GUI);
- testuale (TUI);
- a riga di comando (CLI).

La shell testuale ha un'interfaccia a riga comando (CLI). L'utente digita i comandi da eseguire in un'apposita riga. La shell testuale caratterizza i primi sistemi operativi della storia.

La **shell grafica** è caratterizzata da un'interfaccia grafica detta **GUI** (*Graphic User Interface*). Le shell grafiche rendono più semplice l'interazione tra il software e l'utente. Per eseguire comandi non è necessario digitarli in una riga comandi, è sufficiente cliccare sugli elementi grafici (icone, finestre, pulsanti, ecc.) visualizzati sullo schermo.