

I. Il software

- I.1 [Introduzione](#)
- I.2 [I sistemi operativi](#)
- I.3 [I software applicativi](#)

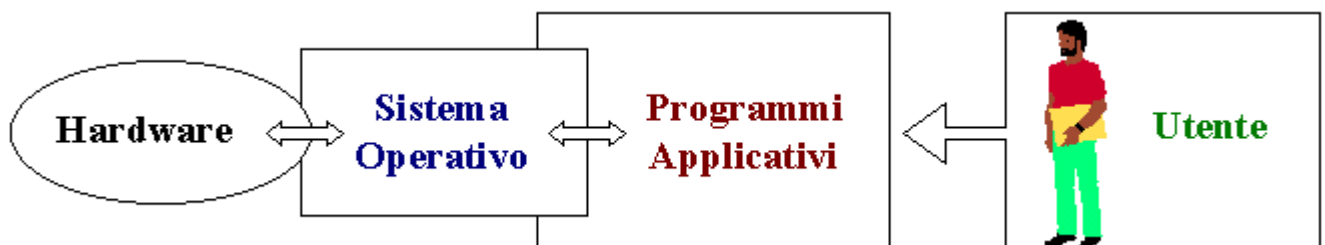
- I.4 [Licenze software: introduzione](#)
- I.5 [Software libero](#)
- I.6 [Software non libero](#)
- I.7 [Software commerciale](#)

I.1 Introduzione

Il software viene usualmente suddiviso dal punto di vista della sua funzionalità. Si usano distinguere tre componenti: il sistema operativo, il software applicativo e i dati, anche se da un punto di vista teorico la terza componente può conglobarsi con la seconda.

Il sistema operativo (SO) o software di sistema è un insieme di programmi che gestisce il funzionamento di base del computer, il software applicativo è un insieme di programmi creati per utilizzare il calcolatore secondo le diverse finalità come ad esempio i programmi di scrittura, di ritocco fotografico, gli antivirus, ecc. (installati sulla macchina dopo che questa contiene già il sistema operativo), i dati sono una serie di informazioni "statiche", utilizzate dai programmi ma incapaci, da soli, di mettere in moto il calcolatore.

In definitiva si può dire che il Software di sistema serve alla macchina per funzionare, mentre il Software applicativo serve all'utente per lavorare.



I.2 I sistemi operativi

Il sistema operativo si occupa dell'interazione tra utente e computer. E' quell'insieme di programmi che rende il computer semplice ed efficiente per l'utente.

Il sistema operativo agisce da intermediario tra l'utente (ed i suoi programmi applicativi) e l'hardware in modo da governare le risorse costituenti il computer.

Lo scopo è quello di fornire un ambiente in cui l'utente possa eseguire i suoi programmi in maniera semplice, conveniente (user friendly) ed efficiente.

Deve essere conveniente in quanto:

- a. deve mostrare in maniera chiara all'utente i dati contenuti nella memoria di massa;
- b. deve curarsi delle cose marginali che non interessano l'utente quali: I/O con le varie periferiche, tramite i device drivers. Non deve essere l'utente ad istruire il computer su come scrivere un dato di output; deve caricare i programmi dalla memoria di massa alla memoria volatile;
- c. deve curarsi che il passaggio di dati tra un programma e l'altro sia semplice;
- d. deve curarsi che passare da un programma in esecuzione all'altro sia semplice;
- e. deve controllare l'esecuzione di un programma e se questo "abortisce" deve dire perché;
- f. deve prevenire l'uso improprio di una qualsiasi componente del sistema

Deve essere efficiente in quanto:

- a. deve allocare le varie risorse hardware tra i vari programmi applicativi. Deve fare in modo che le risorse siano sempre occupate a far lavoro utile e che tutti i programmi siano serviti delle risorse in maniera equa;
- b. deve badare alla sua sopravvivenza.

Funzionamento. I sistemi operativi sono degli Interrupt based systems, ovvero dei programmi sempre in esecuzione (sempre residenti nella RAM) che ascoltano e, quando interrotti dall'utente o dai programmi fanno quello per cui sono stati interrotti. Ciò fa sì che l'ambiente che creano sia interattivo.

Dal punto di vista dell'interfaccia con l'utente esistono due categoria di sistemi operativi: *a linea di comando* e *ad interfaccia grafica*.

Esempio tipico di sistema operativo a linea di comando è Ms-Dos: sullo schermo (di un colore scuro uniforme) non compare nessuna grafica e tutti i comandi devono essere digitati da tastiera.

```

C:\TEMP>dir

Il volume nell'unità C non ha etichetta
Numero di serie del volume: 226D-1E07
Directory di C:\TEMP

.           <DIR>          21/12/98   23.32
..          <DIR>          21/12/98   23.32
_SETUP.DLL  11.264   12/05/97   17.32
      1 file          11.264 byte
      2 dir           638.816.256 byte disponibili

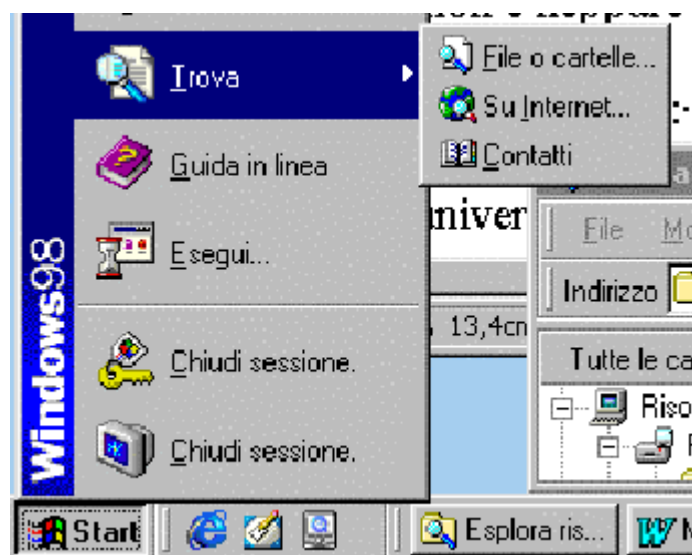
C:\TEMP>

```

Tali sistemi operativi sono estremamente scomodi e difficili da usare, per cui sono stati creati programmi che, appoggiandosi comunque alle funzioni del sistema a linea di comando, forniscono all'utente un'interfaccia grafica semplice da usare.

Esempi tipici di SO a interfaccia grafica sono Windows 98 (e successivi) e MacOS (piattaforma Macintosh). In questi sistemi GUI (Graphical User Interface) tutte le operazioni si svolgono tramite icone e finestre, usando intensivamente il mouse per lanciare comandi, scegliere opzioni, ecc.

Questi sistemi si dicono user friendly (amichevoli verso l'utente) perché anche un utente inesperto riesce, entro certi limiti, ad intuirne il funzionamento (il che non accade certamente con i sistemi a linea di comando).



Tramite il sistema operativo (SO) i programmi applicativi comunicano con l'hardware evitando di dover includere nei programmi stessi tutte le istruzioni necessarie a questa gestione. Nei primi calcolatori e in alcuni casi anche in moderni calcolatori destinati a funzioni particolari non esiste un sistema operativo.

La ragione principale che ha portato alla nascita dei sistemi operativi è la concorrenza di accesso ad una risorsa. Il sistema operativo permette infatti a più programmi, e quindi eventualmente a più utenti, di accedere contemporaneamente alla medesima risorsa hardware: ad esempio alla CPU, alla memoria o al disco.

Loader, ROM e bootstrap. Nei primi calcolatori il sistema operativo svolgeva solamente il compito di caricare degli altri programmi. Questa parte particolare del sistema operativo è rimasta tuttora e viene detta loader.

Il loader viene caricato in memoria all'accensione della macchina usando la memoria ROM (alla partenza del computer, o all'esecuzione di un RESET della macchina, viene messo nel program counter l'indirizzo della ROM che contiene il loader).

Normalmente nella memoria ROM viene caricato quell'insieme minimo di funzioni che permette di far partire il computer caricando da altri dispositivi periferici il resto del sistema operativo. La fase di caricamento del sistema operativo viene detta bootstrap (letteralmente significa alzarsi tirandosi per i lacci delle scarpe, a sintetizzare il fatto che l'avvio avviene a partire dalla piccola parte che è stata precaricata in memoria ROM).

Nel caso dei personal computer IBM compatibili, le ROM contengono quello che viene chiamato BIOS (Basic Input/Output System). Il BIOS si incarica di eseguire una procedura di autodiagnostica (POST, Power On Self Test) che procede a tutta una serie di controlli e verifiche sulle varie componenti e permette di caricare il sistema operativo da hard disk (o da floppy disk).

Nei personal più vecchi, quando si voleva aggiungere un dispositivo hardware che non era stato previsto o sostituirne uno nuovo, era necessario sostituire la ROM. Con i sistemi attuali, invece, ciò non è più necessario in quanto ogni periferica nuova che si collega al computer (stampanti, unità di memoria di massa, schede...) richiede che venga installato il proprio driver (programma che gestisce il funzionamento del dispositivo) nel sistema operativo.

Gestione delle risorse fisiche. Nel caso dei primi sistemi i programmi eseguivano direttamente le istruzioni di gestione delle risorse fisiche e quindi provvedevano a usare direttamente le istruzioni di Input/Output sulle periferiche. Per facilitare lo sviluppo di programmi sono state inserite nel sistema operativo funzioni di lettura e scrittura, rispettivamente, dalle e sulle periferiche che consentono di non conoscere i dettagli di questa gestione. Nei sistemi operativi più moderni la gestione delle periferiche viene fatta attraverso un insieme di funzioni molto limitato ed uguale su tutte le periferiche. Questa gestione è del tutto analoga a quella usata per utilizzare i file del disco.

Interruzioni, device driver e virus. La CPU, dopo aver mandato in esecuzione delle istruzioni di I/O ad una periferica, rimane in attesa della risposta della periferica: ciò è necessario perché la CPU è molto più veloce delle periferiche.

La soluzione che è stata inventata e che è la base dell'intera architettura di funzionamento dei sistemi operativi è la seguente: invece di mandare continuamente segnali alla periferica e disturbarla per chiedere se ha finito, è la periferica a mandare un segnale quando è pronta a spedire una risposta. Questo segnale è un particolare filo del bus di controllo che viene detto Interrupt.

L'idea che viene subito è di sfruttare questo periodo di attesa della CPU per eseguire le istruzioni di un'altro programma (paragrafo successivo).

Il sistema operativo, all'arrivo di una interruzione, stabilisce quale periferica ha provocato l'interruzione e quali istruzioni eseguire per la gestione di quella interruzione. L'insieme delle

istruzioni che vengono eseguite in seguito ad una interruzione viene chiamato device driver di quella periferica.

Questo meccanismo è così comodo che viene usato anche per le chiamate del sistema operativo fatte dai programmi applicativi: quando si desidera una funzione particolare del sistema operativo viene chiamata una istruzione speciale di interrupt detta system call o interrupt software.

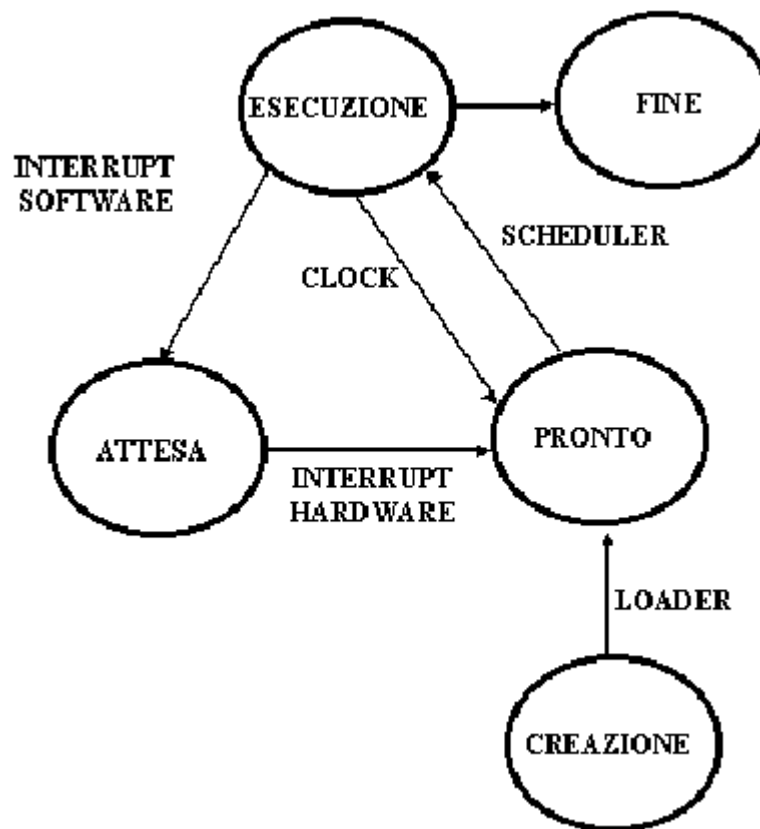
Lo stesso meccanismo è anche utilizzato anche per scopi illegali. I virus informatici sono dei piccoli programmi che, modificando gli indirizzi delle istruzioni di interrupt, si fanno mettere in esecuzione ogni qualvolta un programma richiede un particolare intervento del SO. In questo modo il virus in esecuzione scrive le proprie istruzioni nel disco (infettandolo o danneggiandolo) e se ha l'accortezza di terminare lanciando il vero device driver esso passa del tutto inosservato. I virus diventano in un certo senso una parte del sistema operativo che, essendo unico, consente di far funzionare il virus e diffondere l'infezione su tutte le macchine con quel SO.

Gestione della CPU: Kernel. Vediamo ora quale sia il modo con cui si possono sfruttare i periodi di attesa delle risposte dalle periferiche per poter eseguire più programmi continuamente.

Supponiamo di avere un programma in esecuzione e più programmi caricati in memoria in attesa di essere eseguiti. Il programma in esecuzione rimane l'unico utilizzatore della CPU fino a che non lancia un interrupt software: in quel momento una parte del SO detta scheduler si incarica di capire se è possibile mettere in esecuzione altri programmi (che siano "pronti" per essere eseguiti, cioè non in attesa di una qualche risposta da una periferica) e fra questi viene messo in esecuzione il programma con maggiore priorità rispetto agli altri (decisa, ad esempio, dal periodo che ogni programma ha atteso). Così anche quando una periferica invia un interrupt hardware, il programma che era in attesa di quella risposta viene rimesso nello stato di "pronto" e ritorna in esecuzione appena lo decide lo scheduler.

A turno tutti i programmi sono messi in esecuzione se essi effettuano operazioni di accesso alle periferiche. Tuttavia se un programma in esecuzione non effettua nessun interrupt software, è il clock di sistema a provvedere mandando periodicamente delle interruzioni.

Il grafico sotto illustra sinteticamente il funzionamento dell'esecuzione concorrente di programmi in un moderno sistema operativo (i cerchi sono gli stati dei processi, le frecce sono gli eventi che determinano un cambiamento di stato) a partire dalla creazione del processo fino al suo termine.



La parte del sistema operativo che contiene lo scheduler e la sincronizzazione dei processi si occupa, di fatto, della gestione della risorsa più importante, cioè della CPU, e per questo viene detta Kernel (nocciolo).

Problemi di parallelismo e concorrenza. Se due processi vogliono accedere contemporaneamente alla stessa risorsa, cioè alla stessa periferica od alla stessa zona di memoria, in sola lettura non si pongono problemi di sorta.

La situazione cambia se qualcuno dei processi scrive sulla risorsa. In questo caso dobbiamo aspettare che il processo finisca la scrittura prima che un altro processo possa poter procedere ad una lettura od a una scrittura a sua volta. Il problema viene detto di concorrenza e può essere analogo a quello visto per la CPU.

La soluzione che forniscono quasi tutti i kernel dei sistemi operativi è quella che viene detta dei semafori. Un semaforo è un bit della memoria che viene gestito dal sistema operativo. Se vale 1 allora vuol dire che la risorsa è occupata e occorre che il processo si metta in attesa della sua liberazione. Se invece vale 0 allora la risorsa è libera e mettendo il semaforo a 1 si impedisce ad altri l'accesso.

Gestione della memoria. Il SO si occupa anche della gestione della memoria con attenzione ed efficienza, caricando nelle memorie "ad alto livello" solo i dati e le istruzioni che devono essere modificate ed eseguite. Le altre parti sono invece spostate sul disco per poter essere caricate quando necessario.

Se ad esempio il programma da eseguire non può essere contenuto interamente nella memoria centrale (RAM), si ricorre ad una tecnica denominata memoria virtuale (a paginazione), mediante la quale è possibile gestire in modo ottimizzato ed efficiente la memoria principale. Infatti tale memoria è più veloce rispetto al disco ed è quindi interesse del programmatore sfruttarla al meglio.

Più in particolare, se il programma da eseguire ha dimensioni superiori alla RAM, viene suddiviso in varie parti denominate pagine (paginazione). Inizialmente viene caricato in memoria centrale lo stretto necessario per far lavorare il processore, mentre il resto del programma ripartito in una o più pagine, può aspettare pazientemente nella memoria di massa. Non appena è necessario richiamare un altro sottoinsieme di istruzioni, non presenti in memoria centrale, si crea un interrupt che informa il sistema operativo di provvedere a caricare la pagina richiesta in memoria.

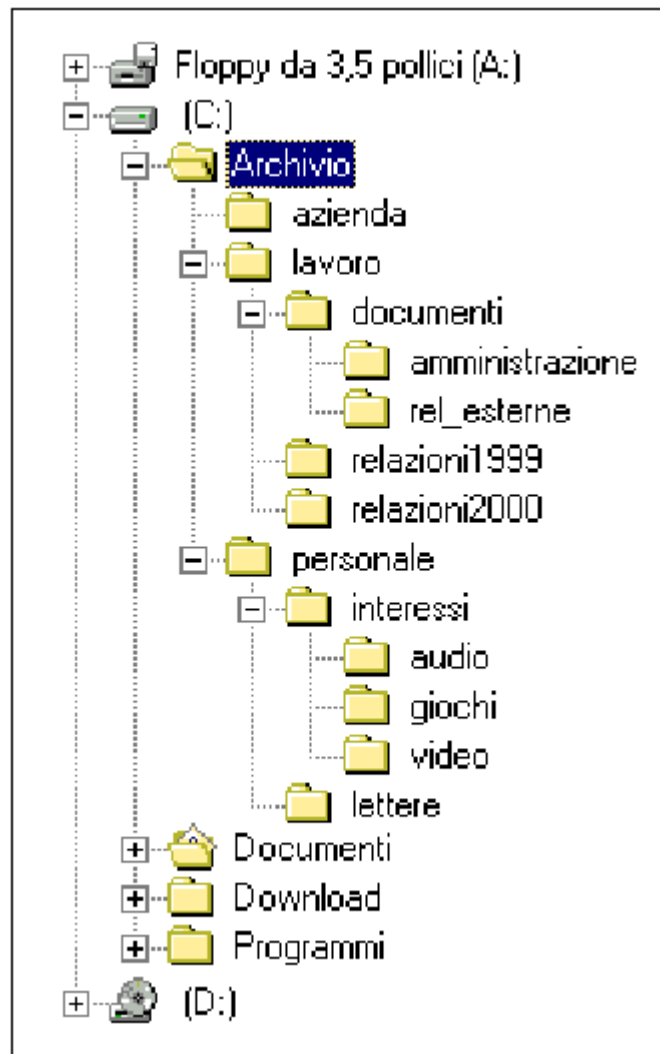
Il File System. Il File System è il modo con cui il sistema operativo gestisce la memorizzazione dei dati sui dischi e sui vari supporti di memoria di massa.

Esistono diversi tipi di file system (a seconda dei vari sistemi operativi), ma comunque tutti organizzano il contenuto dei dischi in Files e Cartelle (o Directory o Folder), seguendo una metafora ripresa dall'archivistica.

Ogni singolo oggetto memorizzato su disco viene detto file (termine che anticamente corrispondeva all'italiano filza, fascicolo, incartamento): un file può essere un programma eseguibile, un insieme di dati numerici, un documento di testo, un'immagine, un documento sonoro, un filmato, una pagina web, un'animazione, o altra cosa.

Sui dischi di un PC possono trovarsi fino a molte decine di migliaia di file e se non ci fosse nessun criterio di ordinamento sarebbe molto complicato riuscire a rintracciare ogni volta il file che ci interessa, per questo motivo il file system permette di creare dei "contenitori", detti cartelle, che permettono di raccogliere i file in gruppi logicamente omogenei. A loro volta le cartelle possono contenere altre sottocartelle e così via, in un gioco di scatole cinesi. Per questo, per sapere dove si trova un file è necessario conoscere tutto il path (percorso) per raggiungerlo. La radice (root) è la cartella di livello gerarchico più elevato, corrispondente a un intero volume (un volume è un contenitore logico di file, corrispondente solitamente a una unità a disco, ad esempio il volume a: indica il floppy disk e c: il disco fisso primario).

Nelle interfacce (GUI) messe a disposizione dal sistema operativo, i file e le cartelle vengono rappresentati con dei simboli grafici detti icone. Le cartelle sono quasi sempre rappresentate con l'immagine di una cartellina d'archivio, mentre i file hanno le icone più varie, a seconda del tipo.



Ogni file e ogni cartella deve possedere un nome che lo distingua dagli altri. Molti sistemi operativi (fra cui Windows) includono nel nome una sigla aggiuntiva detta estensione che caratterizza il tipo di file. L'estensione viene separata con un punto dal resto del nome, così ad esempio in Windows i file che terminano con .exe sono programmi eseguibili, .txt sono file di testo semplice, .doc.xls .mdb .ppt sono documenti rispettivamente di Word, Excel, Access, PowerPoint, .htm e .html sono pagine Web, .wav .mp3 sono file audio, .jpg .gif .bmp .png sono immagini, .mov .avi .mpg sono filmati, e così via. Di solito non si usano estensioni con i nomi delle cartelle (ma sarebbe comunque possibile).

Due oggetti con lo stesso nome (estensione compresa) non possono trovarsi all'interno di una stessa cartella, ma possono invece esistere in due cartelle diverse (anche se contenute una nell'altra).

Per quanto riguarda l'hard disk (su cui si trova il sistema operativo), la maggior parte dei file e delle cartelle che vi si trovano viene creata e gestita direttamente dalle applicazioni senza l'intervento diretto dell'utente (si tratta di file di sistema, di configurazione o di dati).

Ogni disco contiene l'indice di tutti i files, generato e aggiornato automaticamente dal sistema operativo. L'indice memorizza per ogni file il nome, la posizione fisica nel disco, le dimensioni in byte, la data di creazione o modifica, la cartella in cui è contenuto e altre informazioni (a seconda del file system). Il sistema operativo fornisce poi all'utente tutti i programmi per visualizzare

l'indice dei dischi e per organizzarne il contenuto, spostando, copiando o cancellando i file, cambiandone il nome, creando nuove cartelle, ecc.

Alcuni sistemi operativi pongono infine restrizioni e protezioni sui file, impedendo (o perlomeno ostacolando) modifiche di files "delicati" (come i file di sistema) o anche impedendo la visualizzazione di file che contengono informazioni riservate o personali.

Dal punto di vista della memorizzazione su disco, un dato occupa un certo settore in una certa traccia, più dati collegati fra loro possono occupare diversi settori. Per salvare i dati indifferentemente nelle locazioni di memoria, anche in quelle isolate, il sistema operativo crea una tabella sul disco in cui inserire il nome e la lista dei settori usati: ogni riga è un nostro file.

Invece le cartelle sono di fatto tabelle le cui righe contengono nome del file, il numero del file nella tabella delle allocazioni in memoria, la dimensione, l'ultima data di variazione, se si tratta di un file o di una directory.

I file system più diffusi sono: FAT (File Allocation Table), FAT 32 e NTFS (New Tecnology File System) per le piattaforme Windows, HPFS (High Performance File System) per Os/2, EXT2 (Extended Filesystem 2) per Linux.

Classificazione dei SO. Un sistema operativo è *multi-programming* se è in grado di avere più programmi nel processo di essere eseguiti ad ogni istante. In un tale sistema la CPU salta da un lavoro ad un altro a convenienza, ad esempio se occorre aspettare che una qualche periferica compia un certo lavoro.

Un sistema operativo è *time sharing* (o multi-tasking) se la CPU è in grado saltare velocemente da un programma in esecuzione ad un altro e poi ad un altro ancora in modo che tutti i programmi in esecuzione siano effettivamente eseguiti e portati a termine. Il fatto di saltare da un programma all'altro è fatto così velocemente da creare l'illusione che vari lavori si stiano svolgendo simultaneamente, in parallelo. Il time sharing è la caratteristica fondamentale che rende interattivi gli odierni sistemi operativi.

Un sistema operativo è *multi-user* se è in grado di far sembrare un computer come tanti personal computer: uno per ogni utente. Gli utenti comunicano con il computer tramite più terminali.

I principali sistemi operativi.

- Unix. Il più vecchio, sviluppato nel mondo accademico americano per essere usato da supercomputers e mainframes. E' multiprogramming, time-sharing, multi-user ma non è un sistema GUI.
- Linux. Essenzialmente il sistema Unix per PC. E' multiprogramming, time-sharing, multi-user.
- MS-DOS. Il primo sistema operativo per PC. Simile allo Unix ma non è multiprogramming, non è time-sharing, non è multi-user.
- Macintosh. Il primo sistema operativo con GUI. E' multiprogramming e parzialmente time-sharing. La Macintosh ha inventato le finestre, l'uso del mouse ed il concetto di GUI.
- Windows. GUI della Microsoft per MS-DOS. Simile al Macintosh.

- X11. GUI per Unix.
- KDE. GUI per Linux.

I.3 I software applicativi

Numerosi sono i software applicativi con i quali l'utente può lavorare in ambiente Windows. Tale varietà di programmi è caratterizzata dalle differenti caratteristiche degli oggetti da manipolare, per cui ciascuno necessita di apposito software. Infatti ci sono applicazioni dedite a gestire del testo, a creare un foglio di calcolo, a "ritoccare" immagini, a disegnare, ecc. Molto spesso, inoltre, esistono pacchetti di applicativi (o suite) che racchiudono contemporaneamente software specifici, come un programma per scrivere, un foglio di calcolo, un software per le presentazioni, il client di posta, un sistema di gestione di un database e così via. Questa diffusione delle suite di programmi si può spiegare con il fatto che la maggior parte del lavoro che viene generalmente svolto con un computer, a casa o in ufficio, è riconducibile a compiti legati alla produzione di testi, lettere, relazioni, con grafici, tabelle e magari diapositive di presentazione e la loro riunione in un unico pacchetto ne facilita l'utilizzo. Il pacchetto oggi più conosciuto ed utilizzato è Microsoft Office che, pur tra numerose polemiche, ha creato una sorta di standard (gli altri concorrenti lo rincorrono faticosamente, non per la qualità quanto per la diffusione). Tra i pacchetti per l'office automation di libero utilizzo il più conosciuto è, invece, OpenOffice.

Tra i software applicativi, così come per i sistemi operativi, vi è un continuo aggiornamento e/o miglioramento, teso a migliorarne le funzionalità e, a volte, anche per correggere eventuali errori rilevati durante l'utilizzo da parte degli utenti (si pensi, ad esempio, alle varie versioni delle applicazioni Office).

Editor di testo. Il programma più utilizzato (e quindi anche il più rappresentativo di ogni pacchetto di applicativi) è certamente quello per scrivere documenti di testo.

L'applicativo più semplice di questa categoria è l'*editor di testo* o *text editor*. Esso è un programma che consente ad un utente di creare del testo, di salvarlo in memoria e di modificarlo una volta richiamato. Più precisamente tale applicazione consente "l'editing" di testo, ossia il processo di manipolazione di un testo con il quale il testo stesso può essere modificato o risistemato mediante operazioni di aggiunta, di cancellazione o di spostamento di parti.

In Windows 98 è presente ad esempio il text editor Blocco Note che è uno degli "Accessori" del sistema operativo stesso. Questo editor di testo è tra i più semplici da utilizzare, ma ha un numero ridotto di funzionalità; consente comunque di creare file di tipo testo semplice, con formato ".txt".

Con il passare degli anni si sono sviluppati nuovi e sempre più sofisticati programmi che consentono ad un utente di svolgere attività di trattamento testi con l'ausilio di un maggior numero di funzionalità: si tratta dei "text processor" o "word processor". Un esempio è Word Pad (un applicativo di Windows) che, rispetto al Blocco Note, consente di formattare il testo a seconda del tipo di carattere, della sua dimensione, dello stile (grassetto, corsivo, sottolineato). Permette inoltre di stabilire le caratteristiche principali di un paragrafo, tra le quali l'allineamento (a sinistra, centrato, a destra).

Esistono infine software di trattamento testi che permettono la realizzazione di documenti caratterizzati dalla presenza di elenchi, tabelle, immagini, disegni, equazioni, ecc. Il programma più

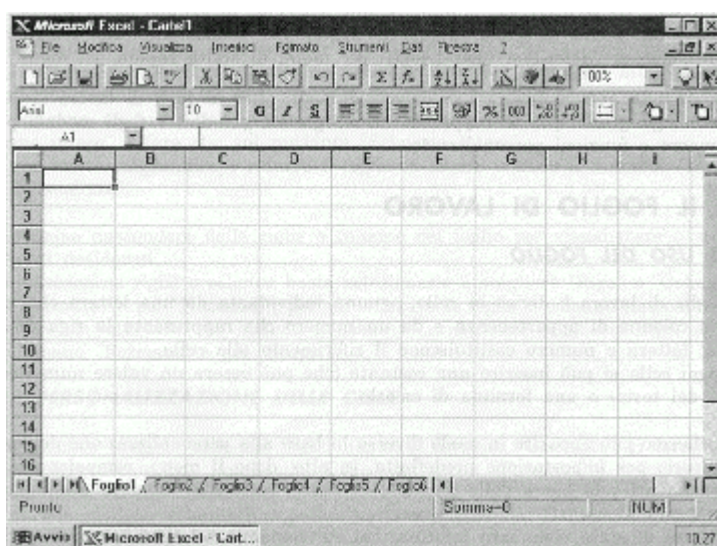
conosciuto è senza dubbio Microsoft Word, un'applicazione del pacchetto Microsoft Office. Mediante tale word processor è possibile creare documenti caratterizzati dalla presenza di elenchi puntati e numerati, di tabelle, di disegni e molto altro. È possibile inoltre integrare informazioni provenienti da applicazioni diverse, per esempio inserire in un documento creato con Word un grafico creato con un'altra applicazione. In Word si possono inserire per di più delle equazioni matematiche complesse, create mediante l'ausilio di un particolare editor di equazioni denominato "Equation Editor".

I file creati con Word vengono salvati per "default", ossia secondo le impostazioni predefinite del programma, come file con estensione ".doc". È comunque possibile salvare il documento, ad esempio, come "Pagina web" (in questo caso l'estensione del file è ".htm" o ".html"), oppure come "Solo testo" (estensione ".txt"). In quest'ultimo caso però il documento potrebbe contenere delle caratteristiche non compatibili con il formato "Solo testo", come ad esempio immagini, disegni, formule. Di conseguenza verrebbe visualizzato un documento non corrispondente all'originale.

Foglio elettronico. La necessità di raccogliere un sempre più elevato numero di dati, di dover definire delle relazioni tra essi e di gestire il tutto in modo semplice, ha portato alla nascita e allo sviluppo del Foglio elettronico. Esso è un quadro di lavoro visualizzato su uno schermo video, costituito da una tabella di celle disposte per righe e per colonne, dove il cambiamento del contenuto di una cella può causare il ricalcolo del contenuto di una o più celle a seconda delle relazioni definite dall'utente.

Uno dei software applicativi che gestisce il funzionamento di un foglio elettronico è Excel, un'altra applicazione del pacchetto Microsoft Office. Avviando tale programma viene presentata una cartella di lavoro che contiene dei fogli di lavoro in cui è possibile immettere i dati. Una cartella può contenere tipi diversi di fogli: oltre ai fogli di lavoro si possono utilizzare fogli grafici per la rappresentazione grafica dei dati.

Come si può osservare dalla figura, il foglio di lavoro è diviso in celle.



Ognuna di queste celle è individuata da una lettera che rappresenta la colonna di appartenenza e da un numero che rappresenta la riga di appartenenza; lettera e numero costituiscono il riferimento alla cella.

In ogni cella si può inserire una costante (che può essere un valore numerico, una data, o del testo) o una formula di calcolo.

Microsoft Excel risulta particolarmente utile ad esempio, quando ci sono dei dati numerici e occorre visualizzare i risultati in maniera grafica. Mediante tale software applicativo è possibile eseguire un'associazione tra la gestione dei dati raccolti e i risultati, espressi anche mediante fogli grafici.

I file creati con Excel vengono salvati, secondo le impostazioni predefinite del programma, come file con estensione ".xls", ossia come "Cartella di lavoro Microsoft Excel". È altresì possibile, come in Microsoft Word, salvare il documento come pagina web.

Gestione database. L'incremento demografico di una città o di una nazione provoca l'aumento del numero di cittadini e un conseguente incremento dei dati anagrafici di ognuno di essi. Inoltre con il trascorrere degli anni ogni persona consegue differenti titoli di studio, cambia professione, si sposa, va in pensione, ..., quindi occorre aggiornare i dati relativi ad ogni persona dopo averli facilmente trovati.

Questo è un esempio in cui si ha a che fare con un insieme di dati da tenere in ordine e aggiornare. Nasce allora la necessità di avere un software applicativo capace di gestire una base di dati o Database.

Un database è una raccolta di dati memorizzata su un supporto dati, organizzata in relazione alle strutture proprie dei dati contenuti e sulla quale è possibile operare o svolgere elaborazioni in modo automatico.

Esistono differenti tipi di database:

- Database archiviato: copia di sicurezza di una base di dati, salvata allo scopo di poter ripristinare la base dati originale;
- Database distribuito: base di dati dispersa fisicamente su diversi supporti, gestita da un apposito sistema per la gestione di base di dati;
- Database gerarchico: base di dati nella quale i dati sono organizzati secondo criteri gerarchici esprimibili mediante una struttura ad albero;
- Database relazionale: base di dati organizzata in funzione di relazioni esistenti tra gli oggetti del database.

Un database è dunque un insieme di informazioni relative ad un particolare argomento o proposito, come ad esempio la gestione degli ordini di clienti o l'organizzazione di una collezione di brani musicali.

Uno dei software applicativi per la gestione di database relazionali è Microsoft Access del pacchetto Office. Con questo programma è possibile gestire tutte le informazioni da un unico file database; tale file ha estensione ".mdb", ossia database di Microsoft.

Tutti gli elementi di Access sono oggetti, quelli fondamentali sono:

- Tabelle: ogni tabella contiene le informazioni relative ad una categoria di dati;
- Maschere: permettono di definire videate per immettere dati in una tabella, ma anche di creare finestre di dialogo;

- Query: permettono soprattutto di effettuare delle selezioni sui dati memorizzati nelle tabelle;
- Report: permettono di predisporre opportunamente le informazioni per la stampa.

Una volta creato un foglio dati è possibile eseguire delle ricerche specifiche sui dati stessi. Si supponga di aver creato un foglio dati contenente nome, cognome e data di nascita dei cittadini di Roma. Mediante le varie funzionalità di Access è possibile eseguire velocemente la ricerca di tutte le persone nate ad esempio lo stesso giorno. È altresì possibile selezionare i cittadini nati in un certo intervallo di tempo (ad esempio dal 1954 al 1960).

In definitiva MS Access permette di gestire abbastanza facilmente basi di dati caratterizzati dalla presenza di relazioni tra determinati oggetti del database stesso.

Editor di grafica. Esistono vari programmi che consentono di creare disegni. Uno di questi è Paint, un'applicazione che fa parte dei sistemi operativi Windows. Con tale software applicativo è possibile disegnare a mano libera, cancellare, cambiare colore, inserire del testo ..., proprio come avviene quando si disegna su carta o tela. Il file creato viene salvato come file Bitmap, ossia con estensione ".bmp".

È possibile inoltre scegliere la risoluzione che dovrà avere il disegno, infatti esistono vari tipi di file bitmap che differiscono a seconda della risoluzione voluta: "Bitmap monocromatica", per dimensioni piccole e disegni in bianco e nero; "Bitmap a 16 colori", per disegni con al più 16 differenti colori; "Bitmap a 256 colori", per disegni con al più 256 differenti colori (aumentando di conseguenza la risoluzione del disegno e la dimensione del file); "Bitmap a 24 bit", per un'ottima risoluzione con conseguente aumento delle dimensioni del file.

In Paint un disegno può essere salvato anche come file con formato ".gif" (Graphics Interchange Format). Questo formato supporta fino a 256 colori per cui è più adatto per la creazione di logo, disegni e qualsiasi altra cosa che contenga solo testo, ma non dispone di un numero sufficiente di colori per riprodurre in modo apprezzabile la maggior parte delle fotografie. Per queste ultime viene utilizzato il file con formato ".tif" che permette un'ottima riproduzione per le foto acquisite da scanner. La dimensione dei file TIFF (Tag Image File Format), data l'alta risoluzione, è relativamente grande e questo rappresenta un inconveniente se si hanno problemi di spazio. Se si vogliono riprodurre foto con una risoluzione accettabile occupando poco spazio, si possono utilizzare file di tipo JPEG (Joint Photographic Expert Group). L'estensione di tale file è ".jpg", ".jpe" o ".jpeg". Un file JPEG rappresenta un'immagine grafica che viene salvata utilizzando tecniche di compressione per ridurne le dimensioni in modo da occupare meno spazio.

Questi file possono contenere fino a 16.700.000 colori e sono più adatte per le fotografie, mentre non si prestano bene ai testi e ai disegni.

Esistono infine i file PNG (Portable Network Graphics), con estensione ".png" o ".ping", che sono un'alternativa ai file GIF.

Per creare disegni, oltre al software Paint, esistono programmi più sofisticati come ad esempio Coreldraw. Come editor per le immagini c'è l'applicazione Imaging di Windows che consente la modifica di file immagine. Se si vogliono inoltre creare fotomontaggi, cartoline, etichette, ..., inserendo immagini proprie, si possono utilizzare per esempio programmi di grafica molto diffusi: Adobe Photoshop, Corel Photo-Paint o Jasc Paint Shop.

Editor audio. Se si desidera registrare la propria voce e successivamente riprodurla si può utilizzare l'applicazione di Windows Registratore di suoni. Il file audio creato mediante tale software applicativo viene salvato come file con formato ".wav". Se però si vuol riprodurre un brano prodotto da un'orchestra sinfonica, bisogna tener conto del fatto che il suono sarà più "ricco": saranno dunque necessarie più informazioni con conseguente aumento della dimensione del file. Di conseguenza, nel caso di riproduzioni musicali il file audio viene compresso secondo uno speciale algoritmo denominato "mp3".

Per i file mp3, in fase di registrazione, vengono utilizzati degli algoritmi di compressione allo scopo di risparmiare spazio su supporto fisico. Successivamente, in fase di riproduzione essi subiscono una decompressione in modo tale da riottenere tutte le informazioni del suono registrato in partenza.

Editor video. Per la riproduzione delle immagini video, il software applicativo presente in Windows è Windows Media Player. Mediante questo programma è possibile riprodurre immagini registrate in vari tipi di file, tra i quali:

- File video (estensione ".avi", ".wmv");
- Filmati MPEG (estensione ".mpeg", ".mpg");

Per creare dei file video si può utilizzare il software applicativo allegato alla vendita di una qualunque telecamera web. Tale programma differisce a seconda del tipo di telecamera e, nella maggior parte dei casi, consente l'acquisizione di immagini e la loro registrazione su file video con formato ".avi". Tra gli altri standard molto utilizzati, in particolare per lo streaming sul WWW, vi è il RealMedia Video elaborato attraverso il RealPlayer (l'ultima versione è RealOne Player), per le sue caratteristiche di compressione dei video.

Navigazione ipertesti. Durante la navigazione sul WWW si ha a che fare con varie pagine Web. Ogni pagina di questo tipo è un file ".html" o ".htm", creato mediante il Linguaggio di Contrassegno Ipertestuale HTML.

L'ipertesto è un sistema non lineare di strutturare le informazioni, infatti consente di passare da un documento all'altro senza che sia necessario seguire un percorso predefinito.

Le immagini che si visualizzano su una pagina non fanno parte del documento HTML, quindi se si vuole salvare una pagina, occorre salvare separatamente tutte le immagini presenti.

Il programma che permette la navigazione in rete si chiama browser. Mediante il browser è possibile visualizzare documenti HTML. I browser più utilizzati sono Internet Explorer e Netscape Navigator.

Per creare un documento HTML non sono necessari particolari software applicativi: è sufficiente un editor di testo. Esistono comunque dei programmi che consentono facilmente di creare pagine Web, ad esempio Microsoft Front Page e Netscape Composer, che appartengono alla categoria WYSIWYG (What you see is what you get).

Programmi autore. Esistono dei software applicativi, a volte vere e proprie piattaforme, che permettono di gestire a proprio piacimento determinati oggetti denominati oggetti multimediali. Si tratta in sostanza di file che consentono di trasmettere conoscenza mediante vari mezzi di comunicazione: audio (nei vari formati WAV, MP3, ecc.), immagini (JPEG, TIFF, GIF, ...), filmati (MPEG, AVI, ...), eccetera.

Tali programmi permettono all'utente di creare nuovo software per applicazioni specifiche e Cd-Rom multimediali.

Caratteristica principale di tali Cd-Rom è l'interattività: è l'utente a gestire le funzionalità del programma in base alle proprie esigenze. Si pensi ad esempio ad una ricerca da effettuarsi all'interno di una enciclopedia multimediale.

Uno dei software applicativi per la multimedialità è Microsoft Powerpoint del pacchetto Office.

Tra i programmi autore sono degni di nota Neosoft Neobook, Toolbook, Macromedia Director e Macromedia Flash (in questi ultimi due casi si tratta di veri e propri standard per creazioni professionali).

I.4 Licenze software: introduzione

Il software è un codice che è trattato dalle leggi sul diritto di autore, in maniera simile a quanto avviene per le opere letterarie. Tradizionalmente, il contratto che regola l'uso del software è la **licenza** ed è sempre importante conoscere i termini di questo accordo per il software con cui si intende avere a che fare.

La legge sul diritto di autore stabilisce già quali sono i diritti di chi produce l'opera e di chi la utilizza, ma nell'ambito del software si è introdotto nella pratica un contratto, non firmato, il cui scopo è spesso quello di limitare ulteriormente i diritti di chi ne fruisce. Questo contratto è la **licenza**.

Il software ha un proprietario (salvo il caso del software di dominio pubblico che viene descritto in seguito), che è tale in quanto «detiene i diritti di autore». Questo proprietario può essere l'autore originale, oppure un altro detentore che ne ha acquisito i diritti in base a un contratto. Il detentore dei diritti di autore è colui che possiede il **copyright**.

L'utilizzo del software può essere concesso gratuitamente o a pagamento, per le operazioni stabilite nel contratto di licenza, o in sua mancanza per quanto stabilito dalla legge. Il pagamento per l'«acquisto» di software, non si riferisce all'acquisizione dei diritti di autore, ma solo delle facoltà stabilite dalla legge, ovvero da quanto indicato nella licenza.

Le licenze software sono documenti legali allegati ai programmi. Senza una tale documento, *un programma non può essere distribuito né modificato senza l'esplicito consenso degli autori*.

All'opposto, una licenza di software libero permette lo studio, la distribuzione, la modifica e la redistribuzione del programma modificato. Questo è possibile in quanto i detentori dei diritti di sfruttamento economico del programma lo consentono, eventualmente a certe condizioni.

I.5 Software libero

Il **software libero** è software che fornisce il permesso per chiunque di utilizzarlo, copiarlo e distribuirlo, in forma originale, o anche dopo averlo modificato, sia gratuitamente che a pagamento. Il software libero può essere tale solo se viene messo a disposizione assieme al codice sorgente, per cui, a questo proposito, qualcuno ha detto: «se non è sorgente, non è software» (*if it's not source, it's not software*).

È importante sottolineare che la «libertà» del software libero non sta tanto nel prezzo, che eventualmente può anche essere richiesto per il servizio di chi ne distribuisce le copie, ma nella possibilità di usarlo senza vincoli, di copiarlo come e quanto si vuole, di poterne distribuire le copie, di poterlo modificare e di poterne distribuire anche le copie modificate.

In alcuni ambienti si preferisce utilizzare la definizione «Open Source» per fare riferimento al software libero nei termini che sono stati descritti, per evitare ambiguità nella lingua inglese. Sotto questo aspetto, nella lingua italiana, come in molte altre lingue, è più appropriato l'uso della definizione «software libero».

Si distinguono quattro punti fondamentali, necessari perché il software possa essere considerato «libero»:

Livello	Descrizione
libertà 0	la libertà di eseguire il programma per qualunque scopo;
libertà 1	la libertà di studiare come funziona il programma e di adattarlo alle proprie esigenze (in tal caso, deve essere disponibile il sorgente);
libertà 2	la libertà di ridistribuire copie del programma;
libertà 3	la libertà di migliorare il programma e di distribuire tali miglioramenti (anche per questo è necessario disporre dei sorgenti).

Un'[elegante definizione](#) di licenza libera fu data per la prima volta da Richard Stallman negli anni Ottanta. Una [definizione più dettagliata](#), equivalente alla prima, fu data negli anni Novanta dalla Open Source Initiative.

Una licenza per il software libero permette a chiunque riceva un programma ad essa soggetto di usarlo per qualunque scopo, di copiarlo, modificarlo e ridistribuirlo. Che la distribuzione sia o meno gratuita non è rilevante. Il codice sorgente deve essere accluso o può essere richiesto al prezzo di costo del trasferimento. Se chi riceve il software può ridistribuirlo solo entro gli stessi termini di licenza, si parla di licenza con [permesso d'autore](#) (*copyleft* in inglese); in caso contrario si parla di licenza *senza permesso d'autore* (*non-copyleft*).

Il software che non può essere commercializzato, pur soddisfacendo i punti precedentemente elencati, viene considerato «software semi-libero», come viene chiarito in seguito.

Vale la pena di elencare alcune definizioni riferite al software libero.

- **Software di dominio pubblico.** Il software di dominio pubblico è software senza copyright. Di per sé, questo tipo di software è libero, solo che, mancando chi può difenderlo, qualcuno può riuscire ad accamparvi dei diritti; pertanto, alcune copie, o varianti di questo software

possono non essere più libere.

- **Software protetto da copyleft.** La parola *copyleft* («permesso d'autore») è un'invenzione e vuole rappresentare il copyright di chi, mentre difende il proprio diritto di autore, vuole difendere la libertà della sua opera, imponendo che questa e le sue derivazioni restino libere. In pratica, una licenza appartenente alla categoria «copyleft» impedisce che chi ridistribuisce il software (originale o modificato che sia) possa aggiungere delle restrizioni ulteriori. Il classico esempio di licenza di questo tipo è la licenza pubblica GNU-GPL.
- **Software libero non protetto da copyleft.** Il software libero non è necessariamente di tipo copyleft e ciò accade quando la licenza non vieta espressamente l'aggiunta di restrizioni da parte di chi lo ridistribuisce. Quando si utilizza software di questo tipo, non è possibile generalizzare: occorre accertarsi dei termini del contratto che riguarda la copia particolare della quale si è venuti in possesso.
- **Software GPL (General Public License).** La **GPL** è il primo e più importante esempio di licenza con [permesso d'autore](#). Chi modifichi software con permesso d'autore e ne distribuisca le modifiche è tenuto a farlo secondo gli stessi termini in cui è distribuito il programma originale. Detto altrimenti, bisogna dare agli altri le stesse libertà che si sono ottenute. Più precisamente, la distribuzione di ogni opera derivata da un programma con licenza GPL deve a sua volta essere soggetta alla licenza GPL o ad una licenza che dia le stesse libertà. La GPL si preoccupa di proteggere il buon nome degli autori imponendo che le versioni modificate di un programma siano chiaramente indicate come tali, se distribuite. È comunque possibile effettuare modifiche per uso personale, o uso interno in un'organizzazione, senza sottostare a formalità particolari. In definitiva, se lo scopo della licenza è di assicurarsi che tutte le copie del programma, comprese quelle modificate, siano accompagnate dal codice sorgente o sia comunque facile procurarselo, allora è bene usare la [GNU-GPL](#). La licenza **GNU-GPL** è l'esempio più importante di licenza che protegge il software libero con il copyleft. Quando si parla di «software GPL» si intende fare riferimento a software protetto con la licenza pubblica **GNU-GPL**. Il [Progetto GNU](#) è stato lanciato nel 1984 per sviluppare un sistema operativo Unix-compatibile completo che fosse [software libero](#): **il sistema GNU**. GNU è un acronimo ricorsivo per “GNU's Not UNIX”; (GNU Non è Unix). Varianti del sistema operativo GNU, che utilizzano il kernel Linux, sono ora ampiamente utilizzate; anche se a questi sistemi ci si riferisce spesso come “Linux”, essi vengono chiamati con più precisione [sistemi GNU/Linux](#).

Per rendere libero un programma bisogna allegare al codice sorgente un file (tipicamente di nome `LICENSE.TXT`) che contenga il testo della licenza scelta. Inoltre, è consigliabile aggiungere all'inizio di ogni file sorgente una riga di copyright. Ad ogni modifica del file bisogna aggiungere l'anno di modifica alla lista degli anni riportata nella riga di copyright. Se risultasse difficile riportare il carattere ©, questo può essere sostituito dai tre caratteri (C). Dopo la riga di copyright bisogna indicare che i termini della licenza di copyright sono contenuti nell'allegato file `LICENSE.TXT`.

Scrivere ex novo una licenza di copyright per il software è una pratica da evitare. È molto difficile scriverne una nuova, anche per un legale specializzato in questa materia. Inoltre, l'uso di una delle licenze il cui uso è consolidato ha diversi vantaggi:

- gli utilizzatori del software le conoscono già, e ne sanno valutare le implicazioni;
- sono scritte da legali;
- hanno provato la loro validità sul campo;

- facilitano lo scambio di codice fra progetti liberi diversi.

I.6 Software non libero

Il software non è libero tutte le volte che non sono soddisfatti tutti i requisiti necessari per poterlo essere. È bene ricordare che il prezzo non è un fattore che limita la libertà, mentre altri dettagli sono più importanti. Anche in questo caso, vale la pena di elencare alcune definizioni che in generale riguardano software non libero.

- **Software semi-libero.** Il software semi-libero è software che permette agli individui di usarlo, copiarlo, modificarlo e distribuirlo, anche modificato, per qualunque scopo, escluso quello di trarne profitto. In altri termini, si potrebbe dire che si tratta di software libero a cui è stata aggiunta la limitazione per la quale questo non può essere usato e distribuito per trarne profitto.
- **Software proprietario.** Il software proprietario è quel software che non è né libero, né semi-libero. Di solito, per «avere a che fare» con questo software è necessario ottenere un permesso speciale, che spesso si limita a concedere l'uso su un elaboratore, o su un gruppo ben determinato.
 - o **Software freeware.** Il termine *freeware* non è abbinato a una definizione precisa, ma viene inteso generalmente come software gratuito, del quale non viene reso pubblico il codice sorgente, che può essere usato e copiato senza poterlo modificare. In questo senso, il prefisso «free» serve solo a evidenziare la gratuità della cosa, ma non la libertà che invece richiede altri requisiti. Un esempio molto noto è il visualizzatore di documenti PDF.
 - o **Software shareware.** Con il termine shareware si fa riferimento a software proprietario che può essere ridistribuito, ma per il quale viene richiesto espressamente il pagamento dopo un periodo di prova. Alcuni di questi programmi smettono di funzionare dopo il periodo di prova o forniscono funzionalità ridotte fino all'acquisto di una chiave che li sblocchi.

I.7 Software commerciale

In base alle classificazioni viste in questo capitolo, il **software commerciale** è tale solo in quanto viene venduto per profitto. Lo sviluppo e la diffusione del software libero dipendono anche dalla possibilità di venderne delle copie, originali o modificate, per trarne profitto. Pertanto, il software che pur offrendo le quattro libertà fondamentali, non consente la commercializzazione per trarne profitto, non viene considerato «libero» in modo completo. In questo senso, è importante evitare di confondere il software proprietario con il software commerciale, perché non sono la stessa cosa.

Si osservi che può esistere anche del software non-libero, che non è nemmeno commerciale.