



# Sistemi Operativi 2

## Breve storia di Linux

Il sistema operativo **Multics (Multiplexed Information and Computing Service)** fu uno dei primi sistemi operativi a condivisione di tempo (time-sharing), ossia multiprocesso e multi-utente, sviluppato attivamente a partire dal 1964 da parte dei centri di ricerca delle compagnie Bell Labs (AT&T Corp., una compagnia telefonica americana) e General Electric, assieme all'università MIT.

Multics mise sul campo tutta una serie di concetti e tecniche costruttive che sono ancora oggi elementi essenziali dei moderni sistemi operativi. Sebbene rivoluzionario, il progetto Multics fu presto abbandonato da Bell Labs, poiché ritenuto troppo complesso da gestire.

A seguito di ciò, Ken Thompson e Dennis Ritchie, due ricercatori di Bell Labs, svilupparono tramite un microcomputer PDP-7 la prima versione di Unics (Uniplexed Information and Computing Service), scritta totalmente in Assembly. Successivamente, sotto proposta di Brian Kernighan, il nome di Unics venne cambiato definitivamente in Unix.

Il sistema operativo Unix si diffuse rapidamente nei successivi 3-5 anni, portandolo allo sviluppo di versioni scritte tramite il linguaggio B e successivamente (e definitivamente) tramite il linguaggio C. Le versioni scritte in tali linguaggio permisero di portare Unix su varie architetture.

Il linguaggio C fu sviluppato da Dennis Ritchie stesso al fine di migliorare il linguaggio B precedentemente sviluppato dal suo collega Ken Thompson. La miglioria principale rispetto al B consiste nell'aggiunta dei tipi di dato (int, float, char, ...) rispetto alle sole generiche word da 4 byte del linguaggio B.

Successivamente, il codice sorgente del sistema operativo Unix venne distribuito ad università e centri di ricerca interamente assieme al proprio codice sorgente, il quale venne anche venduto ad aziende private, portando alla nascita di molte versioni.

Negli anni 80', Richard Stallman sviluppò il sistema operativo GNU (GNU is Not Unix acronimo ricorsivo), basato su Unix ma diverso da esso in quanto non

contenente codice del sistema operativo Unix, e inventò GPL (GNU General Public Licence), una licenza pubblica utilizzata per il software libero. Unix venne inoltre riscritto completamente, aggiungendo pacchetti importanti ad esso, molti presi direttamente da GNU (es: gcc, make, ...).

Negli anni 90', Linus Torvalds sviluppa il kernel Linux, il quale verrà poi utilizzato da altri sistemi operativi basati su Unix o derivati da esso. In particolare, nel 1994 viene definito lo standard Unix, dove un sistema operativo può avere marchio UNIX solo se esso rispetta le SUS (Single Unix Specification) e paga le royalties per l'uso del marchio.

## **Le caratteristiche dei sistemi Unix**

Le caratteristiche di un moderno sistema operativo Unix, indipendentemente dalla sua categoria, sono:

- Multi-utente e multi-processo
- File system gerarchico
- Kernel in grado di gestire la memoria principale, la memoria secondaria, i processi, le operazioni I/O e le risorse hardware in generale
- System call utilizzabili tramite funzioni C che possono essere chiamate per interfacciarsi con il kernel
- Possiedono una shell di sistema, ossia un programma che "esegue programmi" interpretando i comandi dell'utente
- Modularità, programmi di utilità e supporto ad ambienti di programmazione
- Composto da una serie di piccoli programmi che eseguono un compito specifico, limitato, ma in maniera esatta e semplice
- I programmi sono silenziosi, il loro output è minimale e ridotto a ciò che è stato esplicitamente richiesto
- Ogni lavoro complesso può essere svolto come articolazione del lavoro svolto da programmi semplici
- I programmi manipolano solo testo e mai i file binari (es: altri programmi)

## **Utilizzo della Shell di sistema**

## Shell di Sistema



Informalmente, una **shell di sistema** (spesso detta terminale) è un programma che "esegue programmi".

Più formalmente, invece, la shell è un programma interattivo e/o **batch** (ossia "a lotti") che accetta comandi da far eseguire al **kernel**.

Tali comandi non sono necessariamente dei programmi, bensì possono essere anche dei comandi definiti all'interno della shell stessa.

Prima di eseguire un comando, la shell stampa a video un prompt, ossia una stringa nel formato

```
[nome_utente@nome_macchina cwd]$
```

Ogni comando segue la seguente struttura:

```
nome_comando [argomenti_opzionali] argomenti_obbligatori
```

Ad esempio, nel comando `cp -r -i -a -u file_sorgente file_destinazione`, gli argomenti

`-r`, `-i`, `-a`, `-u` sono opzionali, mentre i rimanenti sono obbligatori. Tipicamente, gli argomenti opzionali possono essere utilizzati anche con sintassi alternative (es: per il comando `cp` gli argomenti `-interactive`, `-recursive` sono uguali agli argomenti

`-i`, `-r`). Inoltre, eventualmente essi possono avere un valore aggiuntivo in input (es: l'argomento `-key=1` assegna il valore 1 all'argomento `-key`) e possono essere

raggruppati (es: `cp -ri` è equivalente a `cp -r -i`).

Tutti i comandi lanciati nella shell vengono salvati in una **cronologia**.

Utilizzando le freccette su e giù della tastiera, è possibile scorrere i comandi presenti nella cronologia.

## Super utente (root)



Ogni sistema operativo Linux-based possiede un **super utente** detto **root**, il quale possiede tutti i privilegi di sistema. Pertanto, tale utente possiede accesso ad ogni operazione o comando possibile all'interno del sistema stesso. L'utente root possiede sempre **UID pari a 0**.