

## Sistemi Operativi: Struttura e Servizi

📅 Thu, 6 Oct

### Cos'è un Sistema Operativo?

Un **Sistema Operativo** è un insieme di **software**, il primo programma che viene eseguito quando viene acceso il dispositivo.

- Permette di gestire l'hardware del computer, installare ed eseguire contemporaneamente le altre **applicazioni**.
- È un **insieme di regole** alle quali chi sviluppa applicazioni deve attenersi per far sì che si integrino nell'ambiente del Sistema Operativo.
- Organizza i nostri dati in **file e cartelle**.
- Si occupa del **multiplexing** (o **arbitraggio**) per l'accesso alle risorse hardware da parte delle app.
- Fa da intermediario tra gli **utenti** e il dispositivo stesso.
- Presenta l'hardware in modo più **astratto** e virtuale, più semplice.
- **Protegge e isola** dati e risorse tra utenti e programmi.

### Componenti di un Sistema di Elaborazione

Utenti
Programmi Applicativi
Sistema Operativo
Hardware

### Cosa si chiede ad un Sistema Operativo?

In base al tipo di dispositivo, a un sistema operativo è richiesto un tipo diverso di **efficienza**.

#### ▼ Server e mainframe

Massimizzare le performance e rendere equo l'accesso alle risorse tra i vari utenti.

#### ▼ Laptop, PC e tablet

Massimizzare la facilità d'uso e la produttività del singolo utente.

#### ▼ Dispositivi mobile

Ottimizzare connettività e consumi energetici.

#### ▼ Sistemi embedded

Funzionare con minimo o nullo intervento umano e bassa latenza (*real time*).

### Struttura e Servizi

Com'è strutturato internamente un sistema operativo?

Un sistema operativo si può dividere in tre componenti:

#### ▼ Kernel

Il *programma sempre presente* che gestisce direttamente l'hardware, e offre ai livelli superiori servizi per usarlo più astrattamente.

#### ▼ Middleware

Servizi di alto livello che semplificano l'*interfacciamento (API)* delle applicazioni con le *chiamate di sistema*.

#### ▼ Programmi di sistema

Applicazioni vere e proprie che offrono *funzionalità di supporto* (gestore dei jobs, interfaccia utente...)

### | Servizi offerti da un Sistema Operativo

#### ▼ Controllo processi

Servizi che permettono di caricare in memoria un programma, eseguirlo e verificarne le condizioni di terminazione.

#### ▼ Gestione file

Servizi che rendono possibile scrivere, leggere e in generale manipolare file e directory.

#### ▼ Gestione dispositivi

Servizi che permettono alle app di utilizzare dispositivi periferici.

#### ▼ Comunicazione tra processi

Servizi che permettono a diversi processi applicativi di comunicare tra di loro.

#### ▼ Protezione e sicurezza

Servizi che assicurano di isolare dati e informazioni tra utenti e processi.

#### ▼ Allocazione delle risorse

Allocare le risorse hardware (CPU, RAM, I/O) ai programmi in modo equo.

#### ▼ Rilevamento errori

Rilevare e segnalare errori che avvengono a livello hardware o software.

#### ▼ Logging

Mantenere traccia di quali programmi usano quali risorse.

## Chiamate di Sistema e API

Un **programma applicativo** che vuole utilizzare servizi presenti nel **kernel**, si interfaccia con **Application Program Interface (API)** e libreria del **middleware**, che a sua volta esegue le **chiamate di sistema**.

Le API sono **standard**, mentre le chiamate di sistema cambiano in base al sistema operativo. Per questo motivo sono anche **stabili** e offrono funzionalità più ad **alto livello** e più semplici da usare.

## Duplica modalità di Funzionamento

La CPU può funzionare in due diverse modalità.

### Modalità Utente (*User Mode*)

La modalità utente è limitata: non può eseguire alcune chiamate di sistema e non può accedere ad alcune regioni di memoria.

### Modalità di Sistema (*Kernel Mode*)

La modalità kernel è privilegiata, può accedere alle zone di memoria esclusive del kernel ed effettuare tutte le chiamate di sistema disponibili.

### System Call Interface

Per passare da una modalità all'altra, è necessario impostare il numero di **syscall** tramite i **registri**, ed effettuare la chiamata alla **subroutine** del kernel chiama **System Call Interface**.