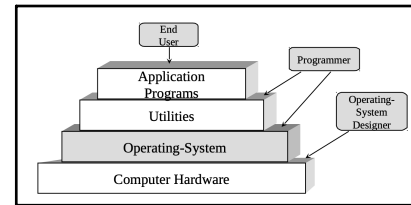


I SISTEMI OPERATIVI SONO COMPONENTI SOFTWARE. QUESTO SOFTWARE HA AL SUO INTERNO UNA CONOSCENZA MOLTO AMPIA ED ELEVATA di quale sia l'HARDWARE SU CUI IL SISTEMA STESSO DEVE ANDARE IN ESERCIZIO.

IL SISTEMA OPERATIVO È AL DI SOTTO di un programma applicativo, questo implica dire che quando noi scriviamo il nostro programma applicativo possiamo sfruttare che esiste già del software che qualcuno ha programmato, che è il sistema operativo.



Quando svilupperemo le applicazioni, saremo in grado di renderci conto che esse vivono in un ecosistema complesso, in cui ci sono tanti altri oggetti software utilizzabili che possono giocare un ruolo importante.

L'utility è una **funzione di libreria** che è al di sotto dell'application programs perché la funzione di libreria è qualche cosa che quando noi sviluppiamo una specifica tecnologia, possiamo utilizzare.

Cio che noi produciamo può sfruttare qualcosa che già esiste.

IL SOFTWARE del sistema operativo è un qualcosa che viene utilizzato dal software applicativo, ma il software applicativo oltre a dover utilizzare il software del sistema operativo, può utilizzare le utilities e va ad usare l'hardware. Quando un programma viene reso compilato e mandato in esercizio noi utilizziamo la CPU.

Mentre un'applicazione che noi stessi abbiamo scritto è in esercizio è possibile che questa applicazione vada a passare il controllo a qualche software che è presente nel sistema operativo.

Quindi c'è una chiamata affinché qualche cosa nell'evoluzione dell'applicazione in realtà venga ad essere implementato e supportato all'interno di software esterno all'applicazione, che è il software del sistema operativo.

IL SOFTWARE del sistema operativo in realtà implementa una serie di servizi (**TASK**) che possono essere triggerati/mandati in esercizio perché l'applicazione ha deciso di farlo.

MA L'APPLICAZIONE può anche non chiamare direttamente i moduli del sistema operativo, ma può chiamare anche qualcosa di intermedio che è una **utilities**.

E LA UTILITIES eventualmente chiamerà il modulo del sistema operativo.

Quindi noi possiamo sviluppare applicazioni che sono basate primariamente sulle utilities o primariamente sul sistema operativo.

QUELLO CHE FAREMO SARÀ CONOSCERE SIA LE UTILITY, SIA I SISTEMI OPERATIVI STESSI.

Le utilities possono essere identificate come moduli di libreria, ossia moduli che non stiamo scrivendo noi, e quando questo modulo prende il controllo potrebbe dover interagire con un altro modulo del sistema operativo.

La presenza dei sistemi operativi ci fornisce dei vantaggi:

- **SEMPLICITÀ**: dentro al sistema operativo, ho già tutta una serie di moduli che mi permettono di andare a pilotare le componenti hardware.

Rende lo sviluppo del software più semplice, mascherando le peculiarità delle piattaforme hardware

Tipicamente un'interfaccia di rete: quando utilizziamo un'applicazione per colloquiare con il mondo esterno, quello che noi facciamo è utilizzare all'interno della nostra applicazione (INTERNET)

Le istruzioni macchina che ci permettono di andare a pilotare un dispositivo di rete, noi chiameremo un software che fa già questo lavoro.

- **EFFICIENZA**: IL SOFTWARE del sistema operativo è stato pre-sviluppato da un **operating system designer**, che conosce molto bene le caratteristiche della macchina sottostante, e quindi codificare all'interno di questo software delle attività che permettano l'interazione con l'hardware e permettano all'hardware di eseguire effettivamente i task che vogliamo, quindi abbiamo un aspetto di efficienza;

- **FLESSIBILITÀ**: se per caso l'hardware sottostante viene ad essere modificato, modifichiamo dei devices con cui la nostra applicazione deve svolgere delle attività, ma per pilotare questi devices abbiamo moduli software all'interno del sistema operativo, bene, la presenza di questo sistema operativo rende il tutto flessibile perché non dobbiamo cambiare l'applicazione per lavorare

con la nuova costruzione dell'hardware, BASTA ^{evocare} il software che ci pensa lui ad interagire in modo ottimizzato, con la nuova istanza dell'hardware.

La sua presenza (del sistema operativo con i suoi moduli software) è una facilitazione per l'utilizzo di un sistema hardware e per lo sviluppo di applicazioni.