

## Sistemi operativi

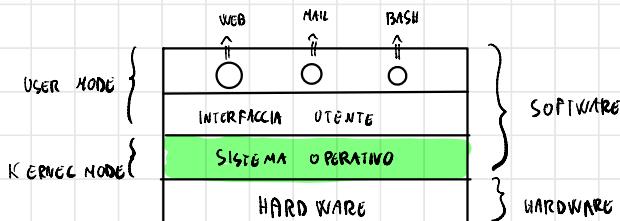
Si ha la necessità di gestire la comunicazione efficace tra processore, memoria centrale, disco, I/O ecc.

Questa gestione avviene tramite un particolare software:  
il sistema operativo.

Il sistema operativo è un software (middleware) che si interpone tra i processi che l'utente manda in esecuzione e l'hardware. Esso fa da tramite attraverso la ricezione e la gestione delle interrupt e delle system call ( librerie / API fornite dal SO al process per poter interagire con l'hardware)

Un programma che lavora in background (demon) non pesa in termini di occupazione di risorse (non consuma unità di elaborazione)

Il SO consente all'utente di interfacciarsi con l'hardware.



Le operazioni dell'utente hanno accessi ristretti e vengono eseguiti in **User Mode**, mentre quelle che devono essere gestite dal SO vengono in modalità **KERNEL MODE** e non ha restrizioni. Il SO viene invocato per prendersi carico di un processo, attraverso le **system call**. Le **utility** sono dei programmi eseguiti in User mode ma che danno accesso a zone che contengono informazioni sensibili es. cambiare la password.

Lo stesso programma, eseguito in momenti diversi, ha diverse tipi di esecuzione in base al contesto (risorse disponibili).

La gestione delle risorse **condivise** è molto delicata, in quanto differenti processi potrebbero necessitare di avere accesso alla medesima risorsa. Si hanno due modi per gestire il problema:

- Mantenere il programma attivo fino a che non si libera la risorsa richiesta
- Mandare in background il processo che attende in modo che non occupi molto di elaborazione

## Quale è la differenza tra Interrupt e System Call ??

Interrupt Request (IRQ) e System Call sono due meccanismi utilizzati dai sistemi operativi per gestire le interruzioni e la gestione dei dispositivi di input/output.

Le interruzioni sono eventi esterni al processore, come ad esempio l'arrivo di un dato da un dispositivo di input/output o un errore hardware. Quando si verifica un'interruzione, il processore interrompe l'esecuzione del programma corrente per gestire l'interruzione stessa e poi riprendere l'esecuzione del programma da dove era stata interrotta.

**Le IRQ sono eventi hardware generati da dispositivi esterni al processore, come ad esempio una richiesta di trasferimento dati da parte di un dispositivo di input/output.**

Quando un dispositivo genera un IRQ, il processore interrompe l'esecuzione del programma corrente e passa alla gestione dell'interruzione. La gestione delle IRQ è affidata al sistema operativo che, in base alla priorità delle richieste, assegna le risorse necessarie per gestire l'interruzione in modo efficiente.

D'altra parte, **le System Call sono un meccanismo utilizzato dai processi per richiedere al sistema operativo di eseguire un'operazione a loro nome.** Le System Call sono un'interfaccia di programmazione (API) tra il processo e il sistema operativo, che permette al processo di richiedere servizi di basso livello, come ad esempio la lettura/scrittura su un dispositivo di input/output o la gestione dei file. Quando un processo richiede una System Call, il sistema operativo interrompe l'esecuzione del processo corrente e passa alla gestione della System Call richiesta. La gestione delle System Call è affidata al sistema operativo che esegue l'operazione richiesta e poi restituisce il controllo al processo che aveva richiesto la System Call.

In sintesi, la differenza tra IRQ e System Call è che le IRQ sono eventi hardware generati dai dispositivi esterni al processore, mentre le System Call sono richieste software effettuate dai processi al sistema operativo. Entrambi i meccanismi sono utilizzati dai sistemi operativi per gestire le interruzioni e la gestione dei dispositivi di input/output.

