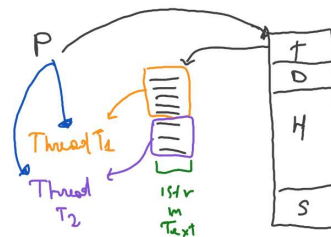


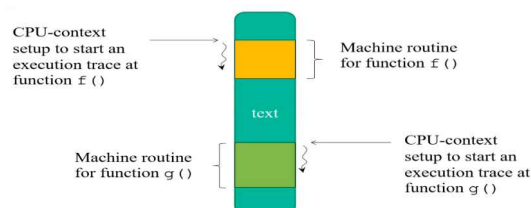
Processi e thread 6 Thread

giovedì 16 ottobre 2025 15:03

Facciamo un riepilogo : per quanto riguarda i processi , si ha che ogni processo ha un proprio spazio di indirizzamento (address space), contesto di CPU , traccia (scheduling CPU-> dispatching) la quale se in sistema time-sharing venga interrotta . Vediamo ora cosa succede nei sistemi operativi moderni (da anni 90) : **si ha che spazio ed indirizzamento e traccia possono essere disaccoppiate**. Quindi abbiamo due entità. L'unità base per il dispatching (traccia istruzioni) la associamo ad **un thread** : ovvero una traccia di esecuzione all'interno dell'address space dell'applicazione attiva. L'unità base delle risorse e dell'address space rimane il processo -> se processo attivo si ha applicazione attiva non parlando della traccia : entità diversa dal processo e questa entità viene denominata thread. Nei sistemi attuali è possibile gestirne una quantità molteplice. Vediamo il dettaglio cosa succede secondo modalità interleaving :

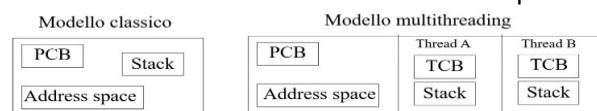


Quindi vediamo in dettaglio :

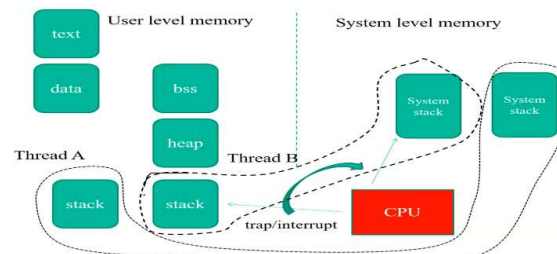


Quindi con questo meccanismo si ha la parallelizzazione delle attività . Vediamo ora le caratteristiche di un ambiente multi threading :

Un processo è caratterizzato da uno spazio di indirizzamento virtuale (immagine del processo) e protezioni e permessi di accesso ; quindi per ogni percorso (thread) si deve conoscere lo stato corrente (running, ready ecc ecc) , una stack area , stack di esecuzione in modo user e stack area modo kernel ed infine un contesto di esecuzione che dovrà essere ripristinato :



In dettaglio :



Quindi ora vediamo delle facility : **thread local storage (tls)**-> variabili per thread . **Quindi è possibile dichiarare variabili globali che afferiscono (a tempo di esecuzione) ad ogni istanza per ogni thread creato**. In **WINDOWS** si dichiarano **_declspec (thread)**, mentre in **UNIX** **__thread**. Nota : qualunque thread accede allo stesso address space, basta usare i puntatori in modo corretto ed avere delle facility di sincronizzazione.