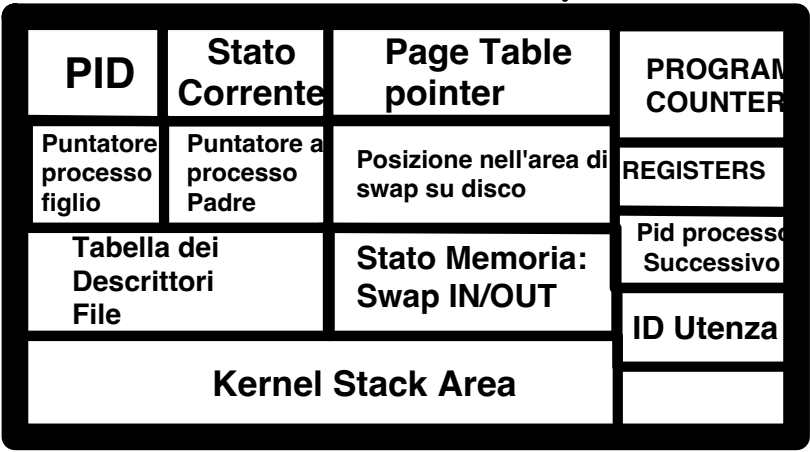


PCB



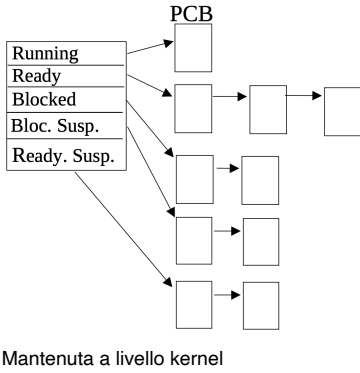
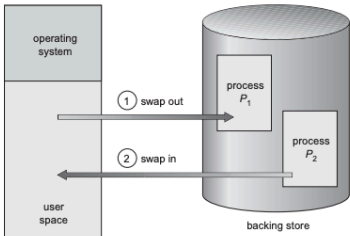
Si trova nel Kernel del Sistema Operativo.

È grazie al PID del processo successivo che si identifica la "coda" dei processi ready, ad esempio.

I processi in RAM sono organizzati in una o più code. Ad ogni coda è associato un descrittore di coda che contiene due campi: indice del primo e dell'ultimo PCB. Ogni PCB ha un campo per identificare il PCB successivo nella coda. La comunicazione tra processo e nucleo avviene tramite una system call.

DISCO

- Un processo in esecuzione può essere temporaneamente rimosso dalla memoria e riversato (**swapped**) in una memoria secondaria (detta **backing store** o **swap area**); in seguito può essere riportato in memoria per continuare l'esecuzione.
- Lo spazio indirizzi di interi processi viene spostato.
- Backing store**: dischi veloci e abbastanza larghi da tenere copia delle immagini delle memorie dei processi che si intende swappare.



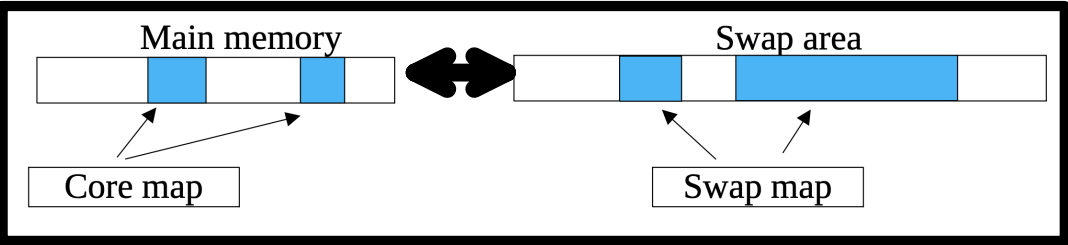
Mantenuta a livello kernel

Ciascuna entry ha il proprio descrittore di coda

RAM

KERNEL O.S
PID = 0
PID = 1

Address Space
processo A



Il descrittore di processo (PCB)

Il PCB viene assegnato ad un processo quando viene posto nello stato new e viene deassegnato quando esce dallo stato terminated e ne viene persa traccia

Tipicamente i PCB sono disposti in code costituite da liste a puntatori. Appartengono ad una stessa coda PCB di processi caratterizzati dallo stesso stato globale (ad esempio coda dei processi ready ad uguale priorità di schedulazione, coda dei processi waiting in attesa di avviare una operazione di accesso ad un disco, coda dei processi waiting in attesa di sincronizzarsi con uno stesso processo di servizio, ecc.)

La swam map ci indica le zone libere e quelle occupate. È chiaro che, essendo la scrittura su disco molto più lenta di quella della RAM, le prestazioni del sistema crollano bruscamente.

>>swapon --show (visualizzare swap on linux)

La swap dovrebbe essere grande almeno quanto la RAM, perché quando mandiamo il computer in ibernazione, il contenuto della RAM viene scritto nella memoria di swap prima di spegnere il computer.

Essa è un'area del disco fisso che interviene in aiuto alla RAM quando questa sta per esaurirsi.

Il S.O esegue lo swapping in maniera proattiva, cioè ben prima dell'esaurimento della RAM, in modo da non restare mai a corto di memoria.

Il processo/applicazione è un'entità caricata in RAM. I pc moderni contengono più processori che condividono una memoria comune, processori distinti eseguono allo stesso tempo istruzioni appartenenti a programmi diversi.

I sistemi operativi time sharing garantiscono un'esecuzione INTERLEAVED delle TRACCE dei singoli processi.

*interleaved=in un intervallo di tempo prefissato una sola CPU può eseguire sequenze di istruzioni appartenenti a programmi diversi.

Durante l'esecuzione di un programma su un determinato processore, l'esecuzione può essere ripresa e sospesa più volte. *

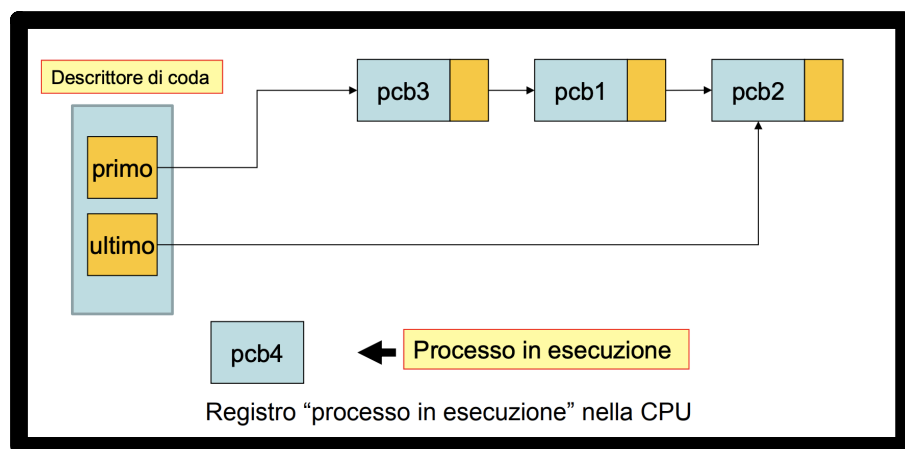
Le tracce sono una sequenza di istruzioni macchina che vengono mandate in fetch.

Il processo è una sequenza di task controllata da uno scheduler, che è un componente di un sistema operativo che implementa una politica di scheduling, che ha come scopo quello di massimizzare il THROUGHPUT, ossia il numero di programmi eseguiti per unità di tempo.

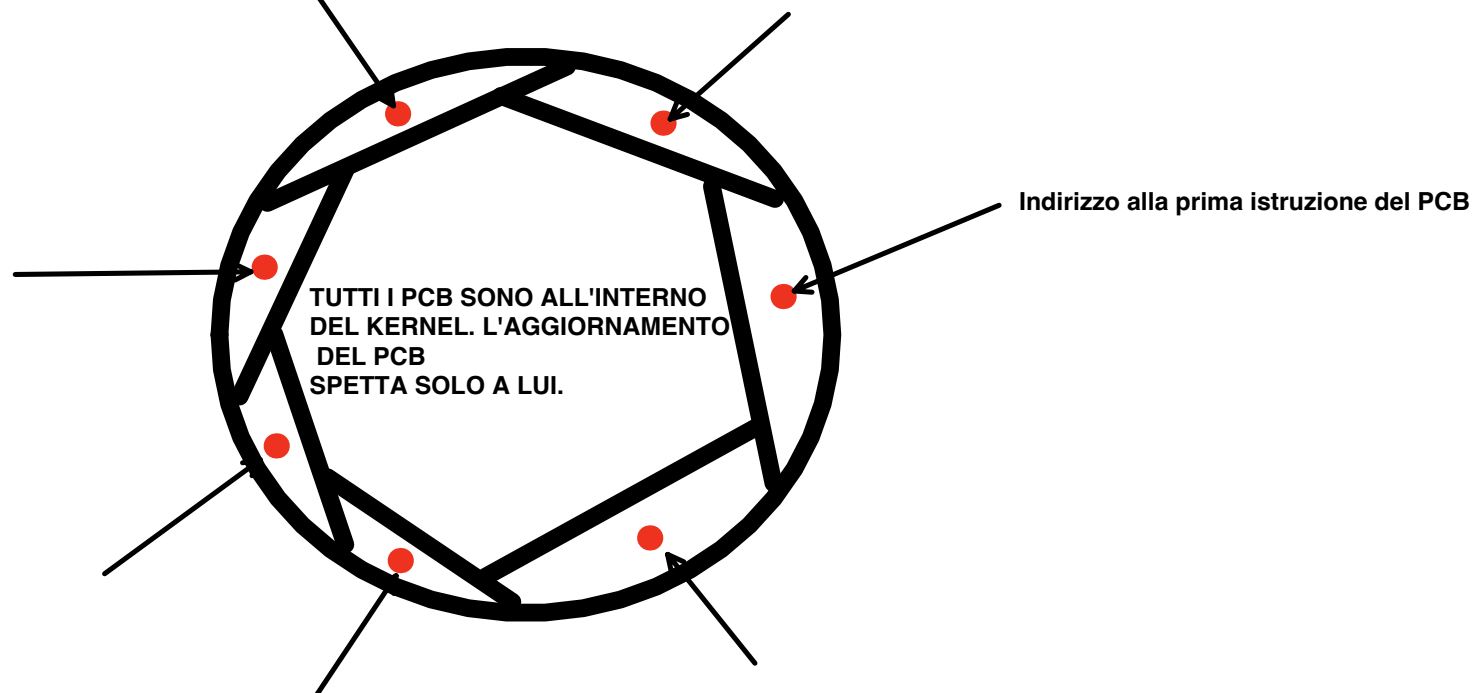
È il sistema operativo che alloca ed inizializza le strutture dati per la gestione dell'esecuzione del processo. Lo stato exit consente il rilascio delle risorse da parte del kernel.

Quando accendiamo un computer, in una fase che viene chiamata BOOTSTRAPPING il kernel viene caricato in RAM, e quindi diventa anche lui un processo in esercizio sulla CPU, tipicamente ha il PID = 0. Il nucleo inizializza le proprie strutture dati e crea alcuni processi di servizio, ossia i processi di tipo login shell che sono in grado di colloquiare con un utente e nel momento in cui quest'ultimo digita la password e l'utenza, il processo PID =1, va a controllare i file /etc/shadow e /etc/passwd.

Le CPU possiedono un registro, ossia il registro del processo in esecuzione, nel quale il S.O memorizza il puntatore al PCB del processo che è in esecuzione.



IL KERNEL:



Quando alloco la struttura dati del processo che va in new, alloco il PCB.

