

# Le risorse

## **Cos'è una risorsa?**

E' qualcosa di necessario al processo per poter procedere nella sua evoluzione (passaggi di stato).

Si possono distinguere:

- Risorse **fisiche** (processore, memoria, disco, stampante,...)
- Risorse **astratte** (un file, una struttura dati,.....)

Le risorse possono essere assegnate al processo

- Per tutta la durata del processo, **in modo statico**, e vengono recuperate dal S.O. solo quando il processo termina.
- Durante l'evoluzione del processo, **in modo dinamico**. In questo caso il processo chiede la risorsa una o più volte quando gli necessita e la rilascia dopo averla utilizzata.

Se il processo *può essere forzato* a rilasciare una risorsa, la risorsa è chiamata **prerilasciabile o preemptive**.

## **Il processore è una risorsa gestita in modo dinamico**

Le risorse possono anche esistere in **più copie** (classi di risorse omogenee), come i blocchi di memoria, insiemi di settori sul disco, diverse stampanti con caratteristiche comuni ecc.... Il S.O. deve decidere **quale** risorsa di una stessa classe assegnare al processo.

**Ogni risorsa** o classe di risorse viene **descritta da tabelle** con le informazioni sullo stato: se è *libera* o *occupata*, a quale processo è assegnata, ecc..... Queste tabelle sono tenute aggiornate dal S.O.

## **La gestione delle risorse è uno dei compiti più difficili del S.O.**

Per questo motivo il S.O. è costituito da diverse parti (strati) ognuno dei quali si occupa alcune tipologie di risorse:

- La risorsa CPU è gestita dal **gestore del processore** (parte del nucleo).
- La risorsa memoria RAM è gestita dal **gestore della memoria**.
- Le periferiche sono gestite dal **gestore delle periferiche** (tastiera, mouse, hard disk, monitor, stampante)
- I file e le directory sono gestiti dal **gestore delle informazioni o File System**.

**Il Gestore del processore:** Decide a quale processo assegnare la CPU.  
Recupera la CPU quando si rende disponibile.  
Gestisce le comunicazioni fra i processi.

**Il Gestore della memoria :** Aggiorna lo stato della aree libere e occupate.  
Sceglie quali aree allocare (assegnare) a ogni processo.  
Le alloca e le recupera quando il processo non ne ha più bisogno.

**Il Gestore delle periferiche :** Assegna la periferica e inizia i processi di I/O.  
Recupera la periferica.

**Il Gestore delle informazioni:** Gestisce le FAT (File Allocation Table).  
Gestisce i privilegi.  
Assegna spazio di disco ai processi (p.es. per creare un file).  
Recupera spazio (p.es. per cancellare un file).  
Crea e cancella file o directories.

## Politiche di assegnazione delle risorse

In genere una risorsa può essere assegnata ad un solo processo per volta, in **mutua esclusione**.

Se più processi chiedono contemporaneamente la stessa risorsa il S.O. deve gestire le richieste con una opportuna politica di assegnazione, stabilendo un ordine tra i processi. Se la risorsa chiesta da un processo non è disponibile il processo passa in *stato di attesa* fino a quando può ottenere la risorsa (richiesta bloccante); per alcune categorie di risorse la richiesta può essere gestita in modo non bloccante, cioè se il processo non può ottenere la risorsa non viene sospeso ma viene solo informato dell'esito negativo.

I processi sono in **competizione** per l'uso delle risorse e quindi **interferiscono** tra di loro.

Il S.O. deve assegnare le risorse ai processi in modo che vengano usate in mutua esclusione; per realizzare questo usa i meccanismi del modello concorrente (per esempio semafori per l'accesso alle risorse).

**La politica di assegnazione** delle risorse stabilisce un **ordine** tra i processi, in modo da garantire che la risorsa sia assegnata in **mutua esclusione** e che **tutti i processi** che la richiedono riescano ad **ottenerla in un tempo finito**.

Il **tempo di attesa** di ogni processo per ottenere la risorsa dipende dalle richieste degli altri processi e dal tipo di politica di gestione adottata; se non si può stabilire un limite al tempo di attesa si può verificare il caso di **attesa indefinita** (o **starvation**) in cui il processo potrebbe non riuscire mai ad ottenere la risorsa.

Tipiche **politiche** di assegnazione sono:

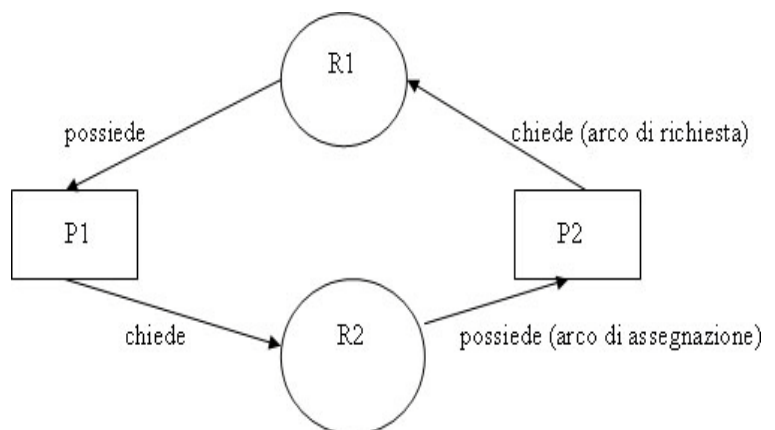
- **in ordine di arrivo** (FCFS - First Come First Served) che soddisfa le richieste nell'ordine in cui sono arrivate; viene realizzata con una coda FIFO (First In First Out) e assicura un valore prevedibile al tempo di attesa poiché l'attesa di ogni processo dipende dal numero delle richieste che la precedono nella coda;
- **a priorità** che soddisfa le richieste in base a un valore di priorità associato a ogni processo; per evitare il problema dell'attesa indefinita, la priorità viene variata in modo dinamico, aumentando la priorità delle richieste che sono in attesa da molto tempo.

## Lo stallo

La presenza di più programmi che competono per l'uso delle risorse può causare situazioni di **stallo o blocco critico (deadlock)**, cioè situazioni in cui non è possibile far avanzare nessun processo.

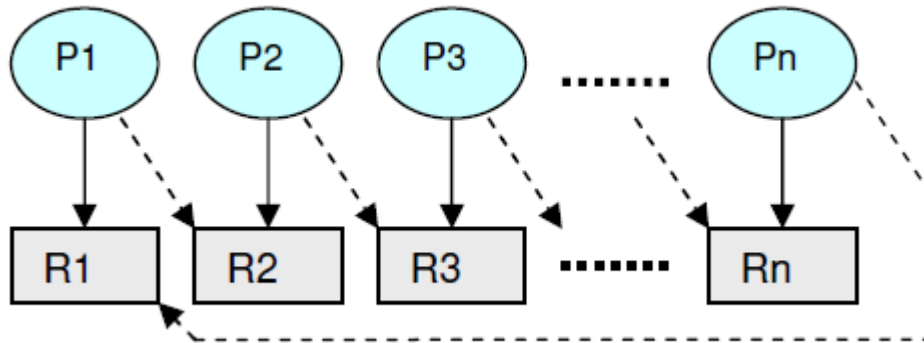
Una situazione di **stallo** si genera quando un processo o più processi rimangono **indefinitamente bloccati** a causa del non verificarsi delle condizioni necessarie per il loro proseguimento.

Si consideri il seguente problema. Due processi **P1** e **P2** utilizzano due risorse R1 e R2; ciascun processo necessita di entrambe le risorse per arrivare a termine. Le due risorse sono acquisite una alla volta e utilizzate in modo esclusivo. Si supponga che ad un certo istante il processo **P1** possieda la risorsa R1 e il processo **P2** la risorsa R2. Se **P1** mantenendo il controllo di R1 chiede R2, esso viene sospeso in attesa della liberazione della risorsa da parte di **P2**; tuttavia se **P2**, mantenendo il controllo di R2, chiede la risorsa R1, entrambi i processi vengono bloccati indefinitamente (**attesa circolare**).



Più in generale dato un insieme di processi  $P_1, P_2, \dots, P_n$  che usano un insieme di risorse  $R_1, R_2, \dots, R_m$  si può verificare una situazione di deadlock se sono vere contemporaneamente le seguenti condizioni:

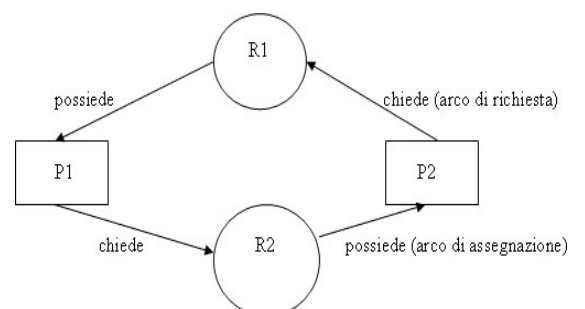
- a) Ciascuna risorsa può essere usata da un solo processo alla volta (**mutua esclusione**)
- b) I processi trattengono le risorse che già possiedono mentre chiedono risorse aggiuntive
- c) Le risorse già assegnate non possono essere sottratte (**senza prerilascio**)
- d) Esiste una situazione di attesa che si chiude in se stessa (**attesa circolare**); ossia alcuni processi formano una catena chiusa tale che ciascun processo della catena controlla almeno una delle risorse richieste dal successivo.



Il S.O. può gestire le situazioni di stallo in due modi:

- **prevenendo** il verificarsi delle condizioni che portano allo stallo
  - Una **prima tecnica di prevenzione** consiste nell'imporre a ciascun processo di richiedere in blocco tutte le risorse di cui necessita; se le risorse non sono tutte disponibili al momento della richiesta il processo non ne ottiene alcuna e viene posto in uno stato di attesa fino a quando tutte le risorse non sono disponibili. Dopo averle usate il processo può rilasciare le risorse in un ordine qualsiasi. Si noti che la condizione b) non è più soddisfatta.
  - Una **seconda tecnica di prevenzione** prevede che se una richiesta per una risorsa viene rifiutata, il processo debba rilasciare tutte le risorse che aveva già acquisito e sospeso; non è più soddisfatta la condizione c)
  - Una **terza tecnica di prevenzione** prevede che a ciascuna risorsa sia assegnato un numero d'ordine e che i processi possano richiedere le risorse una per volta per numero d'ordine crescente. In questo caso non si verifica più una catena circolare, condizione d).
  - Una tecnica più complessa è quella nota come **algoritmo del banchiere**. Per ogni processo deve essere noto il numero massimo di unità di ogni risorsa di cui il processo avrà bisogno. Il S.O. assegna a un processo le risorse richieste solo se ne restano abbastanza disponibili per soddisfare la richiesta massima di almeno uno dei processi attivi; in questo modo si assicura che almeno un processo possa terminare e rilasciare le risorse già occupate.
- **riconoscendo** che si è verificato uno stallo e **risolvendo** il problema.

Per il **riconoscimento** dello stallo il S.O. può utilizzare un sistema basato su un **grafo di allocazione** delle risorse che rappresenta con dei cerchi i processi e con dei rettangoli le risorse. Un arco orientato collega ogni processo alle risorse che sta occupando e a quelle che sta richiedendo. Se si formano dei **cicli** significa che i processi si trovano in *una situazione di attesa circolare* e quindi di **stallo**.



Un altro metodo per il **riconoscimento** dello stallo si

basa sulla **statistica**: il S.O. effettua delle stime dei tempi massimi accettabili di possesso o di attesa di una risorsa e se un processo supera i tempi massimi stimati viene considerato in stallo (non è detto però che lo sia).

Dopo aver riconosciuto uno stallo il sistema deve provvedere a **risolvere** il problema:

- Il modo più drastico consiste nell'interrompere tutti i processi in stallo
- Un metodo migliore consiste nell'interrompere un processo alla volta, finché non si risolve la situazione. Per la scelta del processo da interrompere si può individuare quello con priorità più bassa oppure quello a cui manca più tempo per terminare.