Glossario

- ***Queuing Delay*** (T_w) - Quanto tempo un processo in coda attende di essere processato dal server;
- ***Service Time*** (S_i) - Quanto tempo ci mette il server i a processare un job. È l'inverso della sua velocità:

$$S_i = \frac{1}{\mu_i}$$

  - Nota che questo non include il queuing delay.
- ***Response Time*** (R, a volte T) - La somma di Queuing Delay e Service Time, ovvero il tempo medio che intercorre tra l'ingresso in coda di un job ed il suo completamento;
  - Il ***Think Time*** (Z) è un response time degli utenti.
- ***Completions*** (C) - ***Dato un tempo di osservazione***, il sistema complessivo ha completato C jobs;
  - Posso definirlo per il singolo server all'interno del sistema complessivo. ***Dato un tempo di osservazione***, C_i sono le completions del server i;
  - Se divido per il tempo di osservazione diventa il ***throughput*** X.
- ***Visite*** (V_i) - Se esiste un percorso che collega l'output del server i al suo stesso input, un singolo job potrebbe essere processato più di una volta. Il numero (atteso) di volte che ciò accade è detto numero di visite;
  - Nota che potrebbe essere minore di 1 (e.g. se ho un singolo job diviso al 50% tra due CPU in parallelo e nessun loop, il numero atteso di visite per ognuna sarà $\frac{1}{2}$) o maggiore (e.g. se osservo C completion del sistema e $C_i > C$ completion del server i, significa che ogni job completato dal sistema è stato completato più volte dal server i, ovvero $V_i = \frac{C_i}{C}$);
    - Nota anche che nel 90% dei casi mi riferirò alle visite come dovute ai loop, e quindi $V_i > 1$.
  - Dalla definizione di throughput e di visite discende in modo molto naturale la ***forced flow law***

$$X_i = \langle V_i \rangle X$$

    Visto che il throughput è praticamente la derivata in dt delle completions, questa è una reskin della definizione

$$V_i = \frac{C_i}{C} \Rightarrow C_i = V_i C$$

    La forced flow sta sostanzialmente dicendo che se per completare un job servono più passaggi (visite) dal server i, allora il suo throughput sarà maggiore di quello complessivo del sistema. Di quanto? Del numero di visite.
- ***Demand*** (D_i) - Quanto tempo un singolo job impegna il server i durante una sua singola esecuzione nel sistema complessivo;

$$\langle D_i \rangle = \langle V_i \rangle \langle S_i \rangle$$

- ***Busy Time*** (B_i) - ***Dato un tempo di osservazione***, mentre il sistema complessivo completa C jobs il server i sarà impegnato per un tempo pari alla demand per il singolo job per il numero di completions

$$B_i = D_i C$$

  Analogamente, posso vedere questa grandezza come il service time per il numero di job completati dal server i

$$B_i = S_i C_i$$

  Queste due definizioni sono equivalenti:

$$D_i = V_i S_i \wedge C = \frac{C_i}{V_i} \Rightarrow D_i C = S_i C_i$$

- ***Utilization*** (ρ_i) - Percentuale di tempo in cui, mediamente, il server i è occupato. È una probabilità che discende da una statistica su un lungo periodo di tempo (i.e. se osservo per 10 secondi il server è occupato per 2, se osservo per 100 misuro 22, per 1000 misuro 221... al limite la legge dei grandi numeri ci garantisce che raggiungeremo il "valor vero"):

$$\rho_i = \lim_{\tau \to \infty} \frac{B_i(\tau)}{\tau}$$

  - Dalla prima definizione di B_i trovo la ***bottleneck law***

$$\rho_i = \lim_{\tau \to \infty} \frac{C(\tau)}{\tau} D_i = D_i X$$

  - Dalla seconda trovo la ***utilization law***

$$\rho_i = \lim_{\tau \to \infty} \frac{C_i(\tau)}{\tau} S_i = S_i X_i$$