Dependable Distributed Systems
Master of Science in Engineering in Computer Science

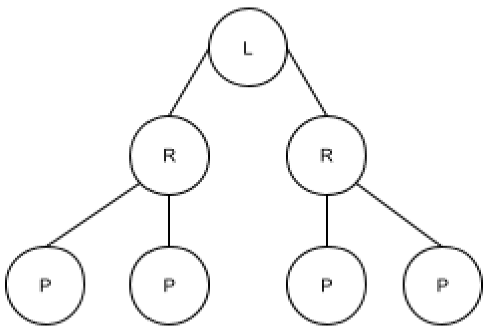
AA 2023/2024

Week 10 – Exercises
November 29th, 2023

Ex1: Answer true or false to the following claims providing a motivation:

1. The performance metrics of a system can be analyzed regardless of its load.
2. Consider a component receiving a single kind of homogeneous requests (i.e., a single workload component); if we are under the stability condition ($\lambda < \mu$), then the expected response time of the system is independent of the arrival pattern.
3. The workload parameters that mostly influence the performance of system are the arrival pattern and the service demands.
4. The M/M/k model assumes that arrivals are deterministically distributed over time.
5. The steady-state availability of a distributed system always increases as the number of processes grows.

Ex2: Consider the distributed systems organized as shown in the figure. Process *P* generates requests *req* (i.e., messages) that are handled by process *L*, processes *R* relay messages between processes *P* and *L*, process *L* computes responses *rep* to the requests *req* generated by processes *P*. In more detail, each process *P* generates requests *req* and sends them to the linked process *R*, which in turn relays them to *L*. Process *L* computes a response *res* when a request *req* is received and then relays it to *R,* which in turn relays it to *P.*



Assuming that each process *P* generates 10 requests per minute, that process *L* takes 0.5 seconds on average to compute a response *rep*, that the computation delay on processes *R* and *P* is negligible and equal to 0, that each process can send/receive 60 messages per minute, and that arrivals and service times are exponentially distributed, evaluate the expected time that elapses between a request generated by a process *P* and the receipt of the response on the same process.
If the steady-state availability of processes *R* and *L* is 0.9, processes *P* are always available (i.e., their availability is equal to 1) evaluate the steady-state availability of the service perceived by a single process *P* (the service is available when a process P is able to communicate with process *L* and receive a response to its requests).

Ex3: Consider a service that is deployed on a single machine. The network interface of such a machine is capable of send/receive 10 requests per second, then the requests are managed by 4 independent threads capable of processing 5 requests per second. Assuming that the arrival and service times are exponentially distributed, estimate the expected response time perceived by a user of the service (i.e., the expected time elapsed between the moment a user sends a request and the time *he receives a response).

Ex3: Consider a service that is deployed on a single machine and a single workload component having an average inter-arrival time equal to 0.125 seconds. The network interface of such a machine is capable of send/receive 10 requests per second, then the requests are managed by 4 independent threads capable of processing 5 requests per second. Assuming that the inter-arrival and service times are exponentially distributed, and that the down-link and up-link capacities are independent, estimate the expected response time perceived by a user of the service (i.e., the expected time elapsed between the moment a user sends a request and the time *he receives a response).

Ex1: Answer true or false to the following claims providing a motivation:

1. The performance metrics of a system can be analyzed regardless of its load.
2. Consider a component receiving a single kind of homogeneous requests (i.e., a single workload component); if we are under the stability condition ($\lambda < \mu$), then the expected response time of the system is independent of the arrival pattern.
3. The workload parameters that mostly influence the performance of system are the arrival pattern and the service demands.
4. The M/M/k model assumes that arrivals are deterministically distributed over time.
5. The steady-state availability of a distributed system always increases as the number of processes grows.

1) F, the perfomace depends heavly on the characteristics of its load because it impact on the performance and is related to the inputs (# of users, requests,..)

2) F, ~~this means that the arrival time~~ $\frac{1}{\lambda}$ ~~is larger than the service time~~ $\frac{1}{\mu}$.

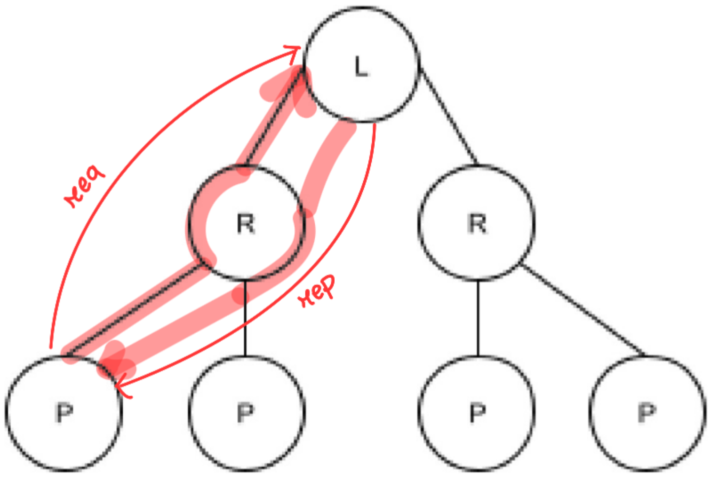
the arrival pattern have a crucial influence on the metrics

3) T, because this two parameters characterize the workload, giving info on who enters in the system (queuing) and on the demand of the system

4) F, are exponentially distributed

5) F, depends on the connection of processes in the system

Ex2: Consider the distributed systems organized as shown in the figure. Process *P* generates requests *req* (i.e., messages) that are handled by process *L*, processes *R* relay messages between processes *P* and *L*, process *L* computes responses *rep* to the requests *req* generated by processes *P*. In more detail, each process *P* generates requests *req* and sends them to the linked process *R*, which in turn relays them to *L*.
Process *L* computes a response *res* when a request *req* is received and then relays it to *R*, which in turn relays it to *P*.
Assuming that each process *P* generates 10 requests per minute, that process *L* takes 0.5 seconds on average to compute a response *rep*, that the computation delay on processes *R* and *P* is negligible and equal to 0, that each process can send/receive 60 messages per minute, and that arrivals and service times are exponentially distributed, evaluate the expected time that elapses between a request generated by a process *P* and the receipt of the response on the same process.
If the steady-state availability of processes *R* and *L* is 0.9, processes *P* are always available (i.e., their availability is equal to 1) evaluate the steady-state availability of the service perceived by a single process *P* (the service is available when a process P is able to communicate with process *L* and receive a response to its requests).

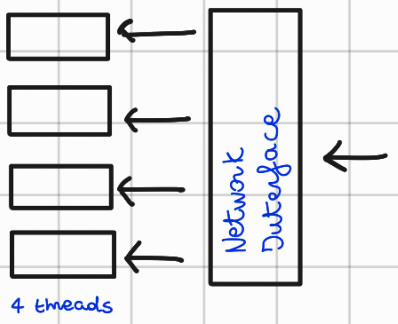


Commento prof. Farina:
Ex2: Partiamo dal comprendere il carico del del sistema, i processi P generano 10 richieste al minuto, questa vanno verso il processo R di riferimento, che le inoltra ad L; valutiamo il solo flusso "in entrata" (in uscita è speculare), i processi R ricevono richieste da due processi P e dal processo L (le risposte), il processo L riceve tutte le richieste generate dai processi P, i singoli P ricevono solo le risposte alle loro richieste; detto questo va calcolato il tempo che una richiesta da tutto il suo percorso nel sistema usando la teoria delle code.

$$R_{\xrightarrow{R}} = \frac{1}{60-(10+10)} = \frac{1}{40} = 0{,}025\ s$$

$$R_{\xrightarrow{L}} = \frac{1}{60-(10}$$

Ex3: Consider a service that is deployed on a single machine and a single workload component having an average inter-arrival time equal to 0.125 seconds. The network interface of such a machine is capable of send/receive 10 requests per second, then the requests are managed by 4 independent threads capable of processing 5 requests per second. Assuming that the inter-arrival and service times are exponentially distributed, and that the down-link and up-link capacities are independent, estimate the expected response time perceived by a user of the service (i.e., the expected time elapsed between the moment a user sends a request and the time *he receives a response).



Dati:

Send/receive = 10 req/s

Service rate = 5 req/s

arrival rate : $\lambda = \frac{1}{0.125\,s} = 8$ req/s

1° queque $\mu = 10$ req/s $\lambda = 8$ req/s

2° queque $\lambda = \frac{8}{4} = 2$ req/s $\mu = 5$ req/s

3° queque $\mu = 10$ req/s $\lambda = 8$ req/s

$$R_{1Q} = \frac{1}{k\mu - \lambda} = \frac{1}{10-8} = 0{,}5\,s$$

$$R_{2Q} = \frac{1}{5-2} = 0{,}33\,s$$

$$R_{3Q} = 0{,}5\,s$$

$$R_{TOT} = R_{1Q} + R_{2Q} + R_{3Q} = 1{,}33\,s$$