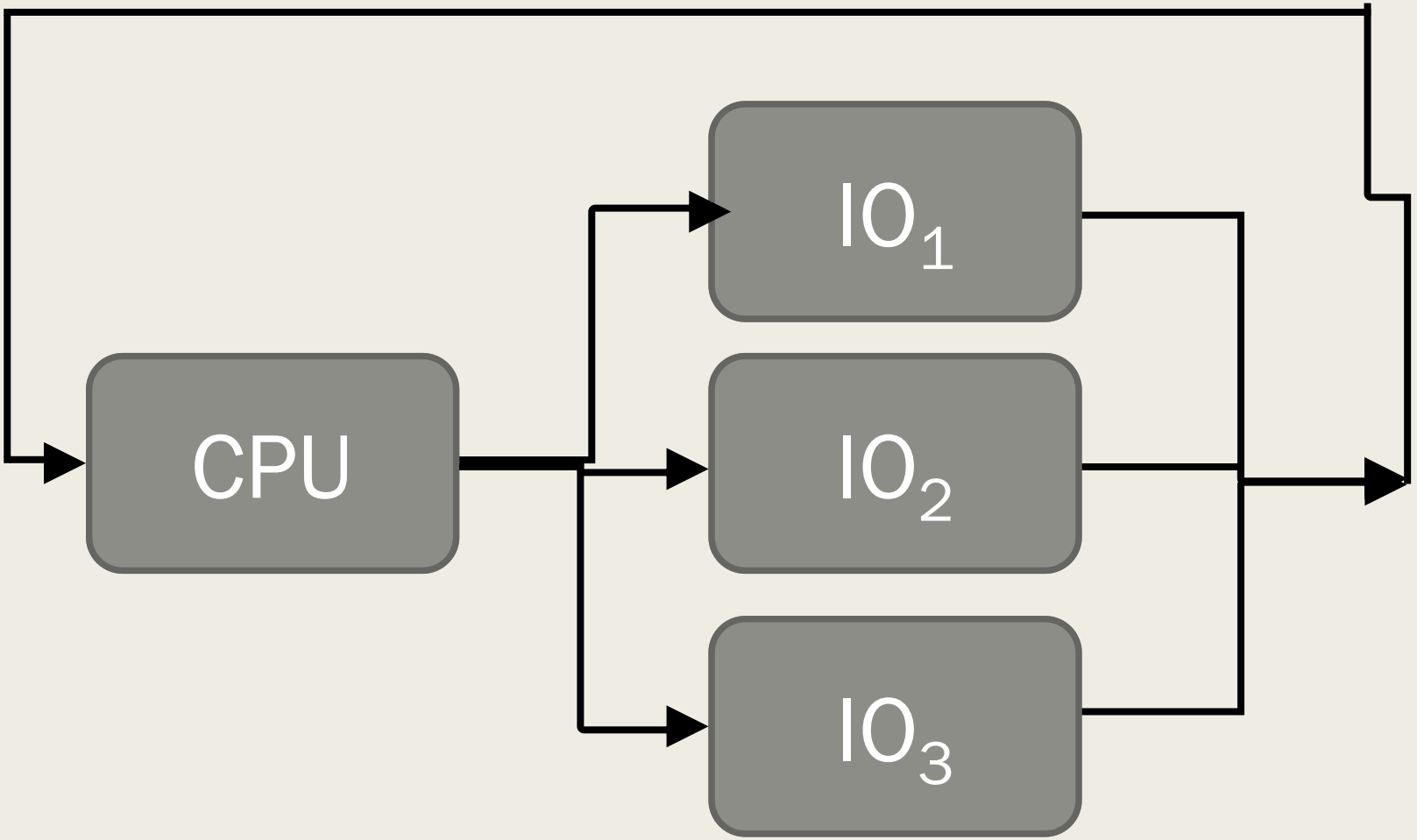
NETWORKS OF QUEUES

Software Performance Engineering

Our analysis thus far

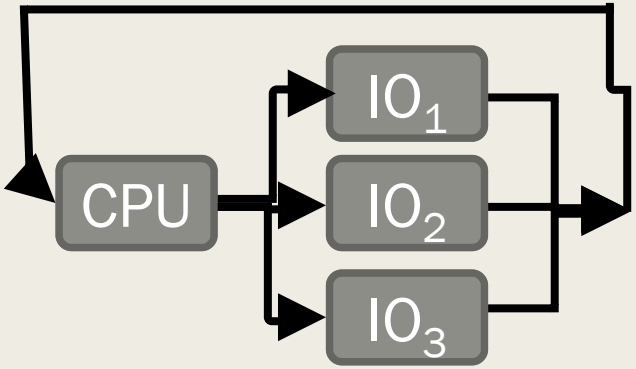
- We define metrics for each system to measure performance
- We use the exponential distribution
  - To analyze inter-arrival times in a Markovian stochastic system
  - For a single random variable, e.g. “customers arriving”
- We use multiple random variables to compute various metrics
  - Customer arrival
  - Service time
- …but…
- Systems have more than one queue!
  - M/M/m queues where m>1

Central Server Model



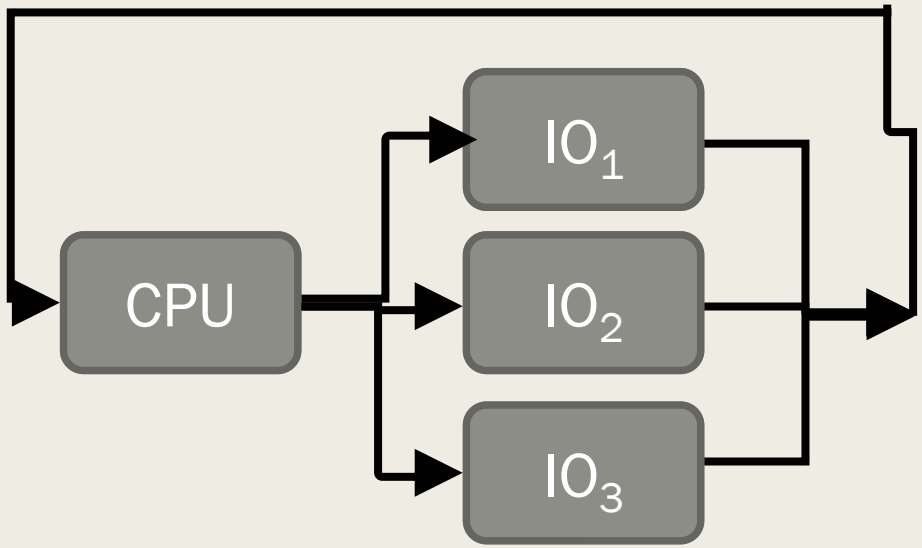
Visit Ratio

- How do we measure a network of queues? How does that translate to our existing mathematical formulas?
- Visit Ratio *V*
  - The relative number of visits to each devices for one job
  - Defined over each device, i.e. V_i
  - Include the processor in that count
  - Could be thought of as “visits per job”, BUT…
  - …typically expressed as a unitless weight,
- For example, our server model
  - $V_{CPU} = 1 + V_{IO1} + V_{IO2} + V_{IO3}$
  - Every visit to an IO is accompanied by a CPU visit



Device Ratios Will Differ

- Note that IO1, IO2, and IO3 need not be the same numbers!
- It all depends on:
  - The service time of the current job on the CPU
  - The service time of the IO device
  - Queuing discipline of each
  - Queuing discipline of CPU
- Thus, in practice, we try to measure utilization on each queue in the queuing network to get an accurate visit ratio



Forced Flow Law

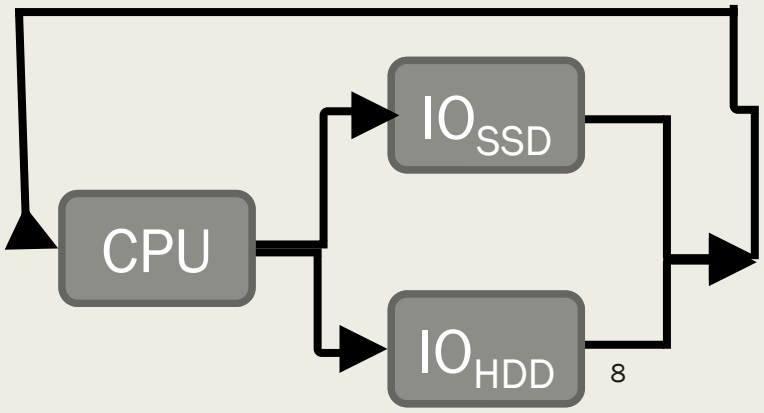
- Throughput of individual devices in the system increase in proportion to the global system throughput of X_{global}
  - $X_i = V_i X_{global}$ for device $i \leq m$
  - Throughput
  - Again, we usually express V_i as unitless so we equate jobs & visits
- Let's define Demand D_i on any device as
  - $D_i = V_i S_i$ for mean service times S_i
  - A weighted time for each job
  - e.g. A disk takes 2ms to write, and is visited 3 times per job, thus the Demand will be 6ms per job
- Translating to Utilization…
  - $X_i S_i = V_i S_i X_{global}$ (multiply by S_i on both sides)
  - $U_i = X_{global} D_i$ (apply Utilization law & sub in Demand)

Upper Bounds on X_{global}

- Saturation: Utilization of any server must always be <1
  - $U_i < 1$ thus
  - $X_{global} D_i < 1$
- And if that's true for all D_i, then it must be true for the highest D_i, call it D_{max}
- Therefore: $X_{global} < \frac{1}{D_{max}}$
  - This is called a *bottleneck*
  - The global throughput of the system will be bounded by the reciprocal of the demand of the bottleneck device

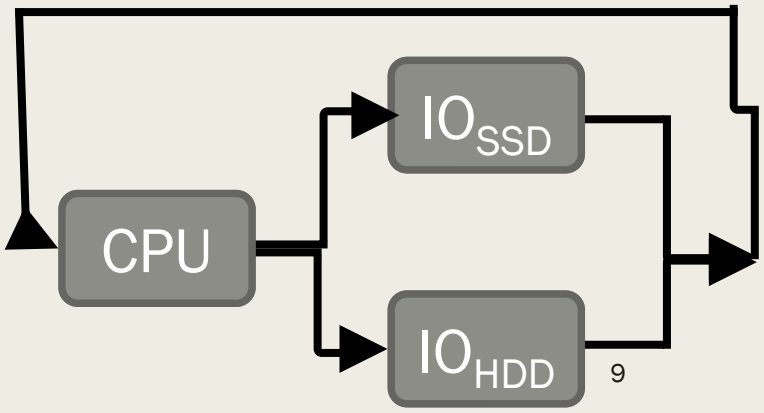
e.g. 2 slow IO devices, fast CPU

- Suppose we have a system with two hard drives and a CPU
- Mean service times:
  - S_{SSD}= 3ms, S_{HDD}= 5ms, S_{CPU} = 1ms
- Visit ratios:
  - V_{SSD}= 6, V_{HDD}= 3
  - So: $V_{CPU} = 1 + 6 + 3 = 10$
- Demands:
  - D_{SSD}=3ms*6=18ms
  - D_{HDD}=5ms*3=15ms
  - D_{CPU} =1ms*10=10ms
- Max system throughput:
  - $X_{global} < \frac{1}{18ms}$
  - $X_{global} < 0.056$ jobs/ms
  - Or $X_{global} < 56$ jobs/s



e.g. slow IO devices, slower CPU

- Suppose we have a system with two hard drives and a CPU
- Mean service times:
  - S_{SSD}=3ms, S_{HDD}=5ms, S_{CPU} =8ms
- Visit ratios: (same as before)
  - V_{SSD}=6, V_{HDD}=3
  - So: $V_{CPU} = 1 + 6 + 3 = 10$
- Demands:
  - D_{SSD}=3ms*6=18ms
  - D_{HDD}=5ms*3=15ms
  - D_{CPU} =8ms*10=80ms
- Max system throughput:
  - $X_{global} < \frac{1}{80ms}$
  - $X_{global} < 0.0125$ jobs/ms
  - Or $X_{global} < 12.5$ jobs/s



General Response Time Law

- Overall system response times:
  - $R_{global} = \sum_i^m V_i R_i$
- Lower bounds? Assume no other jobs in the system, then a job's lower bound is:
  - $R_{global} \geq \sum_i^m V_i S_i$
  - i.e. $R_{global} \geq \sum_i^m D_i$