

## Solution by simulation

A big company, has a SAN based solution for storing files. There are  $N = 200$  users, each one with a think time of  $Z = 2$  minutes (exponentially distributed), represented by an infinite server **terminal station**. Requests are routed to a **network controller** machine, that splits them into blocks. The average service time of the controller is 10 ms per block, and can be considered Erlang distributed with a coefficient of variation of 0.25. Requests are executed in FCFS order, and are routed to the **SAN access controller** with probability 0.9, while they complete and return to the terminal station with probability 0.1. The SAN controller works in FCFS and has a Normal distributed service time with average 12 ms, and standard deviation of 1 ms. It has a finite capacity queue of  $K = 10$  jobs, and performs BAS blocking to prevent overflow. The **SAN controller routes requests to three different disks (disk<sub>1</sub>, disk<sub>2</sub>, disk<sub>3</sub>)**, using the *Join The Shortest Queue* policy. Each disk serves requests in Processor Sharing, and when finished it routes them back to the network controller to proceed with the next block if needed. Disk service times are different, but all exponentially distributed, according to the following averages: **disk<sub>1</sub> -> 30 ms, disk<sub>2</sub> -> 40 ms, disk<sub>3</sub> -> 35 ms**. In this description bold-face has been used to emphasize the different nodes of the model.

The considered the system is clearly no separable: using JMT compute:

1. The throughput of the system ( $X$ )
2. The average system response time ( $R$ ), including also the time spent at the terminal station
3. The number of jobs (queue + service) at the two controllers: the *network controller* and the *SAN access controller*.

For the solutions, consider only the average value, and ignore the confidence intervals.