Load Analysis for Queued Subsystems

1. Analyze Load to find arrival rate
    a. Average message size
    b. Average number of messages per unit time
    c. Processing time for average message
        i. Message transfer time
        ii. Manifest scanning time
        iii. Number of code and documentation files transferred
        iv. Number of manifests transferred

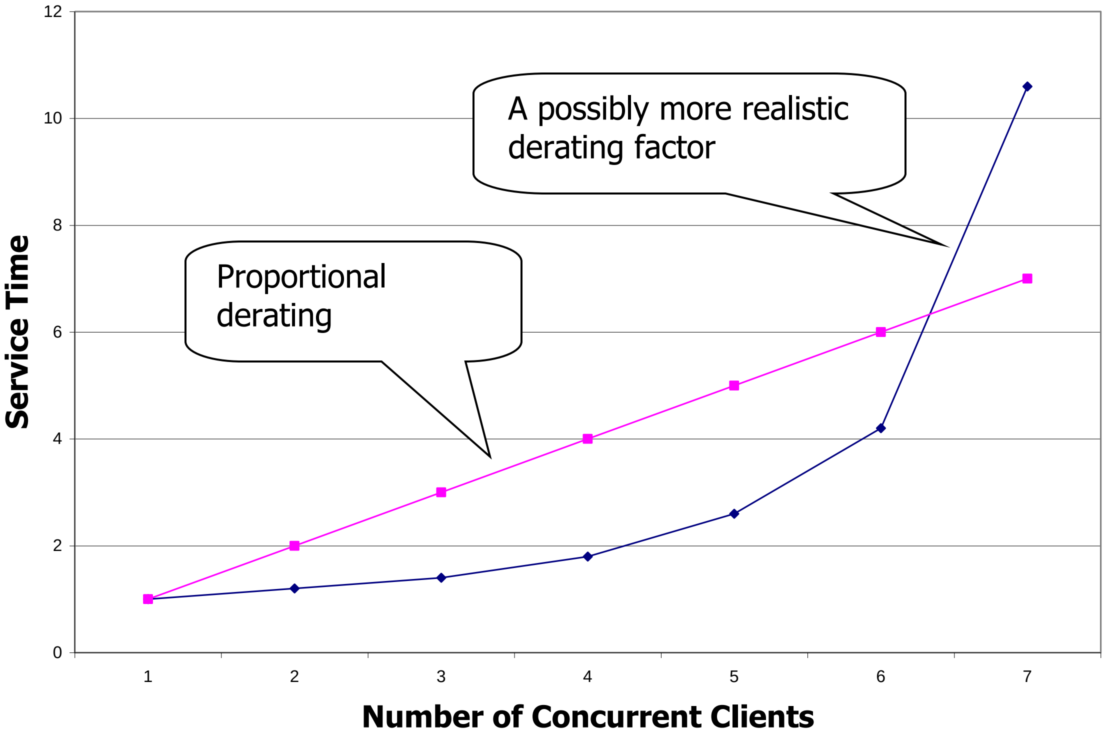
2. Measure Service Time to find processing rate
    a. Build simple prototypes that represent critical processing
    b. Measure their response times.
    c. Calculate or estimate service rate.

3. Calculate $\rho = \lambda/\mu$. If greater than 0.25 one server can't handle the load.

4. If more than one thread serving other clients, then you must derate μ, the service time. It will get smaller when additional clients are being served concurrently. You could assume:

   $\mu_{effective} = \mu$ / number of concurrent clients

   This is pessimistic. I expect the actual service time, $1/\mu_{effective}$ to look something like this:



Because your machine has many threads running even with no clients, adding a second client will not double service time since you are not doubling the entire load. However, eventually as you start to approach full CPU utilization the derating gets worse than proportional since the CPU spends a significant fraction of its time switching between tasks.

The best way to analyze the effect of multiple concurrent clients is to measure service rates when you have 1, 2, 4, 8, … concurrent clients.