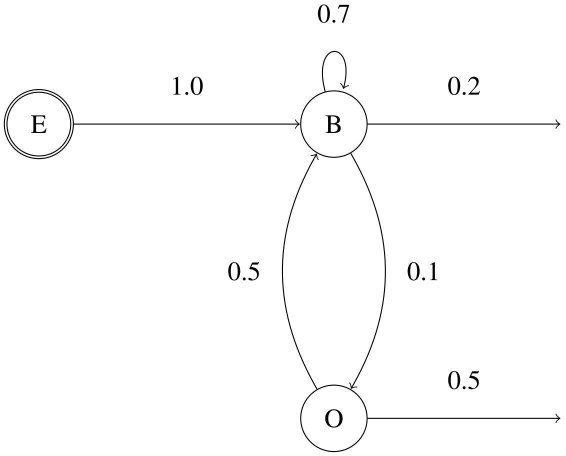
Performance Engineering Tutorial:
Demands and Bottlenecks

Exercise 1. Suppose that the following user behaviour graph models the interactions of a set of users with an IT system:



where B represents the *Browse catalog* service and S corresponds to the *Send order* service.

Question 1.1 Determine the average number of visits to each page and the average session length.
Solution:
We first solve the UBG using the following system of linear equations

$$V_E = 1 \tag{1}$$

$$V_B = V_E + 0.7V_B + 0.5V_O \tag{2}$$

$$V_O = 0.1V_B \tag{3}$$

We find that in a session the calls to each service are on average $V_B = 4$ and $V_O = 0.4$. The average session length is therefore $L = V_B + V_O = 4.4$ requests, where we do not count V_E since this is an artificial node not corresponding to an actual request.

Question 1.2 Assume that the total arrival rate of session to the site is λ sessions per second, each following the UBG defined above. Assume that the site is deployed on a two-tier architecture based on a front server tier, composed of machines that act as both web servers and application servers, and a database tier. The front tier is composed of n identical nodes labelled $F_1, \ldots, F_n$, with identical mean service time $S_{f,c}$ for requests of class $c = B, O$ (as usual, we assume each class to represent a page within the UBG). Similarly, the database tier is composed of a single database node labelled B with service demand $D_{b,c}$ for requests of class c.

Give formulas for the total utilization U_f at an arbitrary node within the front tier and for the total utilization U_b of the node within the database tier.
Solution:
If new sessions arrival at rate λ, then the average arrival rate of requests of the two classes will be $\lambda_B =$

$V_B\lambda$ and $\lambda_O = V_O\lambda$, since every session on average issues V_B requests to B and V_O requests to O. Every request will first hit a front server node and then the database, unless $D_{b,c} = 0$ in which case it hits the front tier only. Since each node i in the front tier processes only $k_{ic} = 1/n$ of the incoming requests of class c, by the utilization law we have

$$U_f = \lambda V_B S_{f,B}/n + \lambda V_O S_{f,O}/n$$
$$U_b = \lambda V_B D_{b,B} + \lambda V_O D_{b,O}$$

Question 1.3 Assume that $n = 2$, $S_{f,B} = 1$, $S_{f,O} = 2$, $D_{d,B} = 3$, $D_{d,O} = 0$. What is the maximum value of λ that the server can sustain before becoming unstable?
Solution:
A system is unstable when one or more resources become bottlenecks, as no spare capacity will be available to process incoming requests, leading to formation of an ever growing backlog. By setting the conditions $U_f \leq 1$ and $U_b \leq 1$, and replacing the parameters with numerical values, we can solve for the maximal value of λ that satisfies both constraints. This is the maximum rate that the server can sustain before becoming unstable. Plugging the numerical values in the formulas obtained in Question 1.2, we write for U_f after a little algebra

$$2.4\lambda \leq 1$$

and for U_b

$$12\lambda \leq 1$$

with the latter being the stricter constraint of the two. Hence $\lambda \leq 1/12$ sessions/sec.

Exercise 2. For a given IT system, we can collect in a demand matrix $D = [D_{i,c}]$, all the service demands at each node i for each service class c. The matrix D therefore shows resources as rows (we here considered resource labels $i = A, B, C, D$) and classes as columns (class labels $c = 1, 2, 3$).

For each of the demand matrices shown below,

- The class bottlenecks, for each class.
- The resources that can never be a bottleneck, i.e., the *dominated* resources.
- The resources that require a linear program (LP) to determine if they are potential bottlenecks.

Question 2.1

$$D = \begin{array}{c} \\ A \\ B \end{array} \begin{array}{c} \begin{array}{cc} 1 & 2 \end{array} \\ \left[\begin{array}{cc} 10 & 9 \\ 5 & 5 \end{array}\right] \end{array}$$

Solution:
In this example, A is the slowest resource for both classes, thus class-1 bottleneck resource: A, class-2 bottleneck resource: A, dominated resources: B, resources that need a LP to determine if they are potential bottlenecks: none.

Question 2.2

$$D = \begin{array}{c} \\ A \\ B \end{array} \begin{array}{c} \begin{array}{cc} 1 & 2 \end{array} \\ \left[\begin{array}{cc} 10 & 5 \\ 5 & 9 \end{array}\right] \end{array}$$

Solution:
In this example, the slowest resource depends on the class, thus class-1 bottleneck resource: A, class-2 bottleneck resource: B, dominated resources: none, resources that need a LP to determine: none.

Question 2.3

$$D = \begin{array}{c} \\ A \\ B \\ C \\ D \end{array} \begin{array}{c} \begin{array}{ccc} 1 & 2 & 3 \end{array} \\ \begin{bmatrix} 10 & 5 & 9 \\ 4 & 7 & 1 \\ 3 & 0 & 10 \\ 0 & 10 & 0 \end{bmatrix} \end{array}$$

Solution:
This is another example where each class has a different bottleneck. It is unclear from visual inspection if B can saturate, as there may be combinations of the request rates λ that push the utilization of this resource to 100%. Note that a resource like D can always saturate, even though it has zero demands on classes 1 and 3, since if the workload consists only of class-2 requests it will saturate for high enough arrival rate of class-2 requests. Therefore, class-1 bottleneck: A, class-2 bottleneck: D, class-3 bottleneck: C, dominated resources: none, resources that need a LP to determine: B.

Question 2.4

$$D = \begin{array}{c} \\ A \\ B \\ C \\ D \end{array} \begin{array}{c} \begin{array}{ccc} 1 & 2 & 3 \end{array} \\ \begin{bmatrix} 10 & 10 & 10 \\ 4 & 2 & 1 \\ 3 & 0 & 9 \\ 0 & 2 & 0 \end{bmatrix} \end{array}$$

Solution:
In this case A is the slowest resource for all classes, so it will always be the bottleneck irrespectively of the request mix. As the demands at the other resources are strictly smaller than the demand on A on all classes, the resource utilization will also be smaller due to the utilization law. Summarising, class-1 bottleneck: A, class-2 bottleneck: A, class-3 bottleneck: A, dominated resources: B, C, D, resources that need a LP to determine: none.

Question 2.5

$$D = \begin{array}{c} \\ A \\ B \\ C \\ D \end{array} \begin{array}{c} \begin{array}{cc} 1 & 2 \end{array} \\ \begin{bmatrix} 10 & 3 \\ 4 & 7 \\ 3 & 0 \\ 0 & 9 \end{bmatrix} \end{array}$$

Solution:
Here we see that A and D can saturate when the request mix is made up of requests of class 1 or 2 only. C is dominated by A, as both of its demands are systematically smaller, whereas there is no single resource that dominates B over all classes. Therefore, class-1 bottleneck: A, class-2 bottleneck: D, dominated resources: C, resources that need a LP to determine: B.

Question 2.6

$$D = \begin{array}{c} \\ A \\ B \\ C \\ D \end{array} \begin{array}{c} \begin{array}{cc} 1 & 2 \end{array} \\ \begin{bmatrix} 10 & 10 \\ 4 & 7 \\ 3 & 0 \\ 0 & 10 \end{bmatrix} \end{array}$$

Solution:
This example shows a special case where on class 2 the demand is identical at A and D, therefore we will

have multiple bottlenecks forming simultaneously under a mix with 100% requests of class 2. Therefore, class-1 bottleneck: A. class-2 bottleneck: A, D, dominated resources: B, C, resources that need a LP to determine: none.

Question 2.7 Choose any of the examples above where you needed to use a linear program (LP) to establish if a resource was a potential bottleneck. Write down the LP formulation. Then explain how would you determine the mix that saturates the potential bottleneck from the optimal LP solution, if one exists.
Solution:
We first illustrate this for resource B in Question 2.3.

$$\begin{aligned} U_B^{\max} = \text{maximize} \quad & 4\lambda_1 + 7\lambda_2 + \lambda_3 \\ \text{subject to} \quad & \\ & 10\lambda_1 + 5\lambda_2 + 9\lambda_3 \leq 1 \\ & 4\lambda_1 + 7\lambda_2 + \lambda_3 \leq 1 \\ & 3\lambda_1 + 10\lambda_3 \leq 1 \\ & 10\lambda_2 \leq 1 \\ & \lambda_1, \lambda_2, \lambda_3 \geq 0 \end{aligned}$$

If the LP has optimal value $U_B^{\max} = 1$, then B can saturate. As the optimization is over the decision variables λ_1, λ_2, λ_3, associated to the optimal value we will have an optimal solution $(\lambda_1^*, \lambda_2^*, \lambda_3^*)$ representing a situation where the system receives a total rate of request equal to $\lambda = \lambda_1^* + \lambda_2^* + \lambda_3^*$ and the mix of the different classes is $(\lambda_1^*/\lambda, \lambda_2^*/\lambda, \lambda_3^*/\lambda)$, so that each entry of the vector gives the percentage of requests of a given class.

Similarly, for resource B in Question 2.5.

$$\begin{aligned} U_B^{\max} = \text{maximize} \quad & 4\lambda_1 + 7\lambda_2 \\ \text{subject to} \quad & \\ & 10\lambda_1 + 3\lambda_2 \leq 1 \\ & 4\lambda_1 + 7\lambda_2 \leq 1 \\ & 3\lambda_1 \leq 1 \\ & 9\lambda_2 \leq 1 \\ & \lambda_1, \lambda_2 \geq 0 \end{aligned}$$

As before, if the LP has optimal value $U_B^{\max} = 1$, then B can saturate and the mix will be given by $(\lambda_1^*/\lambda, \lambda_2^*/\lambda)$, with $\lambda = \lambda_1^* + \lambda_2^*$.