COMP9334
Capacity Planning for Computer Systems and Networks

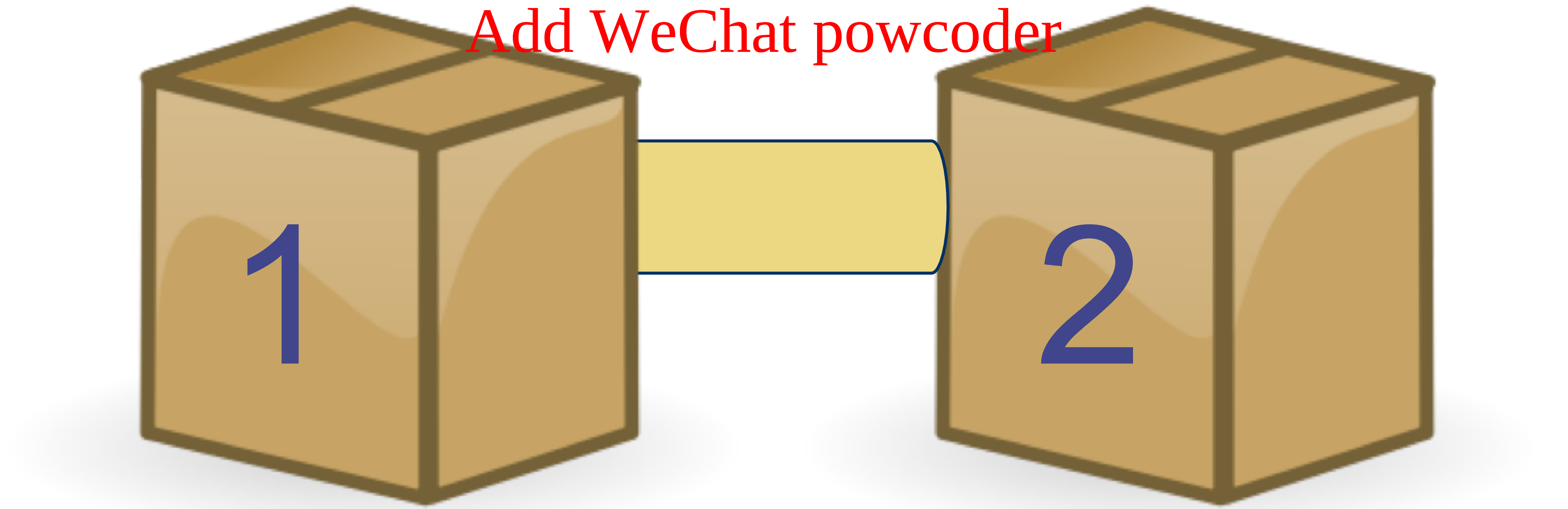
Week 3: Queues with Poisson arrivals

Pre-lecture exercise 1:

- Let X and Y be two events
- Let Prob[X] = Probability that event X occurs
- Let Prob[Y] = Probability that event Y occurs
- Question: Under what condition will the following equality hold?
  - Prob[X or Y] = Prob[X] + Prob[Y]

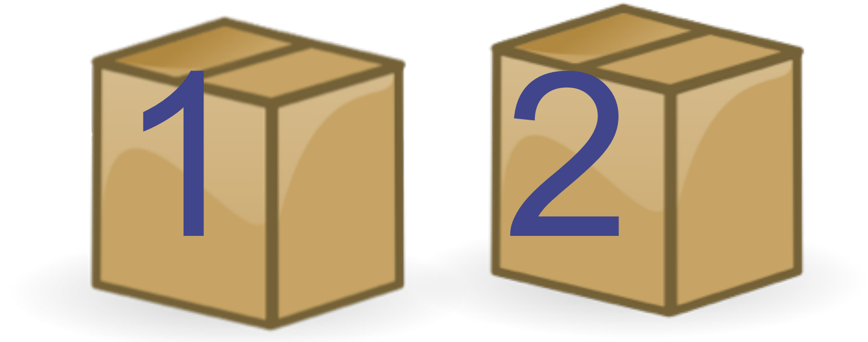
Pre-lecture exercise 2: Where is Felix? (Page 1)

- You have two boxes: Box 1 and Box 2, as well as a cat called Felix
- The two boxes are connected by a tunnel
- Felix likes to hide inside these boxes and travels between them using the tunnel.
- Felix is a very fast cat so the probability of finding him in the tunnel is zero
- You know Felix is in one of the boxes but you don't know which one



Pre-lecture exercise 2: Where is Felix? (Page 2)

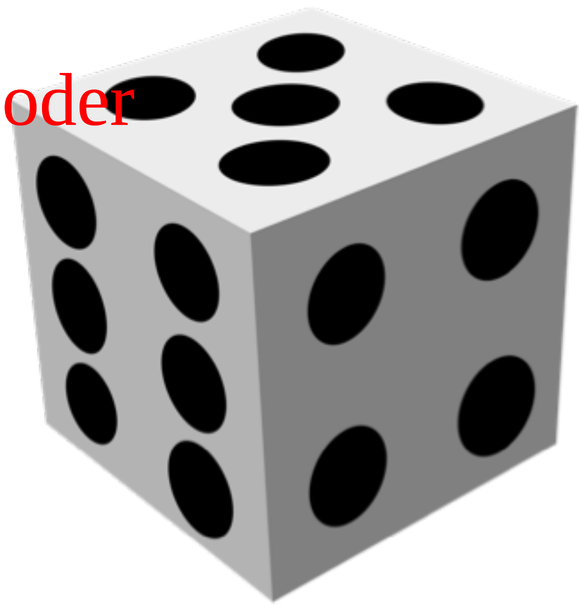
- Notation:
  - Prob[A] = probability that event A occurs
  - Prob[A | B] = probability that event A occurs given event B
- You do know
  - Felix is in one of the boxes at times 0 and 1
  - Prob[Felix is in Box 1 at time 0] = 0.3
  - Prob[Felix will be in Box 2 at time 1| Felix is in Box 1 at time 0] = 0.4
  - Prob[Felix will be in Box 1 at time 1| Felix is in Box 2 at time 0] = 0.2
- Calculate
  - Prob[Felix is in Box 1 at time 1]
  - Prob[Felix is in Box 2 at time 1]



Pre-lecture exercise 3

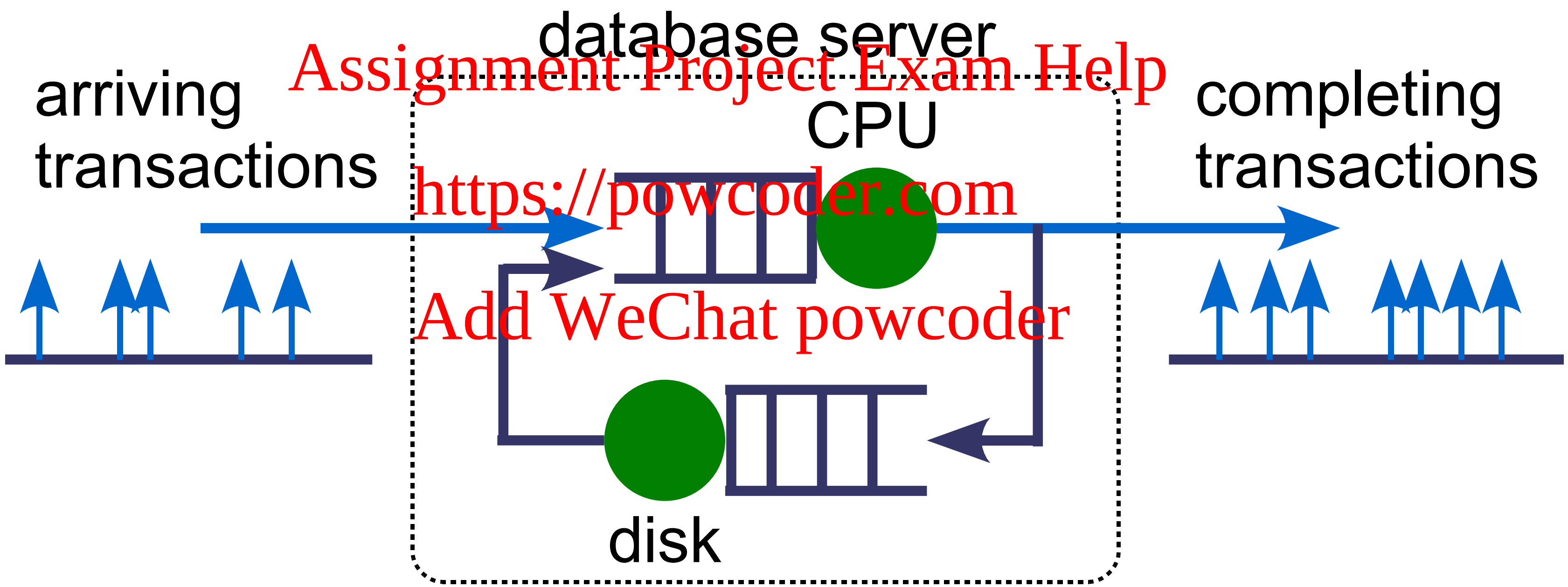
- You have a loaded die with 6 faces with values 1, 2, 3, 4, 5 and 6
- The probability that you can get each face is given in the table below
- What is the mean value that you can get?

Value	**Probability**
1	0.1
2	0.1
3	0.2
4	0.1
5	0.3
6	0.2



Week 1:

- Modelling a computer system as a network of queues
- Example: Open queueing network consisting of two queues



Week 2:

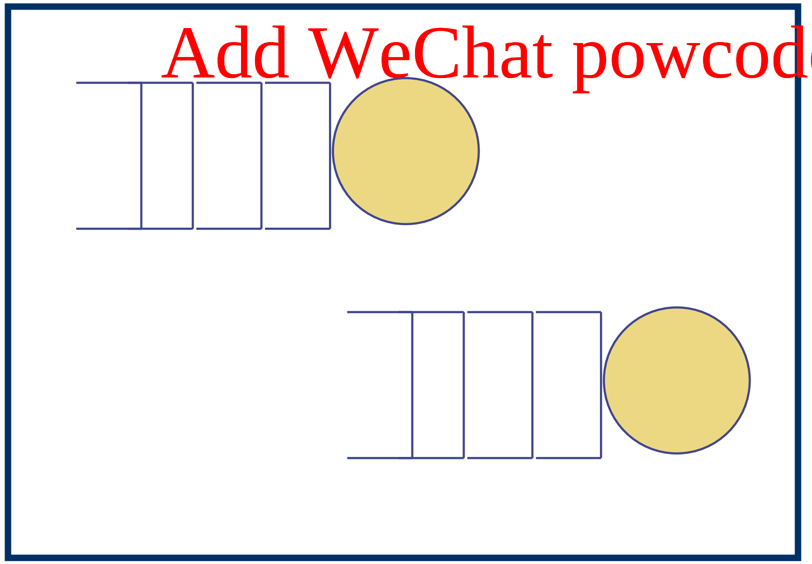
- Operational analysis
  - Measure #completed jobs, busy time etc
  - Operational quantities: utilisation, response time, throughput etc.
  - Operational laws relate the operational quantities

  - Bottleneck analysis

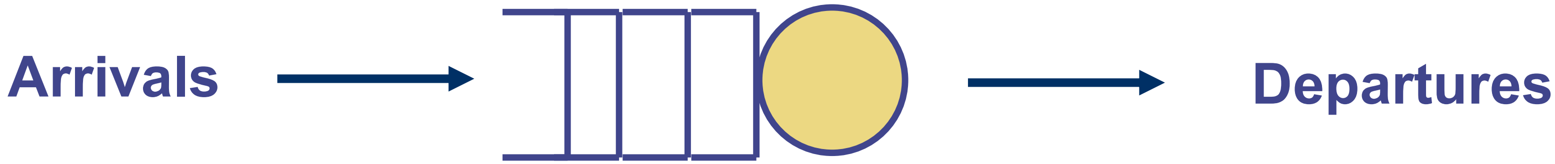
Little's Law

- Applicable to any "box" that contains some queues or servers
- Mean number of jobs in the "box" =

  Mean response time x Throughput
- We will use Little's Law in this lecture to derive the mean response time
  - We first compute the mean number of jobs in the "box" and throughput

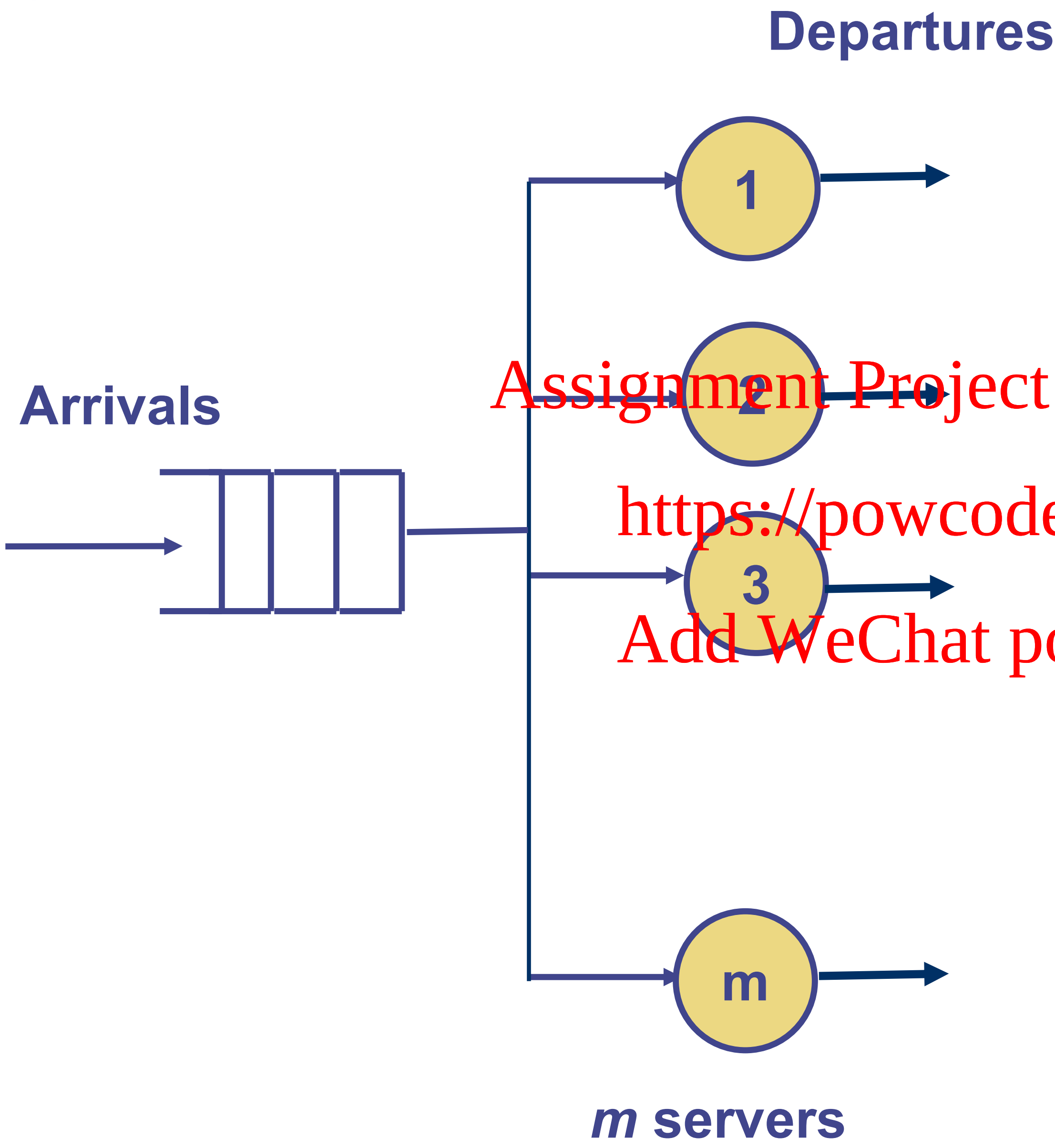


This week (1)



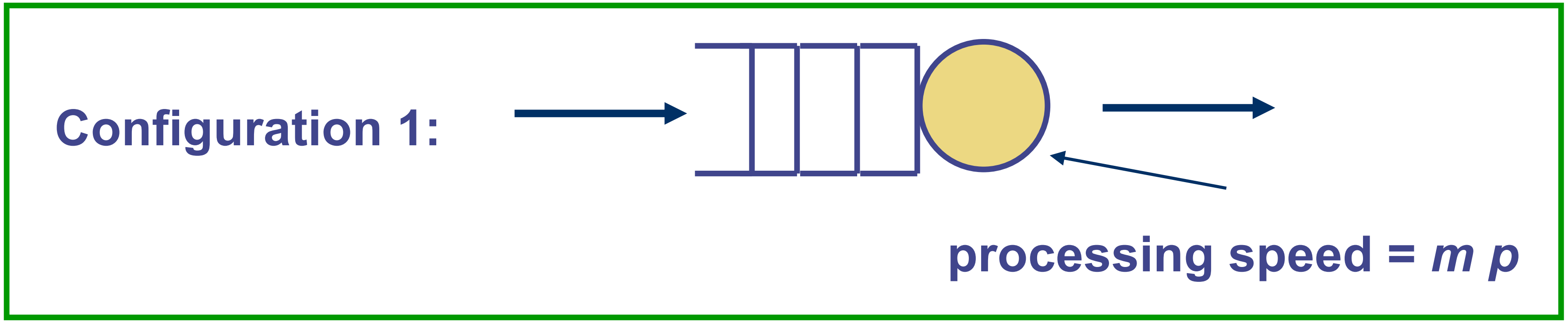
- Open, single server queues and
- How to find:
  - Waiting time
  - Response time
  - Mean queue length etc.
- The technique to find waiting time etc. is called *Queueing Theory*

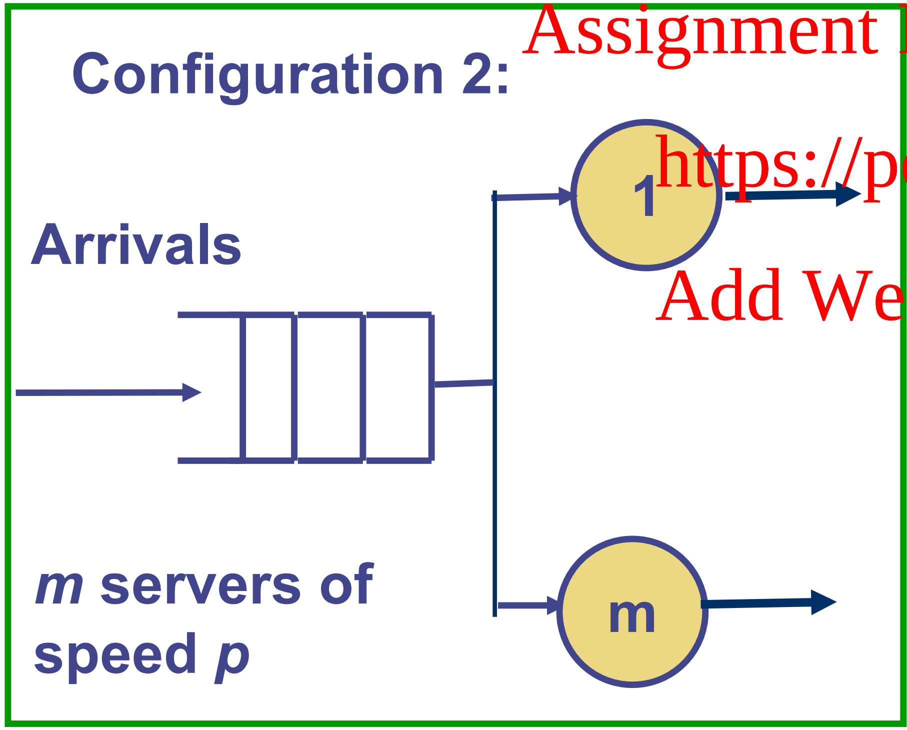
This week (2)

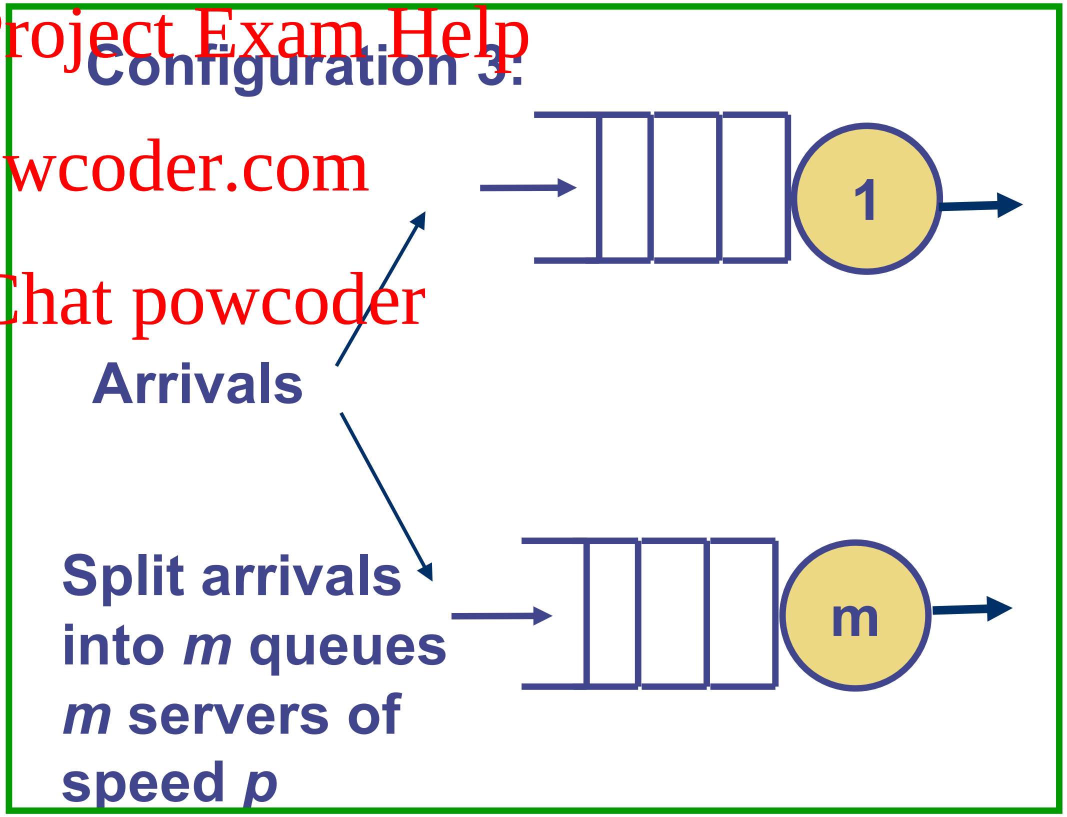


- Open, multi-server queue
- How to find:
  - Waiting time
  - Response time
  - Mean queue length etc.

What will you be able to do with the results?





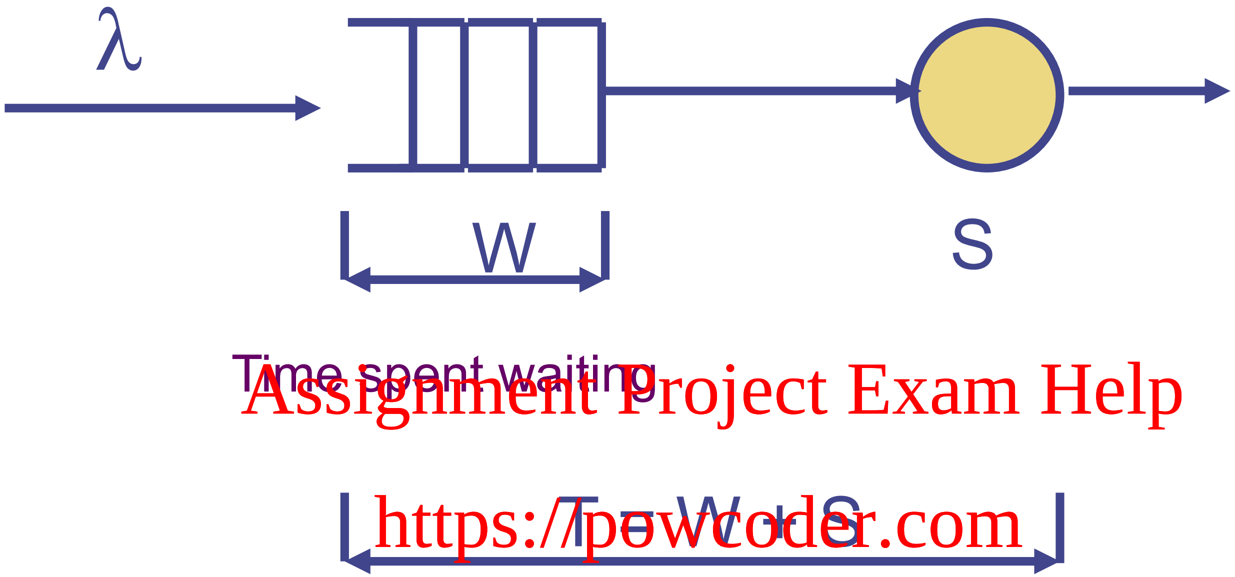


Which configuration has the best response time?

Be patient

- We will show how we can obtain the response time
  - It takes a number of steps to obtain the answer
- It takes time to stand in a queue, it also takes time to derive results in queuing theory!

Single Server Queue: Terminology



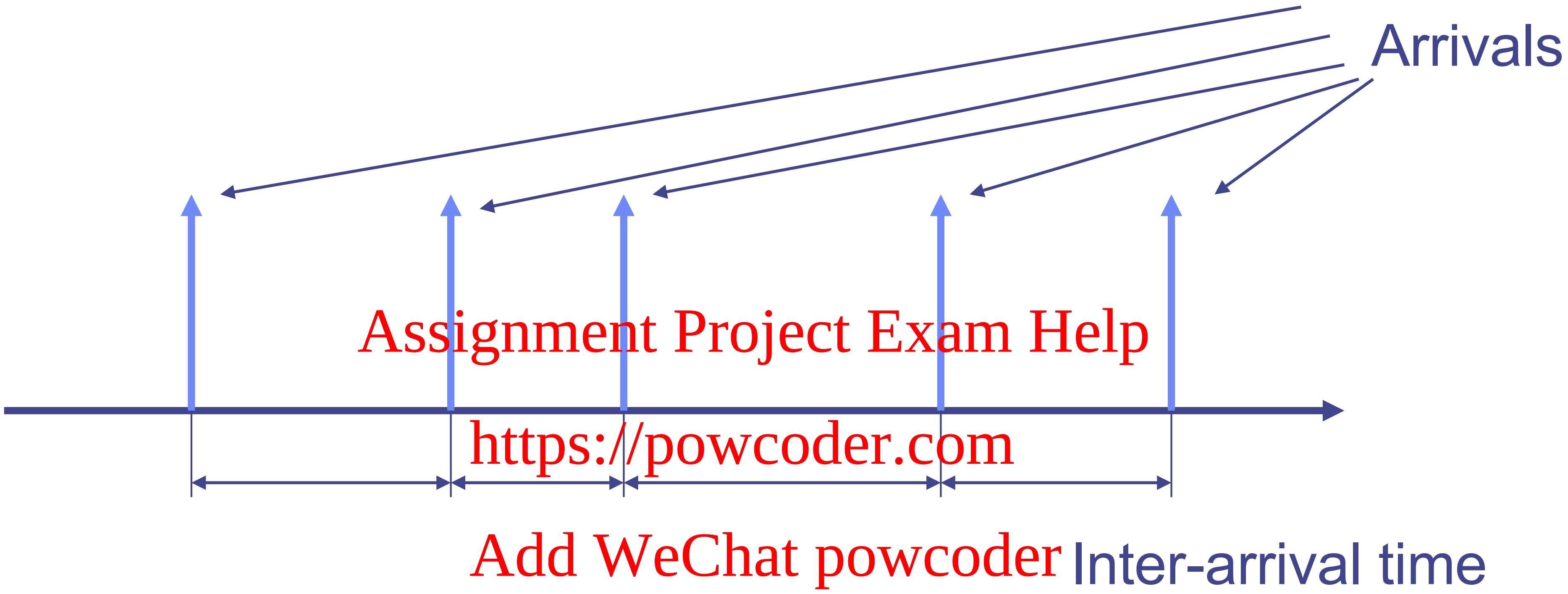
Response Time T
= Waiting time W + Service time S

Note: We use T for response time because this is the notation in many queueing theory books. For a similar reason, we will use ρ for utilisation rather than U.

Single server system

- In order to determine the response time, you need to know
  - The inter-arrival time probability distribution
  - The service time probability distribution
- Possible distributions
  - Deterministic
    - Constant inter-arrival time
    - Constant service time
  - Exponential distribution
- We will focus on exponential distribution

Exponential inter-arrival with rate λ



We assume that successive arrivals are independent

Probability that inter-arrival time is between x and $x + \delta x$
$= \lambda \exp(-\lambda x)\, \delta x$

Poisson distribution (1)

- The following are equivalent
  - The inter-arrival time is independent and exponentially distributed with parameter λ
  - The number of arrivals in an interval T is a Poisson distribution with parameter λ

$$Pr[k \text{ arrivals in a time interval } T] = \frac{(\lambda T)^k exp(-\lambda T)}{k!}$$

- Mean inter-arrival time = 1 / λ
- Mean number of arrivals in time interval T = λ T
- Mean arrival rate = λ

Poisson distribution (2)

- Poisson distribution arises from a large number of independent sources
  - An example from Week 2:
    - *N* customers, each with a probability of *p* per unit time to make a request.
    - This creates a Poisson arrival with $\lambda = Np$

- Another interpretation of Poisson arrival:
  - Consider a small time interval δ
    - This means δ^n (for n >= 2) is negligible

  - Probability [no arrival in δ] = $1 - \lambda\delta$

  - Probability [1 arrival in δ] = $\lambda\delta$
  - Probability [2 or more arrivals in δ] ≈ 0
- This interpretation can be derived from:

$$Pr[k \text{ arrivals in a time interval } T] = \frac{(\lambda T)^k exp(-\lambda T)}{k!}$$

Service time distribution

- Service time = the amount of processing time a job requires from the server
- We assume that the service time distribution is exponential with parameter μ
  - The probability that the service time is between t and t + δt is:

$$\mu \exp(-\mu t)\, \delta t$$

- Here: μ = service rate = 1 / mean service time
- Another interpretation of exponential service time:
  - Consider a small time interval δ
  - Probability [a job will finish its service in next δ seconds] = μ δ
  - Probability [a job will **not** finish its service in next δ seconds] =

Sample queueing problems

- Consider a call centre
  - Calls are arriving according to Poisson distribution with rate λ
  - The length of each call is exponentially distributed with parameter μ
    - Mean length of a call is $1/\mu$ (in, e.g. seconds)

Call centre:

Arrivals

***m* operators**

If all operators are busy, the centre can put at most *n* additional calls on hold.

If a call arrives when all operators and holding slots are used, the call is rejected.

- Queueing theory will be able to answer these questions:
  - What is the mean waiting time for a call?
  - What is the probability that a call is rejected?

Road map

- We will start by looking at a call centre with one operator and no holding slot
  - This may sound unrealistic but we want to show how we can solve a typical queueing network problem
  - After that we go into queues that are more complicated

Call centre with 1 operator and no holding slots

- Let us see how we can solve the queuing problem for a very simple call centre with 1 operator and no holding slots
- What happens to a call that arrives when the operator is busy?
  -
- What happens to a call that arrives when the operator is idle?

  -
- We are interested to find the probability that an arriving call is rejected.

Arrivals →

Call centre:

***1* operator. No holding slot.**

Solution (1)

- There are two possibilities for the operator:
  - Busy or
  - Idle
- Let
  - State 0 = Operator is idle (i.e. #calls in the call centre = ?
  - State 1 = Operator is busy (i.e. #calls in the call centre = ?

$$P_0(t) = \text{Prob. } 0 \text{ call in the call centre at time } t$$

$$P_1(t) = \text{Prob. } 1 \text{ call in the call centre at time } t$$

Solution (2)

We try to express $P_0(t + \Delta t)$ in terms of $P_0(t)$ and $P_1(t)$

- No call at call centre at t + Δt can be caused by
  - **No call at time t and no call arrives in [t, t + Δt], or**
  -

$$P_0(t + \Delta t) = \qquad + $$

Question: Why do we NOT have to consider the following possibility:
No customer at time t & 1 customer arrives in [t, t + Δt] &
the call finishes within [t, t + Δt].

Solution (3)

- Similarly, we can show that

$$P_1(t+\Delta t) = P_0(t)\lambda\Delta t + P_1(t)(1-\mu\Delta t)$$

- If we let Δt ➔ 0, we have

$$\frac{dP_0(t)}{dt} = -P_0(t)\lambda + P_1(t)\mu$$

$$\frac{dP_1(t)}{dt} = P_0(t)\lambda - P_1(t)\mu$$

Solution (4)

- We can solve these equations to get

$$P_0(t) = \frac{\mu}{\lambda+\mu} - \frac{\mu}{\lambda+\mu}e^{-(\mu+\lambda)t}$$

$$P_1(t) = \frac{\lambda}{\lambda+\mu} + \frac{\mu}{\lambda+\mu}e^{-(\mu+\lambda)t}$$

- This is too complicated, let us look at **steady state** solution

$$P_0 = P_0(\infty) = \frac{\mu}{\lambda+\mu}$$

$$P_1 = P_1(\infty) = \frac{\lambda}{\lambda+\mu}$$

Solution (5)

- From the steady state solution, we have
  - The probability that an arriving call is rejected
  - = The probability that the operator is busy
  - =

$$P_1 = \frac{\lambda}{\lambda + \mu}$$

- Let us check whether it makes sense
  - For a constant μ, if the arrival rate rate λ increases, will the probability that the operator is busy go up or down?
  - Does the formula give the same prediction?

An alternative interpretation

- We have derived the following equation:

$$P_0(t + \Delta t) = P_0(t)(1 - \lambda \Delta t) + P_1(t)\mu \Delta t$$

- Which can be rewritten as:

$$P_0(t + \Delta t) - P_0(t) = -P_0(t)\lambda \Delta t + P_1(t)\mu \Delta t$$

- At steady state:

$$\text{Change in Prob in State } 0 = 0$$

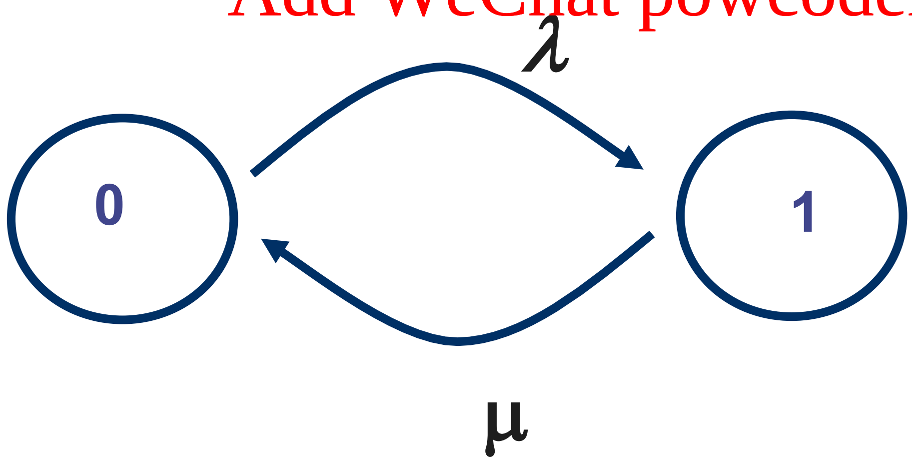
$$\Rightarrow 0 = -\boxed{P_0 \lambda} \Delta t + \boxed{P_1 \mu} \Delta t$$

Rate of leaving state 0

Rate of entering state 0

Faster way to obtain steady state solution (1)

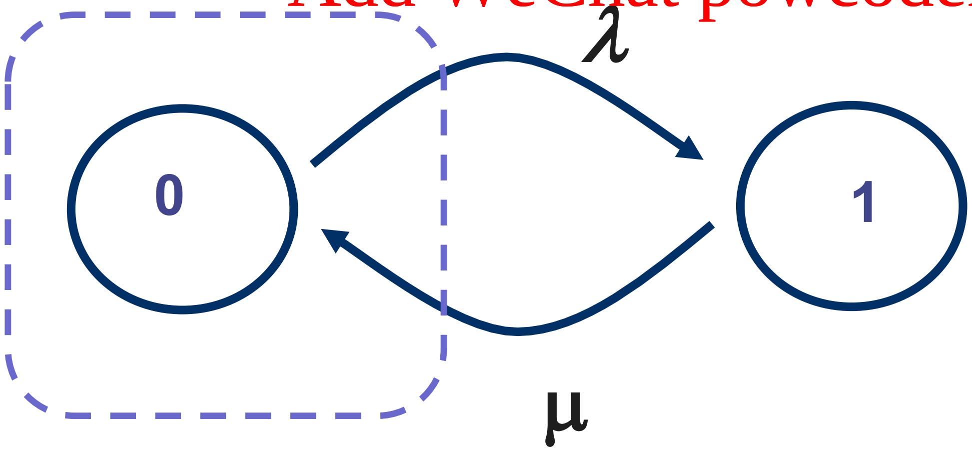
- Transition from State 0 to State 1
  - Caused by an arrival, the rate is λ
- Transition from State 1 to State 0
  - Caused by a completed service, the rate is μ
- State diagram representation
  - *Each circle is a state*
  - *Label the arc between the states with transition rate*



Faster way to obtain steady state solution (2)

- Steady state means
  - **rate of transition out of a state = Rate of transition into a state**
- We have for state 0:

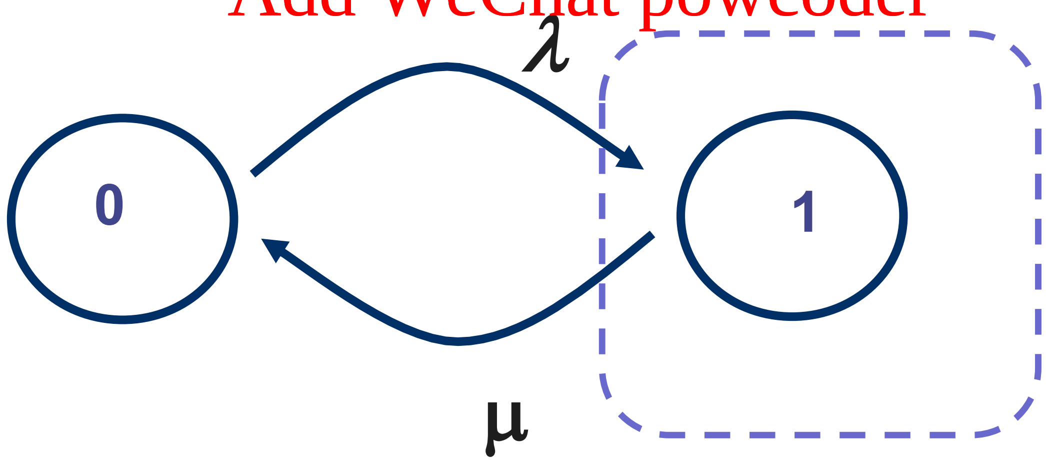
$$\lambda P_0 = \mu P_1$$



Faster way to obtain steady state solution (3)

- We can do the same for State 1:
- Steady state means
  - **Rate of transition into a state** = **rate of transition out of a state**
- We have for state 1:

$$\underline{\lambda P_0} = \underline{\mu P_1}$$



Faster way to obtain steady state solution (4)

- We have one equation $\lambda P_0 = \mu P_1$

- We have 2 unknowns and we need one more equation.
- Since we must be either one of the two states:

$$P_0 + P_1 = 1$$

- Solving these two equations, we get the same steady state solution as before

$$P_0 = \frac{\mu}{\lambda + \mu} \qquad P_1 = \frac{\lambda}{\lambda + \mu}$$

Summary

- Solving a queueing problem is not simple
- It is harder to find how a queue evolves with time
- It is simpler to find how a queue behaves at steady state
  - Procedure:
    - Draw a diagram with the states
    - Add arcs between states with transition rates
    - Derive flow balance equation for each state, i.e.
      - Rate of entering a state = Rate of leaving a state
    - Solve the equation for steady state probability

Let us have a look at our call centre problem again

- Consider a call centre
  - Calls are arriving according to Poisson distribution with rate λ
  - The length of each call is exponentially distributed with parameter μ
    - Mean length of a call is $1/\mu$

Call centre:

Arrivals →

***m* operators**

If all operators are busy, the centre can put at most *n* additional calls on hold.

If a call arrives when all operators and holding slots are used, the call is rejected.

- We solve the problem for $m = 1$ and $n = 0$
  - We call this a M/M/1/1 queue (explanation on the next page)
- How about other values of m and n

Kendall's notation

- To represent different types of queues, queueing theorists use the Kendall's notation
- The call centre example on the previous page can be represented as:

M / M / s (/ B)

Inter-arrival distribution is Markovian i.e. Exponential

Service time distribution is Markovian i.e exponential

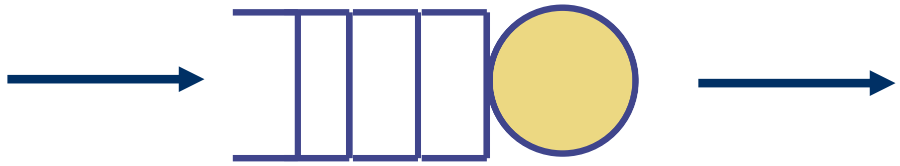
Buffer Positions (wait room)

Number of servers

The call centre example on the last page is a M/M/m/(m+n) queue
If n = ∞, we simply write M/M/m

M/M/1 queue

Exponential
Inter-arrivals (λ)



Exponential
Service time (μ)

Infinite buffer One server

- Consider a call centre analogy

  - Calls are arriving according to Poisson distribution with rate λ

  - The length of each call is exponentially distributed with parameter μ
    - Mean length of a call is $1/\mu$

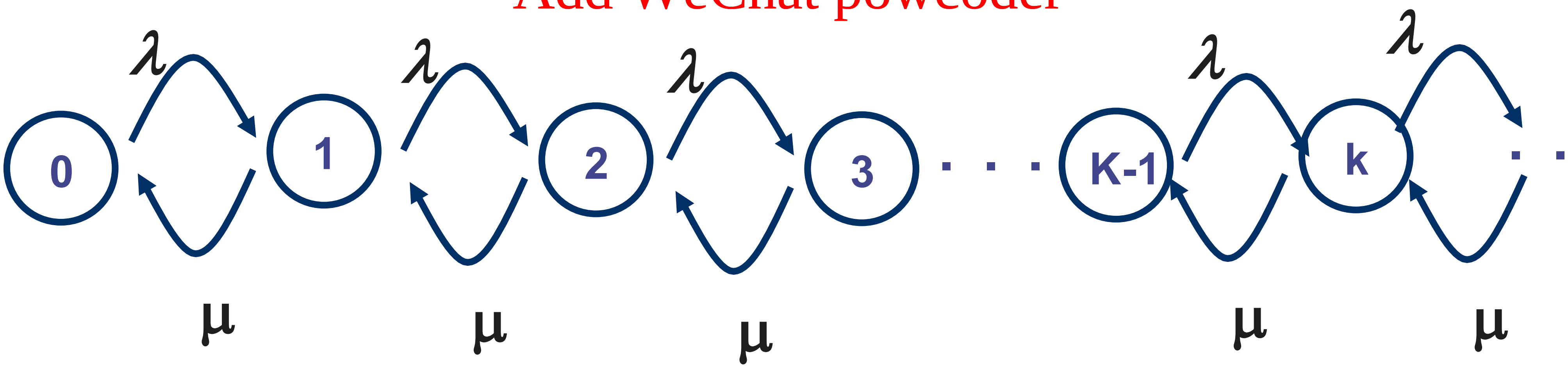
Arrivals

Call centre with *1* operator
If the operator is busy, the centre will put the call on hold.
A customer will wait until his call is answered.

- Queueing theory will be able to answer these questions:
  - What is the mean waiting time for a call?

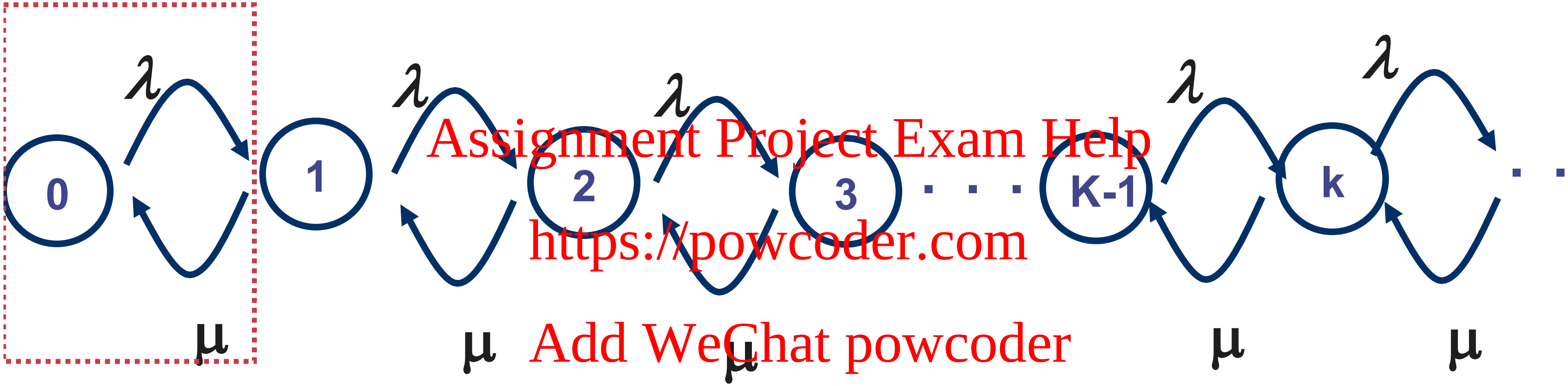
Solving M/M/1 queue (1)

- We will solve for the steady state response
- Define the states of the queue
  - State 0 = There is zero job in the system (= The server is idle)
  - State 1 = There is 1 job in the system (= 1 job at the server, no job queueing)
  - State 2 = There are 2 jobs in the system (= 1 job at the server, 1 job queueing)
  - State *k* = There are *k* jobs in the system (= 1 job at the server, *k-1* job queueing)
- The state transition diagram



Solving M/M/1 queue (2)

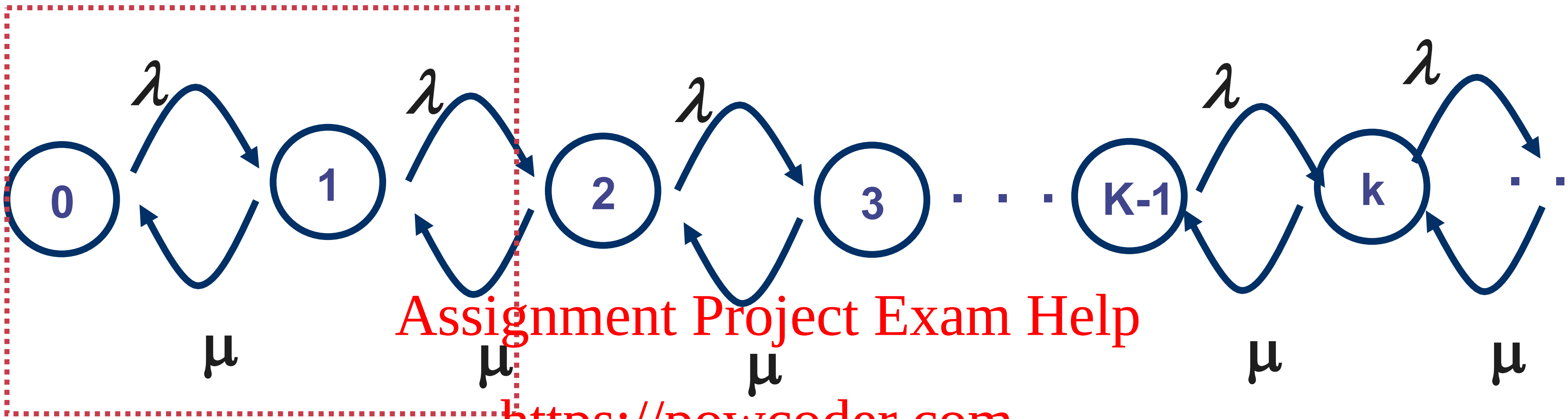
$$P_k = \text{Prob. } k \text{ jobs in system}$$



$$\lambda P_0 = \mu P_1$$

$$\Rightarrow P_1 = \frac{\lambda}{\mu} P_0$$

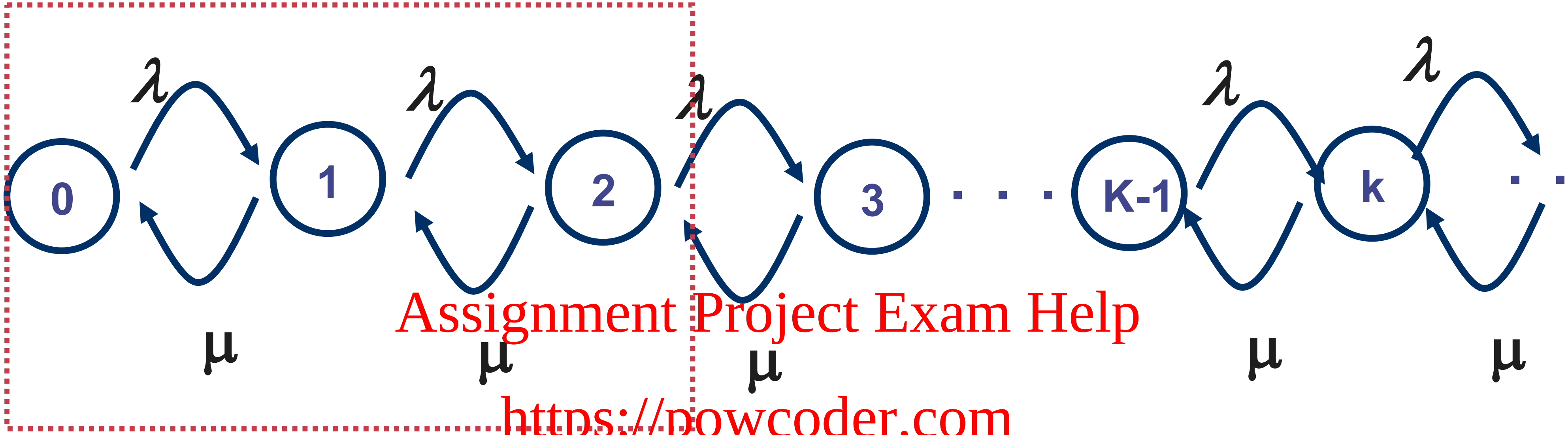
Solving M/M/1 queue (3)



$$\lambda P_1 = \mu P_2$$

$$\Rightarrow P_2 = \frac{\lambda}{\mu} P_1 \quad \Rightarrow P_2 = \left(\frac{\lambda}{\mu}\right)^2 P_0$$

Solving M/M/1 queue (4)



$$\lambda P_2 = \mu P_3$$

$$\Rightarrow P_3 = \frac{\lambda}{\mu} P_2 \quad \Rightarrow P_3 = \left(\frac{\lambda}{\mu}\right)^3 P_0$$

Solving M/M/1 queue (5)

In general $$P_k = \left(\frac{\lambda}{\mu}\right)^k P_0$$

$$\text{Let } \rho = \frac{\lambda}{\mu}$$

We have $$P_k = \rho^k P_0$$

Solving M/M/1 queue (6)

With $P_k = \rho^k P_0$ **and**

$$P_0 + P_1 + P_2 + P_3 + \ldots = 1$$

$$\Rightarrow (1 + \rho + \rho^2 + \ldots) P_0 = 1$$

$$\Rightarrow P_0 = 1 - \rho \text{ if } \rho < 1$$

$$\Rightarrow P_k = (1 - \rho)\rho^k$$

ρ = utilisation
= Prob server is busy
= 1 - P_0
= 1- Prob server is idle

Since $\rho = \dfrac{\lambda}{\mu}$, $\rho < 1 \Rightarrow \lambda < \mu$

Arrival rate < service rate

Solving M/M/1 queue (7)

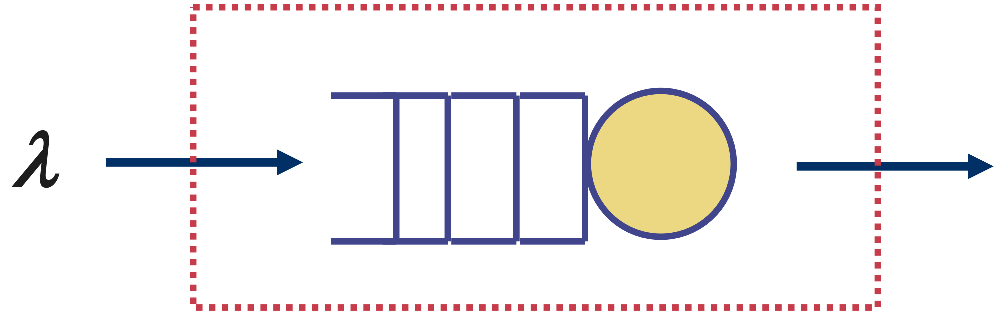
With $P_k = (1-\rho)\rho^k$

This is the probability that there are k jobs in the system.
To find the response time, we will make use of Little's law.
First we need to find the mean number of customers =

$$\sum_{k=0}^{\infty} kP_k = \sum_{k=0}^{\infty} k(1-\rho)\rho^k$$

$$= \frac{\rho}{1-\rho}$$

Solving M/M/1 queue (8)



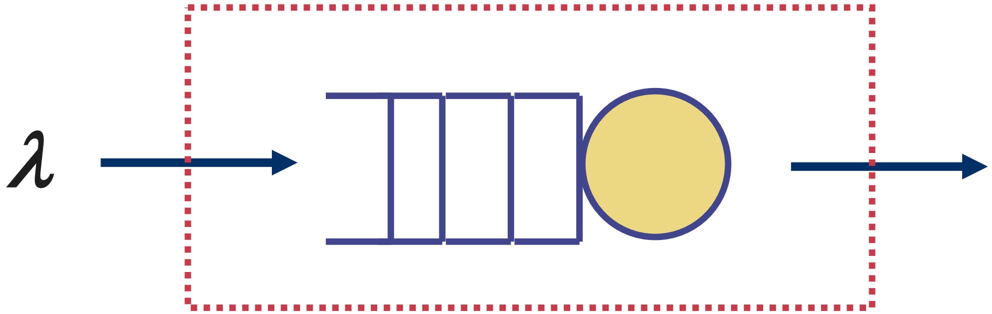
Little's law:

mean number of customers = throughput x response time

Throughput is λ ***(why?)***

$$\text{Response time } T = \frac{\rho}{\lambda(1-\rho)} = \frac{1}{\mu - \lambda}$$

Solving M/M/1 queue (9)



What is the mean waiting time at the queue?

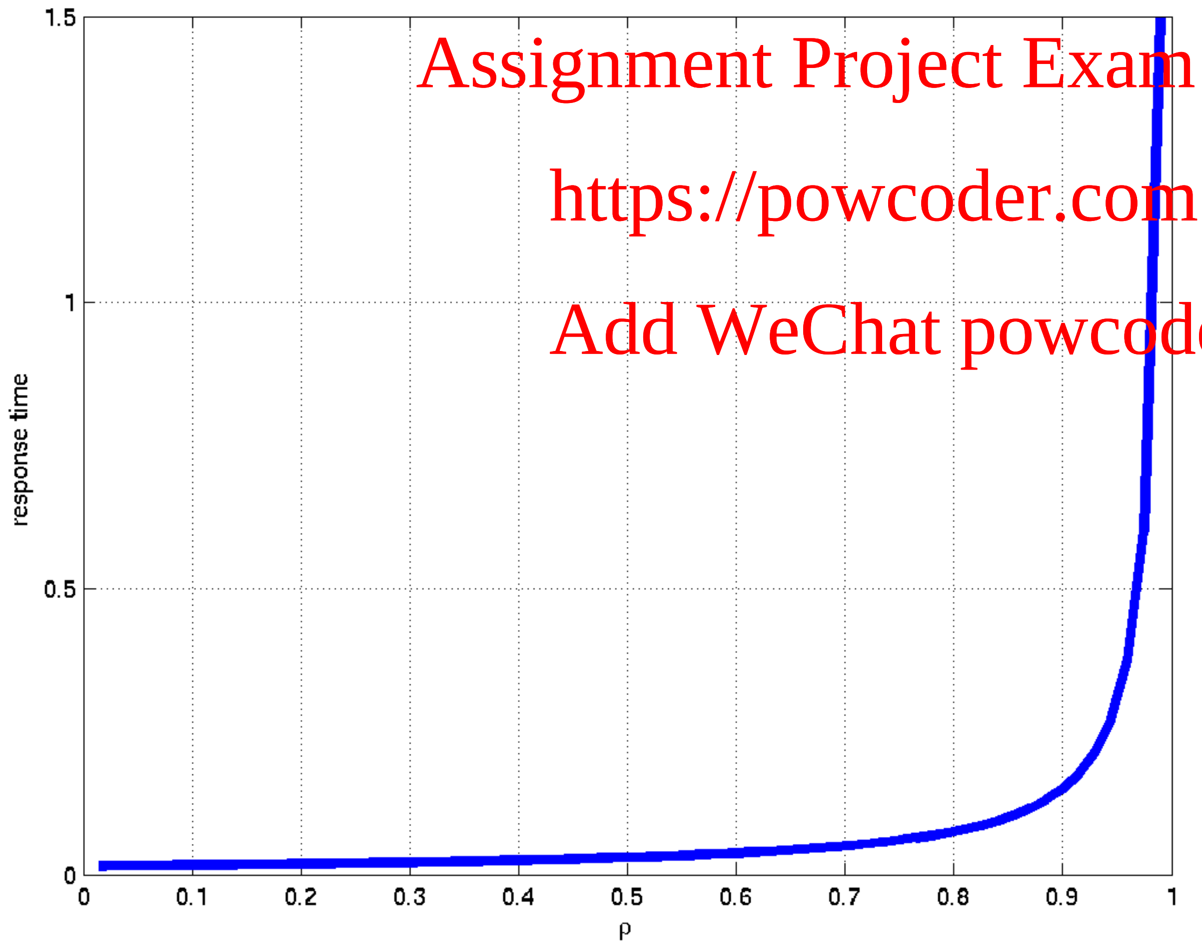
Mean waiting time = mean response time - mean service time

We know mean response time (from last slide)

Mean service time is = $1 / \mu$

Using the service time parameter ($1/\mu$ = 15ms) in the example, let us see how response time T varies with λ

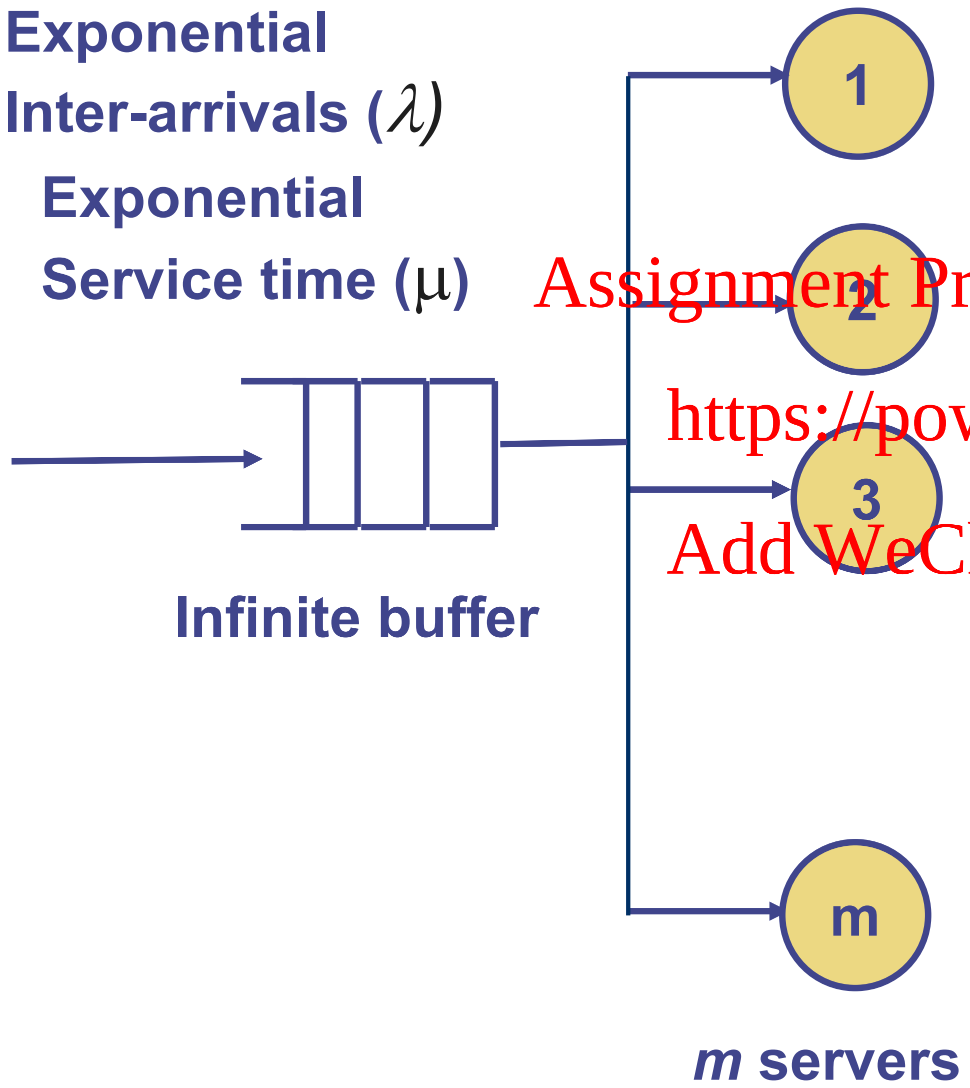
$$T = \frac{1}{\mu(1-\rho)}$$



Observation: Response time increases sharply when ρ gets close to 1

Infinite queue assumption means $\rho \to 1$, $T \to \infty$

Multi-server queues M/M/m



All arrivals go into one queue.

Customers can be served by any one of the *m* servers.

When a customer arrives

- If all servers are busy, it will join the queue
- Otherwise, it will be served by one of the available servers

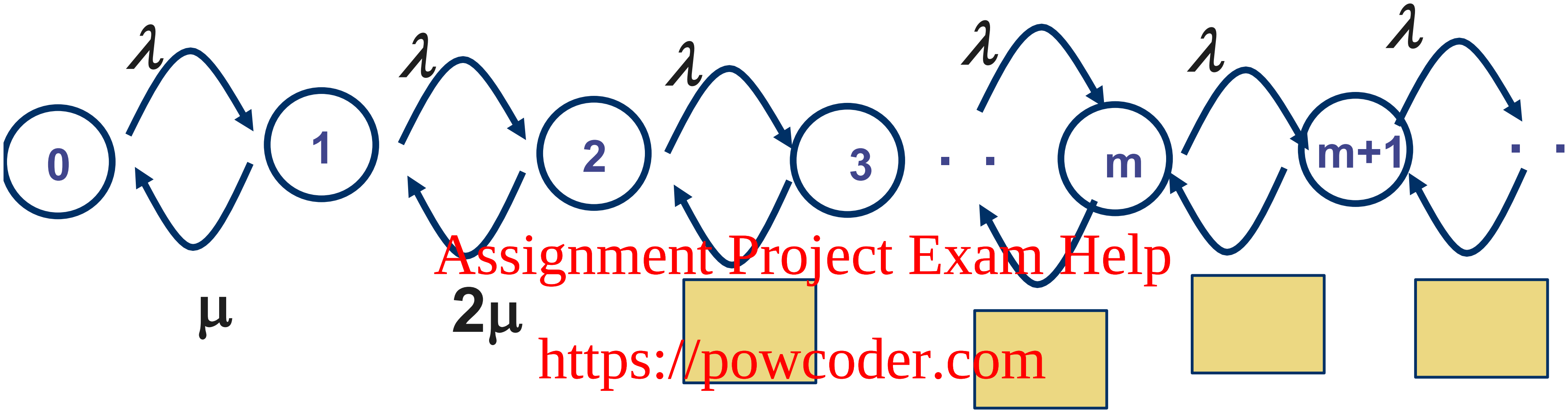
A call centre analogy of M/M/m queue

- Consider a call centre analogy
  - Calls are arriving according to Poisson distribution with rate λ
  - The length of each call is exponentially distributed with parameter μ
    - Mean length of a call is $1/\mu$

Arrivals →

Call centre with *m* operators
If all *m* operators are busy, the centre will put the call on hold.
A customer will wait until his call is answered.

State transition for M/M/m



M/M/m

- Following the same method, we have mean response time T is

$$T = \frac{C(\rho, m)}{m\mu(1-\rho)} + \frac{1}{\mu}$$

where $$\rho = \frac{\lambda}{m\mu}$$

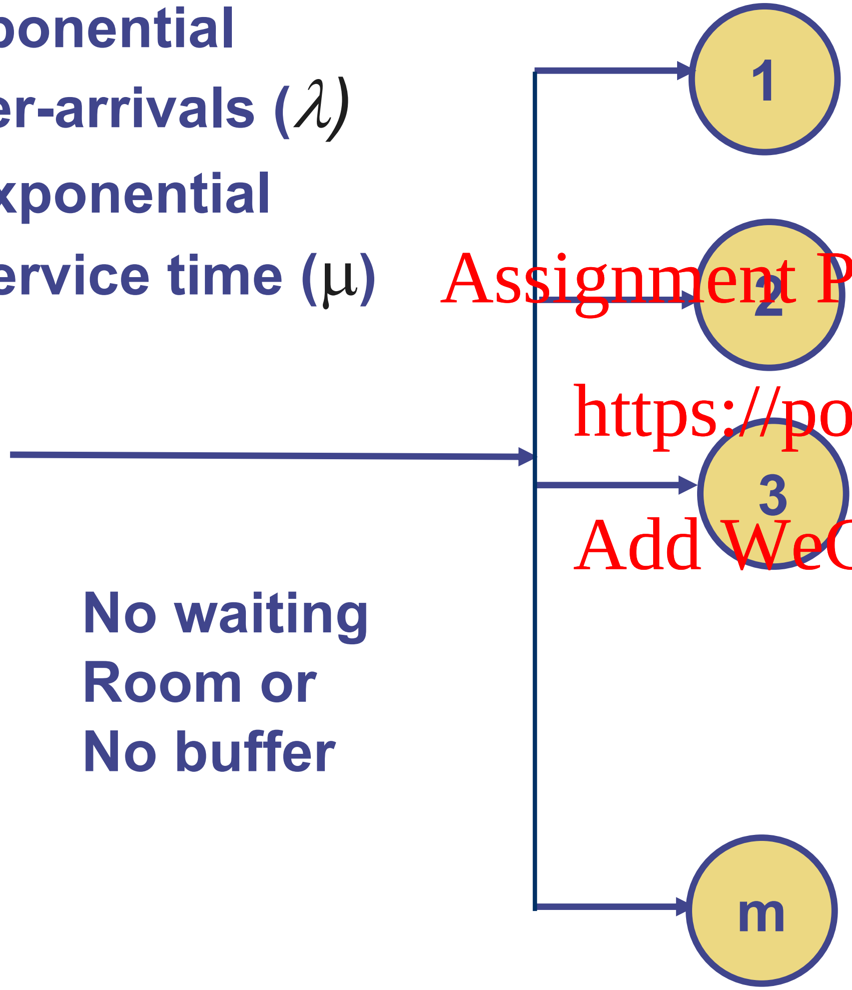
$$C(\rho, m) = \frac{\frac{(m\rho)^m}{m!}}{(1-\rho)\sum_{k=0}^{m-1}\frac{(m\rho)^k}{k!} + \frac{(m\rho)^m}{m!}}$$

Multi-server queues M/M/m/m with no waiting room

Exponential Inter-arrivals (λ)

Exponential Service time (μ)

No waiting Room or No buffer



m servers

An arrival can be served by any one of the *m* servers.

When a customer arrives

- If all servers are busy, it will *depart* from the system
- Otherwise, it will be served by one of the available servers

A call centre analogy of M/M/m/m queue

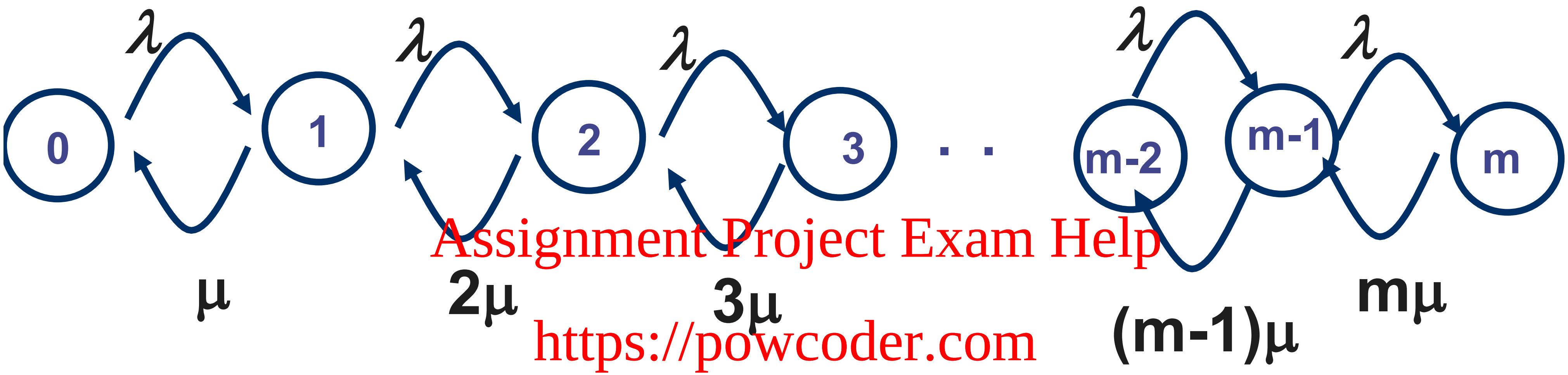
- Consider a call centre analogy
  - Calls are arriving according to Poisson distribution with rate λ
  - The length of each call is exponentially distributed with parameter μ
    - Mean length of a call is $1/\mu$

Arrivals

Call centre with *m* operators

If all *m* operators are busy, the call is dropped.

State transition for M/M/m/m

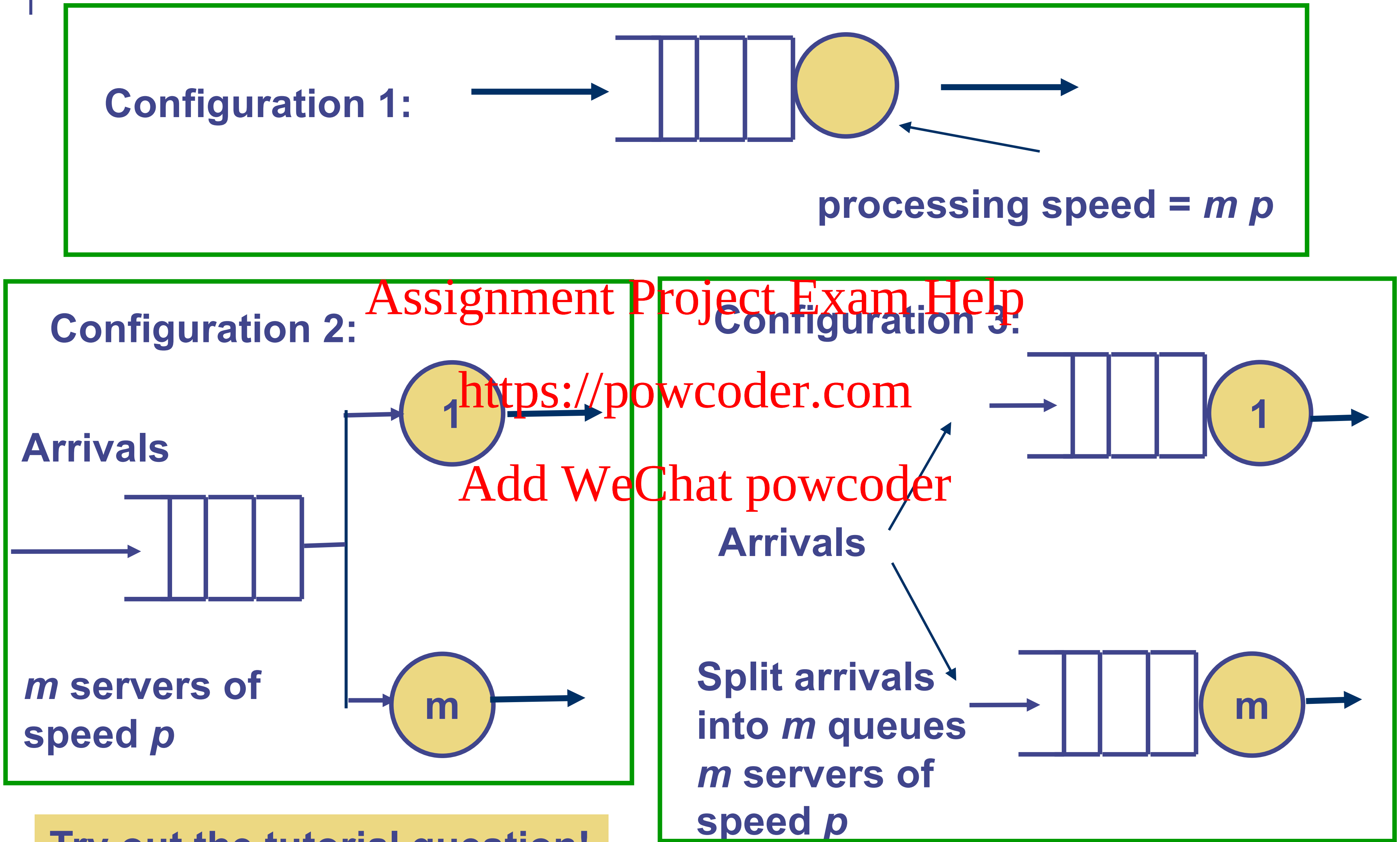


Probability that an arrival is blocked
= Probability that there are m customers in the system

$$P_m = \frac{\frac{\rho^m}{m!}}{\sum_{k=0}^{m} \frac{\rho^k}{k!}} \quad \textbf{where} \quad \rho = \frac{\lambda}{\mu}$$

"Erlang B formula"

What configuration has the best response time?



Try out the tutorial question!

References

- Recommended reading
  - Queues with Poisson arrival are discussed in
  - Bertsekas and Gallager, *Data Networks*, Sections 3.3 to 3.4.3
  - Note: I derived the formulas here using continuous Markov chain but Bertsekas and Gallager used discrete Markov chain
  - Mor Harchal-Balter. Chapters 13 and 14