Performance Analysis

Metrics, Analysis, and Examples

Performance Metrics and Analysis

- Metrics
  - Traditional and extensions
  - Sources of delay
  - Optimizing communication systems
  - Measuring systems
- Basic queueing theory
  - Distributions and processes
  - Single, memoryless queues

Performance Metrics

- Traditional metrics
  - End-to-end latency/RTT
    - Measures time delay
    - Across all layers of network
    - Often abbreviated to “latency” (even for RTT)
  - Bandwidth/throughput
    - Measures data sent per unit time
    - Across all layers of network

Performance Metrics

- Sources of delay
  - Latency: three main components
    - DMA from sending/to receiving host memory
    - Propagation delay in network
    - Queueing delay in routers
  - Overhead: also three main components
    - Data copy between buffers (e.g., into kernel memory)
    - Protocol (TCP, IP, etc.) processing
    - PIO to write description of frame
  - Note that overhead has fixed and per-byte costs

Performance Metrics

- Optimizing communication systems
  - Optimize the common case
    - Send/receive usually more important than connection setup/teardown
      - TCP header changes little between segments
      - Often only a few connections at end hosts
    - Minimize context switches
    - Minimize copying of data

Performance Metrics

- Optimizing communication systems
  - General rule of thumb
    - Most (80-90%) messages are short
    - Most data (80-90%) travel in long messages
  - Focus on bottlenecks
    - Reduce overhead to improve short message performance
    - Reduce number of copies to improve long message performance
  - Thus, CPU speed is often more important than network speed

Performance Metrics

- Optimizing communication systems
  - Maximize network utilization
    - Use large packets when possible
    - Fill delay-bandwidth pipe
  - Avoid timeouts
    - Set timers conservatively
    - Use “smarter” receiver (e.g., with selective ACK’s)
  - Avoid congestion rather than recovering from it

Performance Metrics

- Measuring communication systems
  - Latency
    - Measure RTT for 0-byte (or 1-byte) messages
    - Also report variability
  - Bandwidth
    - Measure RTT for range of long messages
    - Divide by number of bytes sent
    - Report as graph or as value in asymptotic limit
  - Overhead
    - Time multiple N-byte message send operations
    - Be careful of flow control and aggregation

Modeling and Analysis

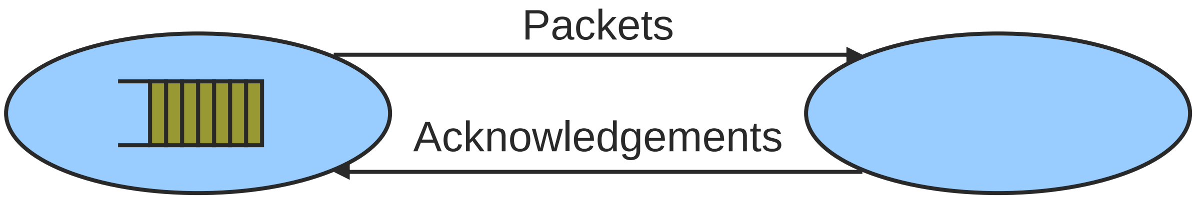
- Problem
  - The inputs to a system (i.e., number of packets and their arrival times) and the exact resource requirements of these packets cannot be predetermined in advance exactly
- But, we can probabilistically characterize these quantities
  - On average, 100 packets arrive per second
  - On average, packets are 500KB
- So, given a probabilistic characterization of these quantities
  - Can we draw some intelligent conclusions about the performance of the system

Delay

- Link delay consists of four components
  - Processing delay
    - From when the packet is correctly received to when it is put on the queue
  - Queueing delay
    - From when the packet is put on the queue to when it is ready to transmit
  - Transmission delay
    - From when the first bit is transmitted to when the last bit is transmitted
  - Propagation delay
    - From when the last bit is transmitted to when the last bit is received

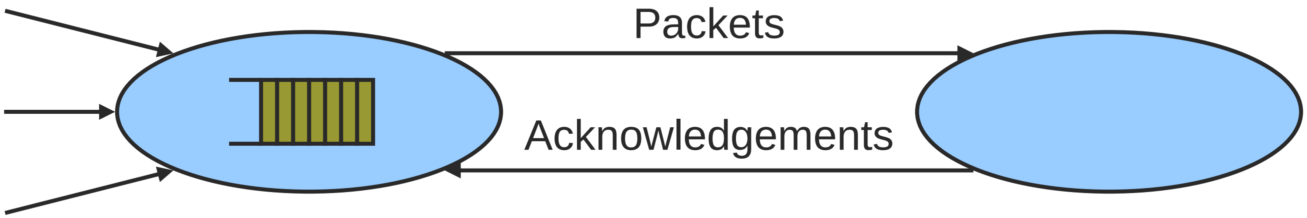
Delay Models

- Consider a data link using stop-and-wait ARQ
  - What is the throughput?
  - Given
    - *MSS* = packet payload size
    - *C* = raw link data rate
    - *RTT* = round trip time (for one bit)
    - *p* = probability a packet is successful



Delay Models

- Calculate the maximum throughput for stop-and-wait
  - *Max throughput = packetlength/(RTT + (packetlength/C))*
  - Could also multiply by *(payload/packetlength)* and *p = probability of correct reception*
- But what about the delay incurred?
  - There may be multiple bursty data sources



Basic Queueing Theory

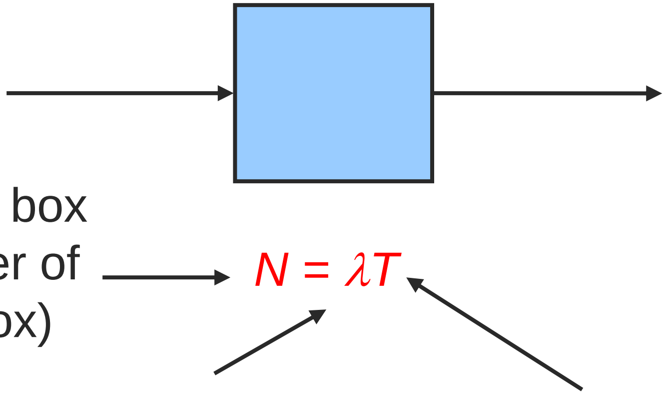
- Elementary notions
  - Things arrive at a queue according to some probability distribution
  - Things leave a queue according to a second probability distribution
  - Averaged over time
    - Things arriving and things leaving must be equal
    - Or the queue length will grow without bound
  - Convenient to express probability distributions as average rates

Little's Law

- Goal
  - Estimate relevant values
    - Average number of customers in the system
      - The number of customers either waiting in queue or receiving service
    - Average delay per customer
      - The time a customer spends waiting plus the service time
  - In terms of known values
    - Customer arrival rate
      - The number of customers entering the system per unit time
    - Customer service rate
      - The number of customers the system serves per unit time

Little's Law

- For any box with something steady flowing through it



$N = \lambda T$

Mean amount in box (average number of things in the box)

Mean arrival to the system (rate at which things enter the box)

Mean time spent in box (average time spent by a thing in the box)

Allows us to express the natural idea that crowded systems (large N) are associated with long customer delays

Little’s Law

$$N = \lambda T$$

Mean amount in box

Mean arrival

Mean time spent in box

- Example
  - Suppose you arrive at a busy restaurant in a major city
  - Some people are waiting in line, while other are already seated (i.e., being served)
  - You want to estimate how long you will have to wait to be seated if you join the end of the line
- Do you apply Little’s Law? If so
  - What is the box?
  - What is N?
  - What is λ?
  - What is T?

Little's Law

$$N = \lambda T$$

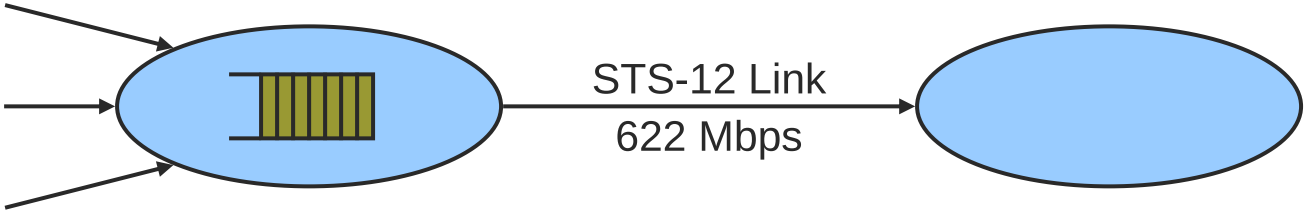
Mean amount in box

Mean arrival

Mean time spent in box

- Box
  - Include the people seated (i.e., being served)
  - Include the people waiting in line (i.e., in the queue)
- Let N = the number of people seated (say 150 seated + 50 in line)
- Let T = mean amount of time a person waits and then eats (say 90 min)
- Conclusion
  - Arrivals (and departures) = 200/90 = 2.22 persons per minute

Little’s Law

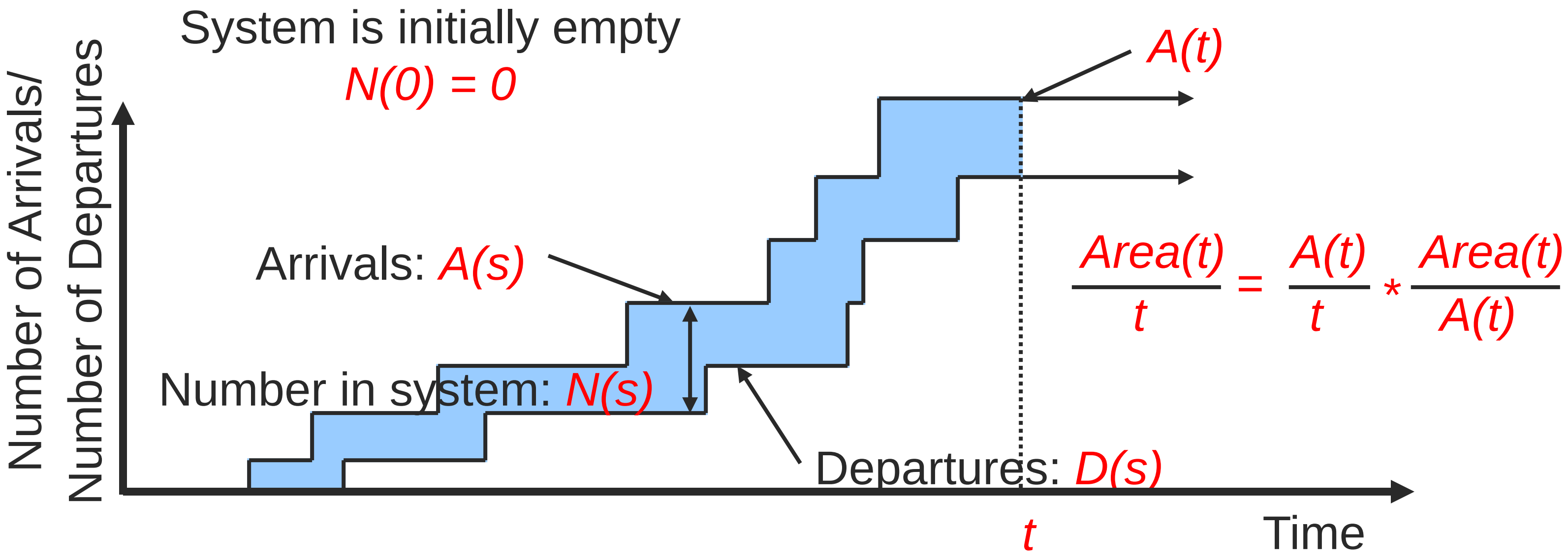


- Suppose data streams are multiplexed at an output link with speed 622 Mbps
- Question
  - If 200 50 B packets are queued on average, what is the average time in the system?
- Answer
  - $T = N/\lambda$
  - T = 200 * 50 * 8 / 622M
  - T = 0.128 ms

Little's Law

- Variables
  - $N(t)$ = number of customers in the system at time t
  - $A(t)$ = number of customers who arrived in the interval $[0,t]$
  - T_i = time spent in the system by the i^{th} customer
  - λ_t = average arrival rate over the interval $[0,t]$

Proof of Little's Law



System is initially empty

$N(0) = 0$

$$\frac{Area(t)}{t} = \frac{A(t)}{t} * \frac{Area(t)}{A(t)}$$

- But this is $N_t = \lambda_t t_t$
  - With time averaging over $[0,t]$
- Let t tend to infinity: $N = \lambda t$

- ○ $N(t)$ = number of customers
- ○ $A(t)$ = number of customers who arrived in the interval $[0,t]$
- ○ T_i = time spent in the system by the i^{th} customer
- ○ λ_t = average arrival rate over the interval $[0,t]$

Memoryless Distributions/ Poisson Arrivals

- Goal for easy analysis
  - Want processes (arrival, departure) to be independent of time
  - i.e., likelihood of arrival should depend neither on earlier nor on later arrivals
- In terms of probability distribution in time (defined for $t > 0$),

$$f(t) = \frac{f(t+\Delta t)}{\int_{\Delta t}^{\infty} f(t')\,dt'} \qquad \text{for all } \Delta t \geq 0$$

Memoryless Distributions/ Poisson Arrivals

solution is:

$$\boxed{f(t) = \lambda e^{-\lambda t}}$$

what is λ?

- it' s the rate of events
- note that the average time until the next event is

$$\int_0^\infty f(t)\,t\,dt = \left(te^{-\lambda t}\right]_0^\infty + \int_0^\infty e^{-\lambda t}dt$$

$$= \left(-\frac{1}{\lambda}e^{-\lambda t}\right]_0^\infty$$

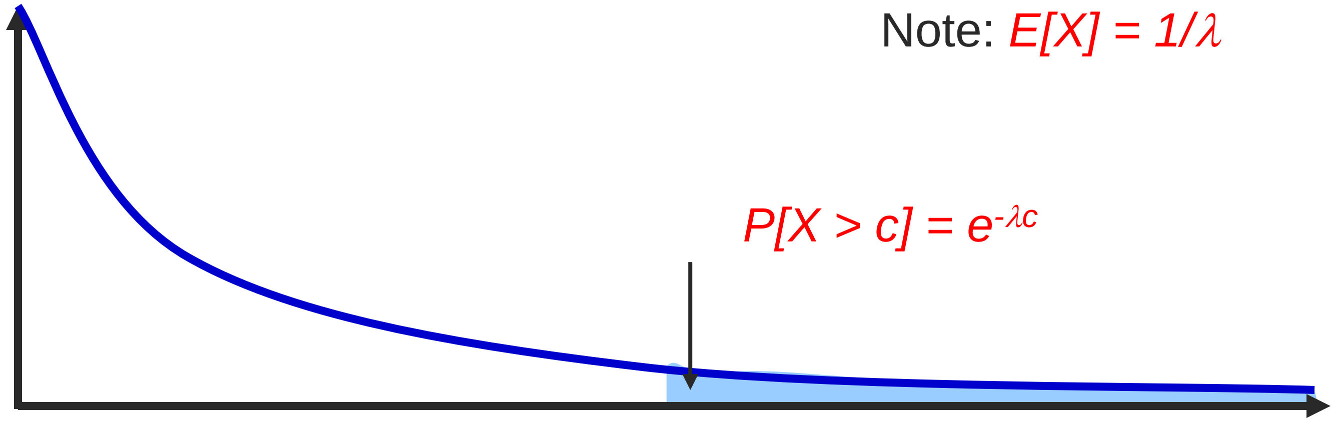
$$\boxed{= \frac{1}{\lambda}}$$

Plan

- Review exponential and Poisson probability distributions
- Discuss Poisson point processes and the M/M/1 queue model

Exponential Distribution

- A random variable X has an exponential distribution with parameter λ if it has a probability density function
    - $f(x) = \lambda e^{-\lambda x}$, for $x \geq 0$



Note: $E[X] = 1/\lambda$

$P[X > c] = e^{-\lambda c}$

Exponential Distribution

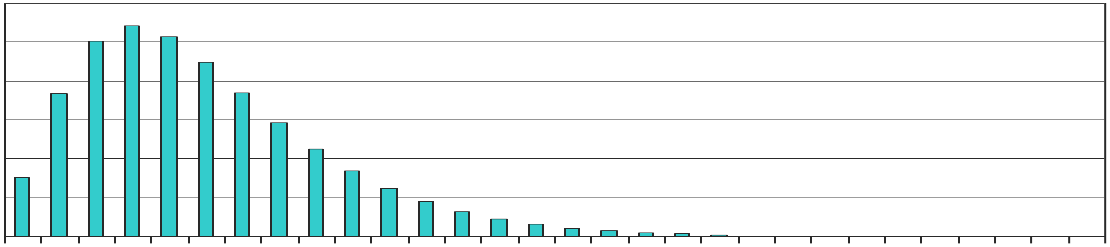
- Suppose a waiting time X is exponentially distributed with parameter $\lambda = 2$/sec
  - Mean wait time is ½ sec
- What is
  - $P[X>2]$?
  - $P[X>6]$?
  - $P[X>6 \mid X>4]$?

Exponential Distribution

- Remember: $\lambda = 2$
- $P[X>2]$
  - $= e^{-2\lambda} = 0.183$
- $P[X>6]$
  - $= e^{-6\lambda} = 6.14 \times 10^{-6}$
- $P[X>6|X>4]$
  - $= P[X>6,X>4]/P[X>4]$
  - $= P[X>6]/P[X>4]$
  - $= e^{-6\lambda}/e^{-4\lambda}$
  - $= e^{-2\lambda}$
  - $= 0.183!$
- Note: this demonstrates the memoryless property of exponential distributions

Poisson Distribution

- The random variable X has a Poisson distribution with mean λ, if for non-negative integers i:
  - $P[X = i] = (\lambda^i e^{-\lambda})/i!$
- Facts
  - $E[X] = \lambda$
  - If there are many independent events,
    - The k^{th} of which has probability p_k (which is small) and
    - λ = the sum of the p_k is moderate
    - Then the number of events that occur has approximately the Poisson distribution with mean λ



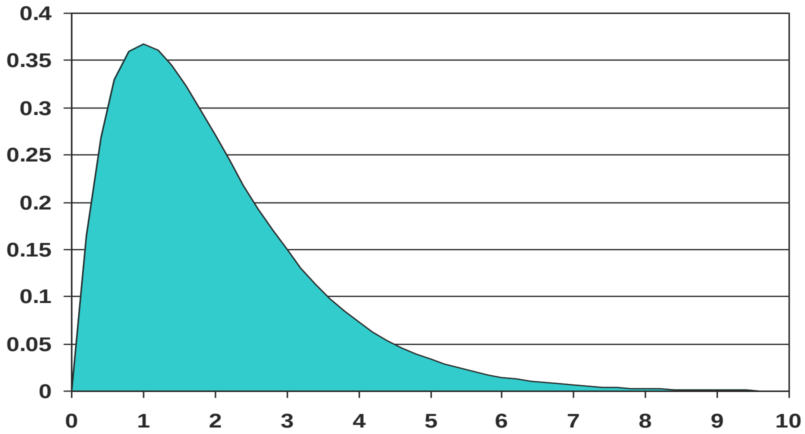
Poisson Distribution

- Example
  - Consider a CSMA/CD like scenario
  - There are 20 stations, each of which transmits in a slot with probability 0.03. What is the probability that exactly one transmits?

Poisson Distribution

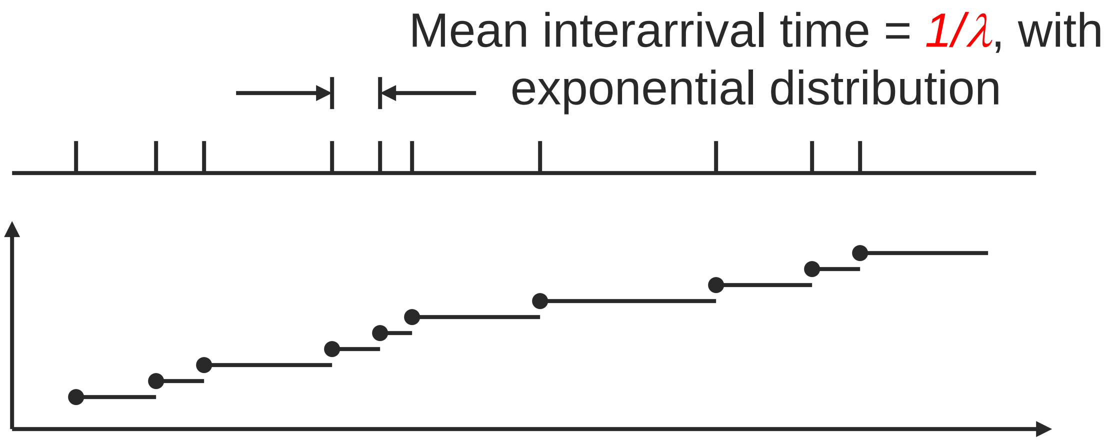
There are 20 stations, each of which transmits in a slot with probability 0.03. What is the probability that exactly one transmits?

- Exact answer
  - $20 * (0.03) * (1 – 0.03)^{19} = 0.3364$
- Poisson approximation
  - Use $P[X = i] = (\lambda^i e^{-\lambda})/i!$
  - With $i = 1$ and $\lambda = 20 * (0.03) = 0.6$
  - Approximate answer = $\lambda e^{\lambda} = 0.3393$



Poisson Point Process

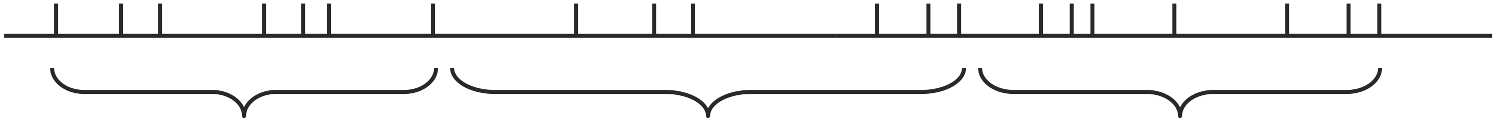
- Definition
  - A Poisson point process with parameter λ
    - A point process with interpoint times that are independent and exponentially distributed with parameter λ.



Mean interarrival time = $1/\lambda$, with exponential distribution

Poisson Point Process

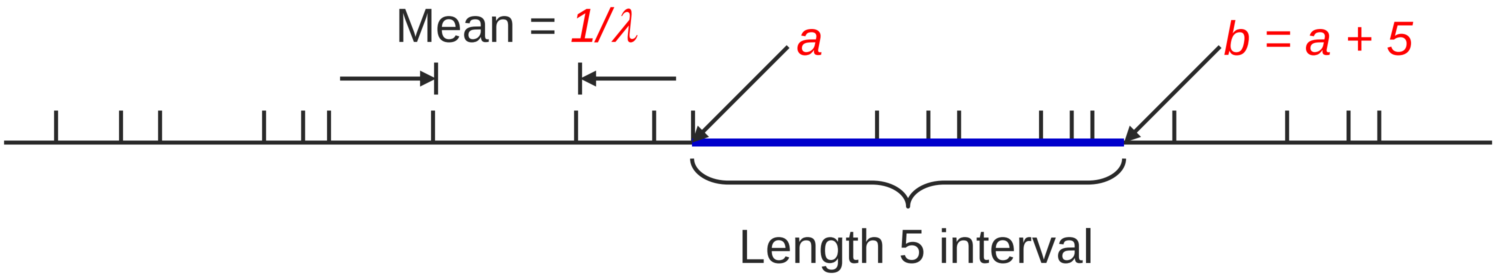
- Equivalently
  - The number of points in disjoint intervals are independent, and the number of points in an interval of length t has a Poisson distribution with mean λt



Shown are three disjoint intervals. For a Poisson point process, the number of points in each interval has a Poisson distribution.

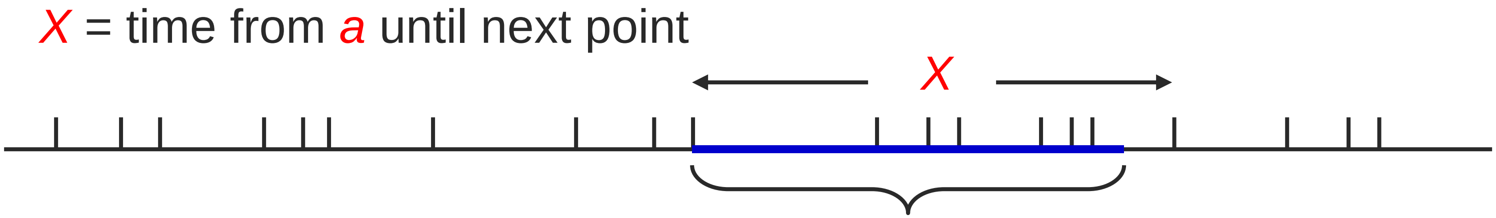
Poisson Point Process

- Exercise
  - Given a Poisson point process with rate $\lambda = 0.4$, what is the probability of NO arrivals in an interval of length 5?



Try to answer two ways, using two equivalent descriptions of a Poisson process

Poisson Point Process



N = number of points in interval

(Poisson with mean 5λ)

Given a Poisson point process with rate $\lambda = 0.4$, what is the probability of NO arrivals in an interval of length 5?

Solution 1: $P[X > 5] = e^{-5\lambda} = 0.1353$

Solution 2: $P[N = 0] = e^{-5\lambda} = 0.1353$

(remember: $P[N = i] = (5\lambda)^i * (e^{-5\lambda}) / i!$, for $i = 0$)

Simple Queueing Systems

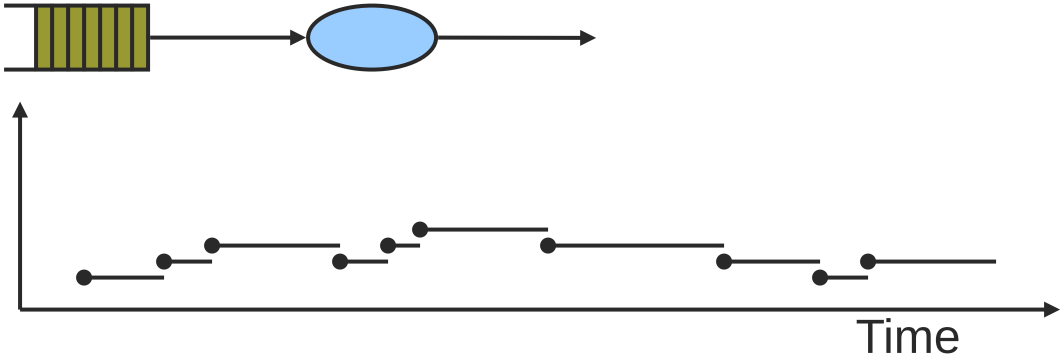
- Classify by
  - "arrival pattern/service pattern/number of servers"
    - Interarrival time probability density function
    - The service time probability density function
    - The number of servers
    - The queueing system
    - The amount of buffer space in the queues
  - Assumptions
    - Infinite number of customers

Simple Queueing Systems

- Terminology
    - M = Markov (exponential probability density)
    - D = deterministic (all have same value)
    - G = general (arbitrary probability density)
- Example
    - M/D/4
        - Markov arrival process
        - Deterministic service times
        - 4 servers

M/M/1 System

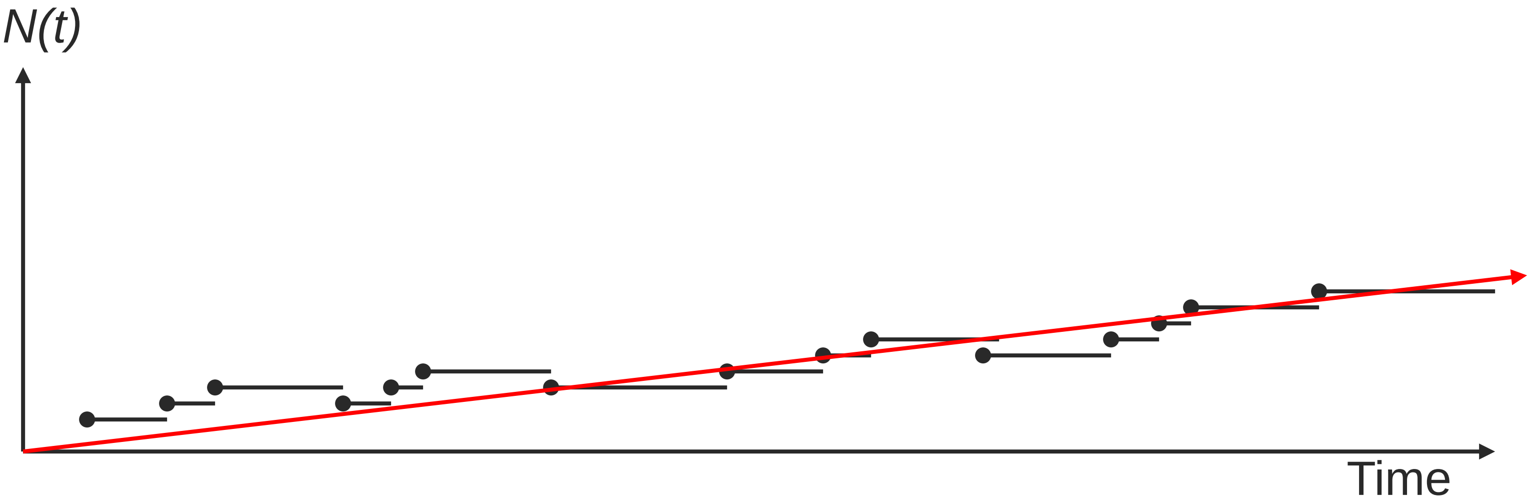
- Goal
  - Describe how the queue evolves over time as customers arrive and depart
- An M/M/1 system with arrival rate λ and departure rate μ has
  - Poisson arrival process, rate λ
  - Exponentially distributed service times, parameter μ
  - One server



$N(t)$ = number in system (system = queue + server)

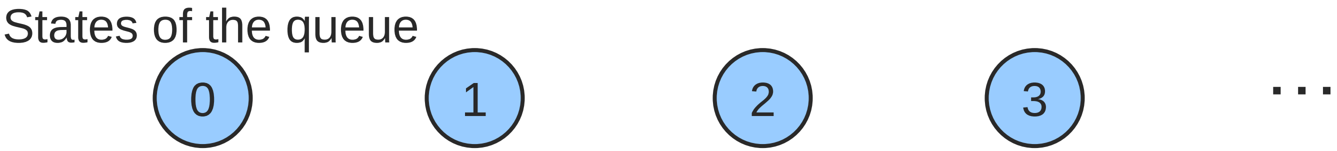
M/M/1 System

- If the arrival rate λ is greater then the departure rate μ
  - $N(t)$ drifts up at rate $\lambda - \mu$



M/M/1 System

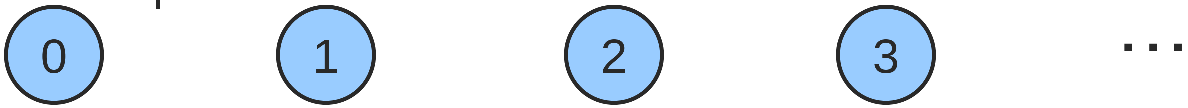
- On the other hand,
  - if $\lambda < \mu$, expect an equilibrium distribution.
- The state of the queue is completely described by the number of customers in the queue
  - Due to the memoryless property of exponential distributions, *N* is described by a single state transition diagram
  - *N* is a Markov process, meaning past and future are independent given present



M/M/1 System

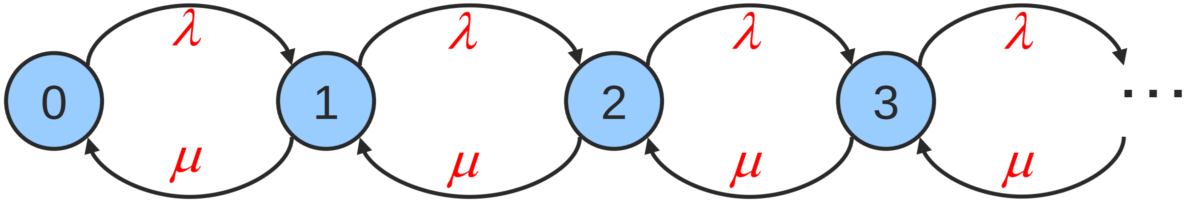
- N is a discrete random variable
  - p_k = probability that there are k customers in the queue
  - Equivalently,
    - p_k = probability that queue is in state k

States of the queue



M/M/1 System

- Goal
  - Find the steady state (long run) probabilities of the queue being in state *i*, *i* = 0, 1, 2, 3, …
- Transitions occur only when
  - A customer finishes service
  - A customer arrives
- Birth-death process
  - Transition from state *i* to state *i+1* on arrival
  - Transition from state *i* to state *i-1* on departure

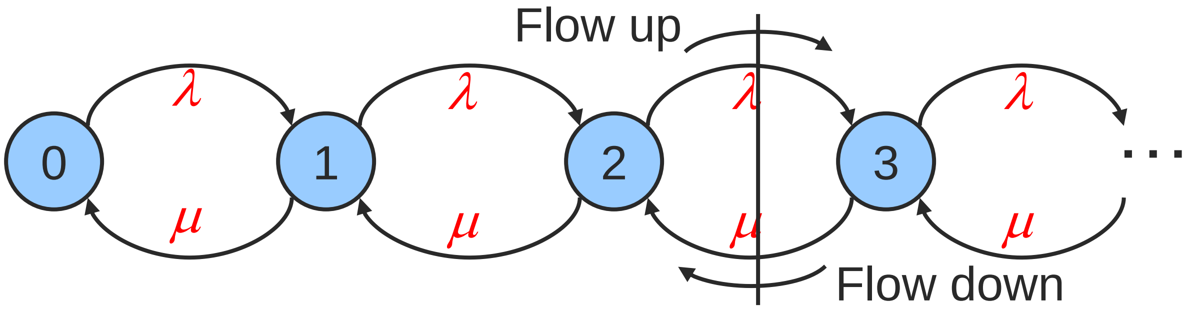


M/M/1: Transition rates

- If the queue is in state *i* with probability p_i
  - Then equivalently , the queue is in state *i* a fraction of p_i of the time
- The number of transitions/second out of state *i* onto state *i+1* is given by
  - (fraction of time queue is in state *i*) * (arrival rate)
  - $p_i * \lambda$
- The number of transitions/second out of state *i* onto state *i-1* is given by
  - (fraction of time queue is in state *i*) * (departure rate)
  - $p_i * \mu$

M/M/1: Steady State

- Claim
  - For the steady state to exist, # of transitions/sec from state *i* to state *i+1* must equal # of transitions/sec from state *i+1* to state *i*
- Result
  - Net flow across boundary between states must be zero
- Basic idea (not a real proof)
  - Otherwise, in the long run, the net flow of the system would always drift to the higher state with probability 1



M/M/1 System

- Given that we must balance flow across all boundaries,
  - $\lambda p_i = \mu p_{i+1}$ for all $i \geq 0$
- Balance Equations

$$\lambda p_0 = \mu p_1 \quad \Rightarrow \quad p_1 = (\lambda/\mu)\, p_0$$

$$\lambda p_1 = \mu p_2 \quad \Rightarrow \quad p_2 = (\lambda/\mu)\, p_1 \quad \Rightarrow \quad p_2 = (\lambda/\mu)^2\, p_0$$

$$\lambda p_2 = \mu p_3 \quad \Rightarrow \quad p_3 = (\lambda/\mu)\, p_2 \quad \Rightarrow \quad p_3 = (\lambda/\mu)^3\, p_0$$

$$\ldots \qquad \ldots \qquad \ldots$$

$$\lambda p_i = \mu p_{i+1} \quad \Rightarrow \quad p_{i+1} = (\lambda/\mu)\, p_i \quad \Rightarrow \quad p_{i+1} = (\lambda/\mu)^{i+1}\, p_0$$

M/M/1 System

- Problem
  - To solve the balance equations, we need one more equation:
    - $\sum_{i=0}^{\infty} p_i = 1$
- Thus
  - $p_k = (\lambda/\mu)^k \, p_0$ (1)
  - $\sum_{i=0}^{\infty} p_i = 1$ (2)
- Plugging 1 into 2, we get
  - $\sum_{i=0}^{\infty} p_0 * (\lambda/\mu)^i = 1$
- Result (for $\lambda < \mu$)
  - $p_0 = 1 / (\sum (\lambda/\mu)^i) = \ldots = 1 - \lambda/\mu$
  - $p_k = (\lambda/\mu)^k * (1 - \lambda/\mu)$

M/M/1 System

- So What?
  - We now know the probability that there are 0, 1, 2, 3, … customers in the queue (p_i)
- Define N_{avg}
  - = average # of customers in queue
  - = expected value of the # of customers in the queue
- N_{avg}
  - $= \sum_{\text{all possible \# of cust}} i * P[i \text{ customers}]$
  - $= \sum_{i=0}^{\infty} i * p_i = \sum_{i=0}^{\infty} (1 - \lambda/\mu) * (\lambda/\mu)^i * i$
  - $= (\lambda/\mu)/(1 - \lambda/\mu)$

M/M/1 System

- Define Q_{avg}
  - = average # of customers in waiting area of the queue
- Q_{avg}
  - $= \sum_{\text{all possible \# of cust in waiting area}} i * P[i \text{ customers in waiting area}]$
  - $= \sum_{i=0}^{\infty} i * P[i+1 \text{ customers in queue}]$
  - $= \sum_{i=0}^{\infty} (1 - \lambda/\mu) \; * (\lambda/\mu)^{i+1} * i$
  - $= (\lambda/\mu)/(1 - \lambda/\mu) - \lambda/\mu$
  - $= N_{avg} - \lambda/\mu$

M/M/1 System - Utilization

- Utilization
  - The fraction of time the server is busy
  - = *P[server is busy]*
  - = *1 – P[server is NOT busy]*
  - = *1 – P[zero customers in queue]*
  - = $1 - p_0$
  - = $1 - (1 - \lambda/\mu)$
  - = λ/μ
- Since utilization cannot be greater then 1,
  - Utilization = $min(1.0, \lambda/\mu)$

M/M/1 System - Utilization

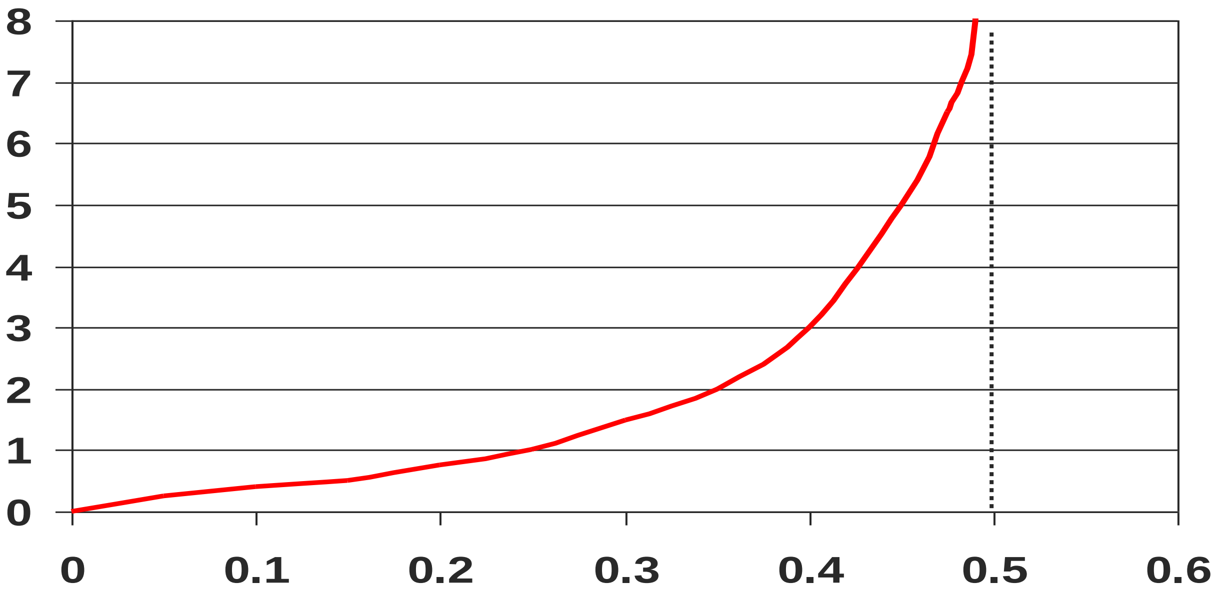
- Utilization example
  - Packets arrive for transmission at an average (Poisson) rate of 0.1 packets/sec
  - Each packet requires 2 seconds to transmit on average (exponentially distributed)
  - *What are* $N_{avg,}$ Q_{avg} *and* ρ*?*

M/M/1 System - Utilization

- Utilization example
  - Packets arrive for transmission at an average (Poisson) rate of 0.1 packets/sec
  - Each packet requires 2 seconds to transmit on average (exponentially distributed)
  - $N_{avg} = (\lambda/\mu)/(1 - \lambda/\mu) = 0.1*2 /(1 – 0.1*2) = 0.25$
  - $Q_{avg} = N_{avg} - \lambda/\mu = 0.25 – 0.1*2 = 0.05$
  - $\rho = \lambda/\mu = 0.2$

M/M/1 System - Utilization

- Intuitively, as the number of packets arriving per second (λ) increases, the number of packets in the queue should increase



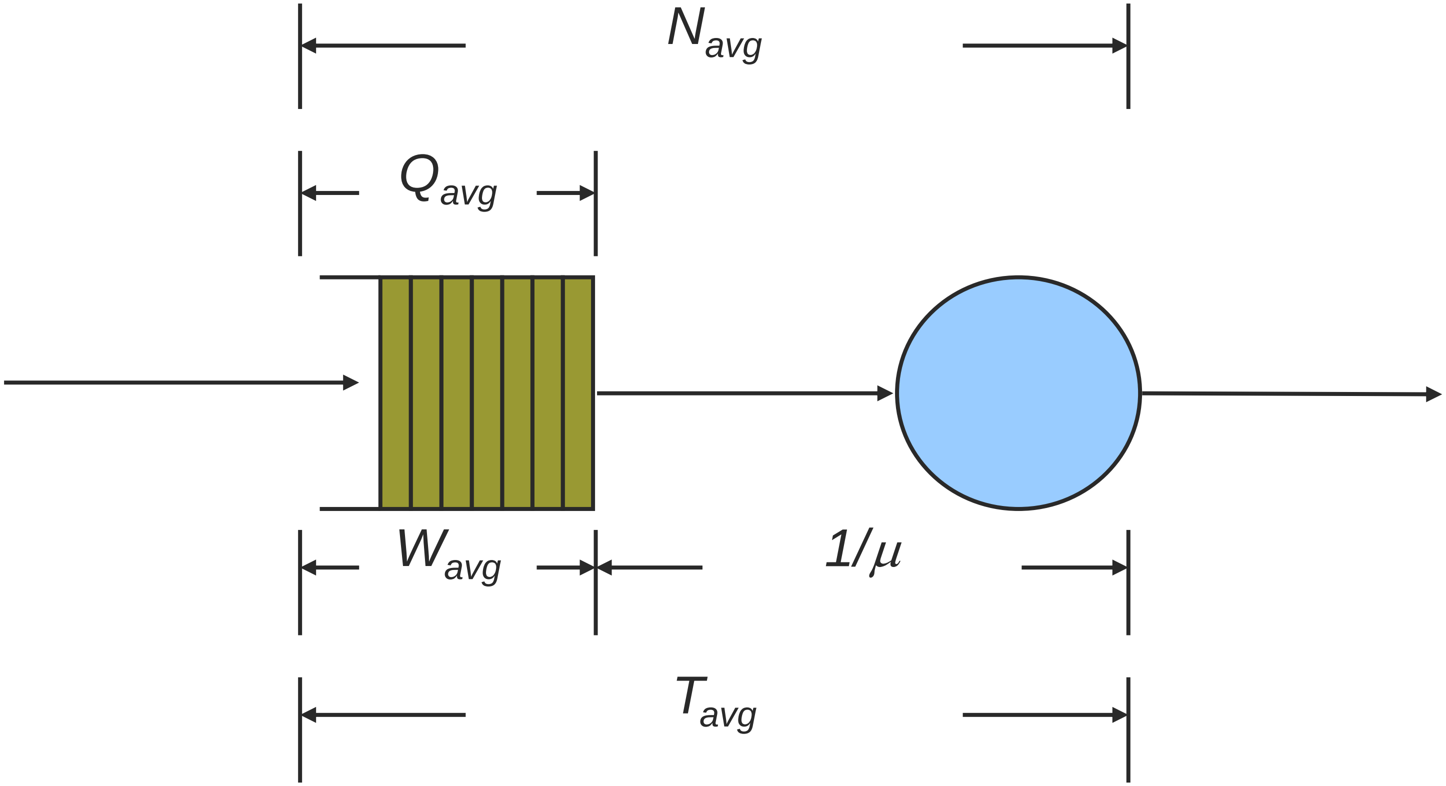
M/M/1 System - Utilization

- Normalized Traffic Parameter (ρ)
  - Note that N_{avg} and Q_{avg} only depend on the ratio λ/μ
  - Define ρ
    - = (avg arrival rate * avg service time)
    - = $\lambda * 1/\mu = \lambda/\mu$
  - Intuitively, if we scale both arrival rate and service time by a constant factor, N_{avg} and Q_{avg} should remain the same
  - Note
    - If $\lambda > \mu$ (i.e. $\lambda/\mu > 1$), then more packets are arriving per second than can be serviced
    - Thus, N_{avg} and Q_{avg} are unbounded when $\rho \geq 1$!

M/M/1 System – Time Delays

- Given $\{p_0, p_1, p_2, \ldots\}$, we can derive N_{avg} and Q_{avg}
- We may also want to know the following
  - T_{avg} = average time from when a packet arrives until it completes transmission
  - W_{avg} = average time from when a packet arrives until it starts transmission

M/M/1 System – Time Delays



M/M/1 System – Little's Law

- Now we can use Little's Law to relate N_{avg} and Q_{avg} to T_{avg} and W_{avg}
  - $N_{avg} = \lambda T_{avg}$ $\Rightarrow T_{avg} = N_{avg}/\lambda$
  - $Q_{avg} = \lambda W_{avg}$ $\Rightarrow W_{avg} = Q_{avg}/\lambda$

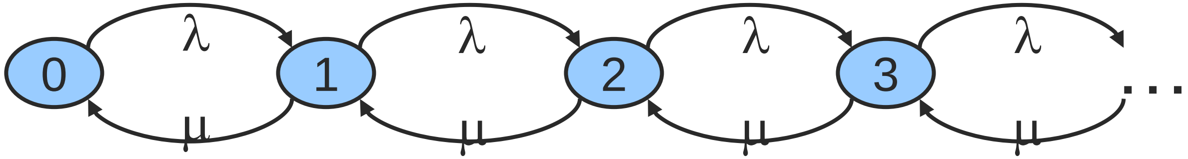
  - Also note: $W_{avg} + 1/\mu = T_{avg}$

M/M/1 System

- Packets arrive with the following parameters
  - λ = 2 packets per second
  - $1/\mu$ = ¼ sec per packets
  - ρ = 0.5
- Utilization = $\rho = \lambda/\mu = 2/4 = 0.5$
- $N_{avg} = \rho/(1 - \rho) = 0.5/1\text{-}0.5 = 1$ packet
  - $\Rightarrow T_{avg} = N_{avg}/\lambda = ½ = 0.5$ sec
- $Q_{avg} = N_{avg} - \rho = 1 – 0.5 = 0.5$
  - $\Rightarrow W_{avg} = Q_{avg}/\lambda = 0.5/2 = 0.25$ sec

M/M/1 System - Summary

1. Draw state diagram



2. Write down balance equations

flow “up” = flow “down”

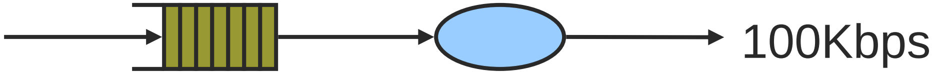
3. Solve balance equations using

$\sum_{i=0}^{\infty} p_i = 1$ for $\{p_0, p_1, p_2, \ldots\}$

4. Compute N_{avg} and Q_{avg} from $\{p_i\}$

5. Compute T_{avg} and W_{avg} using Little’s Theorem

M/M/1 System - Example



- Packets arrive ant an output link according to a Poisson process
  - The mean total data rate is 80Kbps (including headers)
  - The mean packet length is 1500
  - The link speed is 100Kbps
- Questions
  - What assumptions can we make to fit this situation to the M/M/1 model?
  - Under these assumptions, what is the mean time needed for queueing and transmission of a packet?

M/M/1 System - Example

- Answer Part 1:
  - “Customers”
    - Packets
  - “Server”
    - The transmitter
  - Service times
    - The transmission times
  - Packets sizes
    - Variable lengths, with a exponential distribution
    - Packet lengths are independent of each other and independent of arrival time

M/M/1 System - Example

- Remember
  - The mean total data rate is 80Kbps
  - The mean packet length is 1500
  - The link speed is 100Kbps
- Answer Part 2: Find λ, μ and T
  - Need to convert from bit rates to packet rates
    - λ = 80Kbps/12Kb = 6.66 packets/sec
    - μ = 100 Kbps/12Kb = 8.33 packets/sec
  - So, T = mean time for queueing and transmission
    - $T = 1/(\mu - \lambda) = 1/1.67 = 0.6$ *sec*

M/M/1 System - Example

- Also
  - The mean transmission time is
    - $1/\mu = 0.12$ sec,
  - So the mean time spent in queue is
    - $W = T - 1/\mu = 0.6 – 0.12 = 0.48$sec
  - The mean number of packets is
    - $N = \rho/(1 - \rho) = 0.8/(1 – 0.8) = 4$ packets

M/M/1 System in Practice

- The assumptions we made are often not realistic
- We still get the correct qualitative behavior
- Simple formulas for predictive delay are useful for provisioning resources in a network and setting controls
- Real traffic seems to have bursty behavior on multiple time scales
    - This is not true for Poisson processes