CS144
An Introduction to Computer Networks

Packet Switching

Queue models

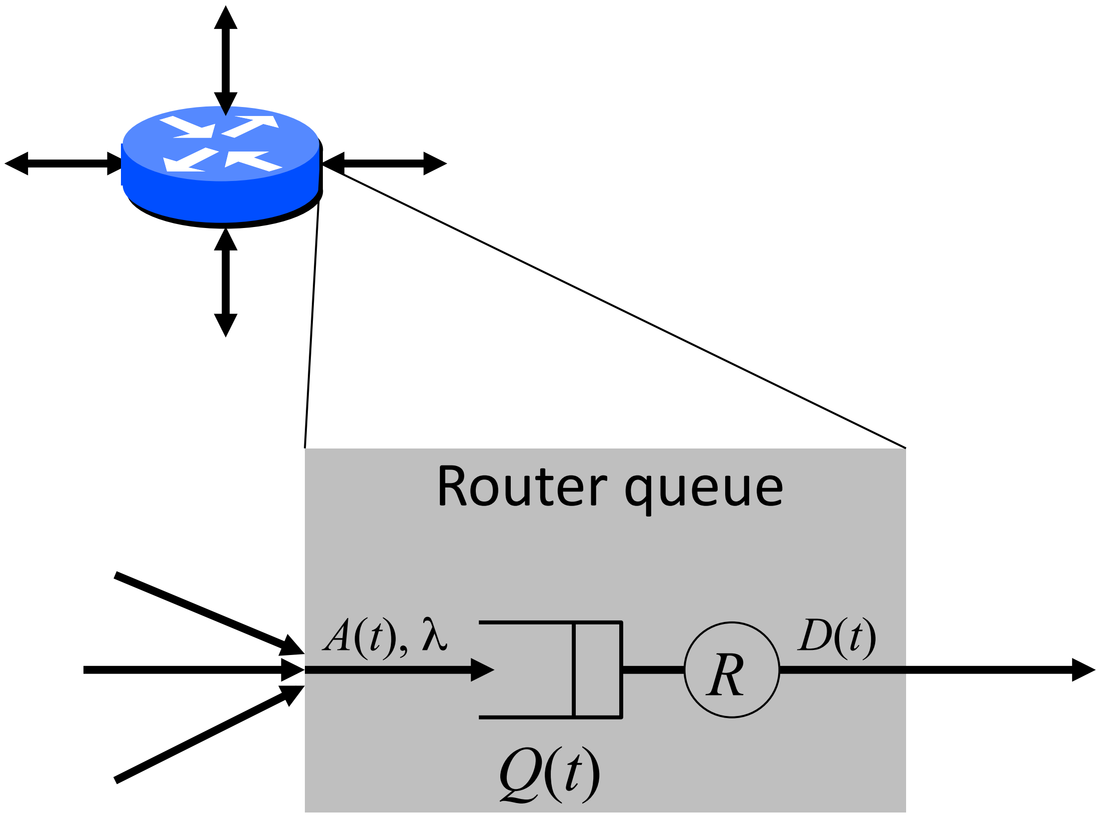
Nick McKeown
Professor of Electrical Engineering
and Computer Science, Stanford University

Outline

1. Simple deterministic queue model
2. Small packets reduce end to end delay
3. Statistical multiplexing

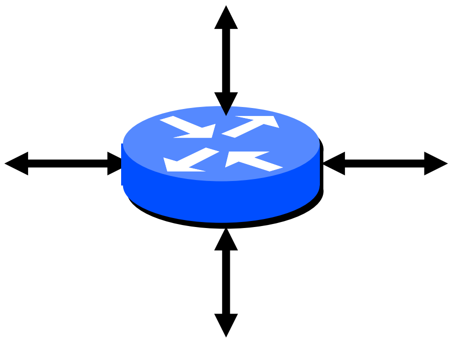
HIDE

Simple model of a router queue



Properties of $A(t)$, $D(t)$:
- $A(t)$, $D(t)$ are non-decreasing
- $A(t) >= D(t)$

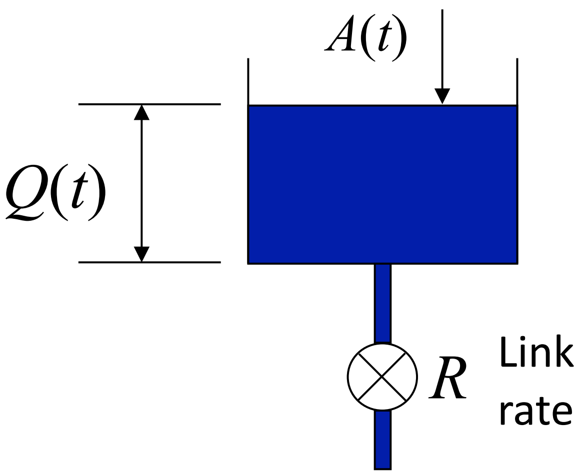
Simple model of a router queue



HIDE

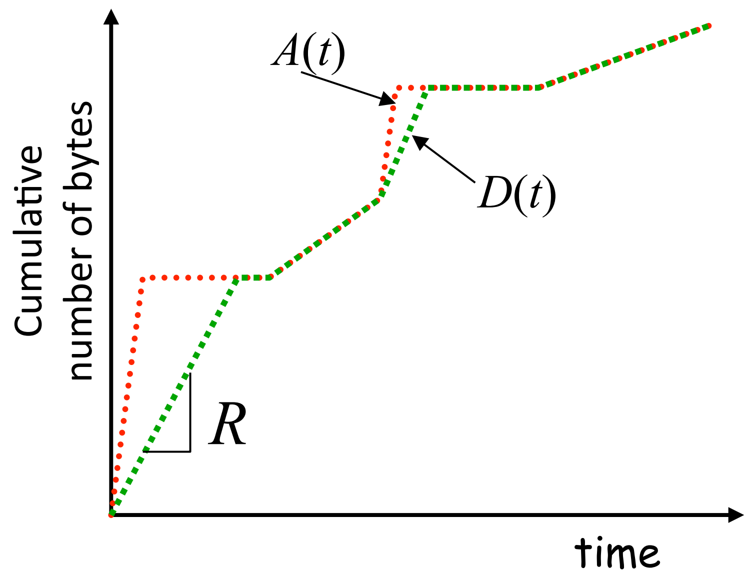
Simple model of a queue

Cumulative number of bytes arrived up until time t.



$D(t)$

Cumulative number of bytes departed up until time t.

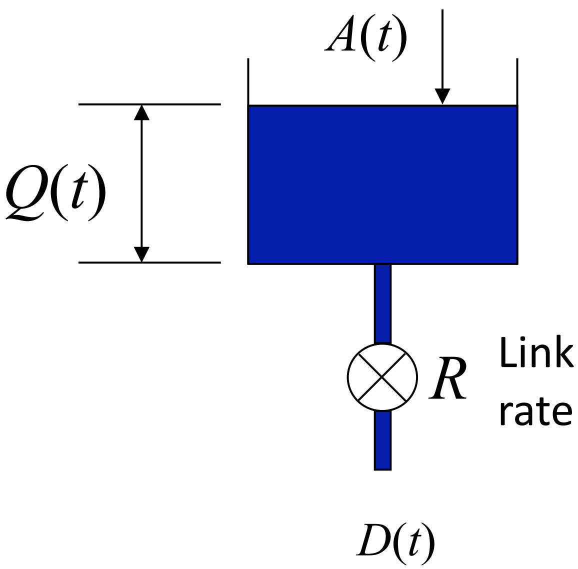


Properties of $A(t)$, $D(t)$:

- $A(t)$, $D(t)$ are non-decreasing
- $A(t) >= D(t)$

Simple model of a queue

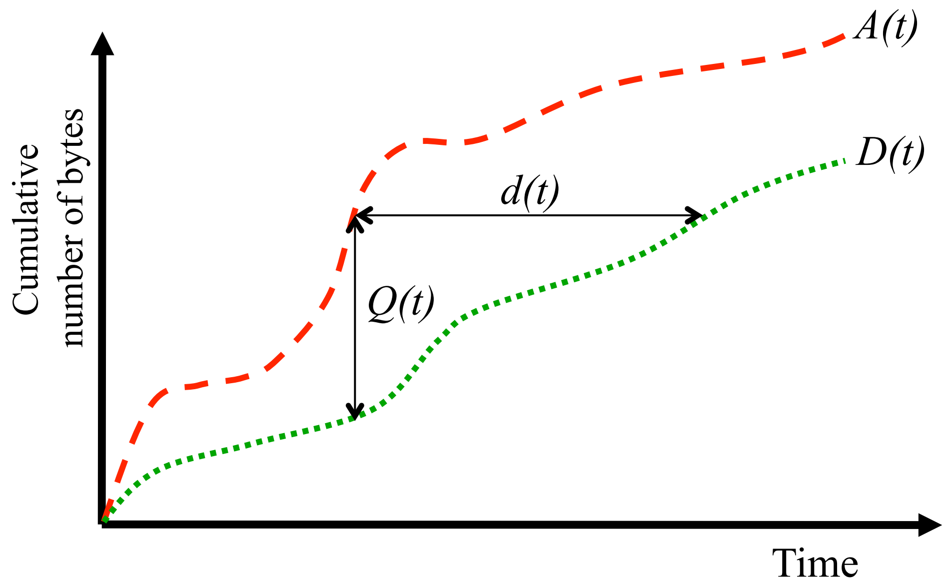
Cumulative number of bytes arrived up until time t.



Cumulative number of bytes departed up until time t.

HIDE

Simple model of a queue



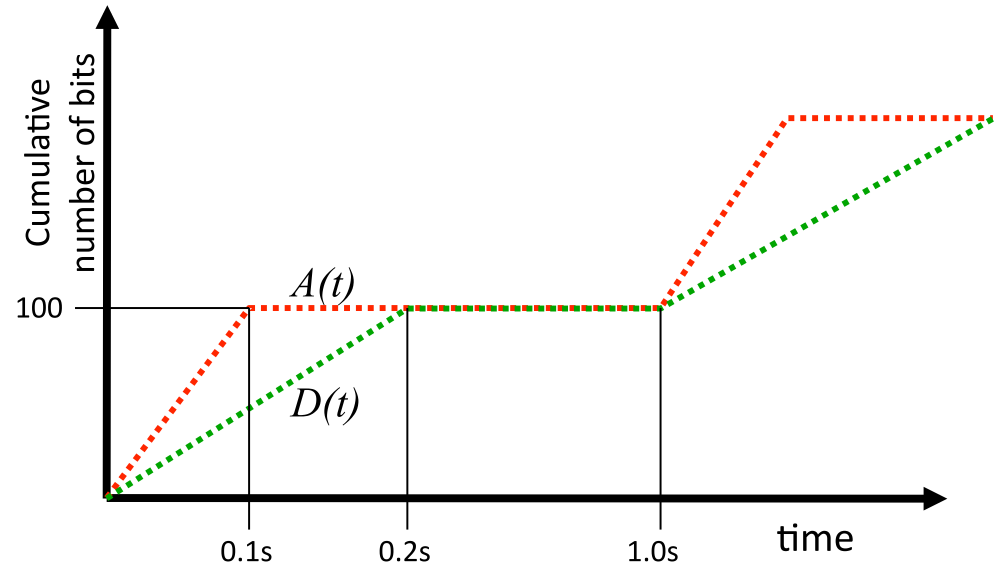
Queue occupancy: $Q(t) = A(t) - D(t)$.

Queueing delay, $d(t)$, is the time spent in the queue by a byte that arrived at time t, assuming the queue is served first-come-first-served (FCFS).

Simple model of a queue

Example

Every second, a 100 bit packet arrives to a queue at rate 1000b/s. The maximum departure rate is 500b/s. What is the average occupancy of the queue?



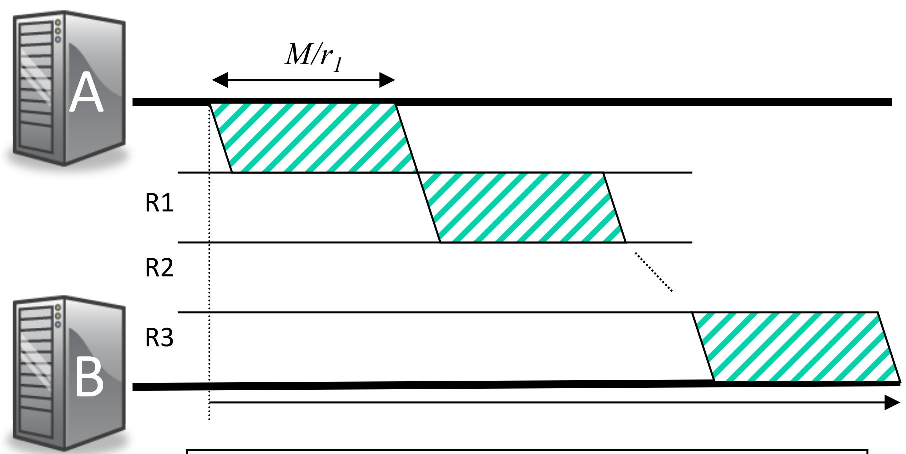
Solution: During each repeating 1s cycle, the queue fills at rate 500b/s for 0.1s, then drains at rate 500b/s for 0.1s. Over the first 0.2s, the average queue occupancy is therefore $0.5 \times (0.1 \times 500) = 25$ bits.
The queue is empty for 0.8s every cycle, and so average queue occupancy: $\bar{Q}(t) = (0.2 \times 25) + (0.8 \times 0) = 5$

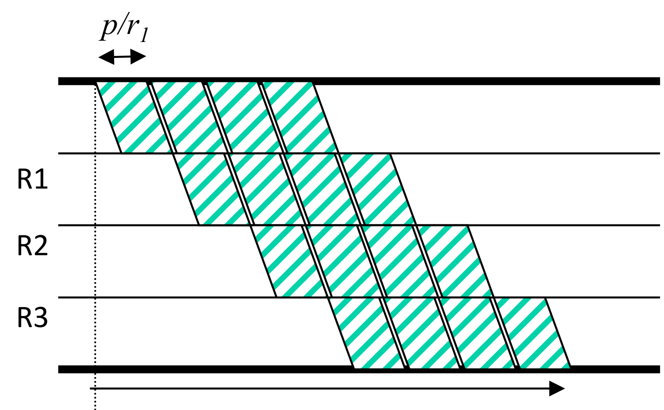
Outline

1. Simple deterministic queue model
2. Small packets reduce end to end delay
3. Statistical multiplexing

Packet Switching

Why not send the entire message in one packet?





$$\text{End-to-end delay},\ t = \sum_i \left(\frac{M}{r_i} + \frac{l_i}{c} \right)$$

$$\text{End-to-end delay},\ t = \sum_i \left(\frac{p}{r_i} + \frac{l_i}{c} \right) + \left(\frac{M}{p} - 1 \right) \frac{p}{r_{\min}}$$

Breaking message into packets allows parallel transmission across all links, reducing end to end latency.

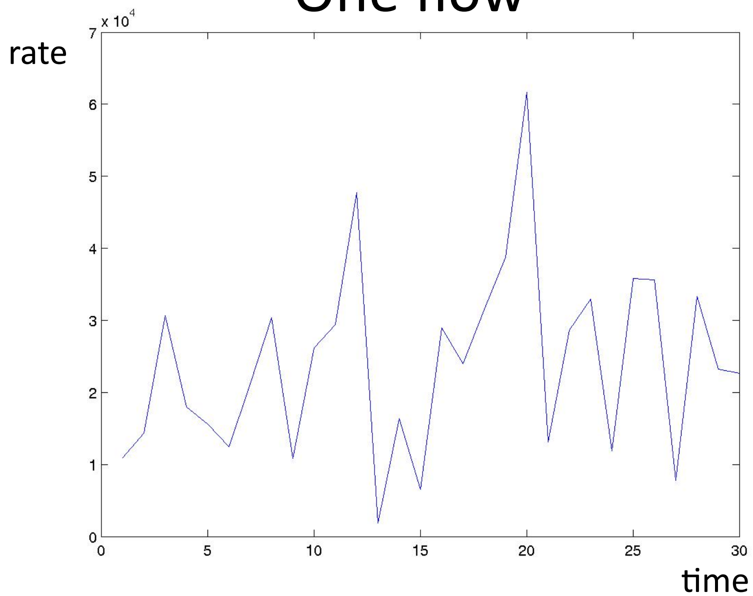
Outline

1. Simple deterministic queue model
2. Small packets reduce end to end delay
3. Statistical multiplexing

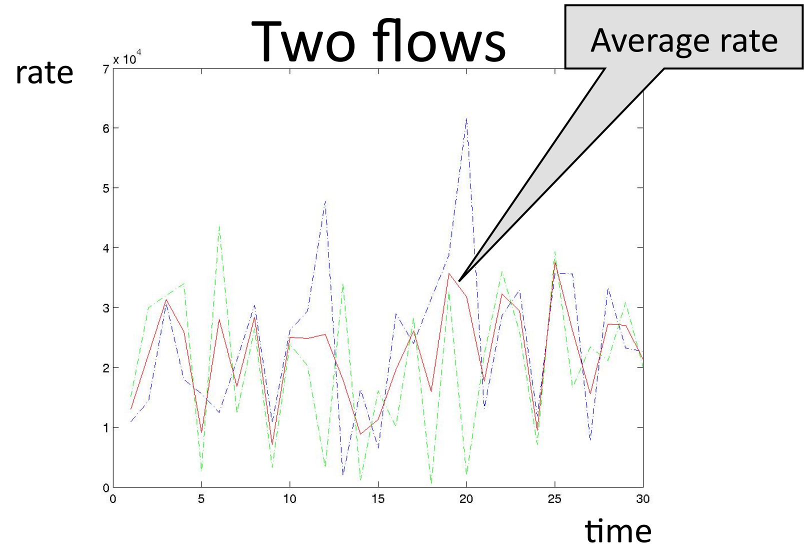
Statistical Multiplexing

Basic idea

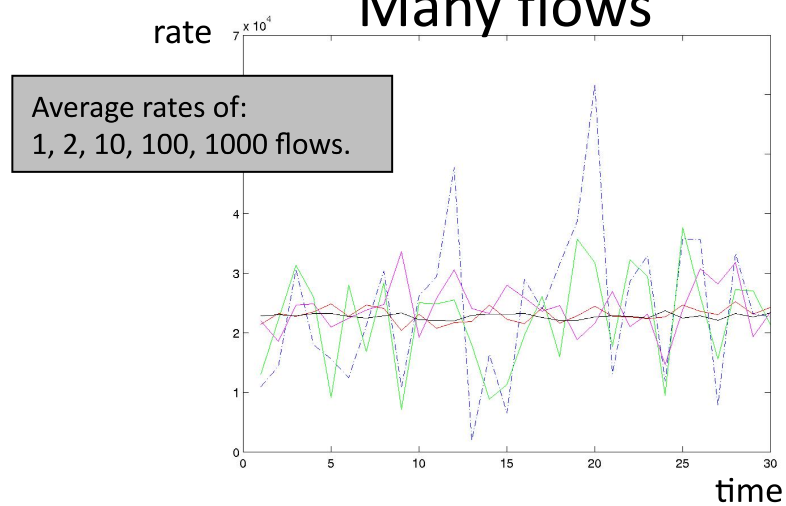
One flow



Two flows



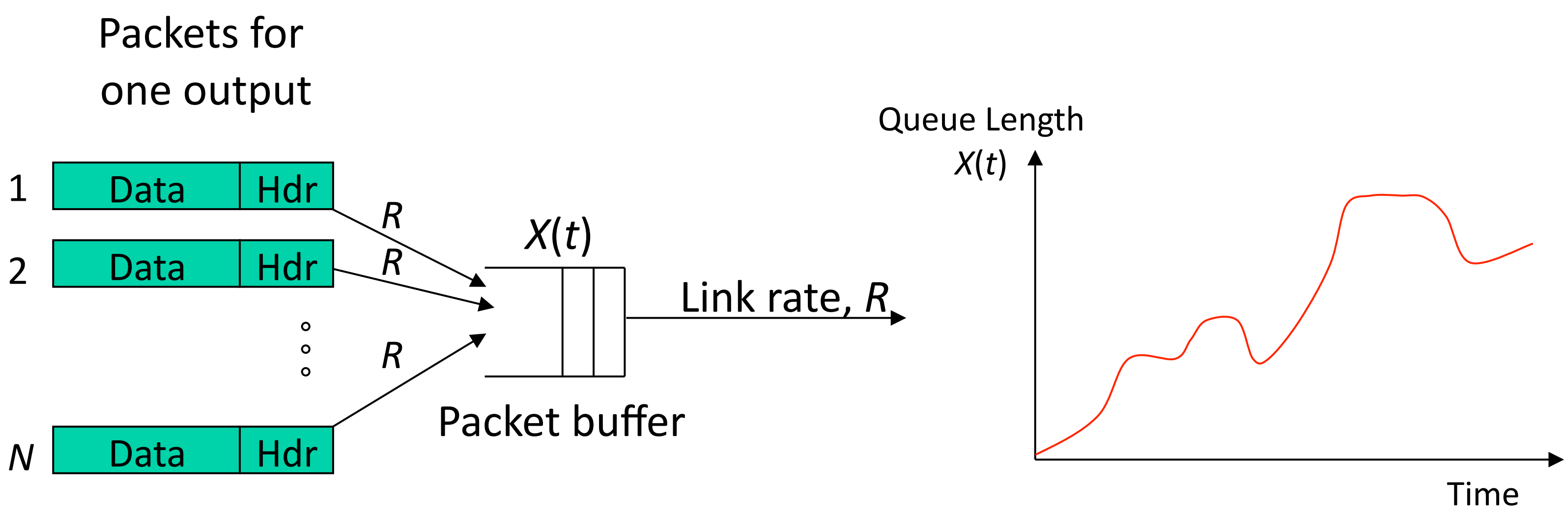
Many flows



HIDE

Packet Switching

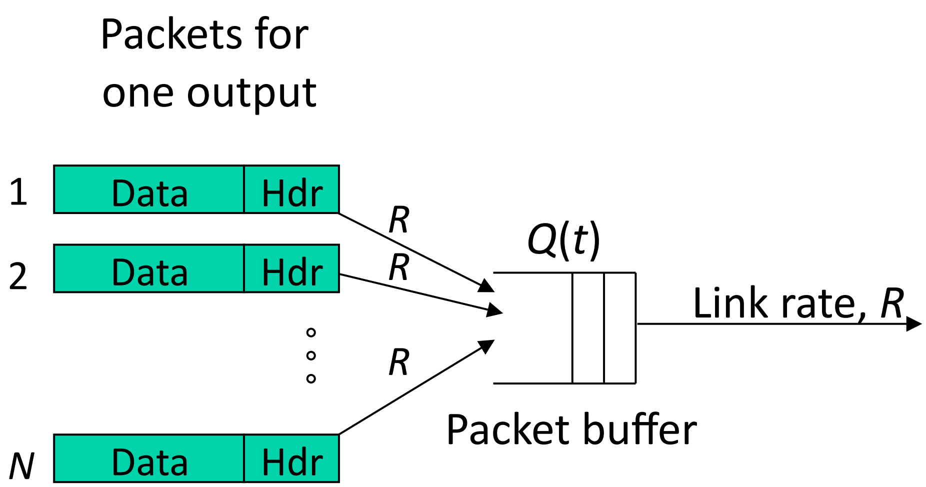
Statistical Multiplexing



- Because the buffer absorbs temporary bursts, the egress link need not operate at rate $N.R$.
- But the buffer has finite size, B, so losses will occur.

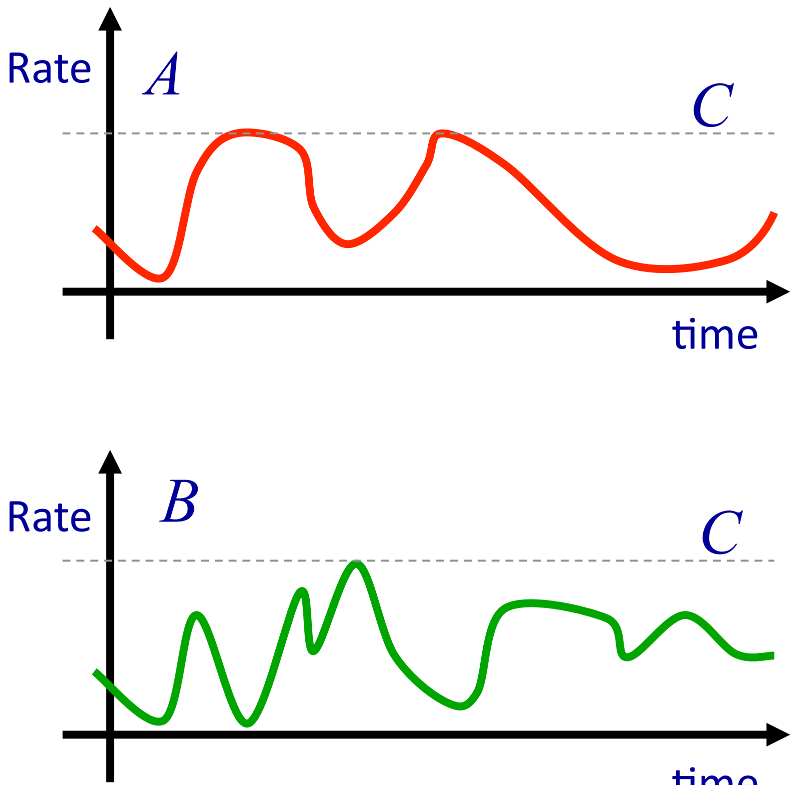
Packet Switching

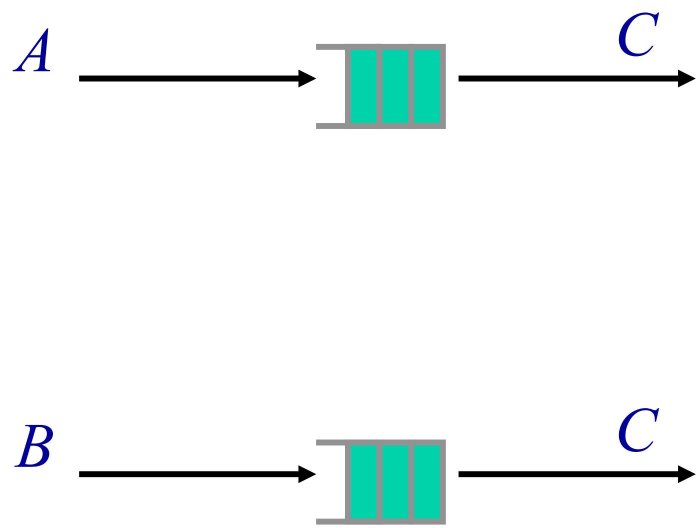
Statistical Multiplexing



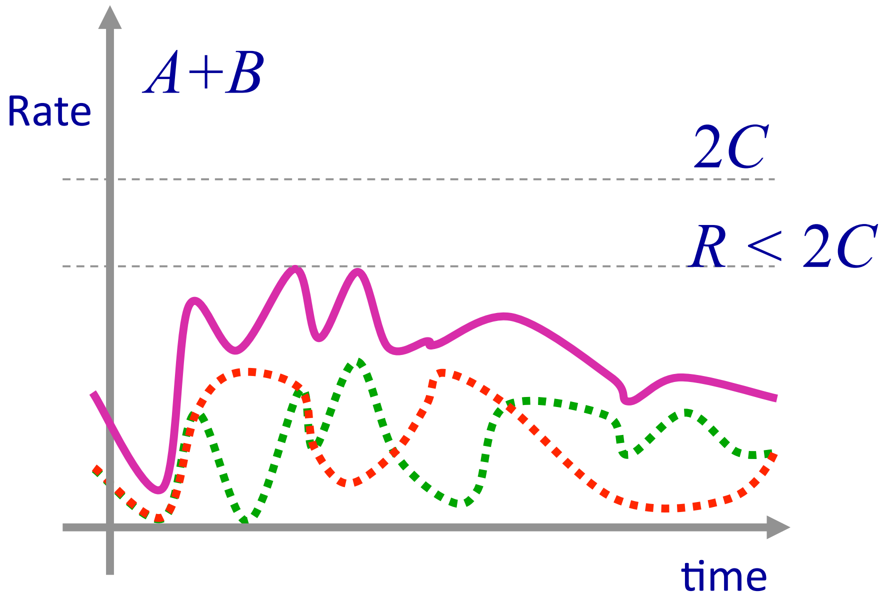
- Statistical multiplexing means the egress link need not run at rate *NR*.
- The buffer absorbs brief periods when the aggregate rate exceeds *R*.
- Because the buffer has finite size losses can occur.

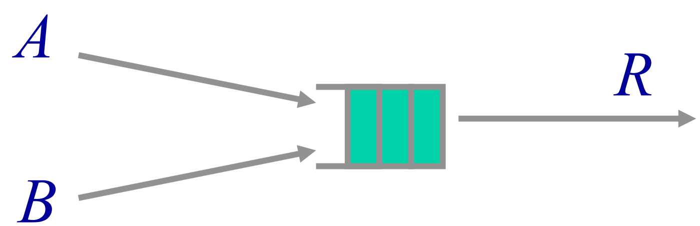
Statistical Multiplexing





Statistical Multiplexing Gain





Statistical multiplexing gain = $2C/R$

Summary

Often, we can use a simple deterministic model of a queue to understand the packet dynamics in a network.

We break messages into packets because it lets us pipeline the transfer, and reduce end to end delay.

Statistical multiplexing lets us carry many flows efficiently on a single link.

<end>