CS519: Computer Networks

Lecture 1: Jan 24, 2004

Intro to Computer Networking

Lets start at the beginning…

CS419

- What is a network for?
  - To allow two or more endpoints to communicate
- What is a network?
  - Nodes connected by links

Lets start at the beginning…

CS419

- Is this a network?

Lets start at the beginning…

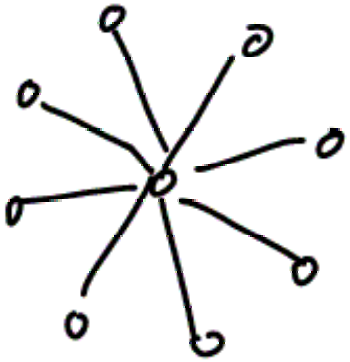
CS419

- Is this a network?

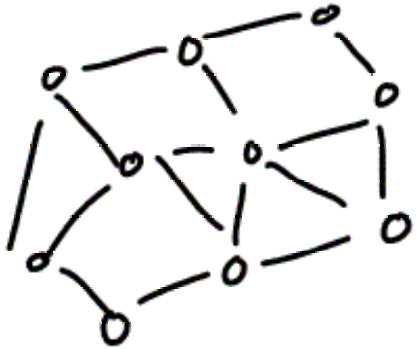
- Of course it is!
  - Just not very interesting

Other “networks” (network topologies)

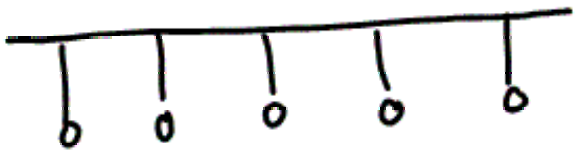
CS419



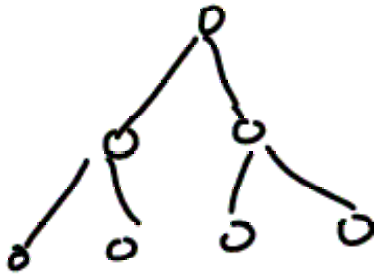
Star, or
Hub and spoke



Mesh



Broadcast



Hierarchical

What is a data network?

CS419

- The answer is NOT “a network that carries data”
  - Cause you can send “data” (e.g. a fax) over the “voice network”
- “Data network” is often a euphemism for “packet network”
  - And “voice network” is often a euphemism for “circuit network”

Packet network versus circuit network

CS419

- Historically, a circuit network was a network that literally established a physical wired connection between two points
  - With relays, plus amplifiers and stuff
- Before computers, this was the only way to do networks

Packet network versus circuit network

CS419

- But these days voice is modulated and digitized in numerous ways as it works through the network
  - Very few physical circuits
- So nowadays we consider a circuit network one that appears to establish a fixed “pipe” (amount of bandwidth) between two points

Types of circuits

CS419

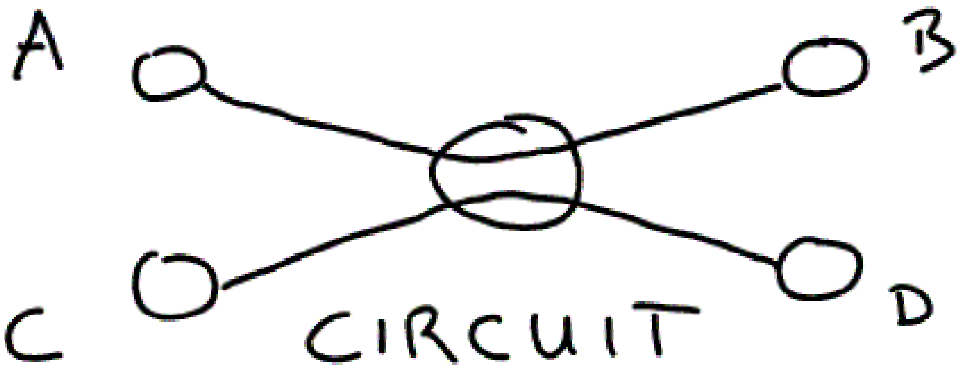
- Synchronous time-division multiplexing (STDM)
  - Each circuit is given a slice of time
- Frequency-division multiplexing (FDM)
  - Each circuit is given a transmission frequency

Packet network versus circuit network

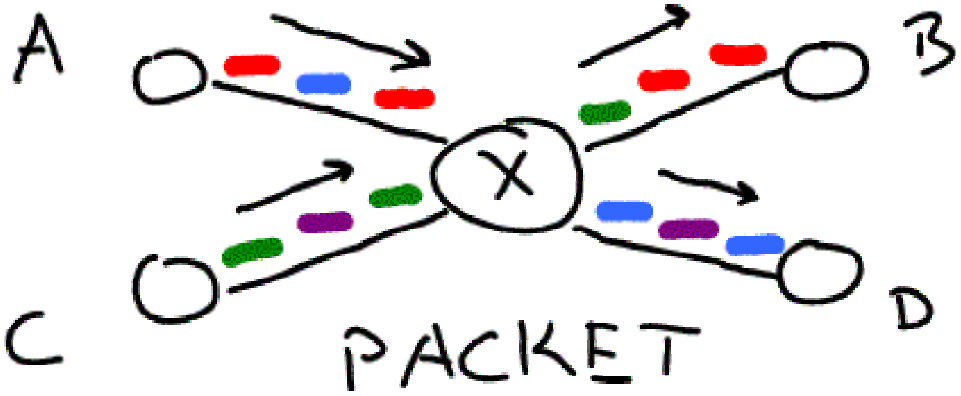
- By contrast, a packet network allows small units of data (packets) to be individually sent to different destinations

Packet network versus circuit network

CS419



C can't talk to B while A is talking to B



A and C can both talk to B and D

Packet network versus circuit network

- So clearly packet switched is better than circuit switched, right?

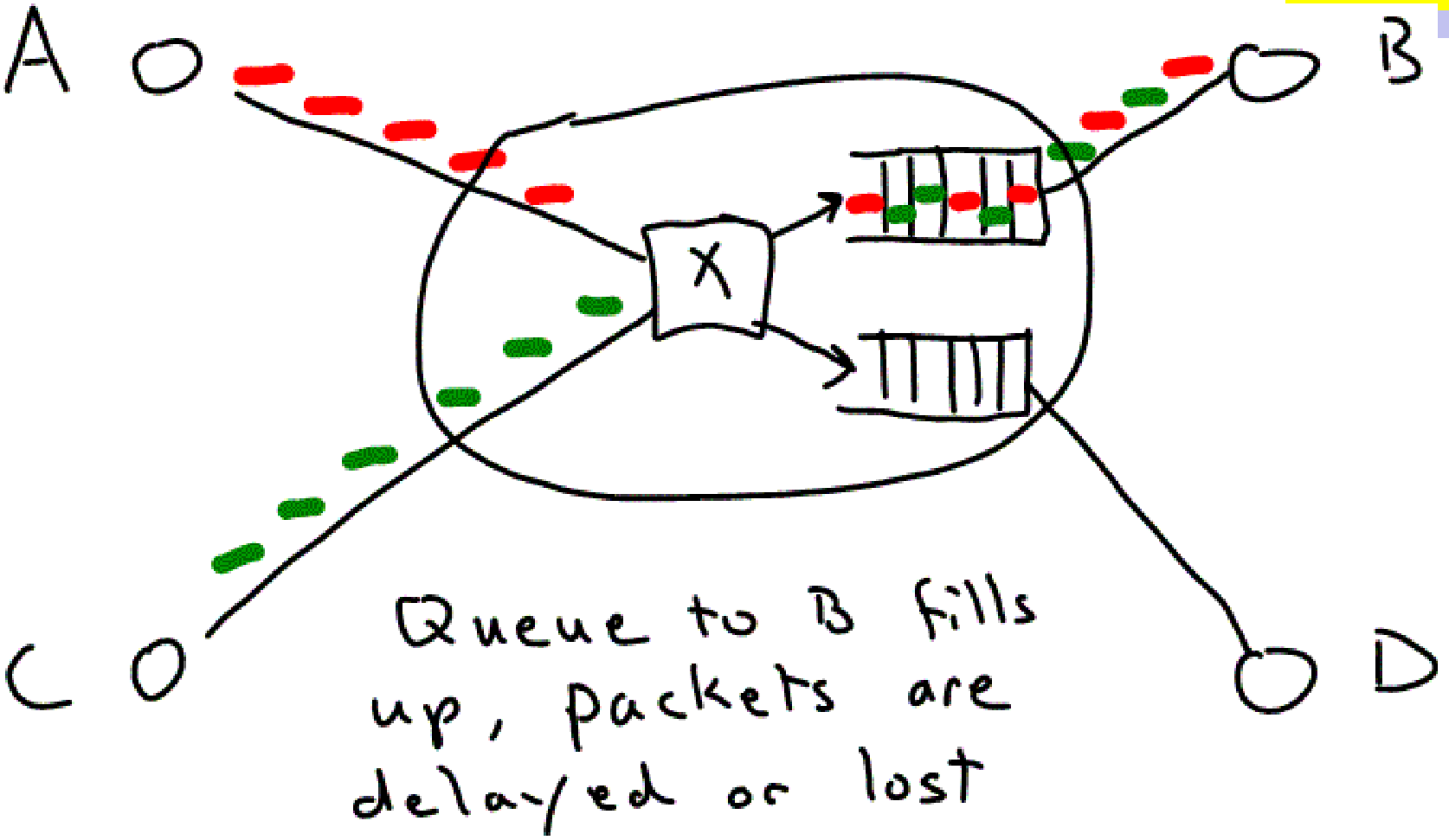
Packet network versus circuit network

CS419

- So clearly packet switched is better than circuit switched, right?
- Well, as with so much in this world, *it depends*
- What if A and C try to talk exclusively to B at high speed at the same time?

Delay and packet loss in packet networks

CS419

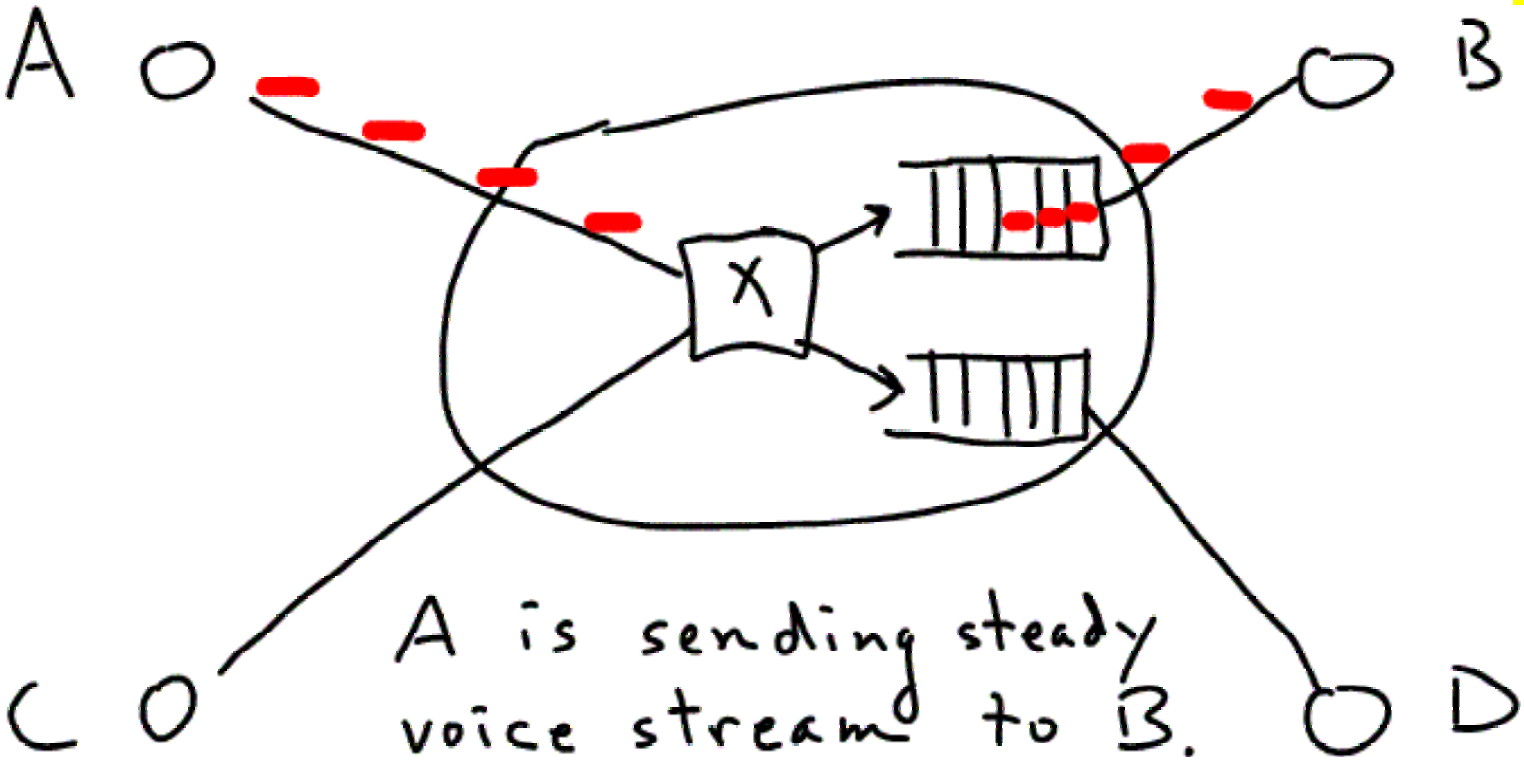


Delay and packet loss in packet networks

CS419

- Can happen any time multiple links feed into a single link
  - And incoming volume exceeds outgoing volume
- Larger queues can reduce packet loss at the expense of more delay
- Ultimately the sources have to slow down (congestion control)
- By contrast, circuit networks can block (busy tone)

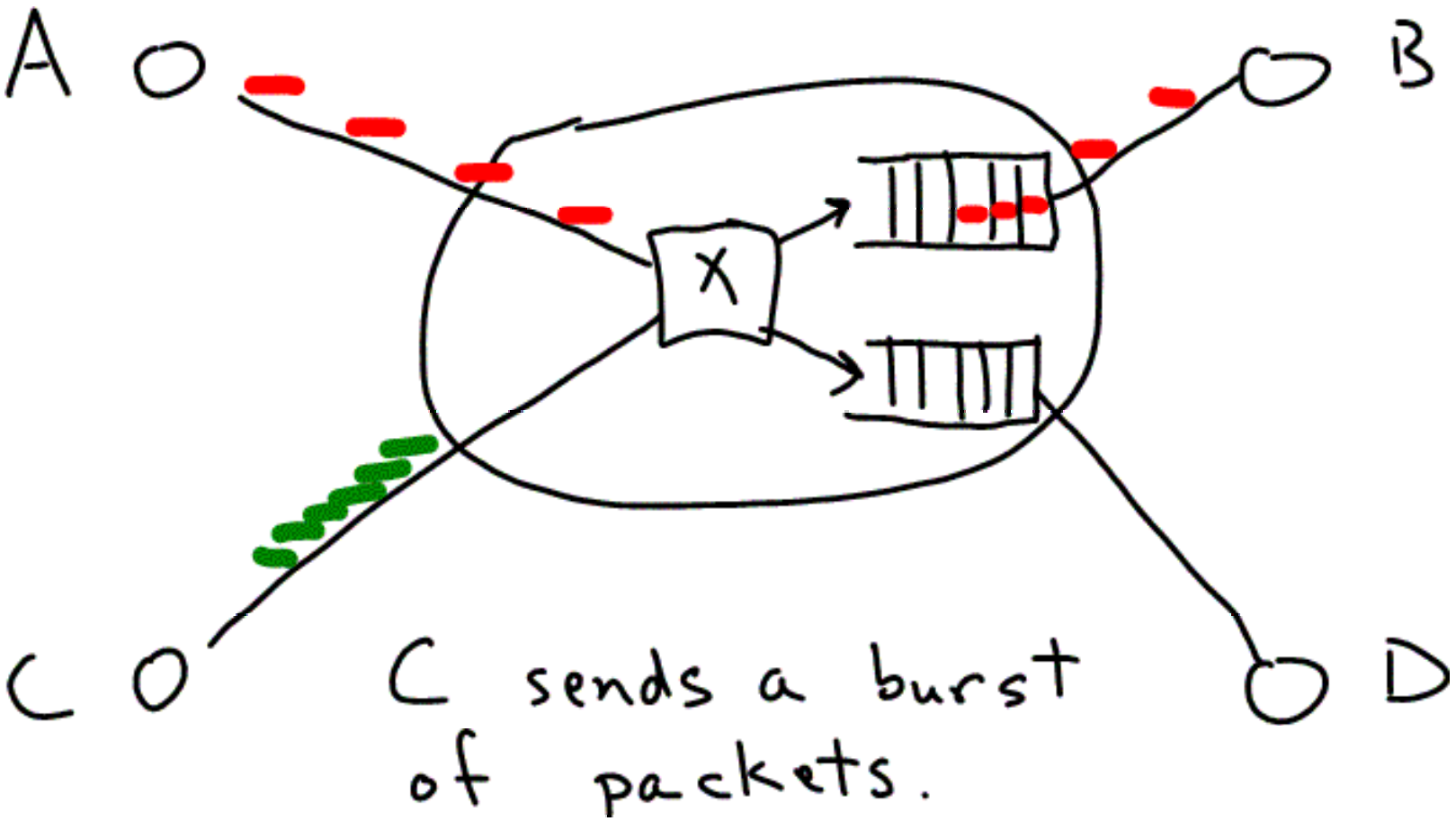
Also Jitter



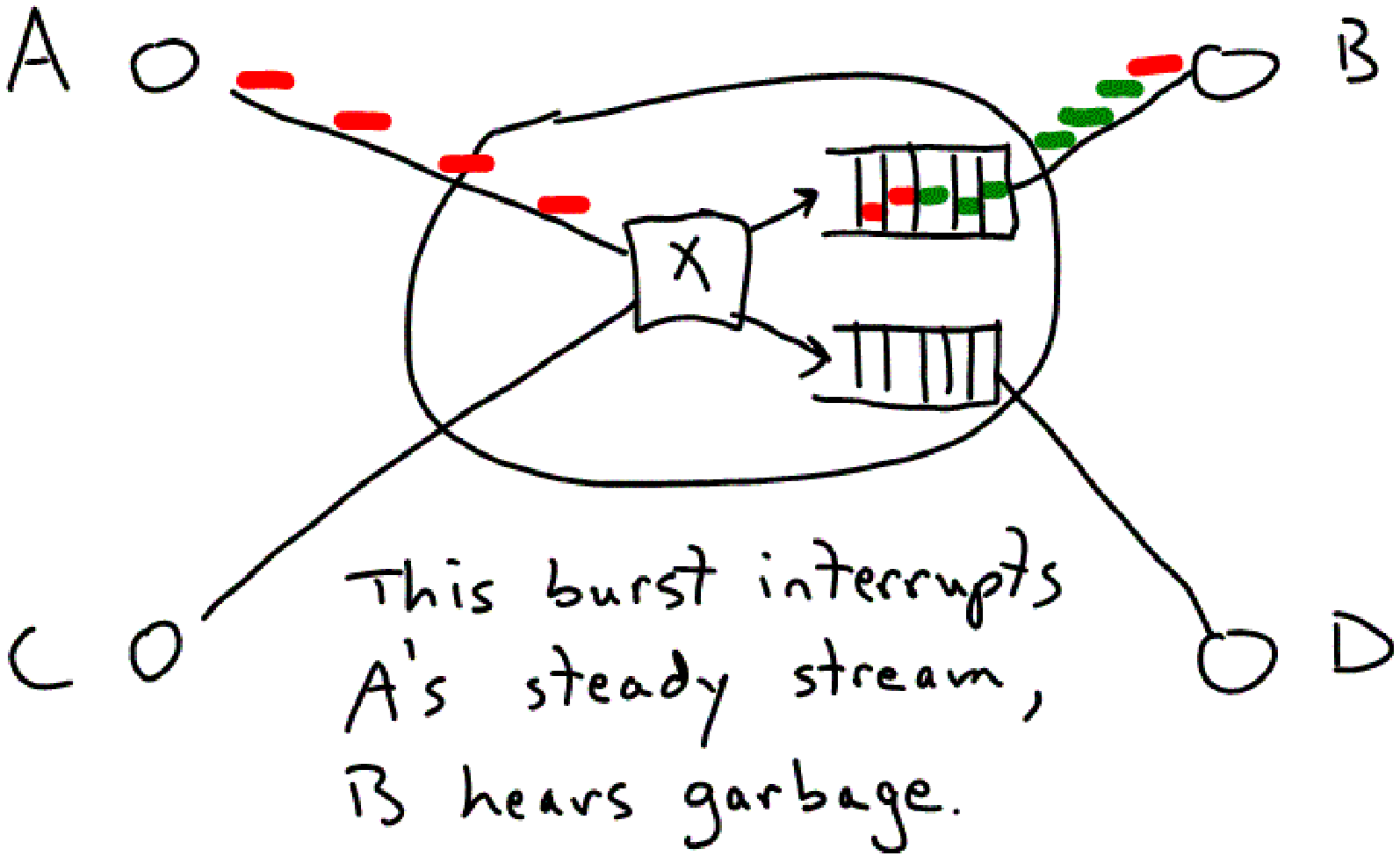
A is sending steady voice stream to B.

Delay is bad. Variation in delay is bad too. → Jitter

Also Jitter



Also Jitter


A
B
C
D
X
This burst interrupts
A's steady stream,
B hears garbage.

Circuits versus packets

CS419

- Circuits are an all or nothing proposition
  - Give good quality, if you can get yourself a circuit in the first place
  - Efficient only if the application keeps the circuit full (I.e. a voice stream)
- Packets are more flexible
  - Can send a little or a lot
  - But other traffic can interfere at any time
  - More efficient when traffic is bursty

Can a packet network emulate a circuit?

CS419

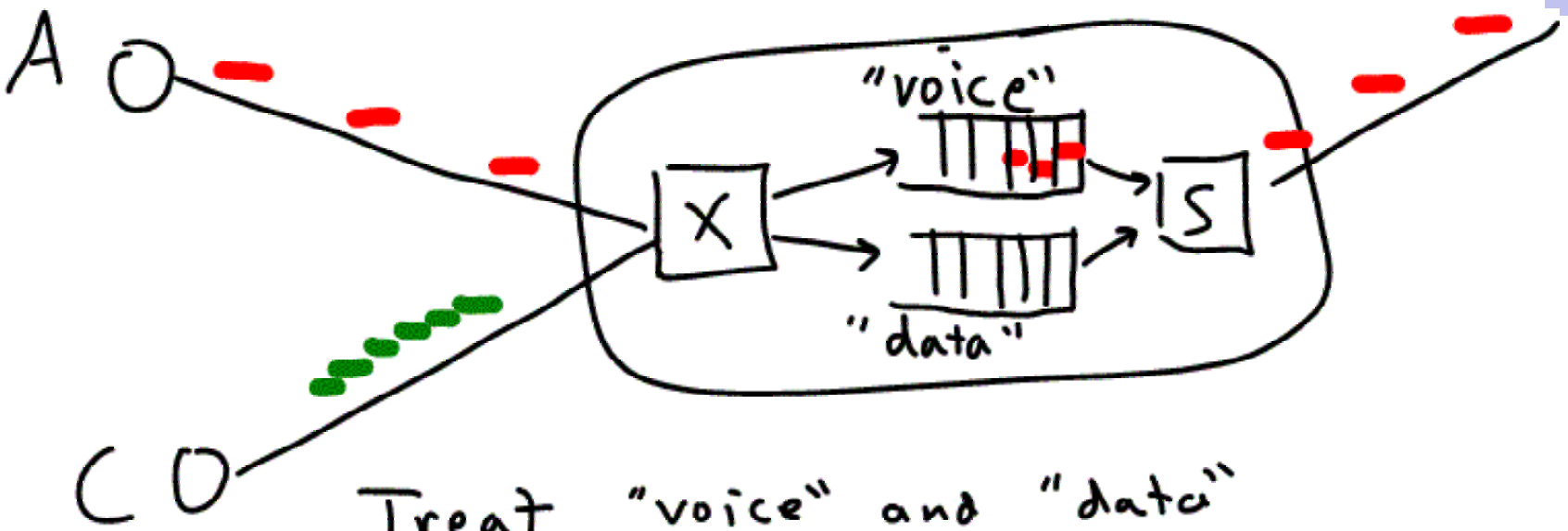
- After all, our STDM circuit sent data over the wire in “chunks”

Can a packet network emulate a circuit?

CS419

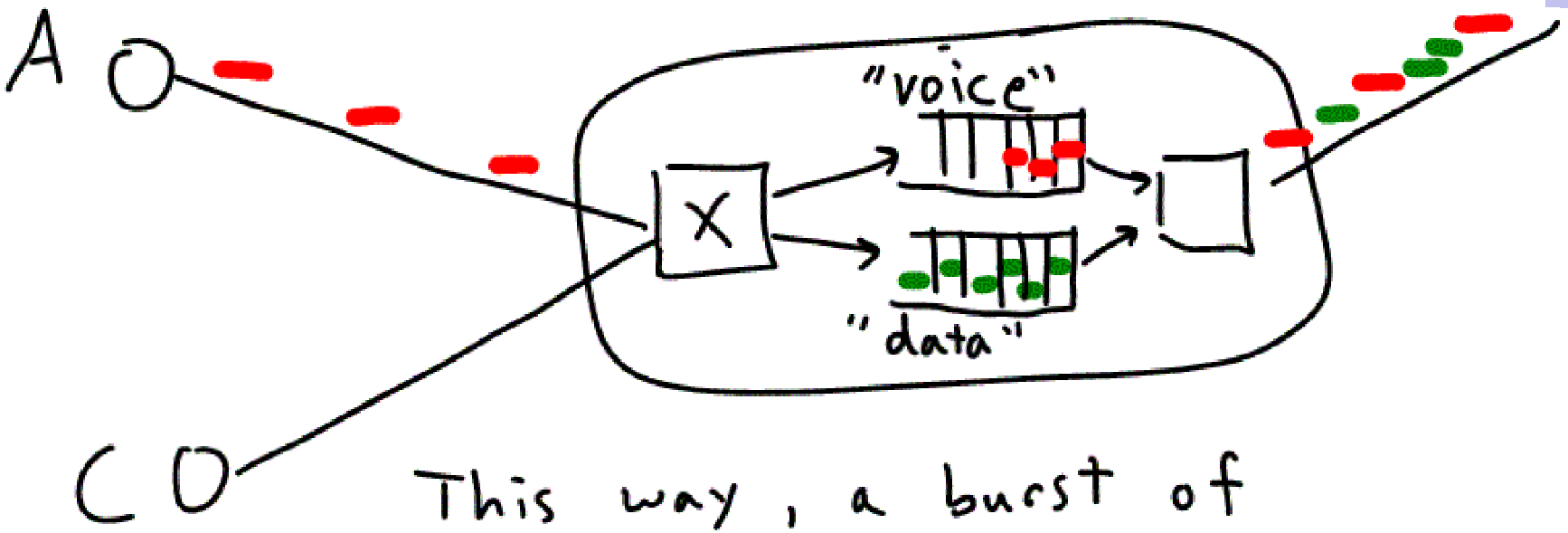
- After all, our STDM circuit sent data over the wire in “chunks”
- The answer is yes, it can
- And indeed, the first packet networks offered “services” that very much emulated circuits

One way to do it



Treat "voice" and "data" packets differently, for instance with different queues and schedulers

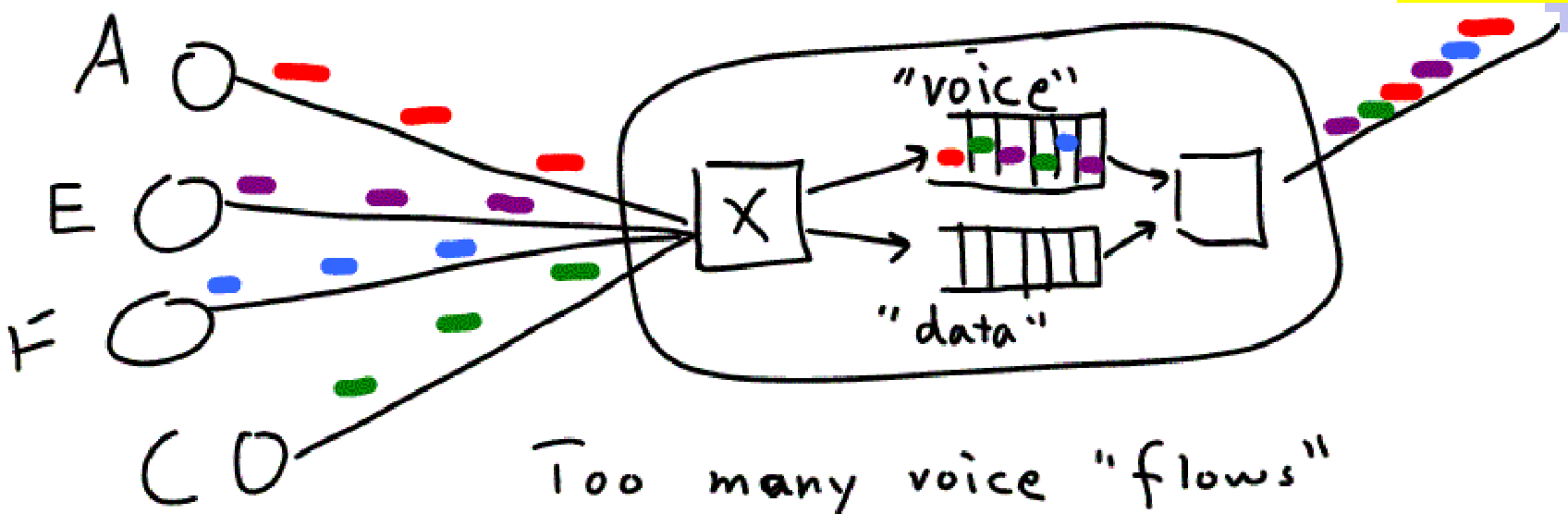
One way to do it



This way, a burst of data doesn't add jitter to the voice, but rather gets delayed itself

But this has complications too

CS419



Too many voice "flows" can still overflow the queue. Must be a way to reserve queuing resources ("call setup")

“Datagram” versus “virtual circuit” networks

CS419

- Both are packet networks
  - (We won’t discuss pure circuit networks any more in this course)
- Virtual circuit networks have the notion of call setup and blocking
  - But much more complex traffic models than our simple two-queue example
- Datagram networks is how the Internet ultimately got built!

But virtual circuit networks still important

CS419

- We don't see virtual circuit networks to our desktop
  - Though this was the vision for many folks
- But virtual circuit networks formed the unpinning of the Internet
  - Something called ATM
  - Being replaced with MPLS

This class focuses on the Internet

CS419

- Which is a datagram network
- One big topic will be how queues in the Internet manage not to become hopelessly overloaded
  - Many of you know, the answer is TCP, but we'll look at this in detail

Some terms introduced so far

CS419

- Network, node, link, queue
- Circuit and packet networks
  - a.k.a. data and voice networks
- Virtual circuit and datagram networks
- Delay, latency, loss, drop, jitter, blocking

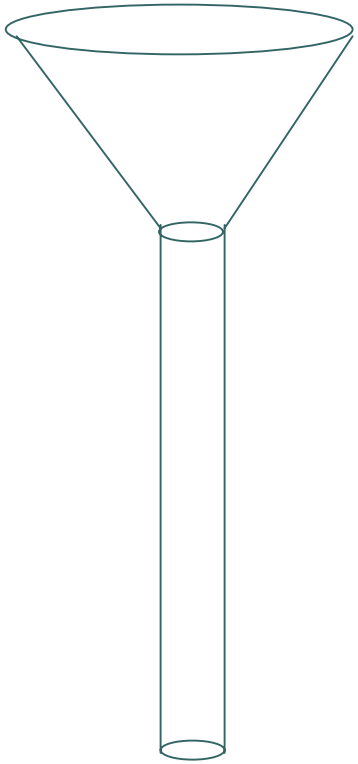
Bandwidth and Latency

CS419

- We looked at delay due to queuing
- But there are three main components to delay:
  - Propagation delay
  - Transmit delay
  - Queuing delay

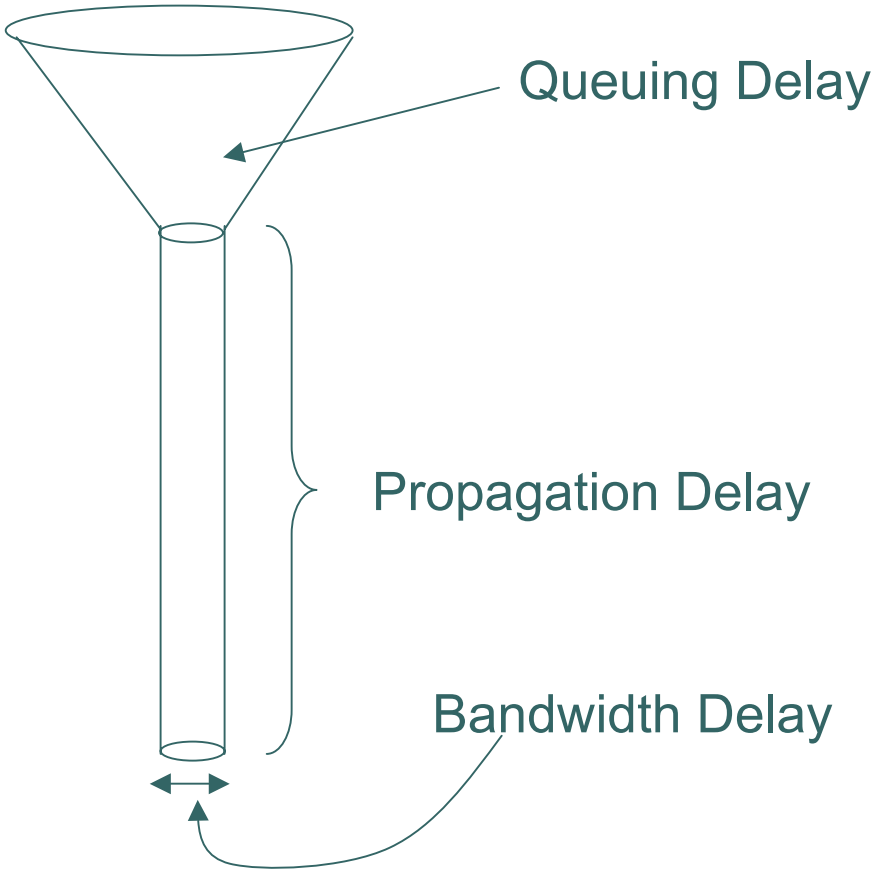
Queuing, transmit, and propagation delays

CS419



Queuing, transmit, and propagation delays

CS419



Total latency

CS419

- Total latency =
  - Propagation + Transmit + Queue
- Propagation =
  - Distance / Speed of light
- Transmit =
  - Packet size / Bandwidth

Delay x Bandwidth Product

CS419

- Refers to the number of bits you can have "in the pipe" at the same time
  - Or, how many bits you can stuff in the pipe before the first bit comes out the other end
  - Like hot water getting from the water heater to your shower!
- As bandwidth increases (and distance doesn't change) this is becoming an issue

An extreme (but realistic) Delay x Bandwidth Example

CS419

- Coast-to-coast propagation delay = 15ms
- OC192 link = 10 Gbps
- 10 Gbps x 15ms = 150,000,000 bits = 19 Mbytes = 7 songs (MP3 files)
- You could stuff 7 songs into an OC192 pipe at Boston before the first song starting arriving in LA!!!

A more common Delay x Bandwidth Example

- 50ms coast to coast delay (mainly from queuing)
- 100 Mbps Ethernet
- This is about 600Kbytes…still a decent sized file
- Delay x Bandwidth is starting to dominate our thinking about protocol performance

Common provider bandwidth units

- DSO = 64 Kbps
- DS1 = 1.544 Mbps
- DS3 = 44.736 Mbps
- OC3 = 155.52 Mbps
- OC12 = 622.08 Mbps
- OC48 = 2.488 Gbps
- OC192 = 9.953 Gbps
- OC768 = 39.813 Gbps

Bandwidth and throughput and goodput

CS419

- Bandwidth is the maximum theoretical speed of a pipe
- Throughput is the actual measured speed
  - Vague term because depends on where you measure
- Goodput is the throughput seen by the application
  - Throughput over the pipe can be more than goodput because of dropped and retransmitted packets, control packets, and headers