

# 15-441/61 Computer Networks

15-441 Spring 2019

Profs Peter Steenkiste & **Justine Sherry**



Carnegie  
Mellon  
University

# What makes up a network?

- A **communication medium** is the means by which information (the message) is transmitted between a speaker or writer (the sender) and an audience (the receiver).
- **Data transmission** is the process of sending digital or analog data over a communication medium to one or more computing, network, communication or electronic devices
- **Data encoding** is the process of applying a specific code, such as letters, symbols and numbers, to data for conversion into an equivalent cipher





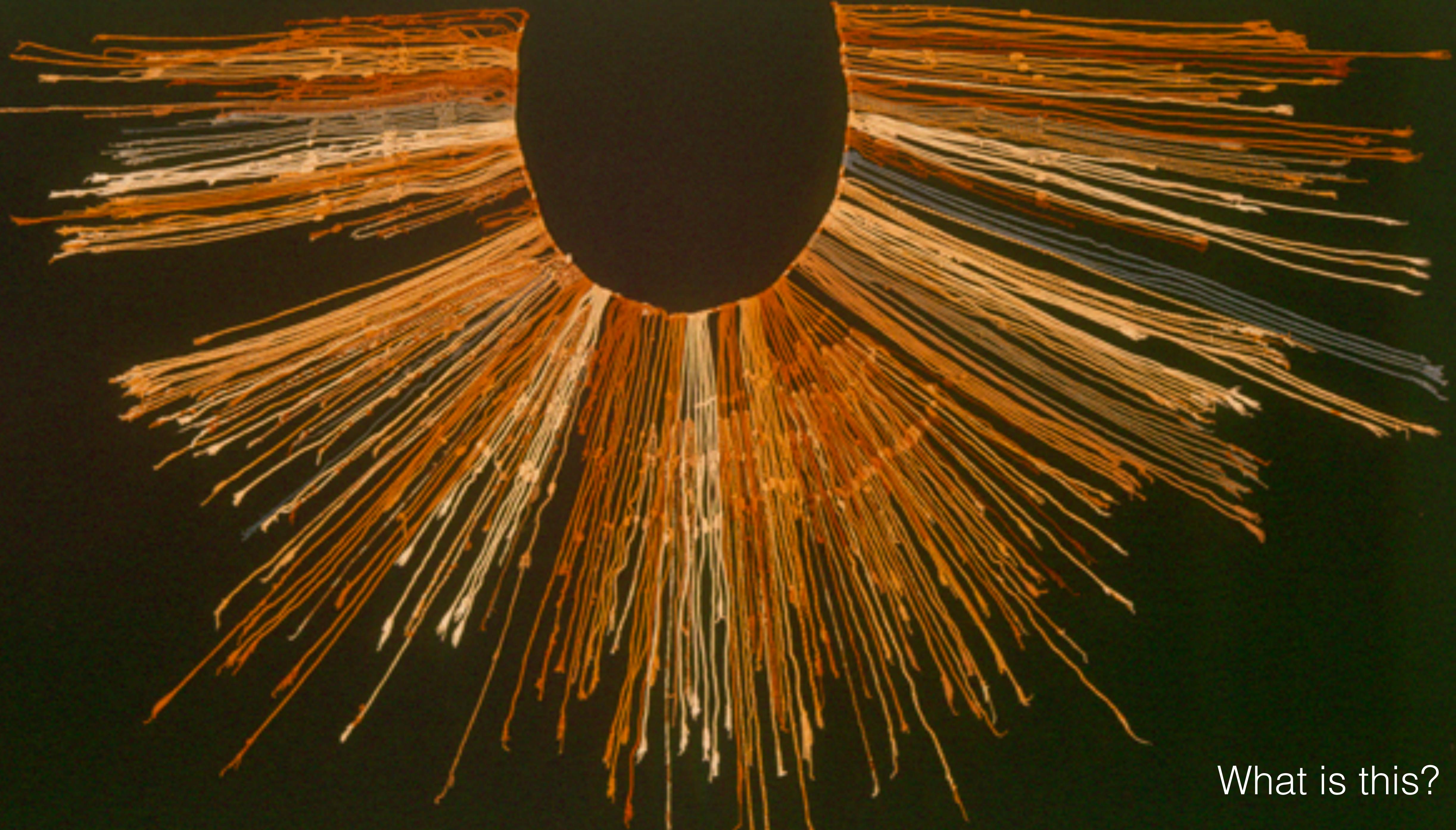
Do you  
remember this  
movie?



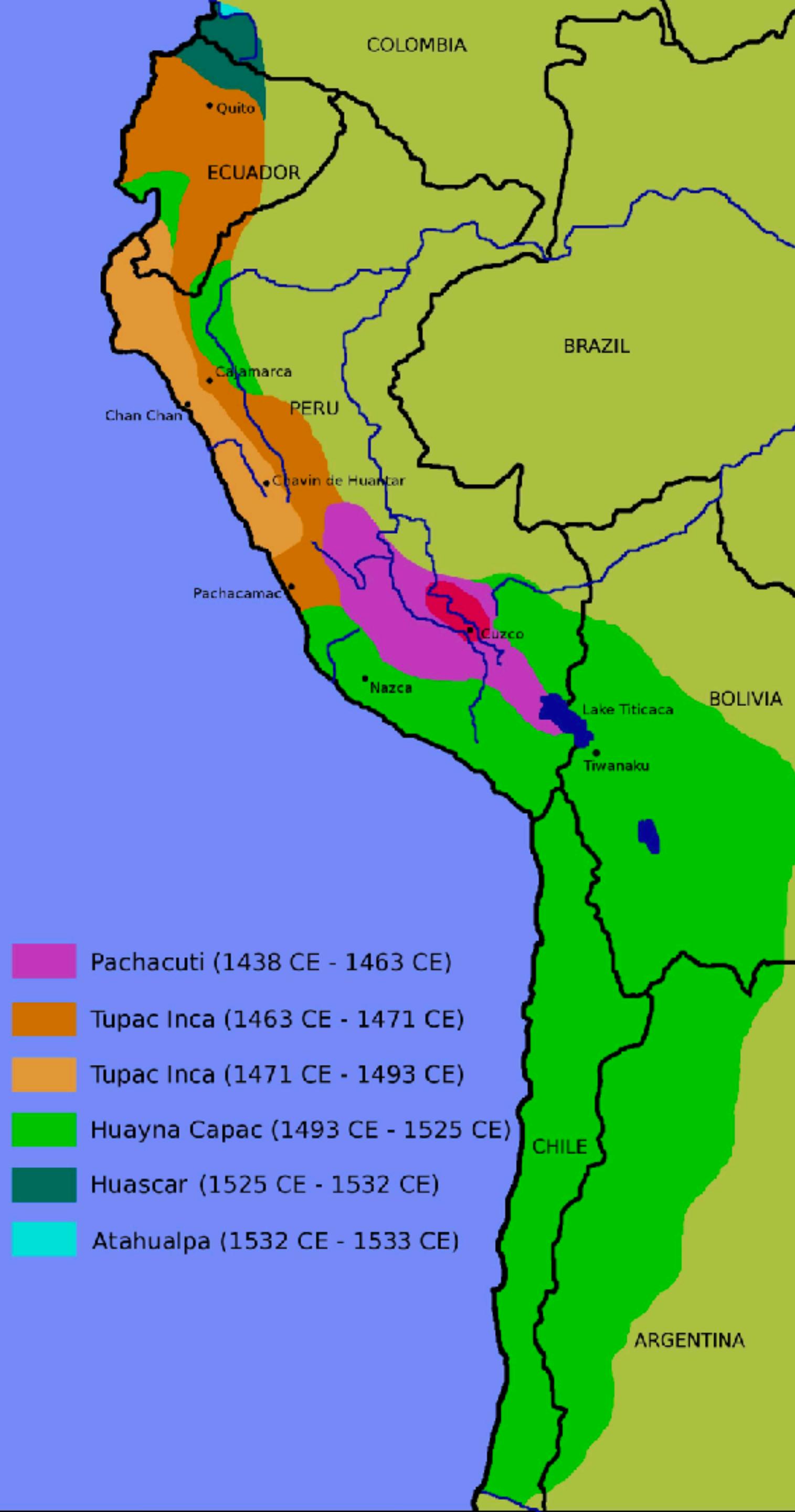
Why did I just show you this  
video?



- 
- The image shows the Great Wall of China stretching across a range of mountains. The wall is a light-colored stone or brick structure, and the surrounding hills are covered in dense green vegetation. In the foreground, there's a tall, rectangular beacon tower with a staircase leading up to its platform. A few people can be seen standing on the tower and walking along the wall. The sky is filled with white and grey clouds.
- The Great Wall of China used **beacon towers** to transmit information
    - The *communication medium* was light and darkness
    - The *data encoding* had only two values:
      - Light means danger! No light means safe!
    - *Data transmission* occurs as the light travels from outpost to outpost — at the speed of light!



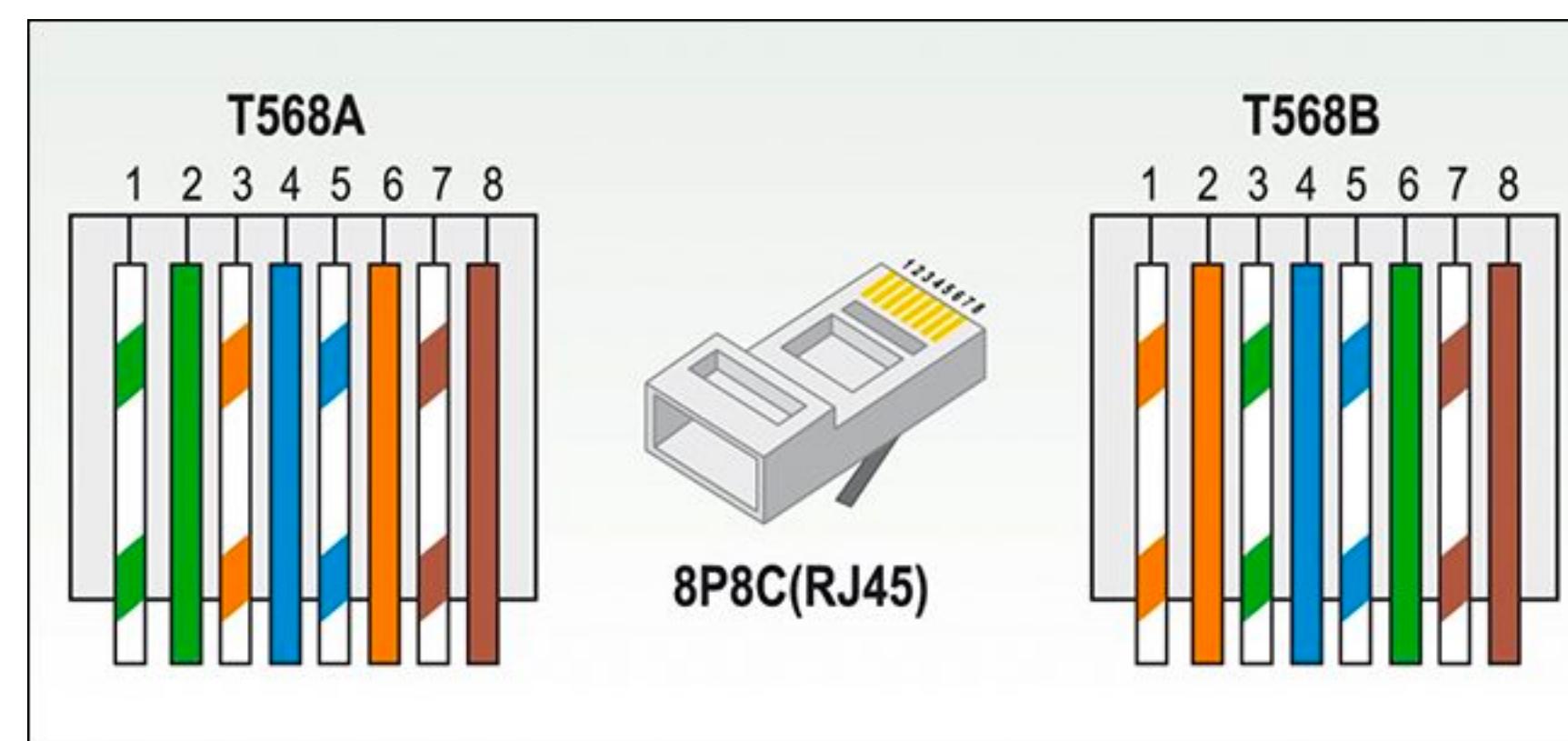
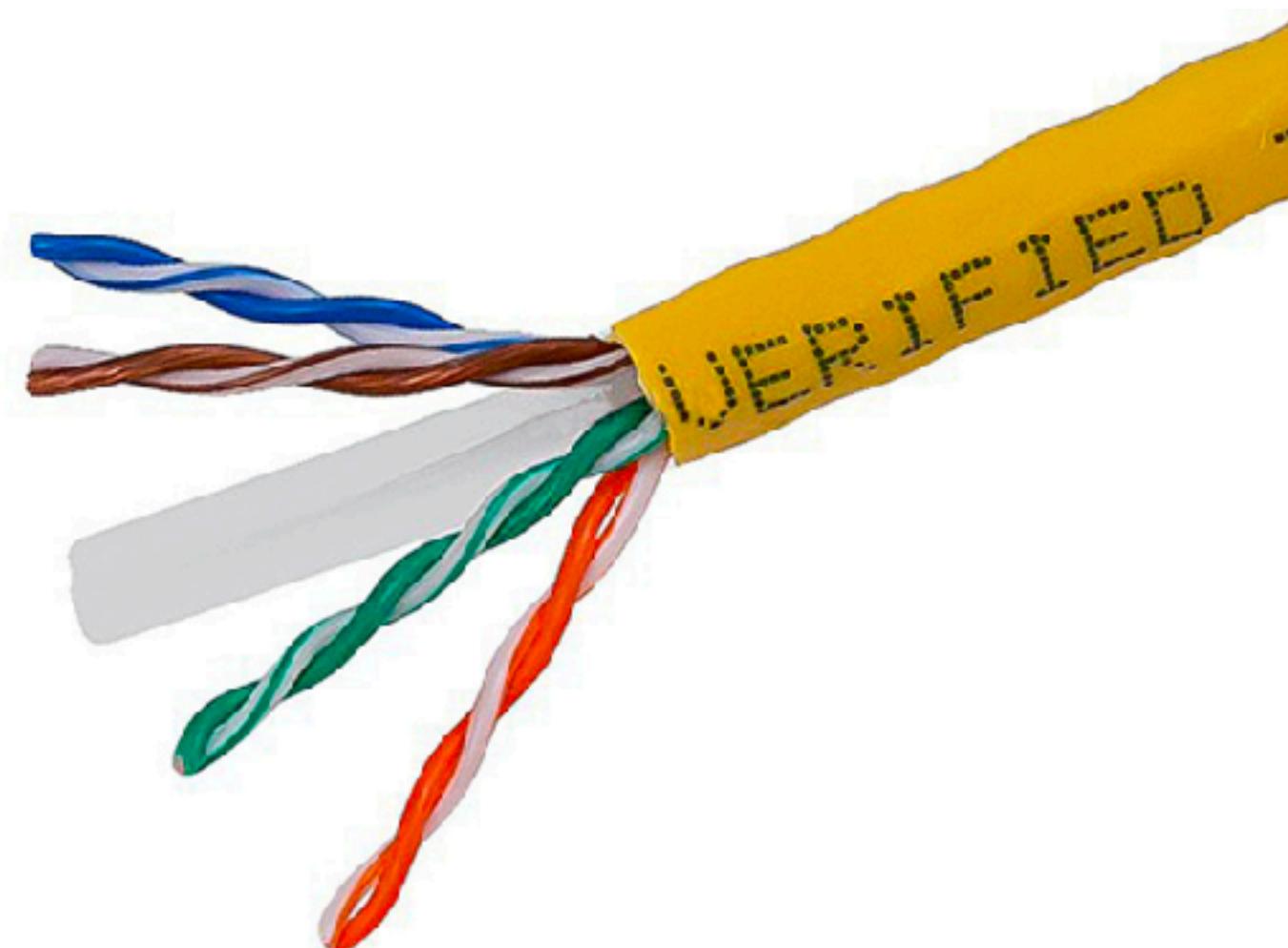
What is this?



- A **quipu** was a tool for information storage and transmission in the Incan empire
  - The *communication medium* consisted of up to thousands of strings
  - The *data encoding* consisted of tying knots in the strings in a special pattern.
    - A lot more capacity than “on” or “off” lights.
  - *Data transmission* required carrying the knotted strings from one place to another.
    - A lot slower than the speed of light.

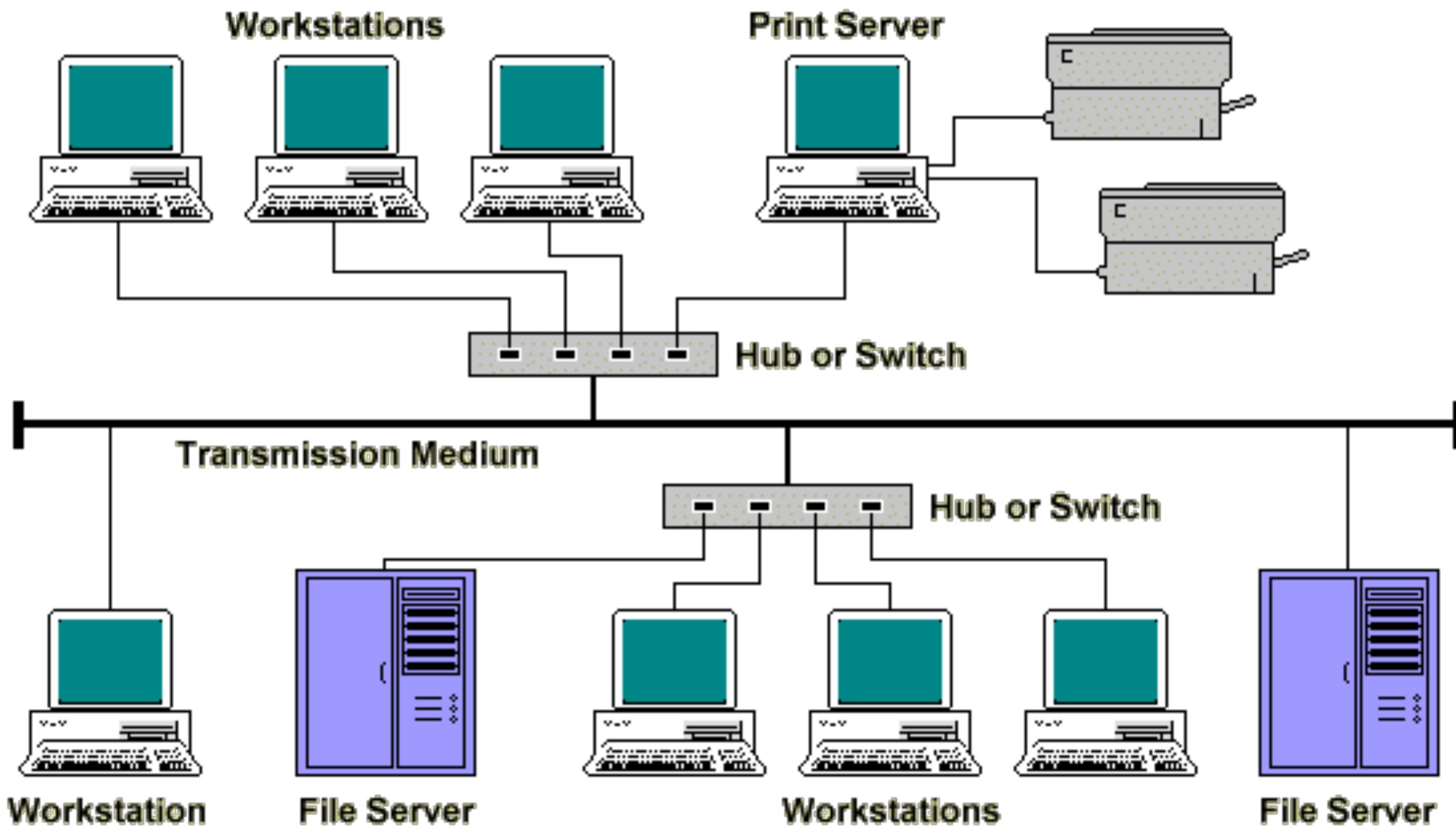


# A modern communication medium



- **Ethernet over copper**
  - What is the medium?
  - How is data encoded?
  - How is data transmitted?

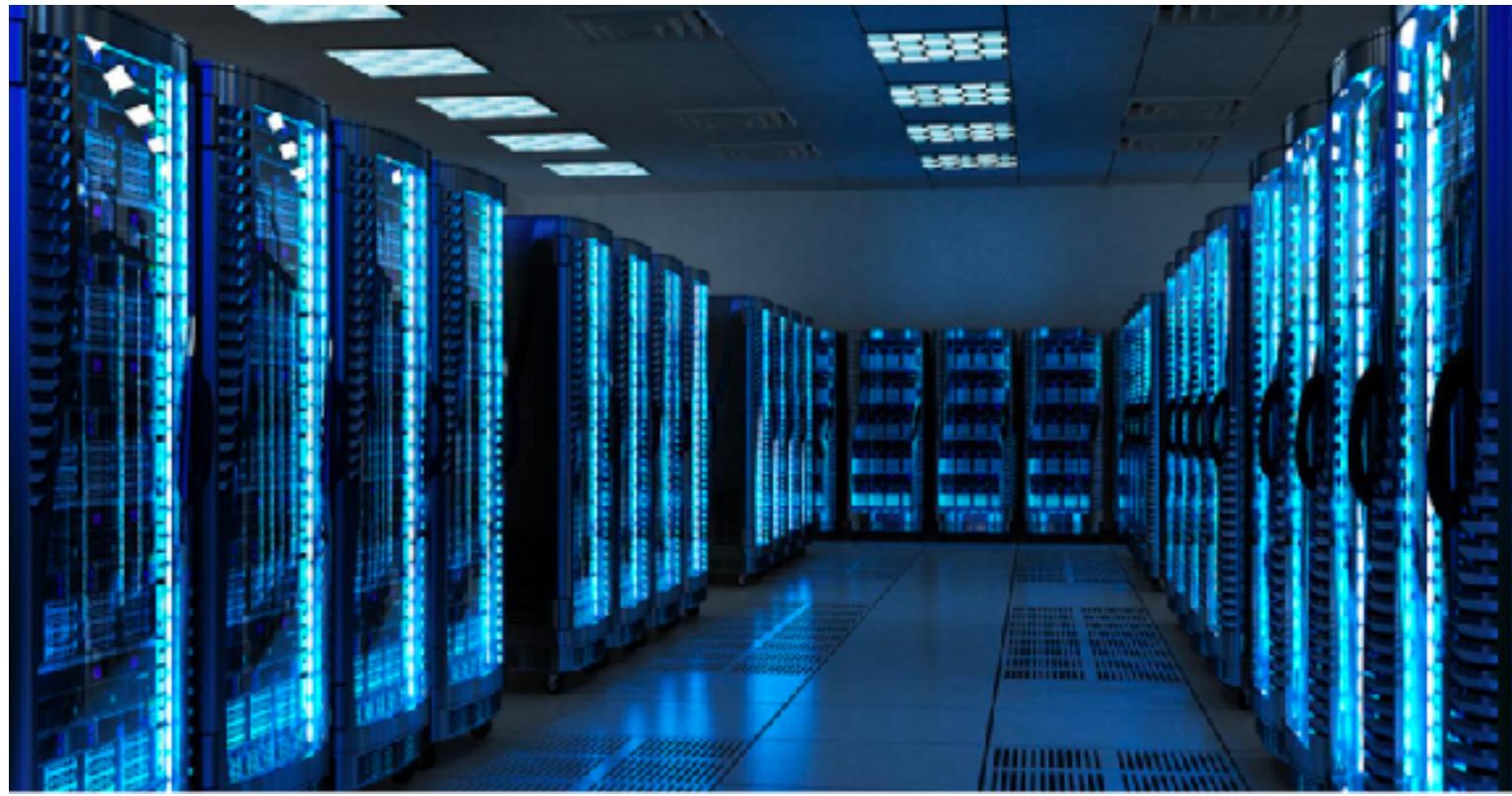




When you connect two or more endpoints with a communication medium, you have a network.



# Modern Networks



Datacenters



Low Earth Orbit (LEO)



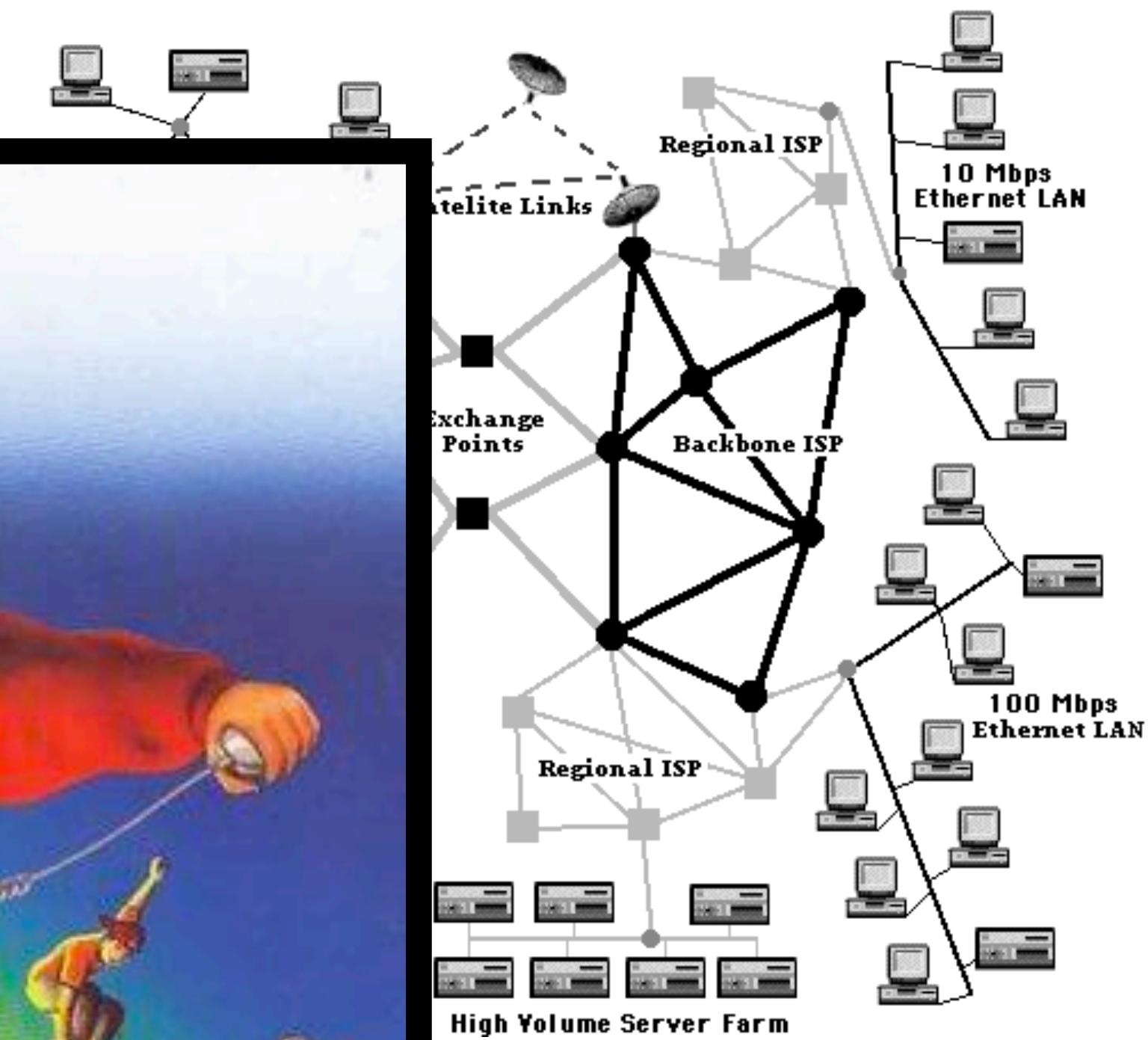
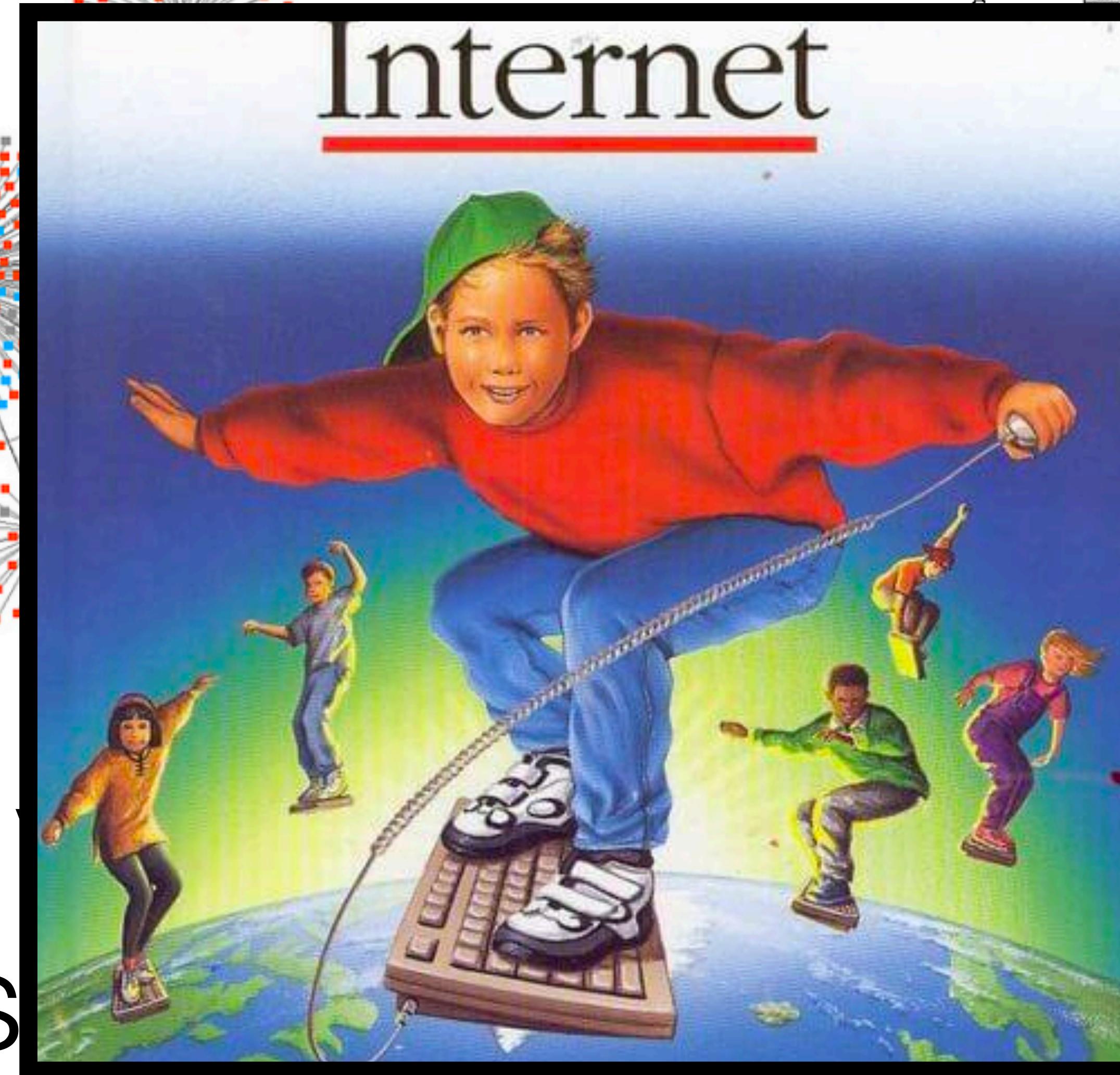
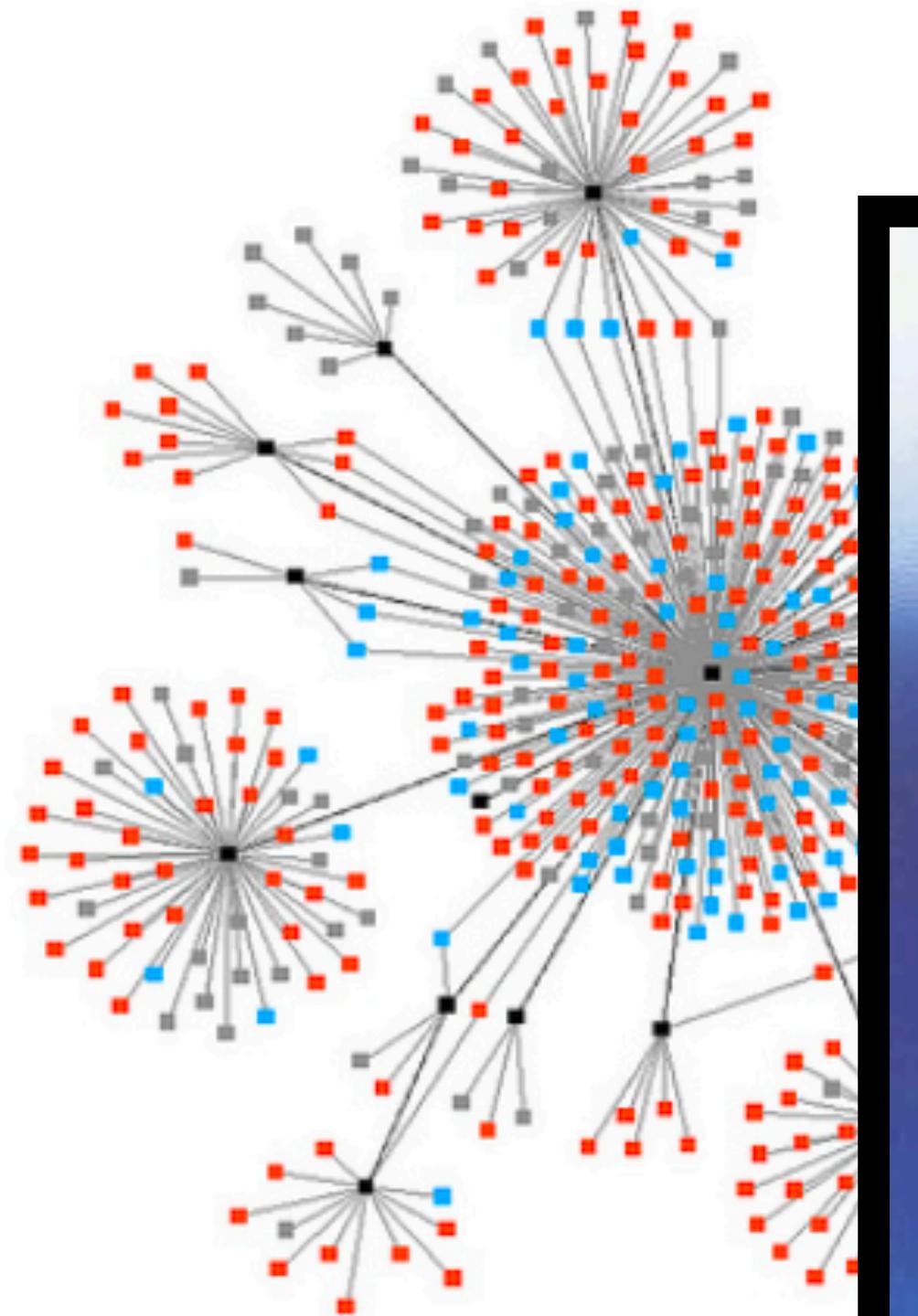
University Campus

**How is building each of these networks different from each other?**

**What might make it more challenging to develop, say, web applications for each scenario?**



When  
networks



or more  
networks.



Why should I care about this  
class?



Would you rather have a super computer  
without a network connection, or a simple  
cell phone with connectivity to the Internet?



Networking is central to all computer systems that matter.



# This Class

- We will learn about lots of kinds of networks.
  - *How to implement and manage networks*
  - *How to design applications that use networks*
- And we will do this in a hands-on way
  - *You're going to write a lot of code (sorry, not sorry!)*
    - *You'll really be able to impress recruiters and grad schools after this.*



## Lectures:

- ◆ Nuts and bolts — how networks work today.
- ◆ A bit of a history lesson, too: how did we get here?

## Project #1:

- ◆ Build a Web Server
- ◆ “Get your feet wet” with big C projects
- ◆ Think about all the things you rely on in designing applications and services that *use* the Internet

# Part 1: Internet Basics



# How does the Internet know where to send my data when I connect to a server in say, Pakistan?

Where do domain names like google.com come from? Who decides who gets to own what name?

How does my computer make sure data that I send is not lost or corrupted?

Why are some networks “slow” or “fast”?

# Part 1: Internet Basics



## Lectures:

- ◆ The Web and HTTP, Content Distribution Networks, Overlays (and P2P and Bitcoin), network security.
- ◆ Guest lecture from CTO at Fastly, a startup Content Distribution Network

## Project #2:

- ◆ Build a transport protocol for basic file transfer
  - ◆ Make sure data doesn't get lost or corrupted, and make sure data is transferred fast!
- ◆ Adapt your protocol for use in:
  - ◆ A datacenter
  - ◆ A CDN
  - ◆ The moon

# Part 2: Building Applications that Use Networks



What is a CDN and how does it make the web faster?

What is the difference between HTTP 1.5 and HTTP 2.0?

How do we send data over the Internet privately?

Why does the underlying structure of the network impact application performance?

## Part 2: Building Applications that Use Networks



# Part 3: Building Network Infrastructure

## Lectures:

- ◆ Modern network infrastructure and challenges: LTE vs 5G, datacenter networks, middleboxes and router architecture
- ◆ Guest lecture from Jitu Padhye at Microsoft Azure

## Project #3:

- ◆ Build Netflix
  - ◆ Seriously
- ◆ Integrate everything you learned in Part 2 (online video, CDNs) with some routing.
- ◆ This project will make you feel so powerful.



# Part 3: Building Network Infrastructure

What happens inside a datacenter at Facebook or Microsoft?

How do routers and switches work on the inside?

What algorithms and architectures make them “fast”?

What is the difference between “4G” and  
“5G” cell service?



Any questions about what we  
will learn in this class?



# Course Policies

**Mostly outlined in the  
syllabus - this is an  
overview**



## MOST IMPORTANT WAY TO SUCCEED IN THIS CLASS:

The majority of your grade comes from class projects

- **45%** for Projects I, II and III
- 18% for Midterm exam
- 27% for Final exam
- 10% for Homework

**This means: START EARLY! Use office hours to ask for help! Debug your code with your own testing scripts!**



# Late Work

- We will give you two “late days” for free.
  - You don’t need to tell us which days you are using — we calculate late days at the end of the semester to your advantage (e.g., if you turn in both a project and a homework two days late, we will give you full credit on the project since that is worth more points.)
- Any other late assignments are penalized 15% per day late. No assignments are accepted more than 48 hours after the deadline.
- See the syllabus about dealing with emergencies.



# Don't Cheat. Seriously.

- Working together is important
  - Discuss course material in general terms
  - Work together on program debugging, ..
  - Collaborating on projects P2 and P3
- Final submission must be your own work
  - Homeworks, midterm, final, projects....
- Submitting or using someone else's work is an academic integrity violation (i.e., cheating)
  - We will follow the university policy on reporting violations
  - Voluntarily sharing your work is also a policy violation
- Web page has details, e.g., university policy, etc.



# REALLY don't cheat on the projects

- The project code you submit must be your work!
  - Exception is the starter code provided by us, standard libraries, packages mentioned in the project handout
  - If in doubt, ask the course staff
- We use tools to compare submissions
  - These tools are very good
  - Don't compete with them (the odds are against you)
- Some students have put their projects on the web
  - Posting and using the code is a form of cheating
  - If you can find the code, so can we



# CMU's Disability Services Office is Great

- If you need their help — for any reason — we do what they tell us to, no questions asked (we don't need to know why you need accommodations).
- Please email the professors a copy of the accommodations sheet for us to sign, or bring a paper copy to either of us in Office Hours.



# CAPS is also great.

- Whenever Prof Sherry is worried about a student, she calls CAPS and they give great advice.
- Many people think CAPS is just for people with severe mental health troubles. You can also go just because you're feeling a little stressed about *anything* and you need someone to talk to.
  - Seriously, no problem is too small.
  - If you think about visiting them, just go ahead and do it.



# How do I get off of the waitlist?

- Email the professors and tell us your name, program, and whether or not you have taken 15-213/15-513 (and what grade you got).
  - **213/513 is a pre-requisite for this course.**



Any policy questions?



# Your TAs are amazing.



Back to technical stuff!  
(Fun!)



# Today/Thu: Networking Basics

- Handling heterogeneity is one of the major challenges in networking.
- It's part of what makes the Internet such a marvelous system
  - Connecting very different networks, with different goals and designs, across the globe.
- It's part of what makes designing applications that use networks challenging
  - As you will see building your projects this semester!
- We'll learn the basic vocabulary for talking about how networks perform



Flash back to ten minutes ago:  
What do we already know about how  
networks differ?



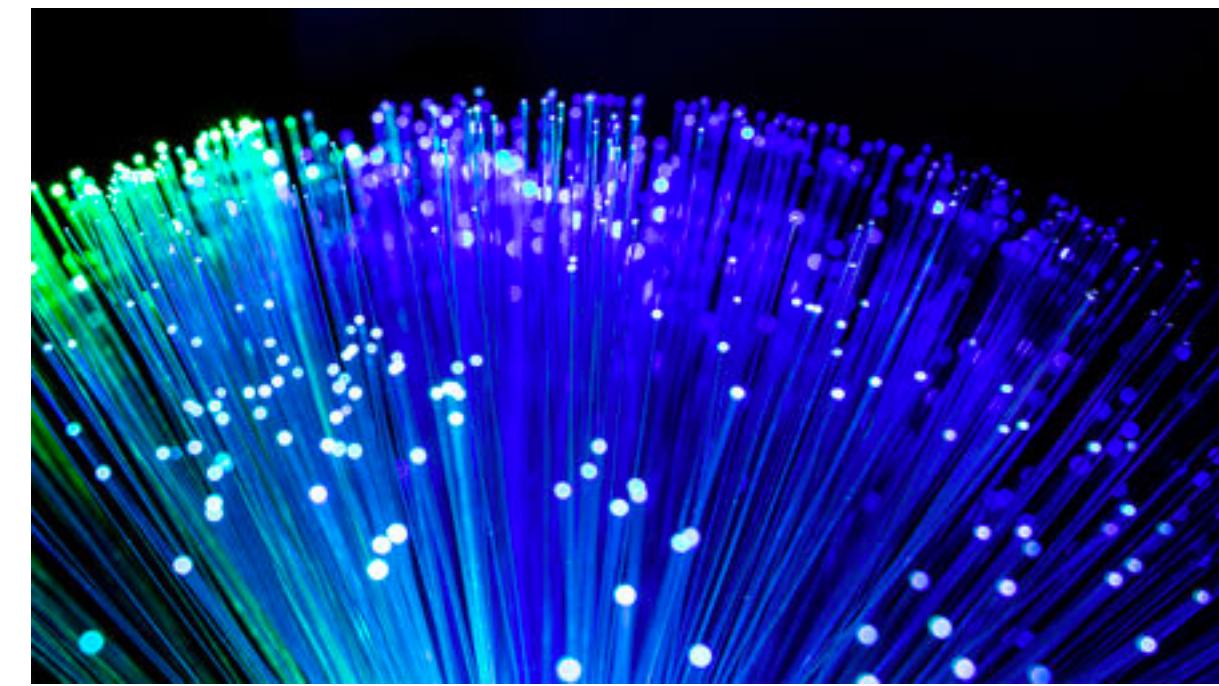
# Physical Medium



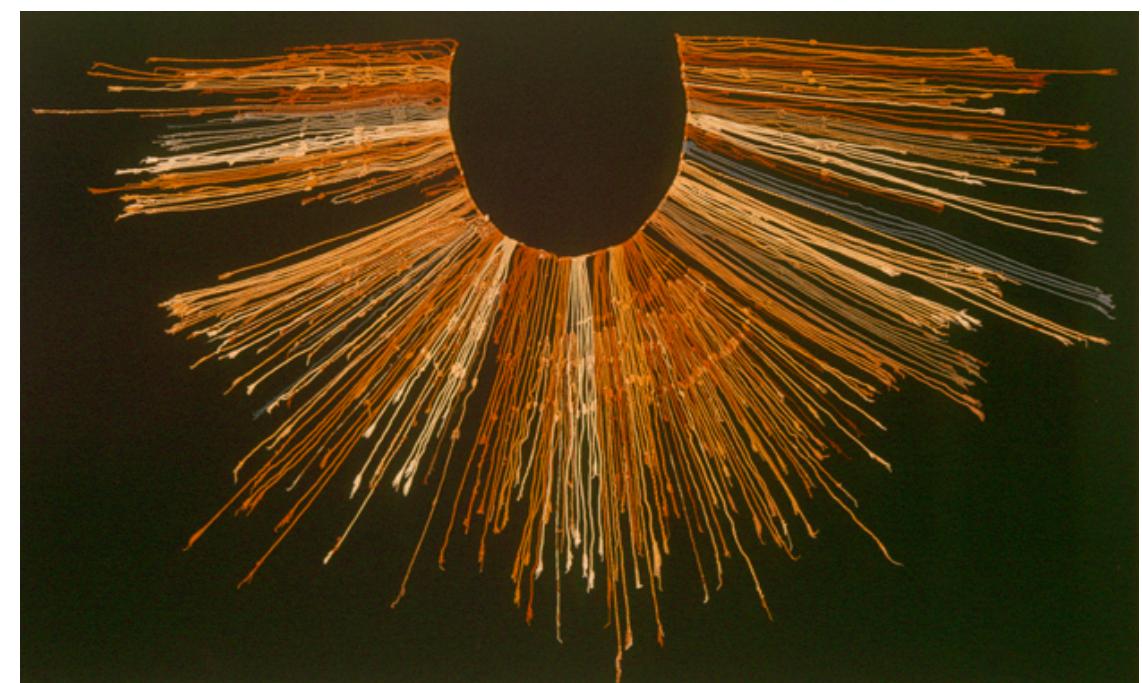
Light/Air



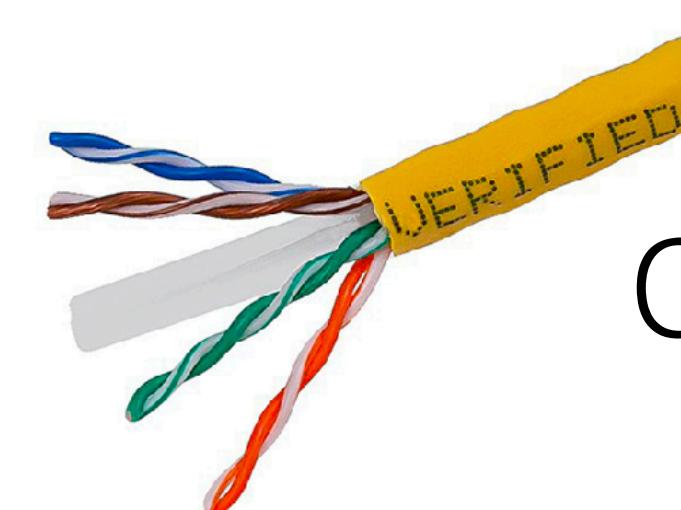
Radio



Fiberoptics (light again!)



String



Copper



The Internet does allow for  
many physical mediums.



Every network has a different  
bandwidth and latency.



# Regardless of Medium: Nodes and Links



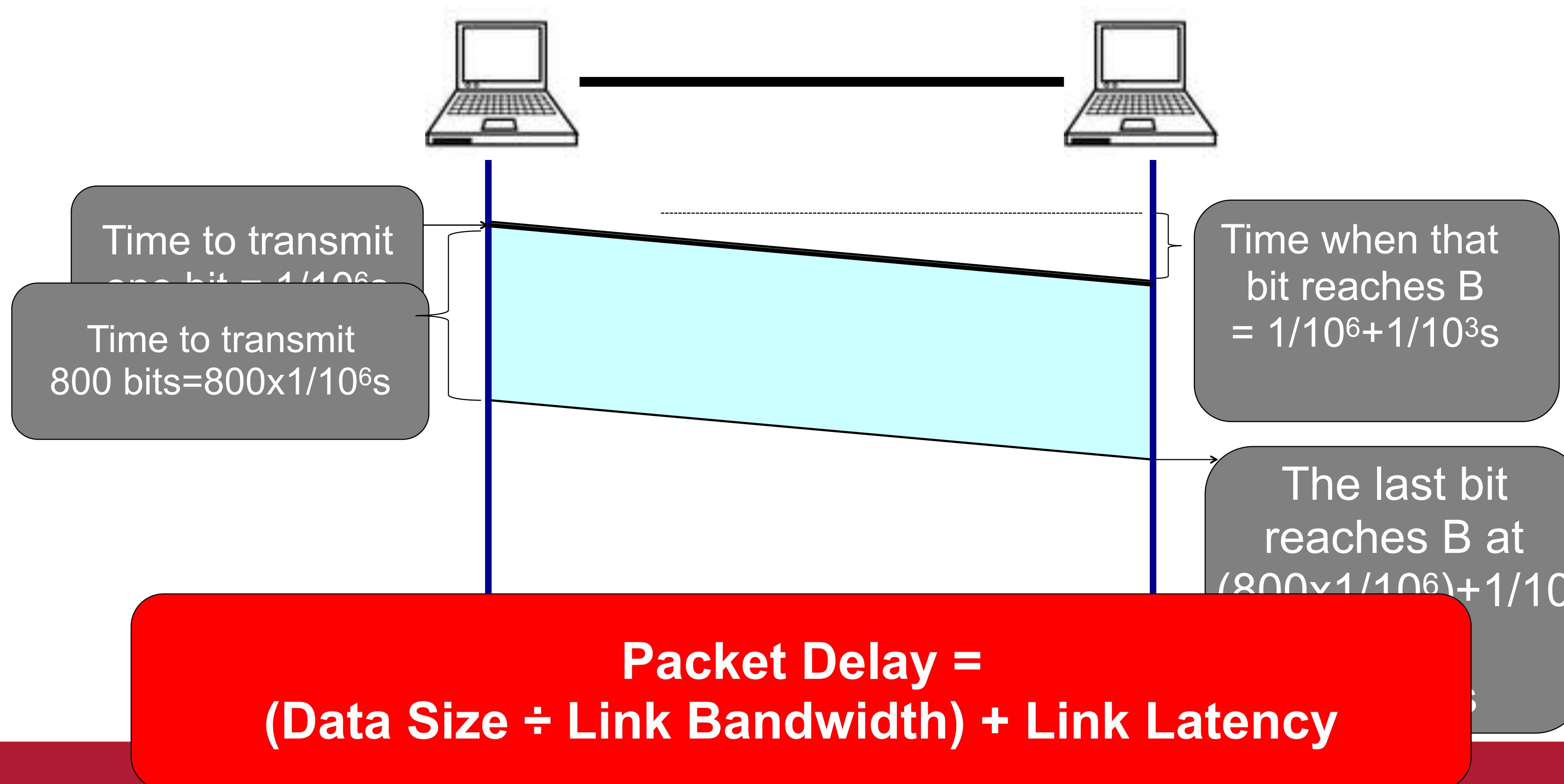


- Bandwidth (capacity): “width” of the link
  - number of bits sent (or received) per unit time (bits/sec or bps)
- Latency (delay): “length” of the link
  - propagation time for data to travel along the link (seconds)



# Packet Delay

*Sending 100B from A to B?*



# Your Turn

Packet Delay = (Data Size ÷ Link Bandwidth) + Link Latency

## **Left Side of Room:**

You are working in a datacenter with a 10 microsecond latency and 40 Gbps links. How long does it take to send 1500 bytes from sender to receiver?

## **Right side of Room:**

You are working at NASA with a satellite exploring the solar system. The latency is 8 seconds and the bandwidth is 32kbps. How long does it take to send 1500 bytes from sender to receiver?



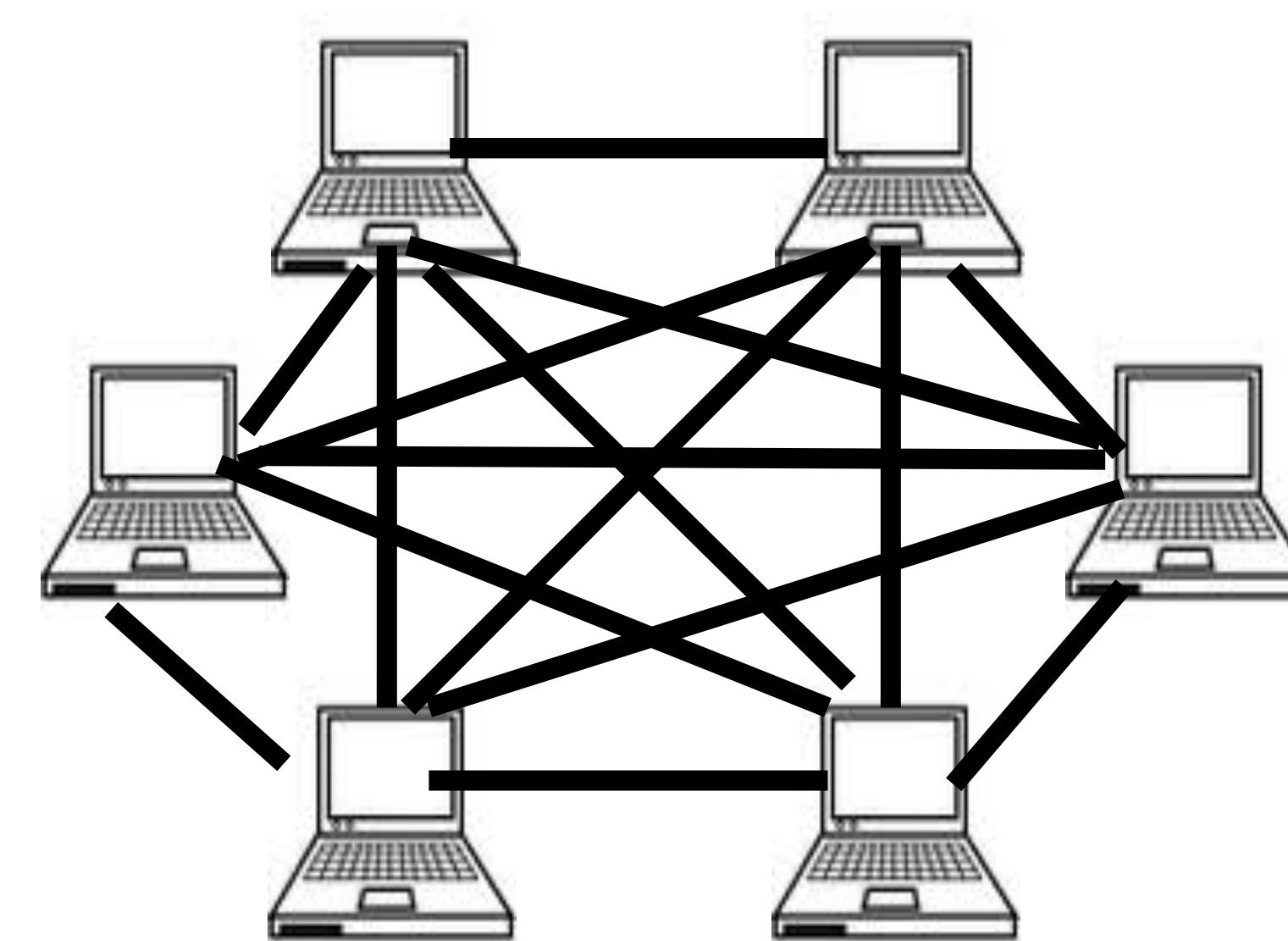
The Internet supports arbitrary sending rates and latencies.



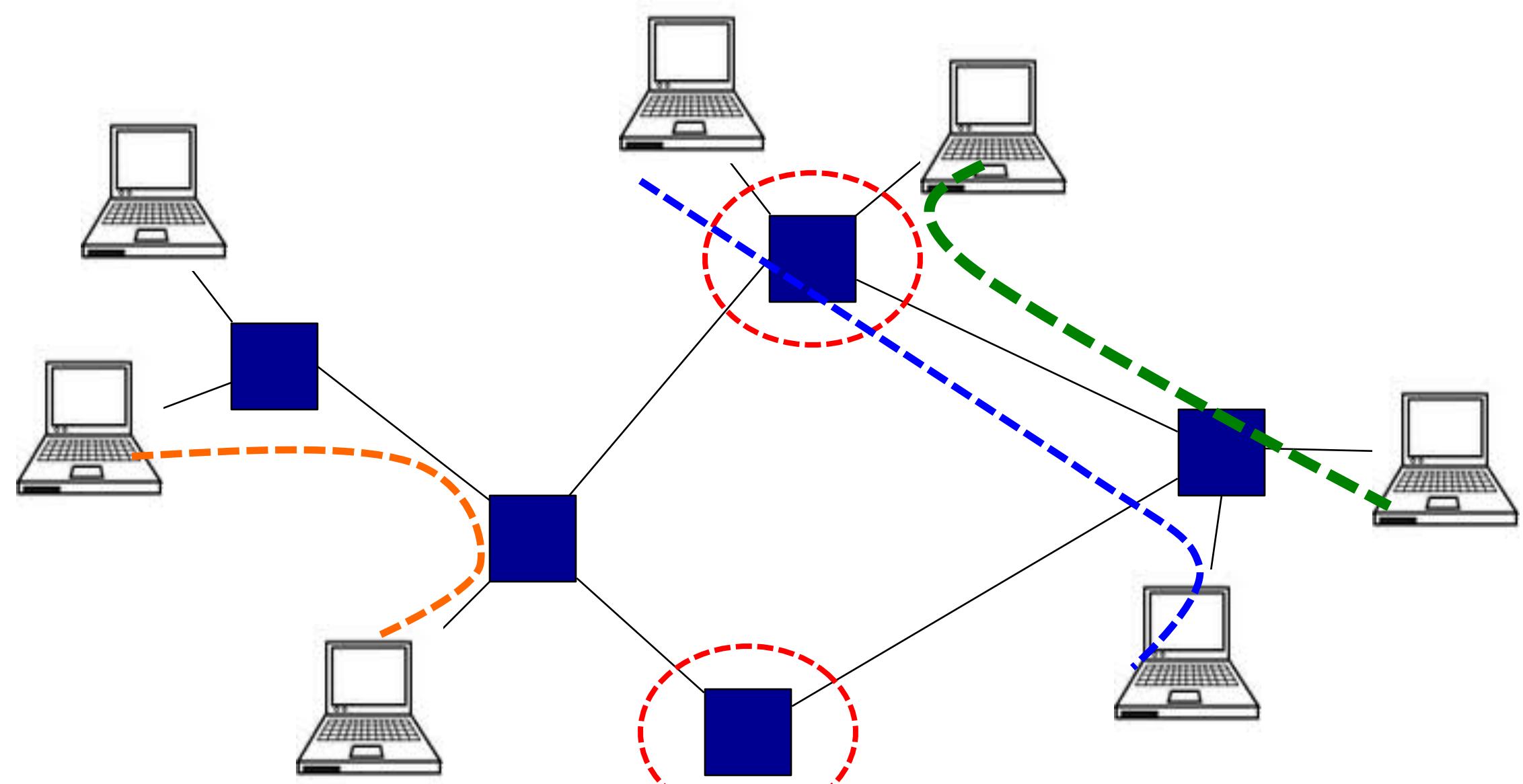
# Nodes and Links



# What if we have more nodes?



# Solution: A switched network

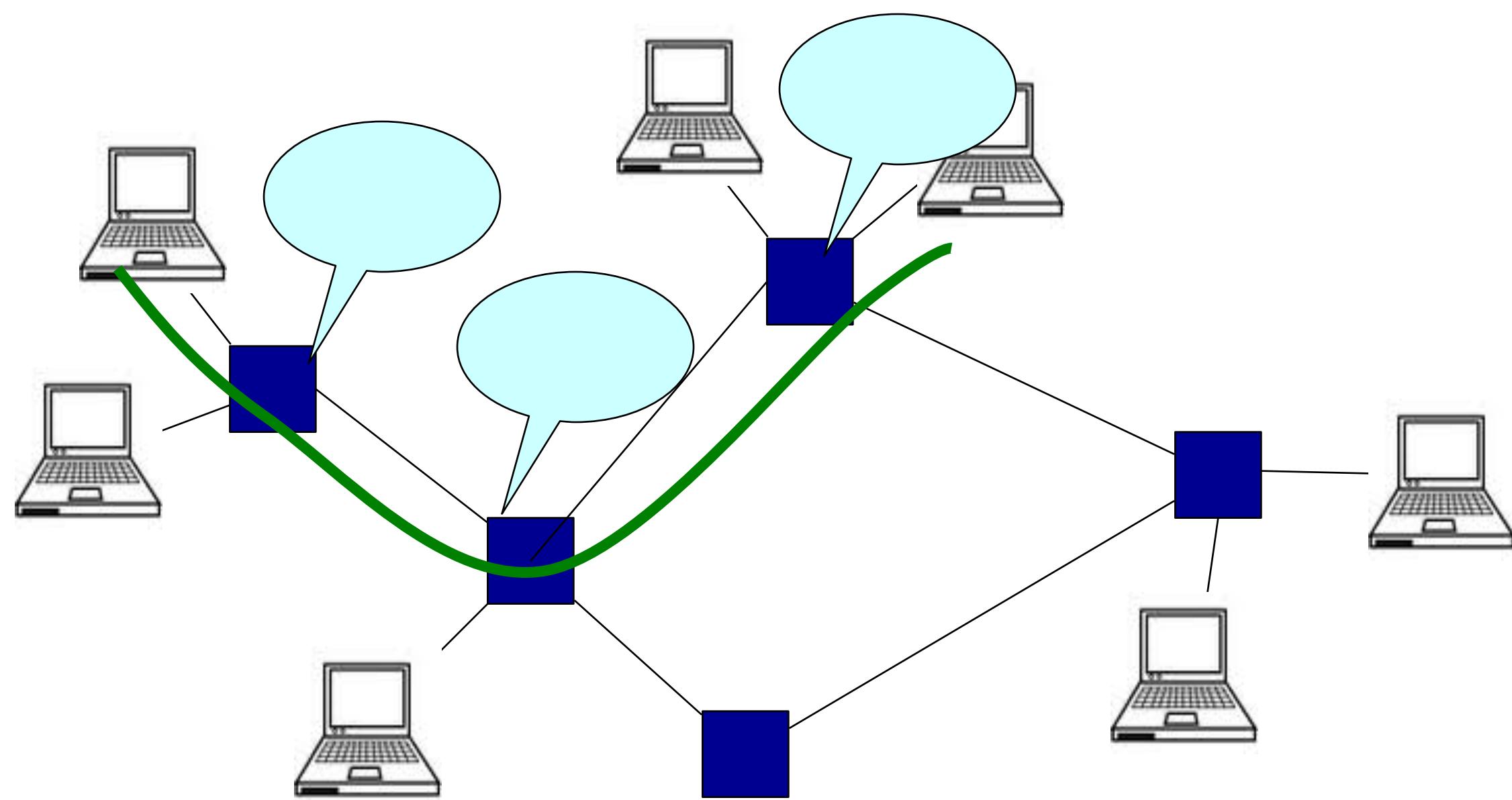


# Two forms of switched networks

- Circuit switching (used in the telephone network)
- Packet switching (used in the Internet)
- **Spoiler Alert: The Internet is Packet Switched. It does not support Circuited Networks. Let's see why.**



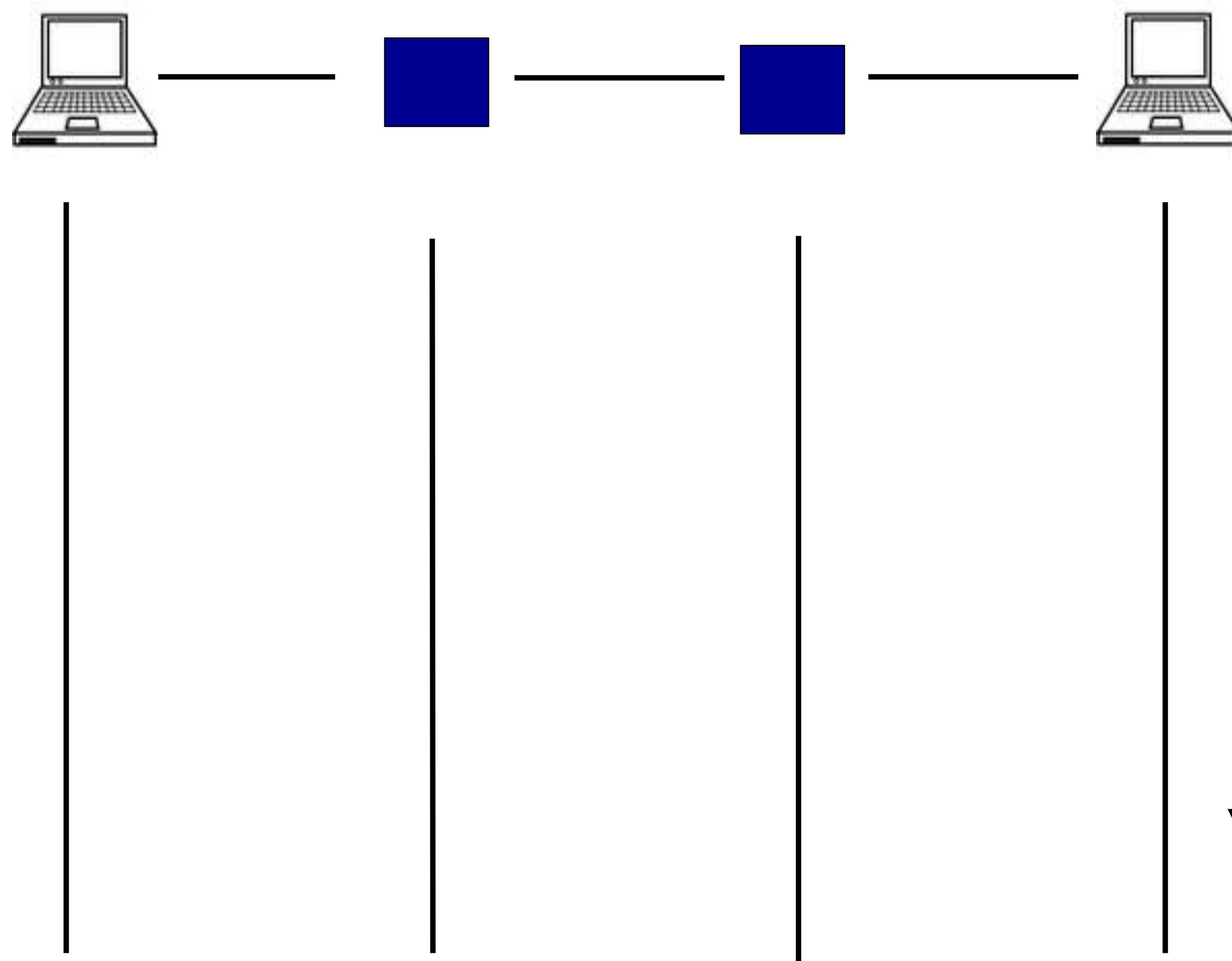
# Circuit switching

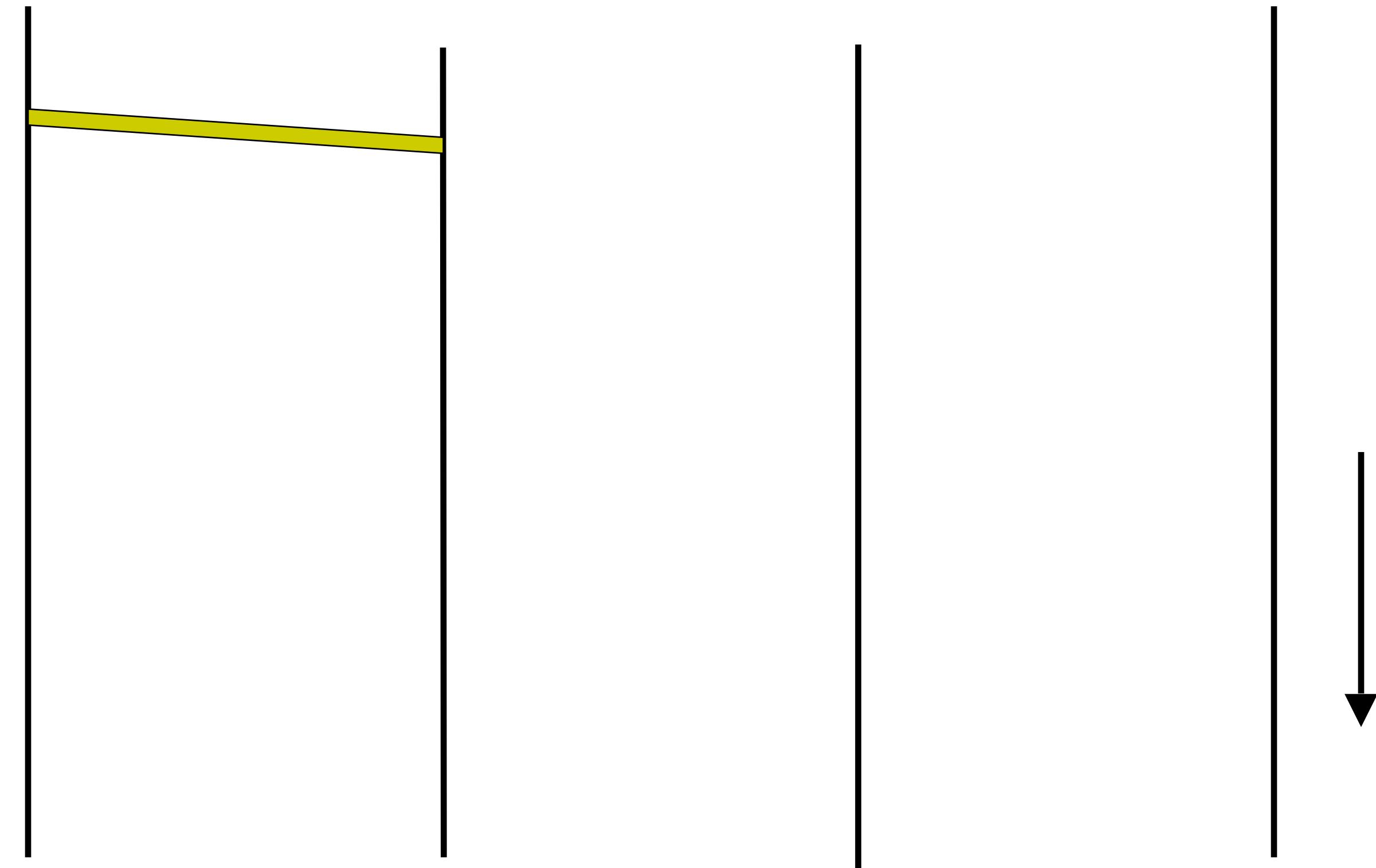
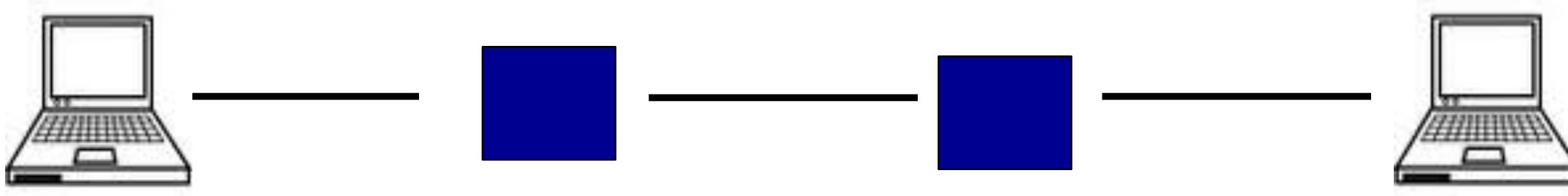


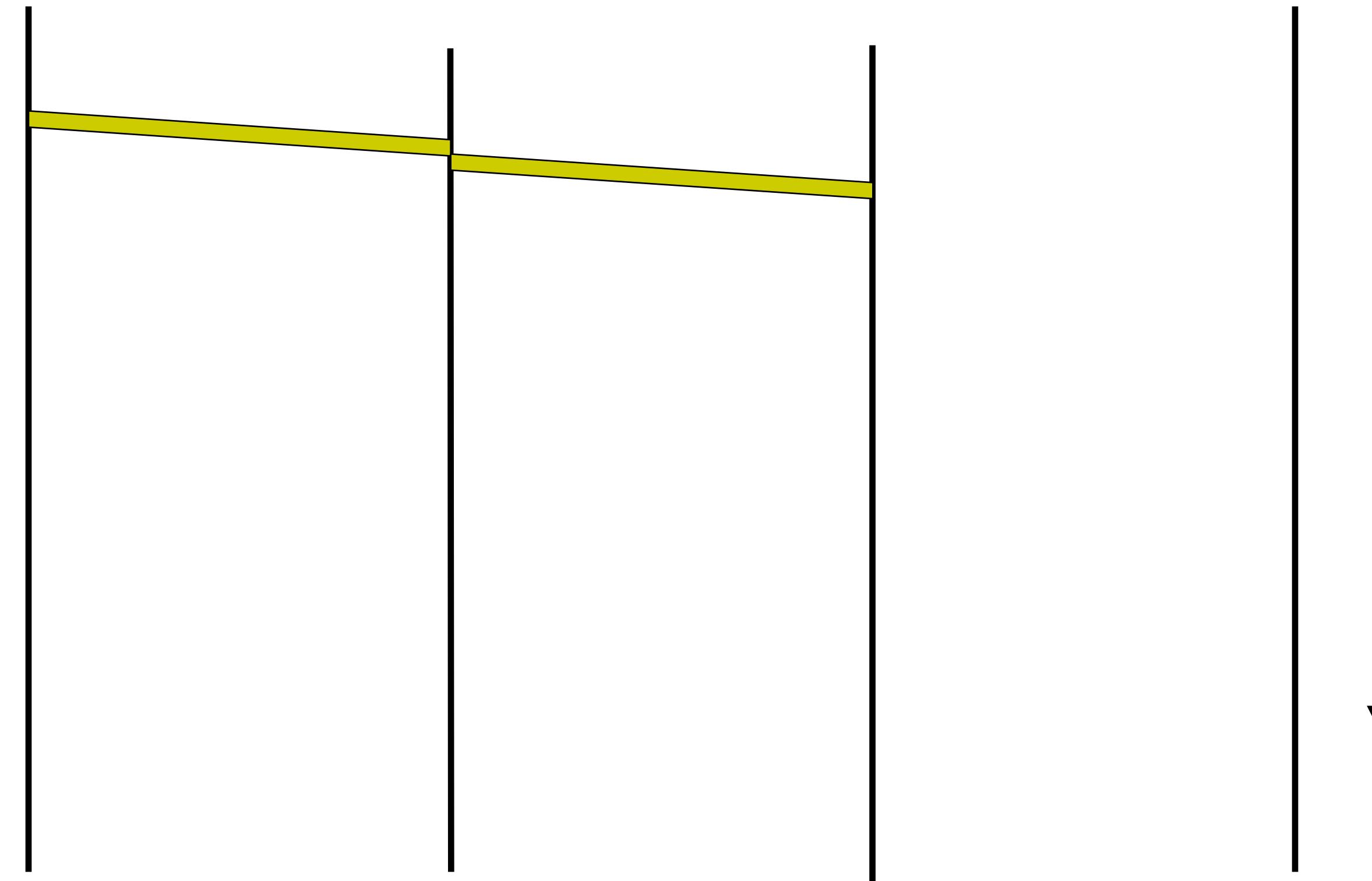
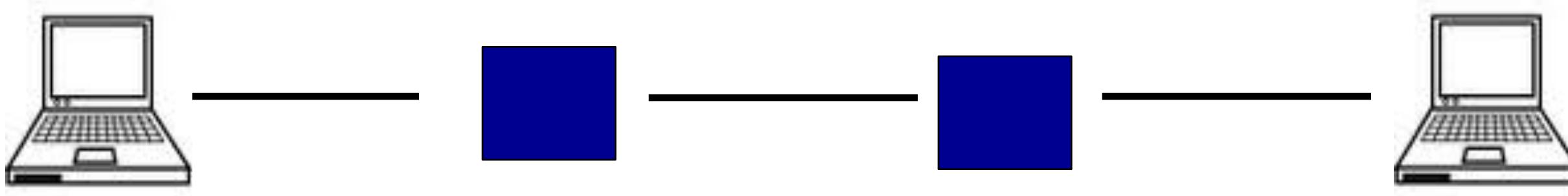
- (1) Node A sends a reservation request
- (2) Interior switches establish a connection -- i.e., “circuit”
- (3) A starts sending data
- (4) A sends a “teardown circuit” message

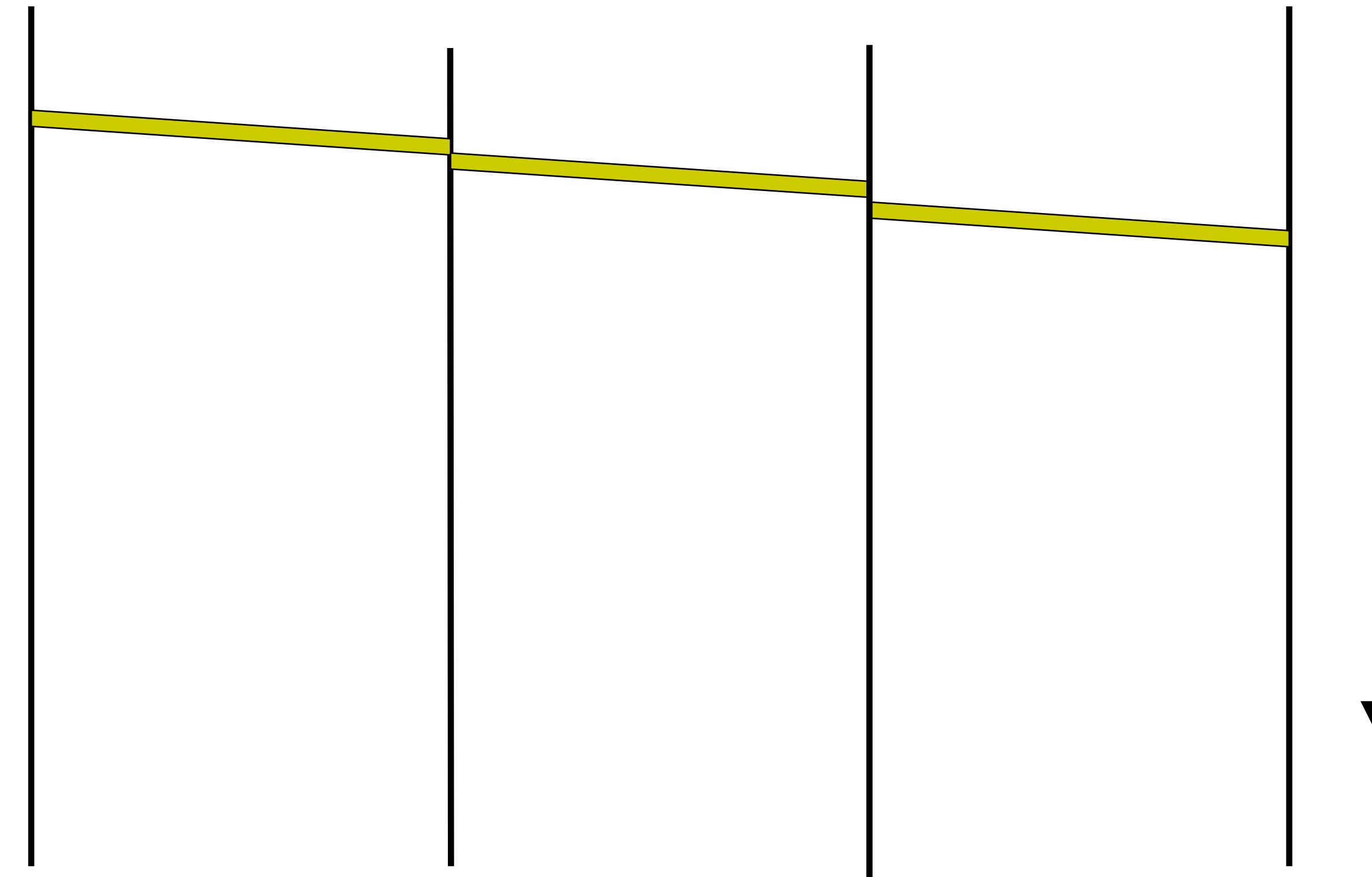
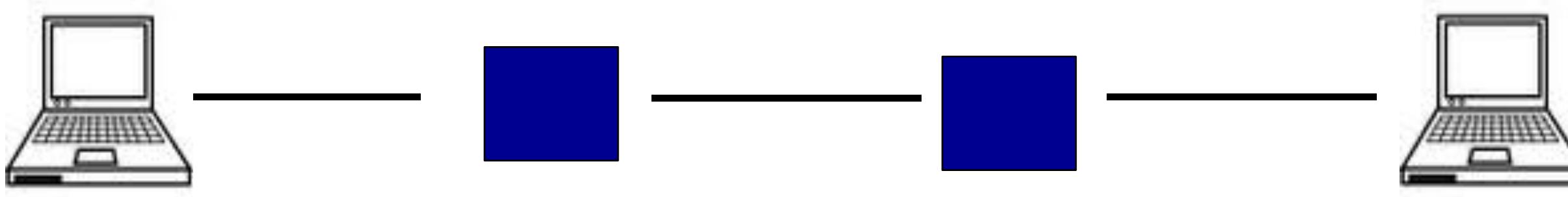


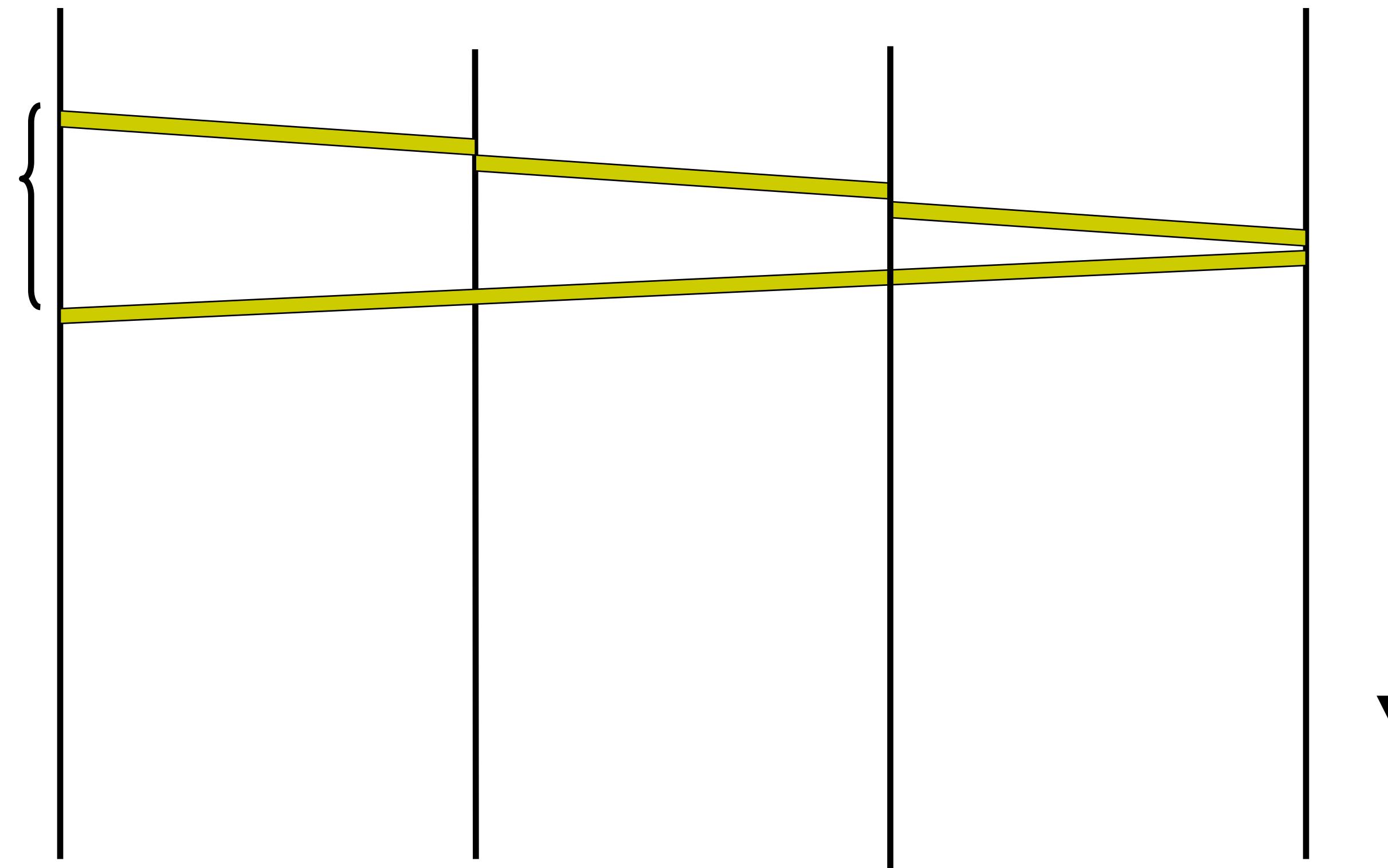
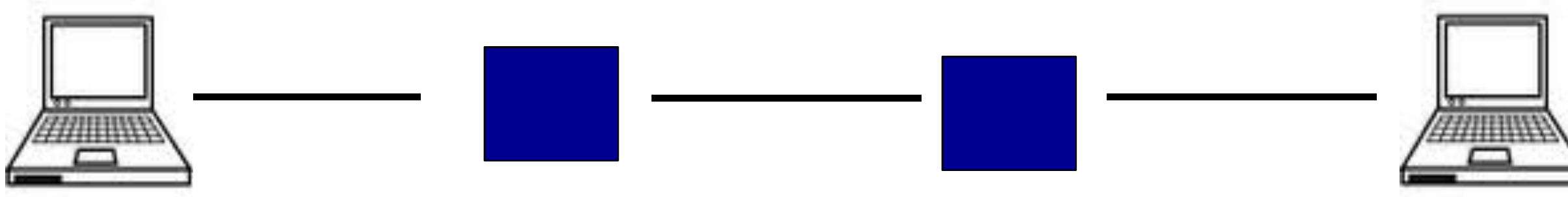
# Timing in Circuit Switching

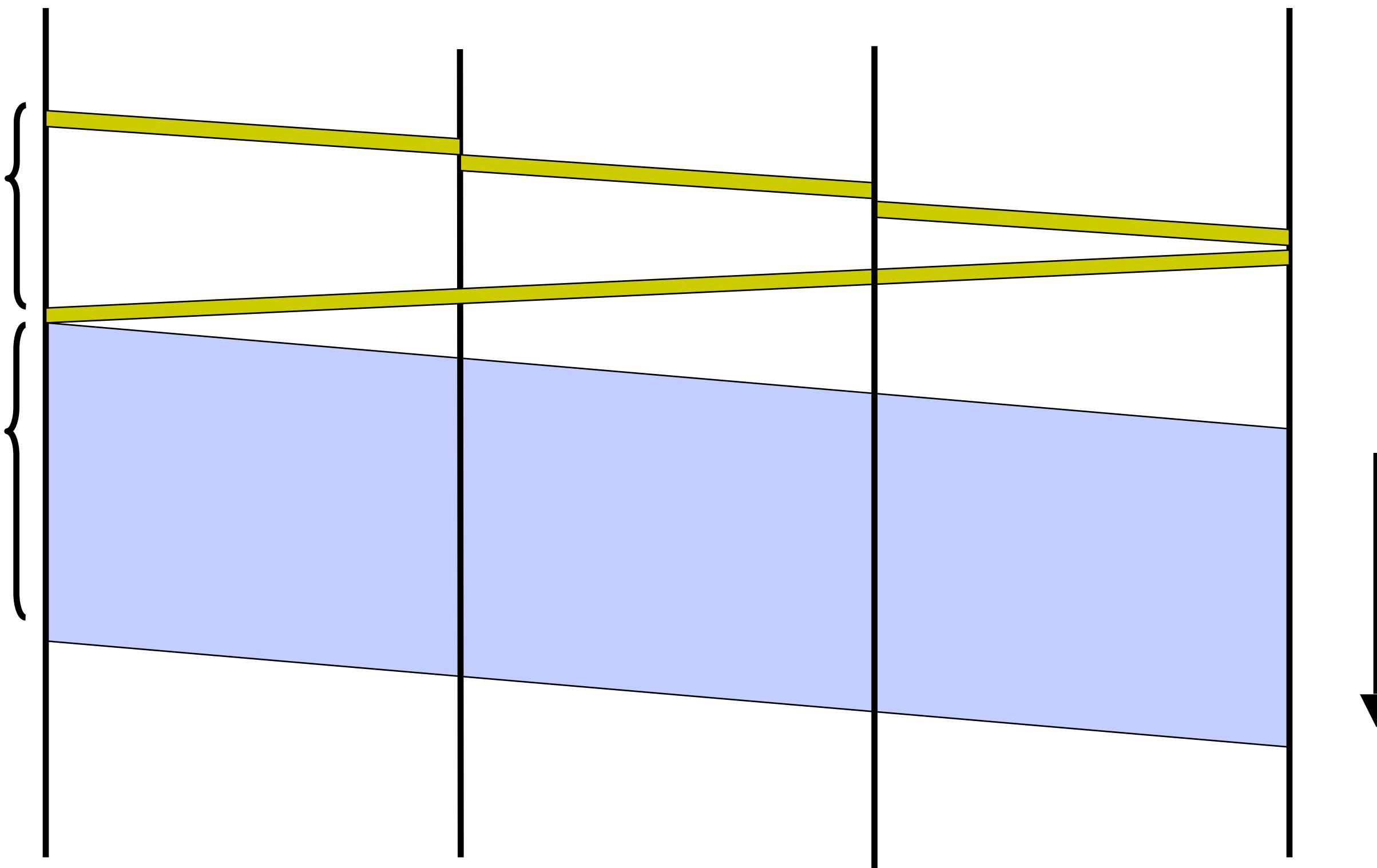
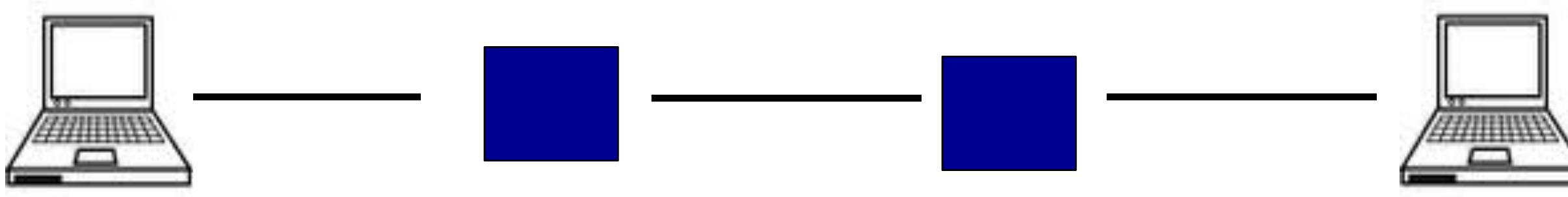


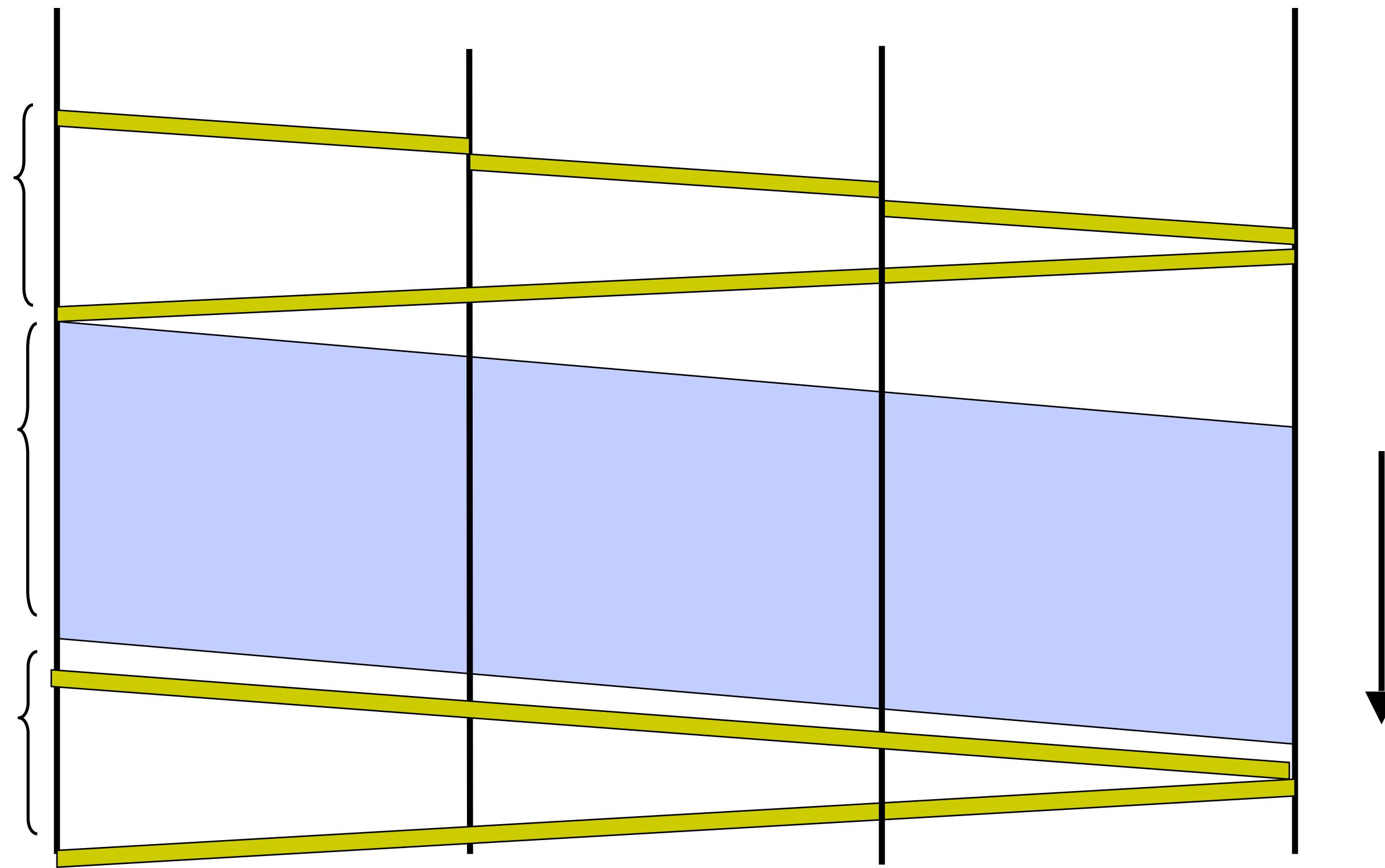
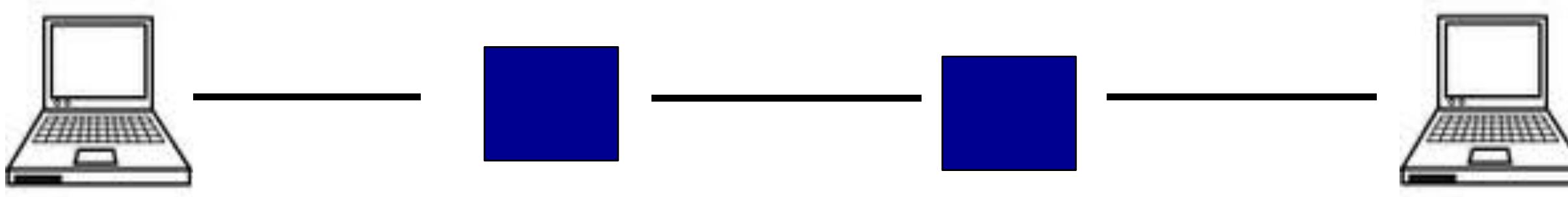








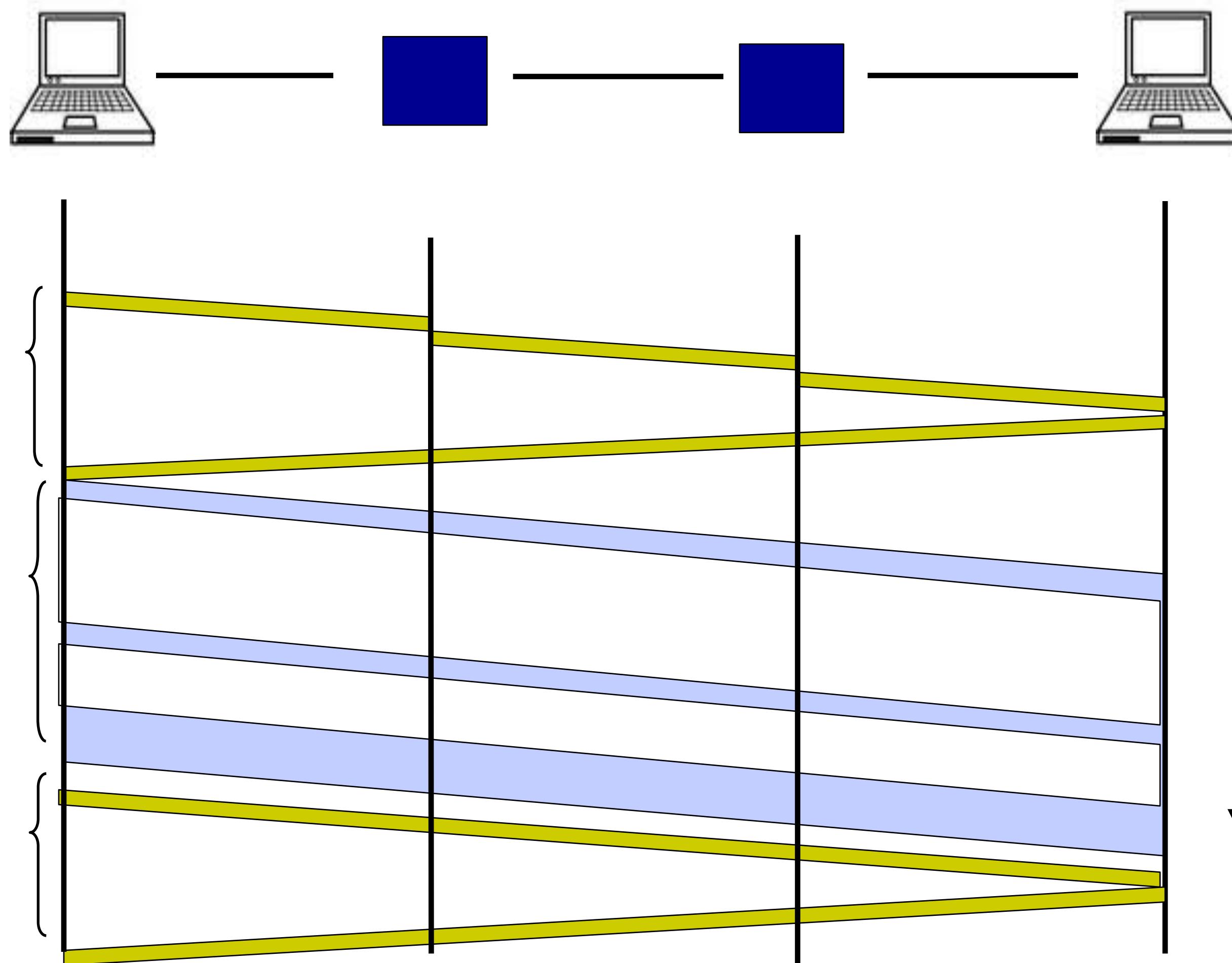




# Circuit switching: pros and cons

- Pros
  - guaranteed performance
  - fast transfer (once circuit is established)
- Cons

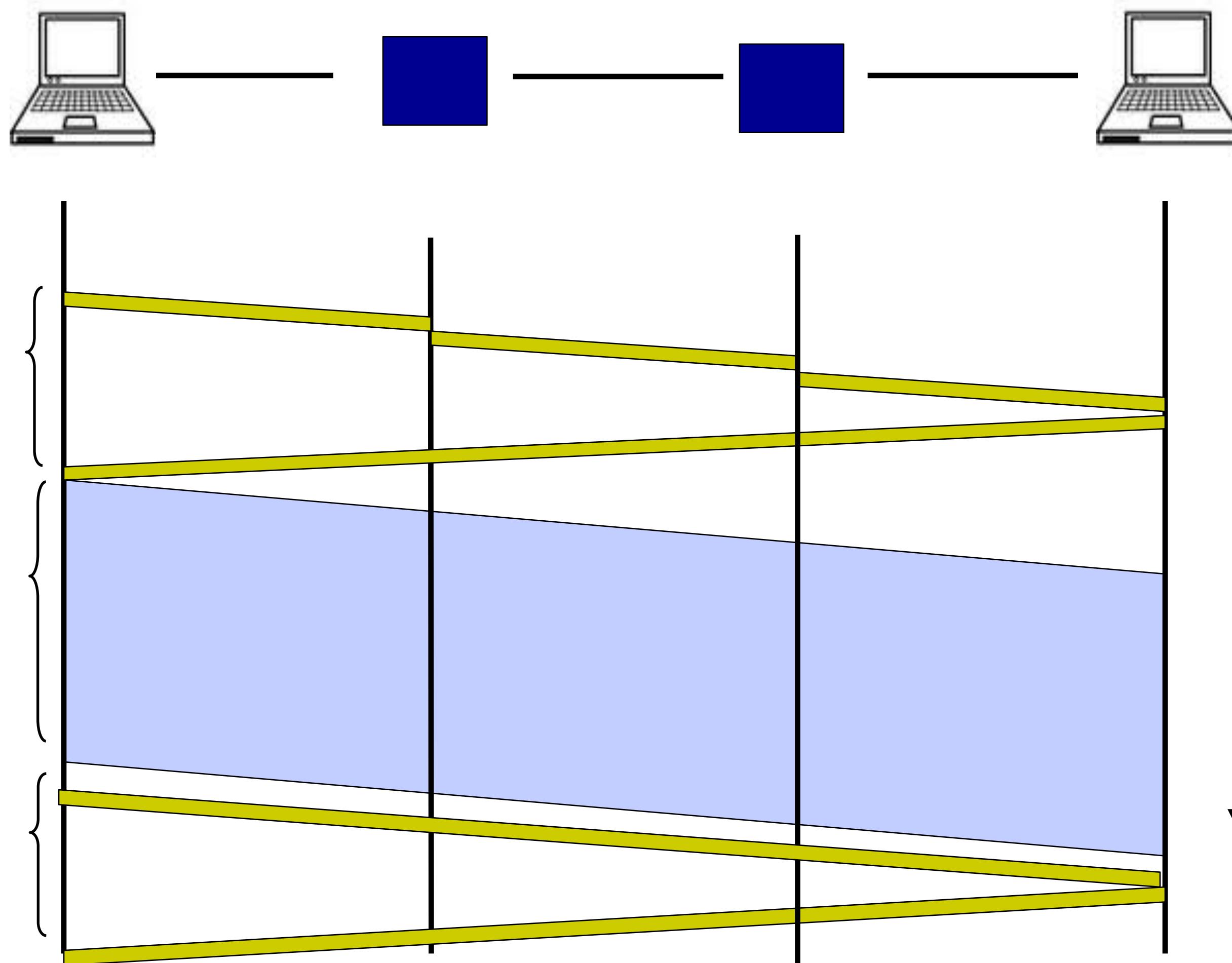
# Timing in Circuit Switching



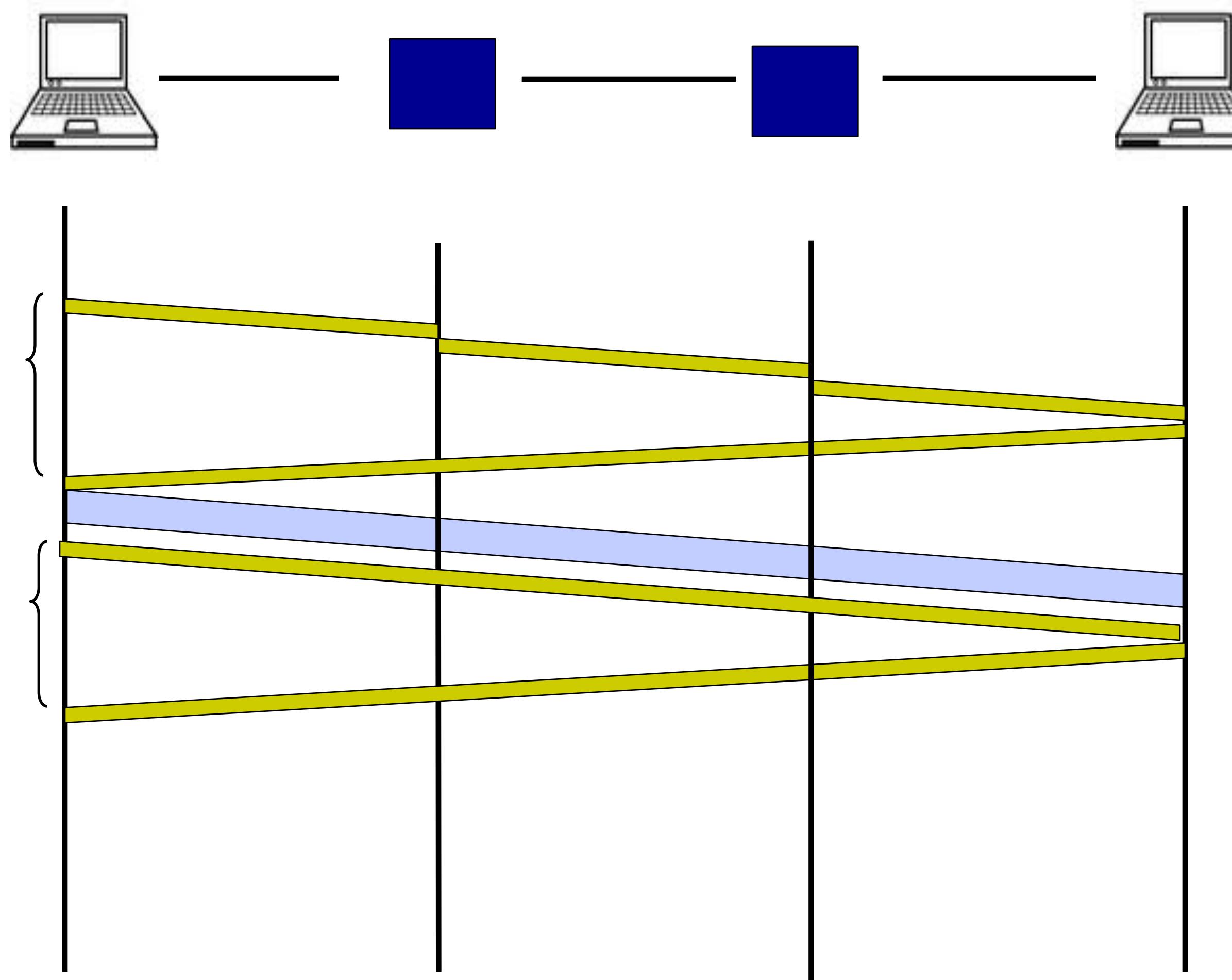
# Circuit switching: pros and cons

- Pros
  - guaranteed performance
  - fast transfer (once circuit is established)
- Cons
  - **wastes bandwidth if traffic is “bursty”**

# Timing in Circuit Switching



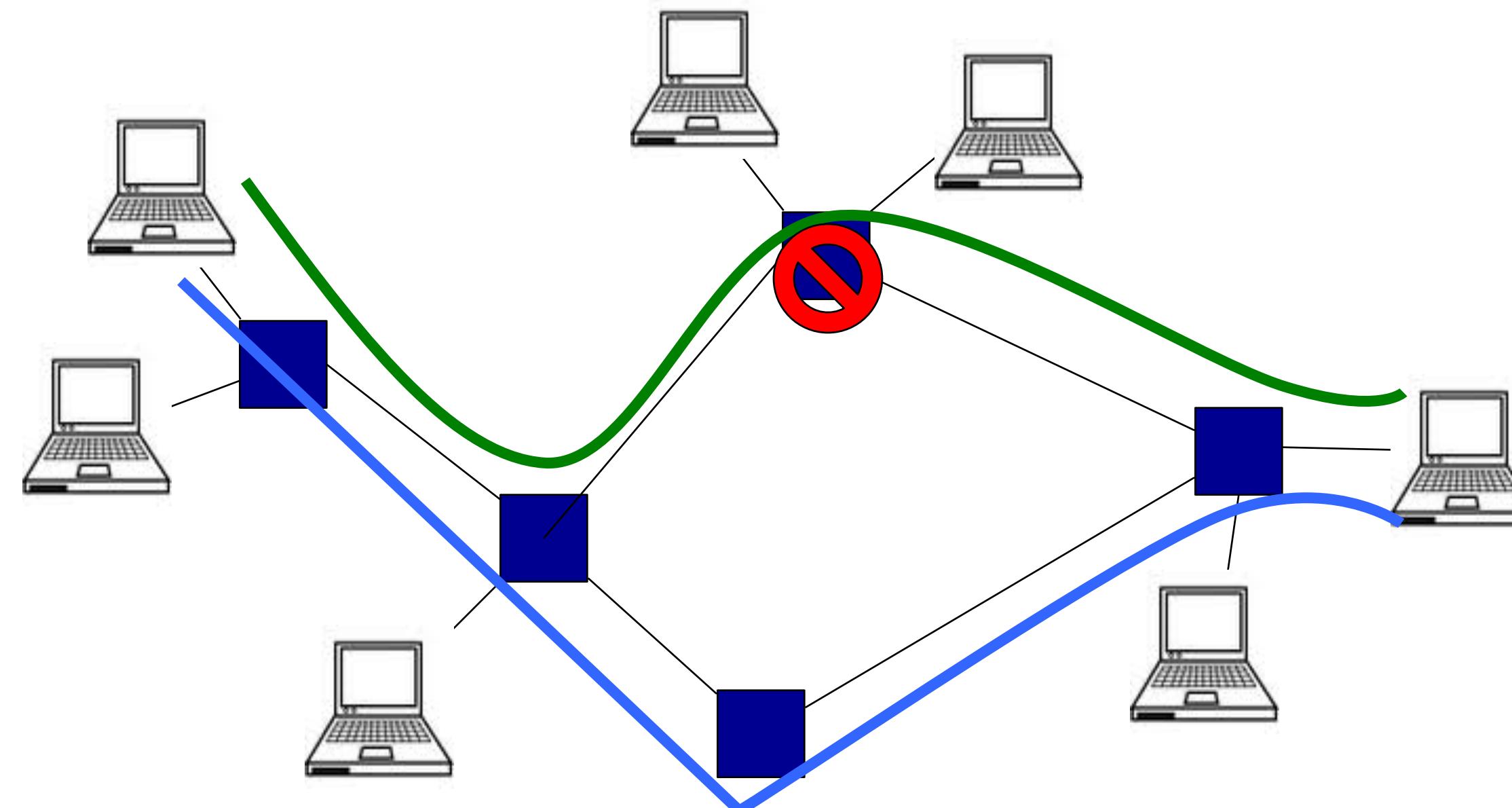
# Timing in Circuit Switching



# Circuit switching: pros and cons

- Pros
  - guaranteed performance
  - fast transfers (once circuit is established)
- Cons
  - wastes bandwidth if traffic is “bursty”
  - **connection setup time is overhead**

# Circuit switching



Circuit switching doesn't “route around trouble”

# Circuit switching: pros and cons

- Pros
  - guaranteed performance
  - fast transfers (once circuit is established)
- Cons
  - wastes bandwidth if traffic is “bursty”
  - connection setup time is overhead
  - **recovery from failure is slow**



# Two forms of switched networks

- Circuit switching (e.g., telephone network)
- Packet switching (e.g., Internet)



# Packet Switching

- Data is sent as chunks of formatted bits (**Packets**)
- Packets consist of a “**header**” and “**payload**”\*
  - payload is the data being carried
  - header holds instructions to the network for how to handle packet
- For now, think of the header as an envelope
  - To: Professor Steenkiste
  - From: Professor Sherry

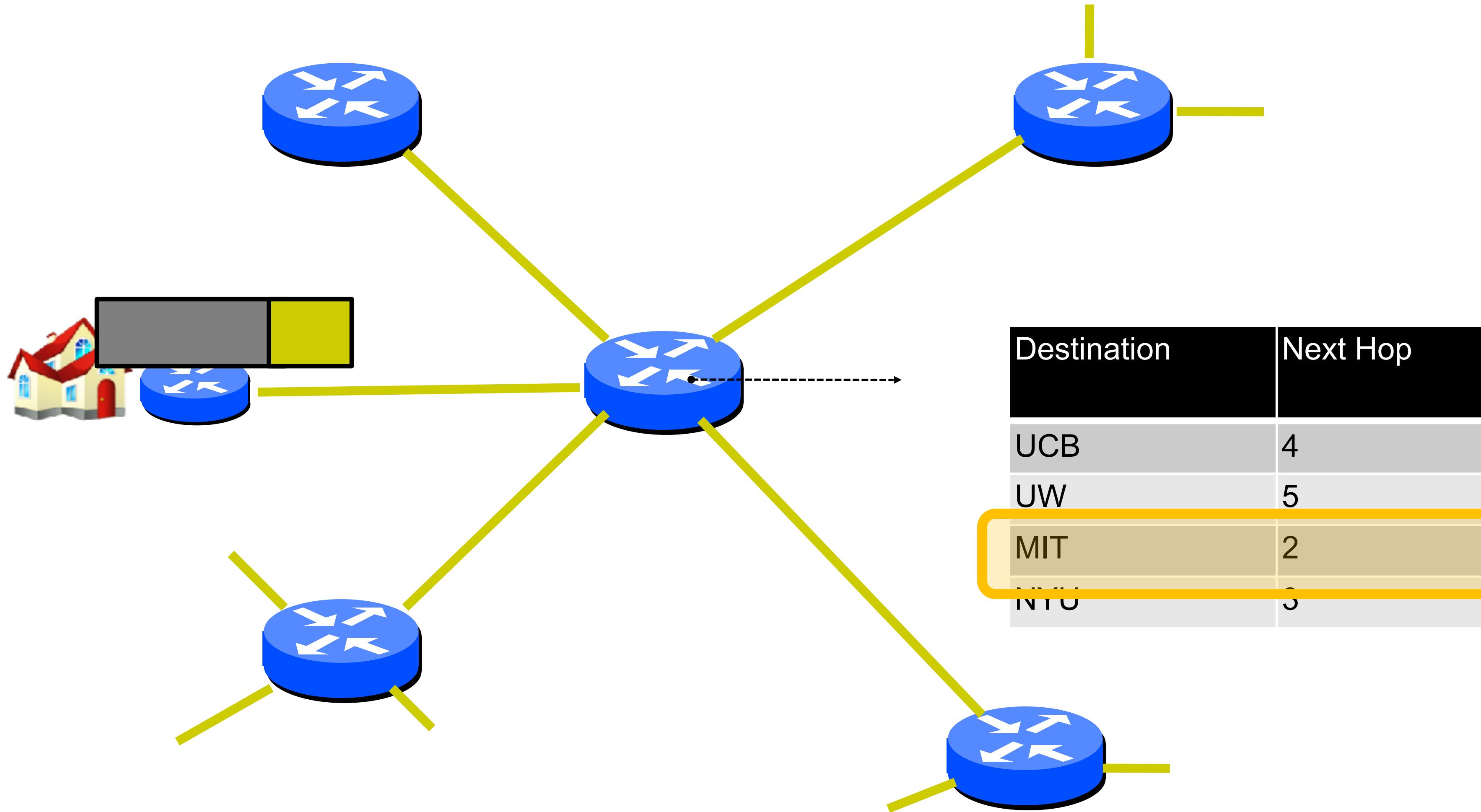


# Packet Switching

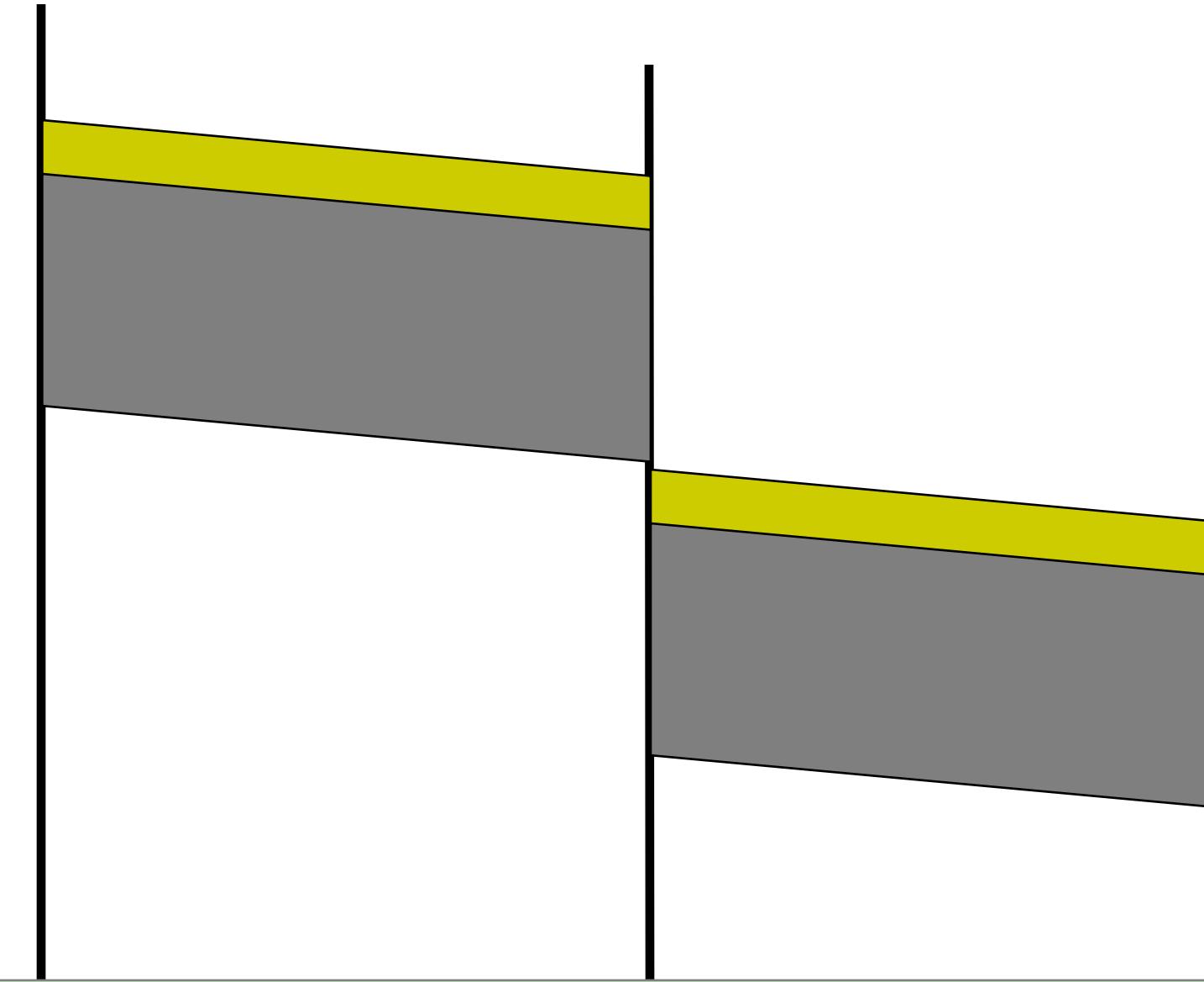
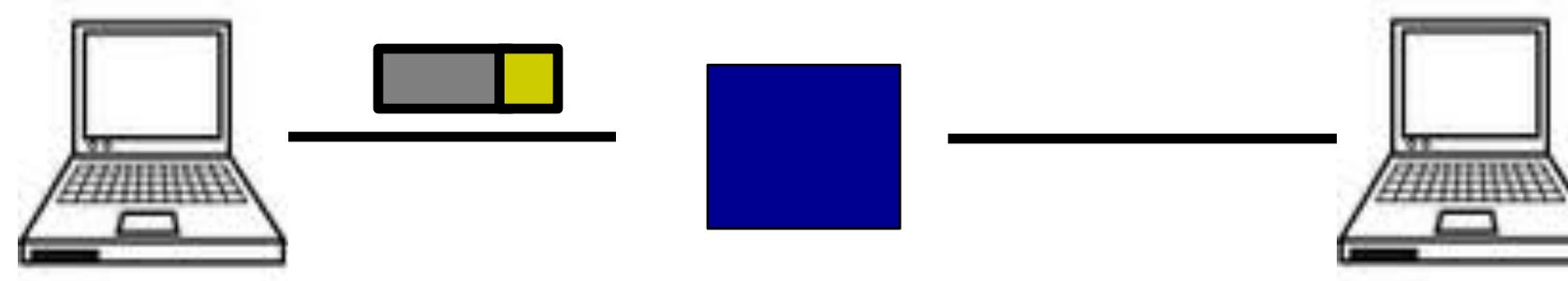
- Data is sent as chunks of formatted bits (Packets)
- Packets consist of a “header” and “payload”
- Switches “forward” packets based on their headers



# Switches forward packets



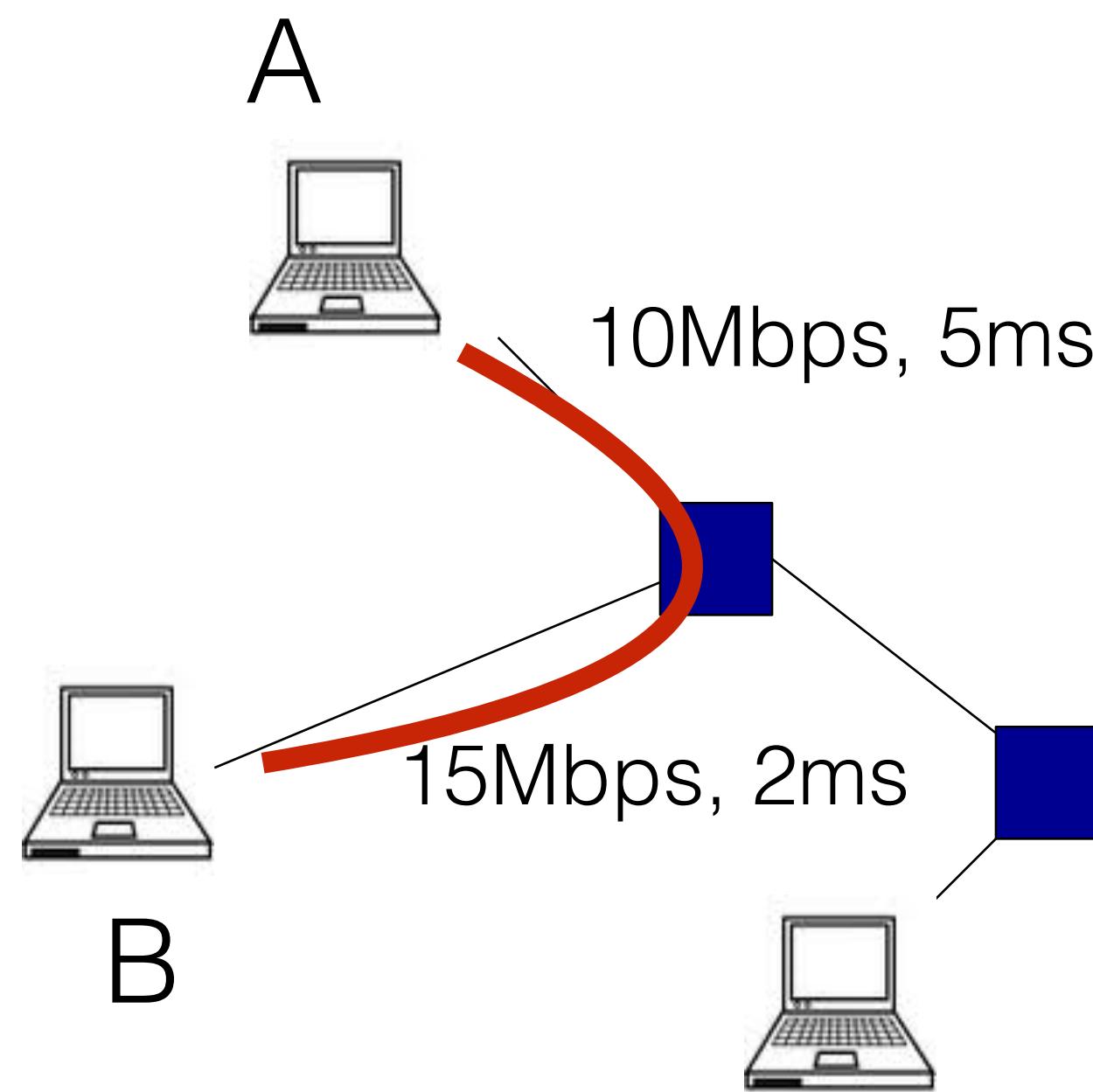
# Timing in Packet Switching



The switch in the middle waits for the whole packet to arrive. Then, it reads the packet header. Then, it re-sends the packet to the next stop.

# Your Turn

Packet Delay = (Data Size ÷ Link Bandwidth) + Link Latency



How long does it take for a 1000 Byte packet to travel from A to B?

The packet will traverse two links, the first is 10Mbps with 5ms delay, and the second is 15Mbps with 2ms delay.



# Packet Switching

- Data is sent as chunks of formatted bits (Packets)
- Packets consist of a “header” and “payload”
- Switches “forward” packets based on their headers



# Packet Switching

- Data is sent as chunks of formatted bits (Packets)
- Packets consist of a “header” and “payload”
- Switches “forward” packets based on their headers
- Each packet travels independently
  - no notion of packets belonging to a “circuit”

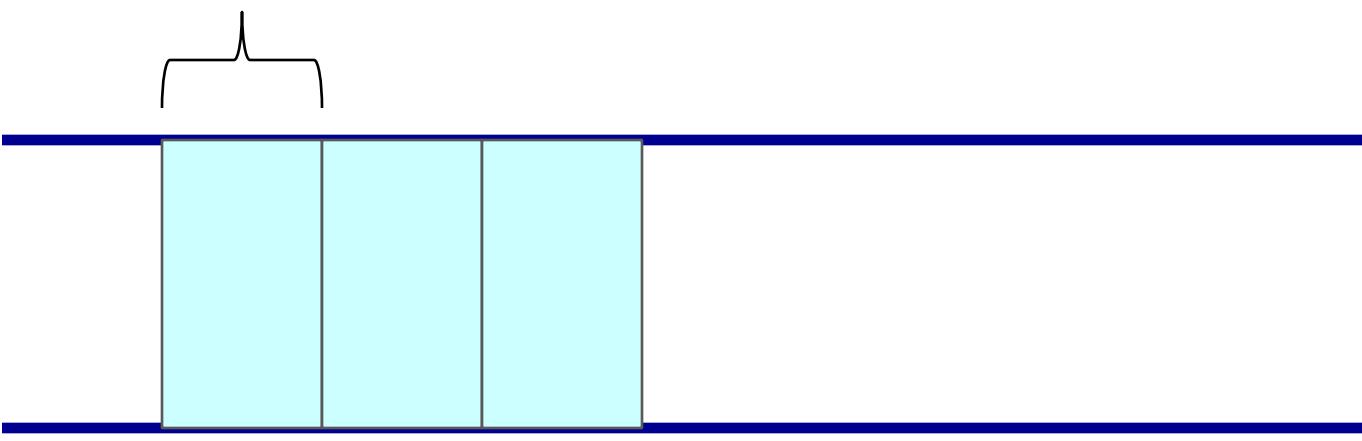


# Packet Switching

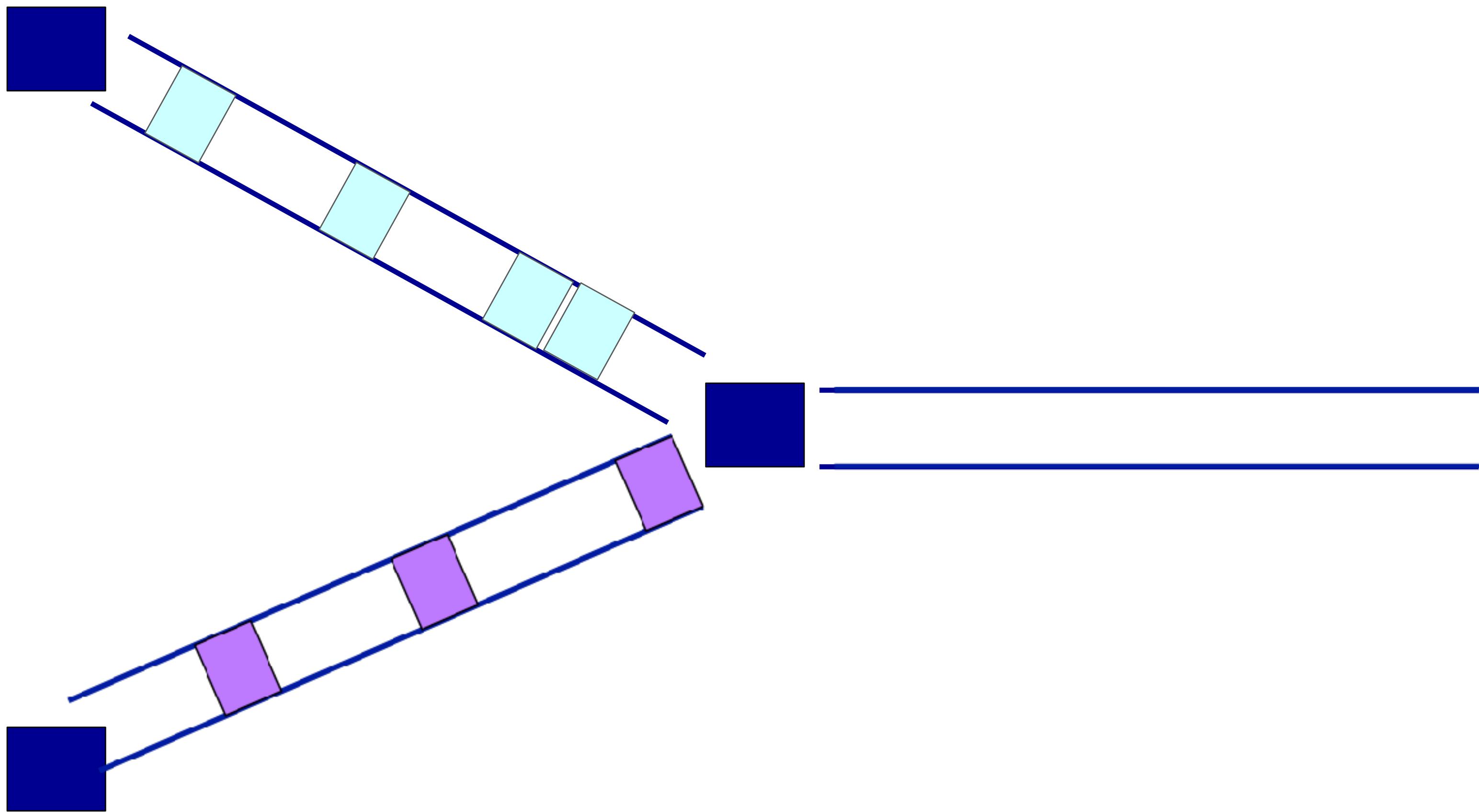
- Data is sent as chunks of formatted bits (Packets)
- Packets consist of a “header” and “payload”
- Switches “forward” packets based on their headers
- Each packet travels independently
- No link resources are reserved in advance. Instead packet switching leverages **statistical multiplexing**



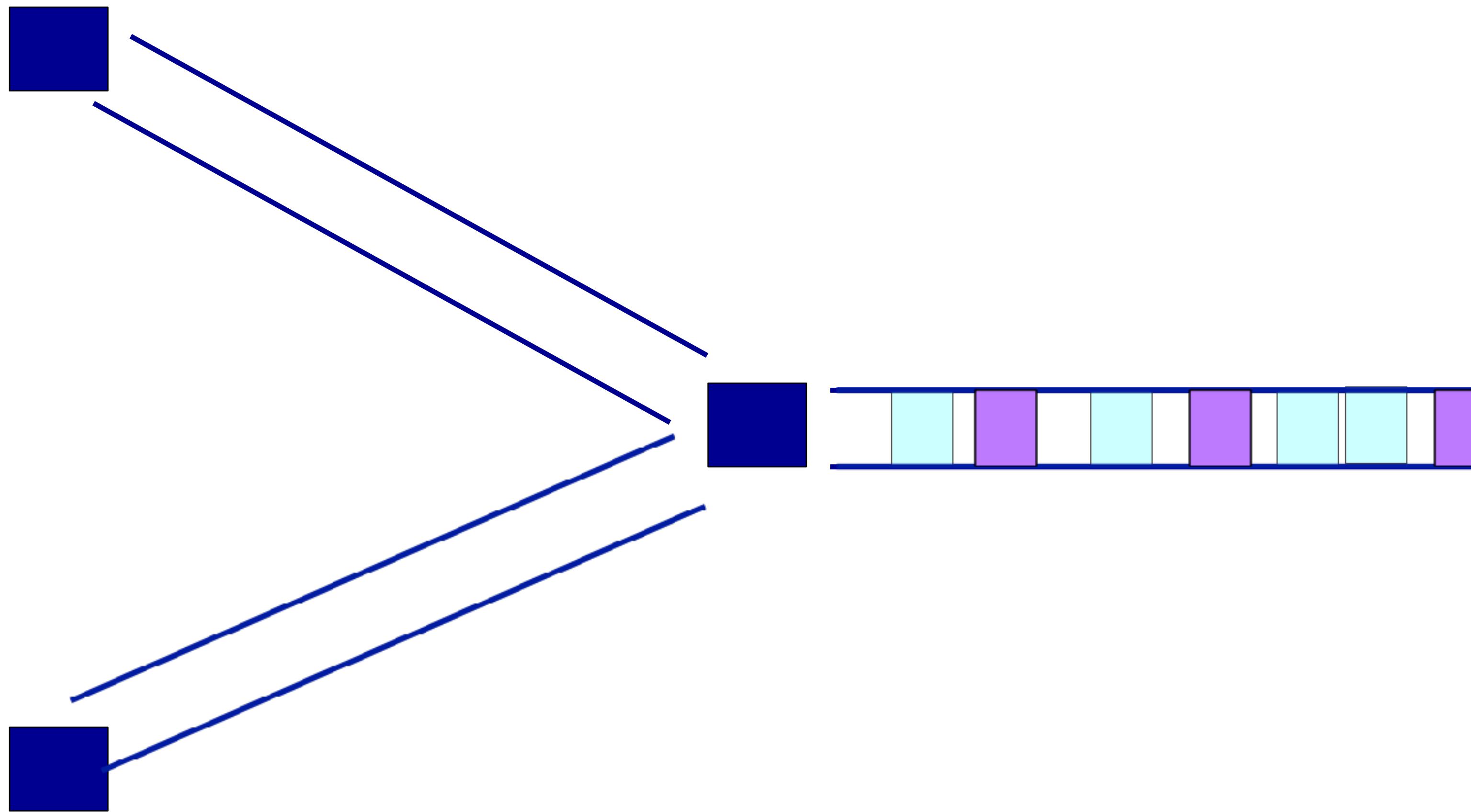
# Statistical multiplexing: pipe view



# Statistical multiplexing: basic view



# Statistical multiplexing: basic view



# Statistical Multiplexing is a recurrent theme in computer science

- Circuit vs Packet Switched Networks
  - Packet switched won for the Internet
- Cloud computing reserved servers vs shared servers and VMs
  - Few people buy and reserve racks of computers anymore...



# Back To Networks

- What do we know now?
  - **Networking connects all computer systems that matter.**
  - But “networks” are very different!
    - Different physical media (... the Internet connects them anyway)
    - Different bandwidth and latency properties (... the Internet connects them all anyway)
    - Packet vs Circuit Switching (... the Internet only supports packet switching!)



# The Rest of This Course

- Part 1: How does the Internet interconnect such diverse networks?
  - At scale?! Without just breaking?! Without central authority?!
- Part 2: How do we build applications that cope with diverse networks?
- Part 3: How do we build and manage different kinds of networks?



Homework 1 is so easy. Turn it in by Friday!

