

# **CS 168**

# **Introduction to the Internet: Architecture and Protocols**

Fall 2016

Scott Shenker

CS168.io

<http://inst.eecs.berkeley.edu/~cs168/fa16/>



Sssshhhhhhhhhhhhhhhhhhh

# Today: answer 8 basic questions

- What is this course about?
- What does the Internet look like?
- Why study the Internet?
- Why are networking courses so bad?
- What is the course workload, grading, etc?
- Who am I and how do I teach?
- What do I expect from you?
- **Is CS168 the right class for you?**

# Question I Won't Answer

- How do I get off the waiting list?

# Questions Answered in Real-time!

- As an **experiment**, one or more TAs will be on Piazza during lecture
- If you have quick question and don't want to ask me, then ask on Piazza
- Don't use this for deep conceptual questions:
  - Ask those of me, because you can't be the only one who is confused
- But if you missed something in passing, ask online

**What is this course about?**

# There are many kinds of networks

- Telephone (landline) networks
- Cellular networks
- WiFi networks
- Ethernets
- Infiniband
- Supervisory control and data acquisition
- Frame relay networks, optical networks, ATM.....
- **We won't study any of them....**

Class will focus almost exclusively on the **Internet**

# Networks versus “The Internet”

- The Internet is not a particular kind of network
  - It is not just one more technology on the list
- The Internet ties different networks together
  - The *Internet*
- ***Why does this matter?***



# Goals for a network technology

- Speed (bits per second)
- Cost
- Port-density (# of links into a single switch)
- Reliability (mean time between failures)
- Other “features” specific to that network
  - Infiniband: no packet drops
  - Ethernet: no configuration (L2)
  - ...

# Goals for the Internet

- Ability to connect many different networks
  - Ethernet and optical and ATM and ....
- Ability to scale to entire world
  - Geographically and numerically
- Ability to tolerate and recover from failures
  - Which are inevitable...
- .....

***These are vague but more interesting goals!***  
*(more architectural than engineering)*

# Architecture vs Engineering

- Architecture:
  - The allocation of functionality and definition of interfaces among elements
- The Internet “architecture” is the decision about what tasks get done, and where:
  - **What**: Delivering packets or files? Reliably or not?
  - **Where**: In the network, or in the hosts?
- Engineering is about how tasks get done
  - The particular algorithms or protocols

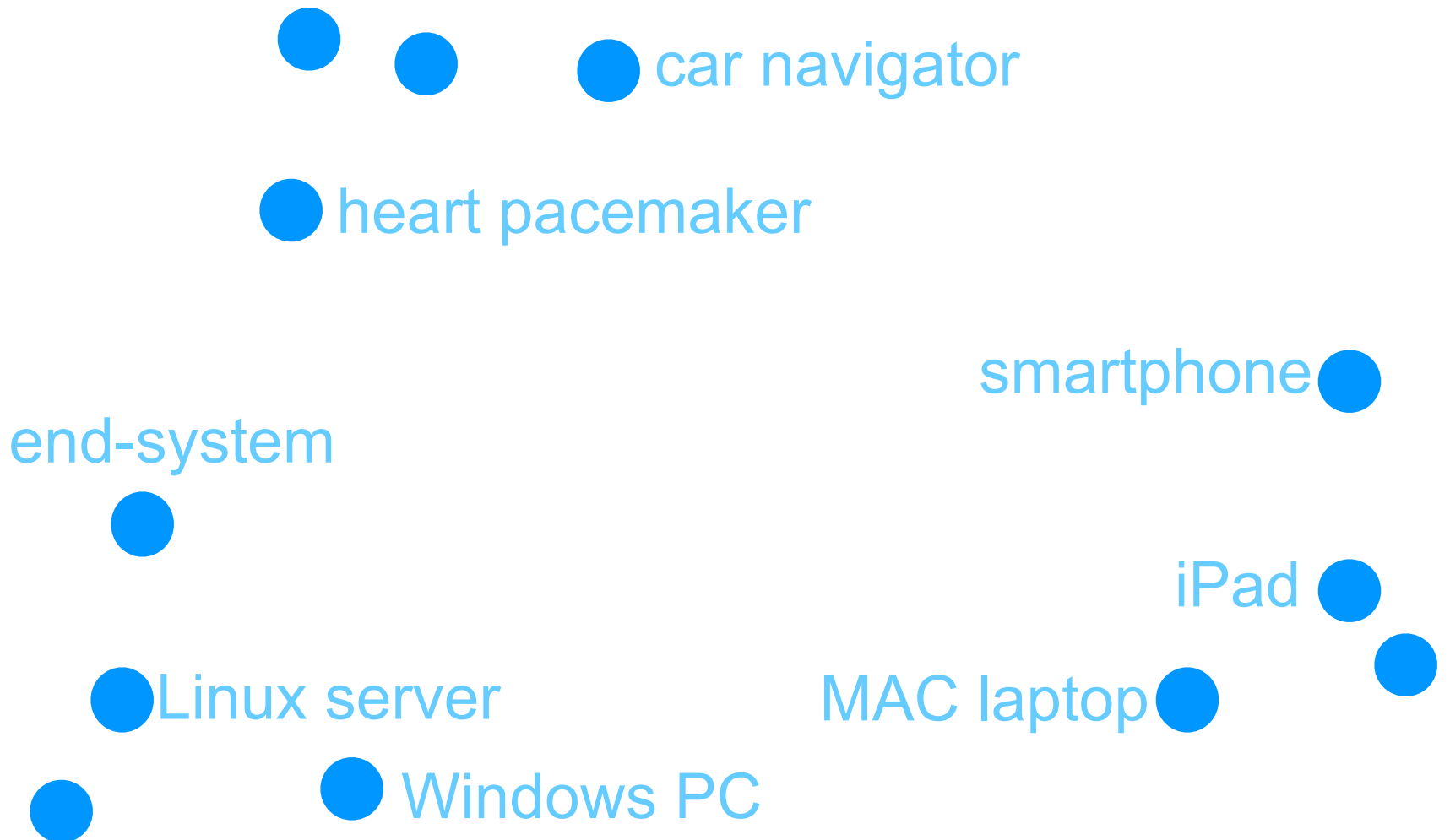
# Architecture: My Obsession

- Architectural decisions play a crucial role in scaling, heterogeneity, robustness, etc...
- True for Internet, and systems more generally
- ***This is what I spend my life worrying about***
  - It is what interests me
  - And I'm not good at the other stuff (algorithms, etc.)

# **What Does the Internet Look Like?**

# Basic Networking Components

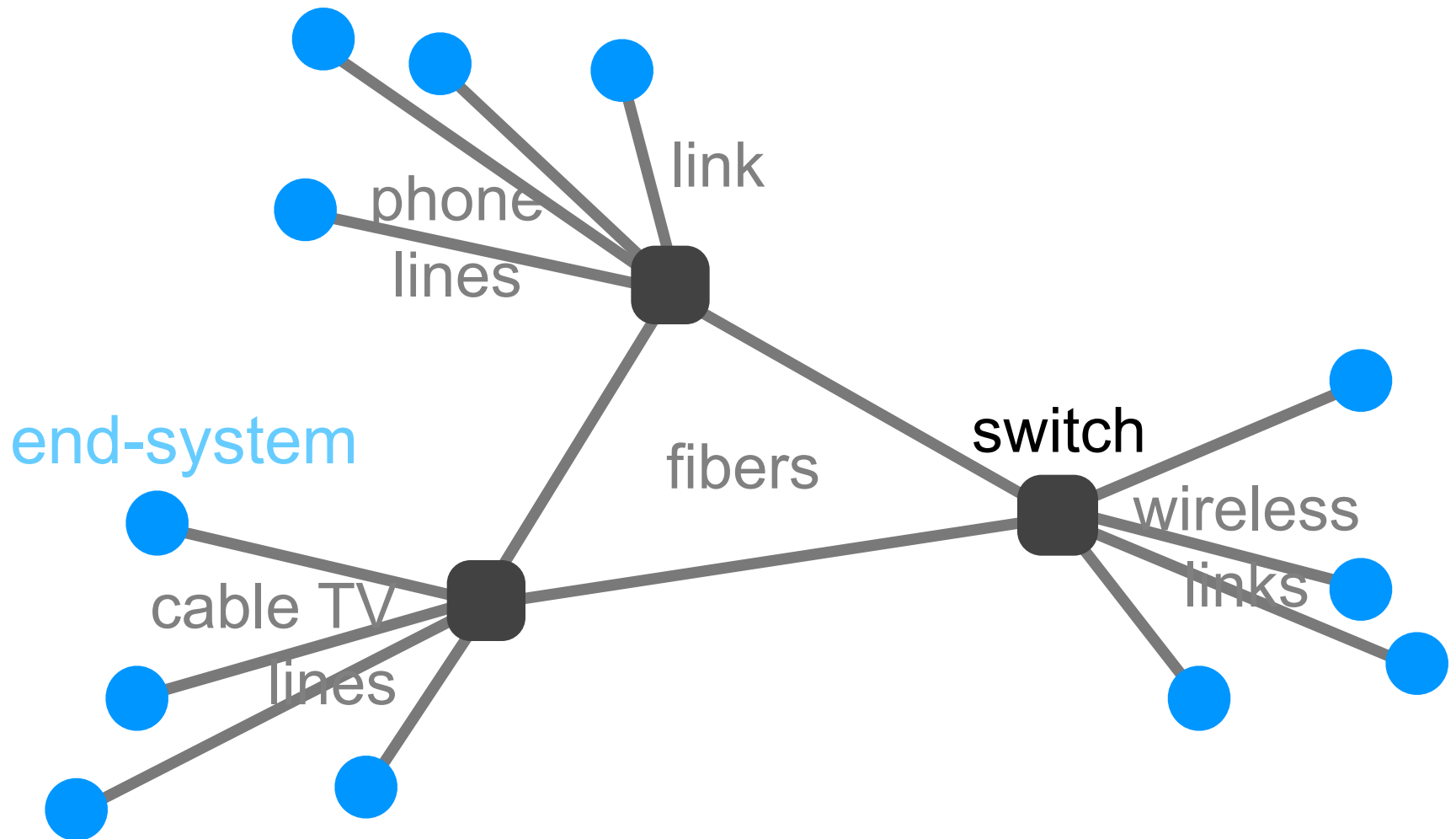
- **End systems:** they send/receive packets
- **Switchers/routers:** they forward packets
  - Routing decisions ensure packets reach destination
  - Will use terms **switch** and **router** interchangeably
    - Next lecture will explain technical difference
- **Links:** connect end systems to switches, and switches to each other



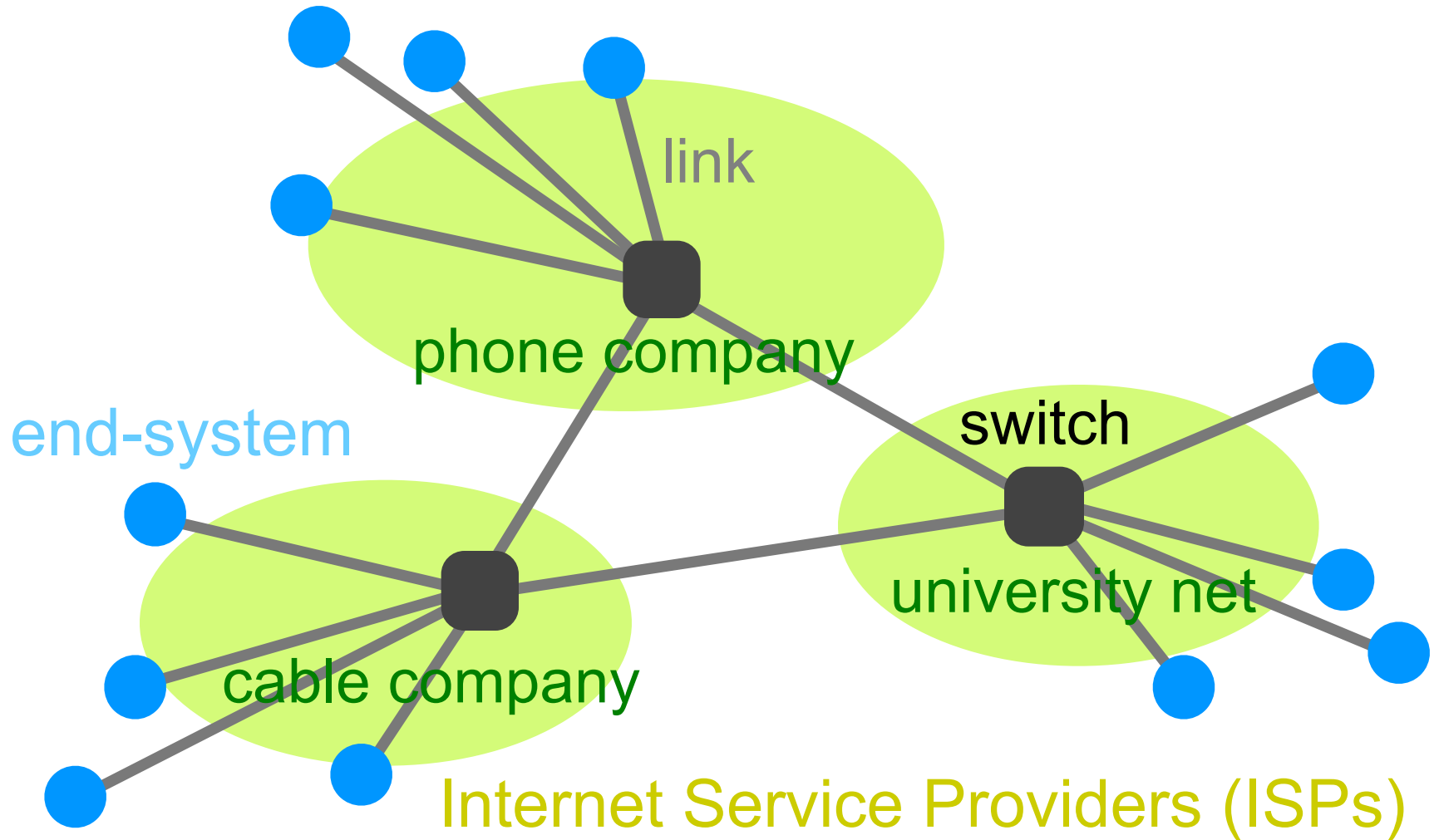
end-system

switch

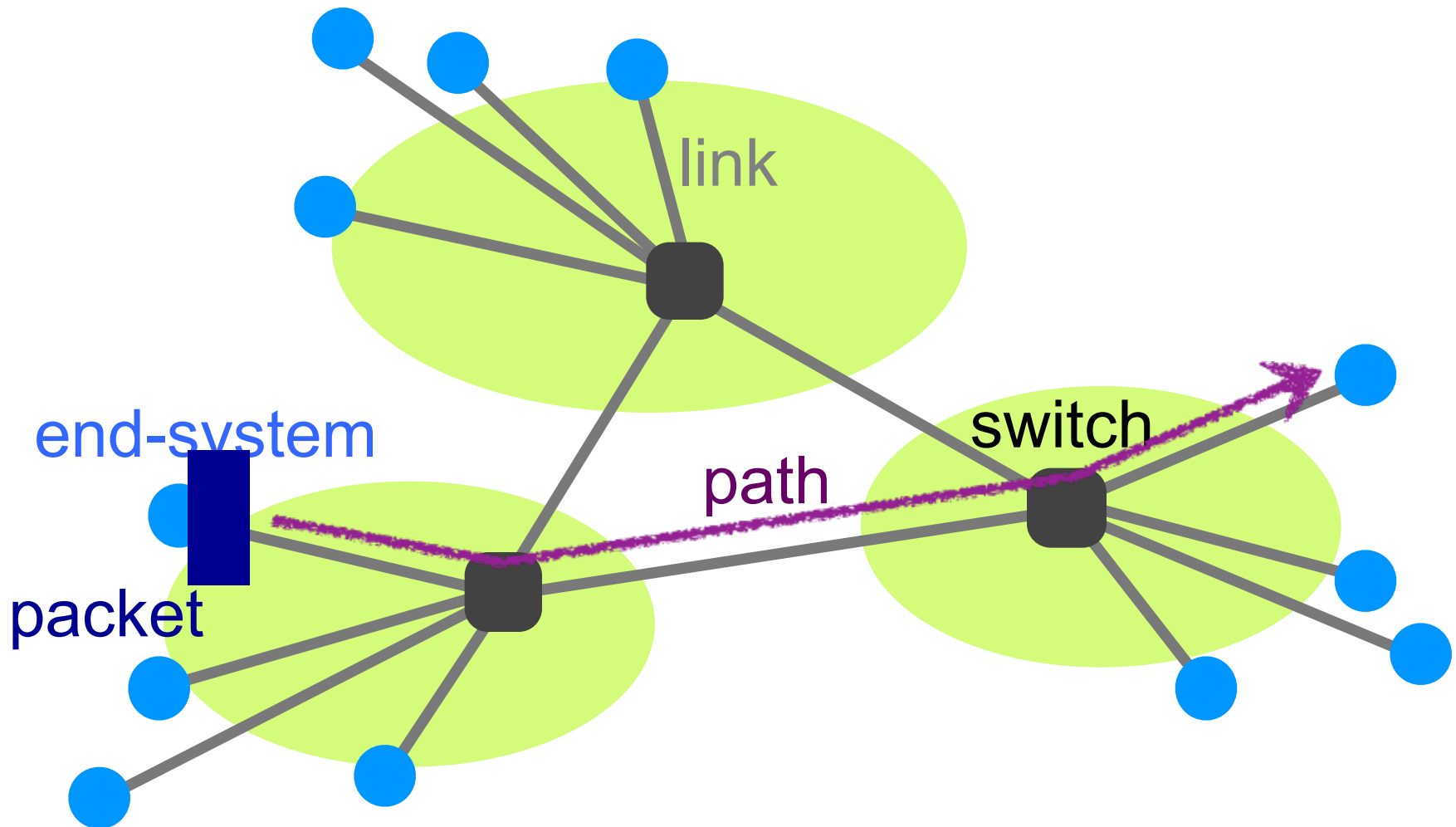




# Who runs the network?



# What's the purpose of the Internet?



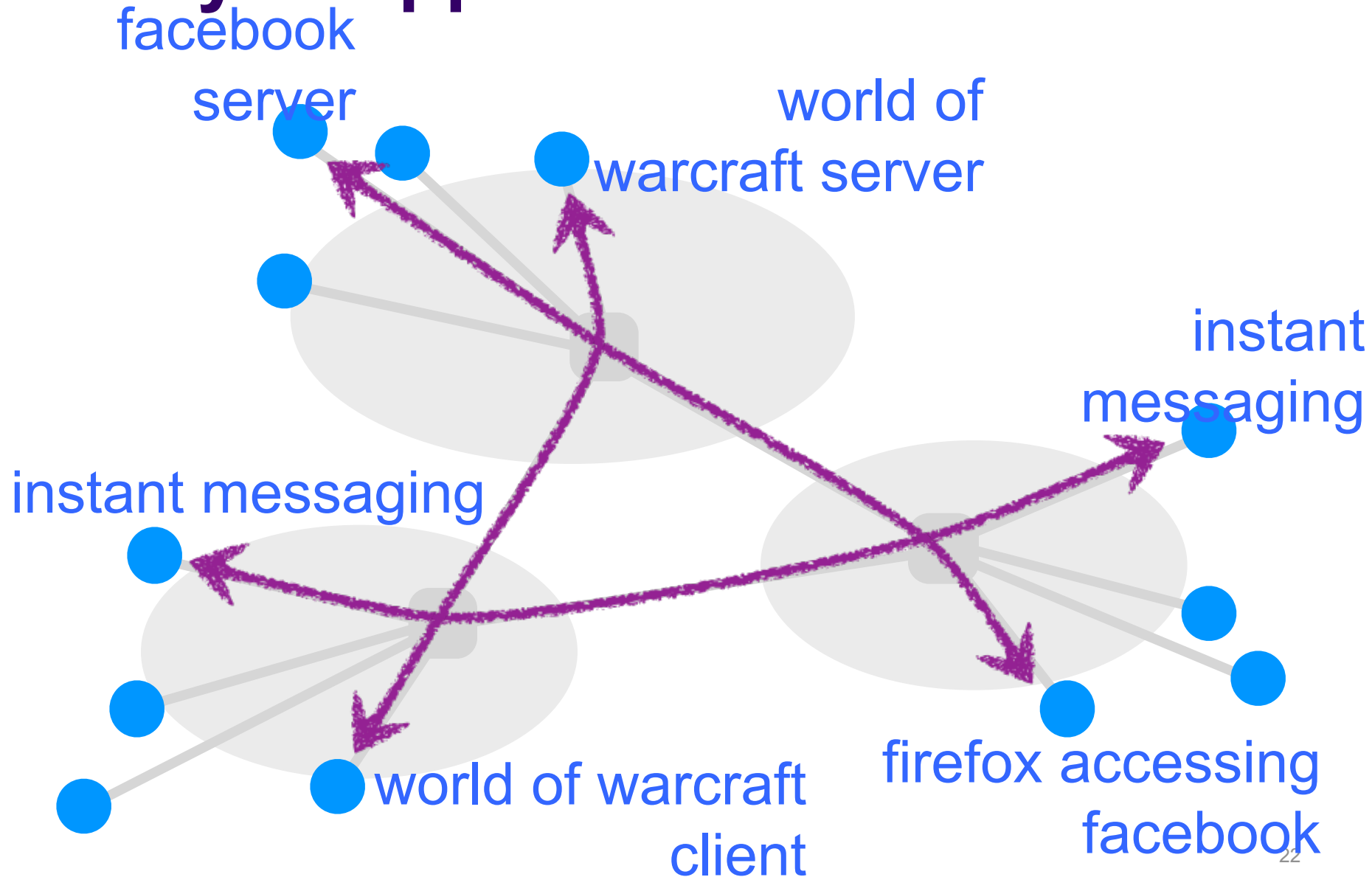
# Underlying purpose of networks...

- Support the logical equivalent of IPC
  - (John Day)
- IPC is set of mechanisms that allow processes on the same host to exchange data
- The network merely allows two processes on different hosts to exchange data
- This is the entire purpose of networking
  - *Everything else is just commentary....*

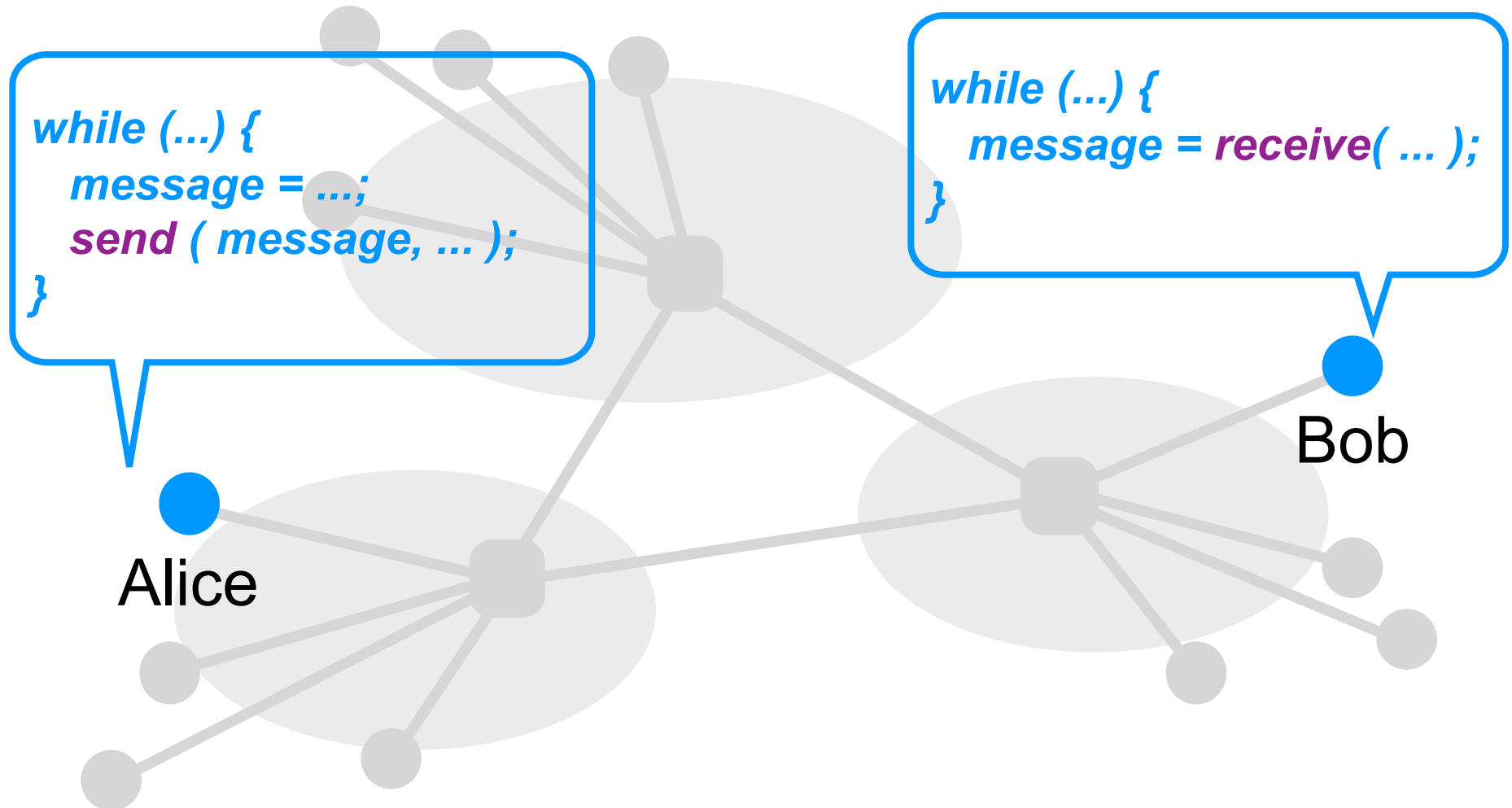
# Why Does This Matter?

- Clean separation of concerns:
  - Networking delivers data
  - Applications figure what to do with data
- Keeps network fully general (any app works!)
- When people complain about network security
  - Mostly the fault of application/OS, not network
- Will return to this later in course.....

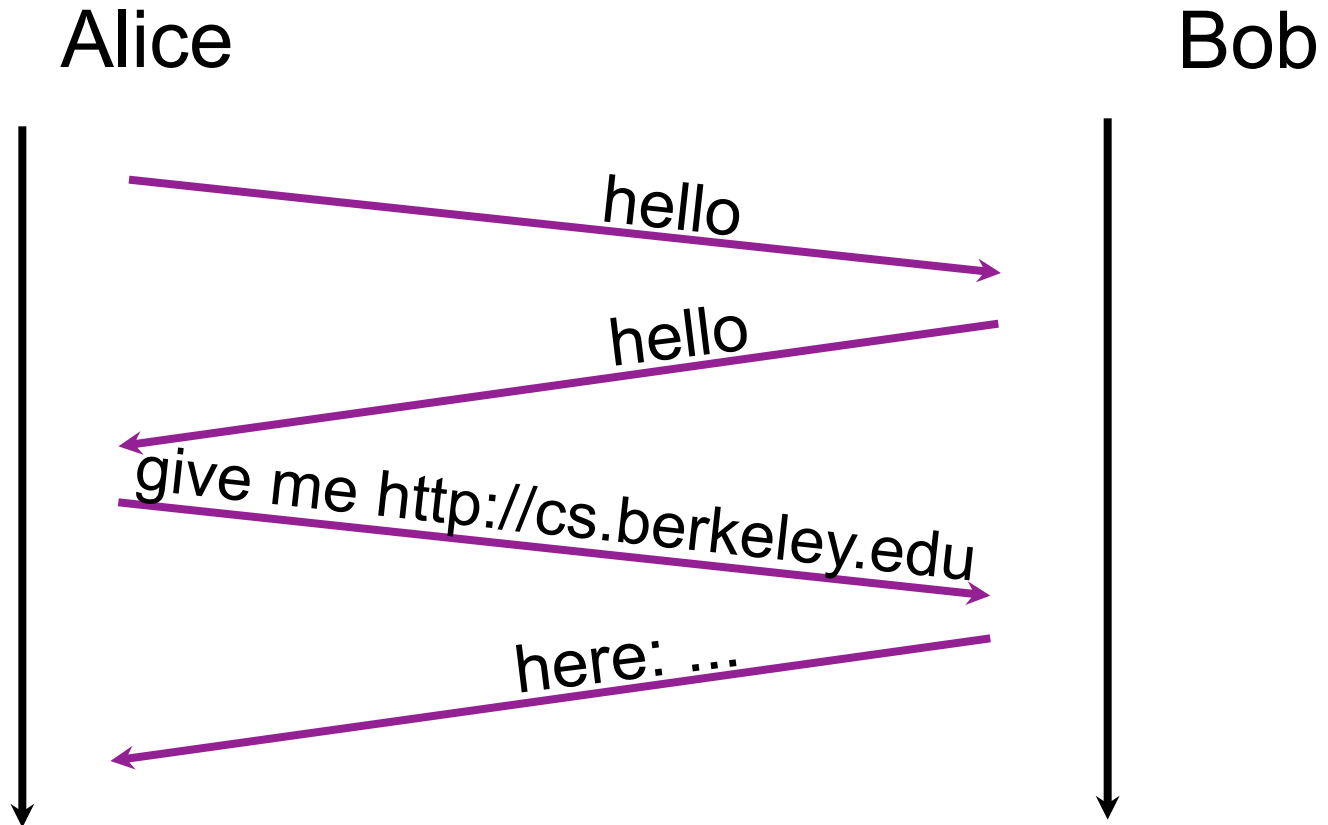
# Variety of applications



# Using the network....



# Structure of a Protocol



*Protocols are like conversational conventions....  
determining who should talk next, and how they should respond*



# **Why study the Internet?**

- The way we do business
  - E-commerce, advertising, cloud-computing

- The way we have relationships



## search e

and view

copyright

ease

diagnose

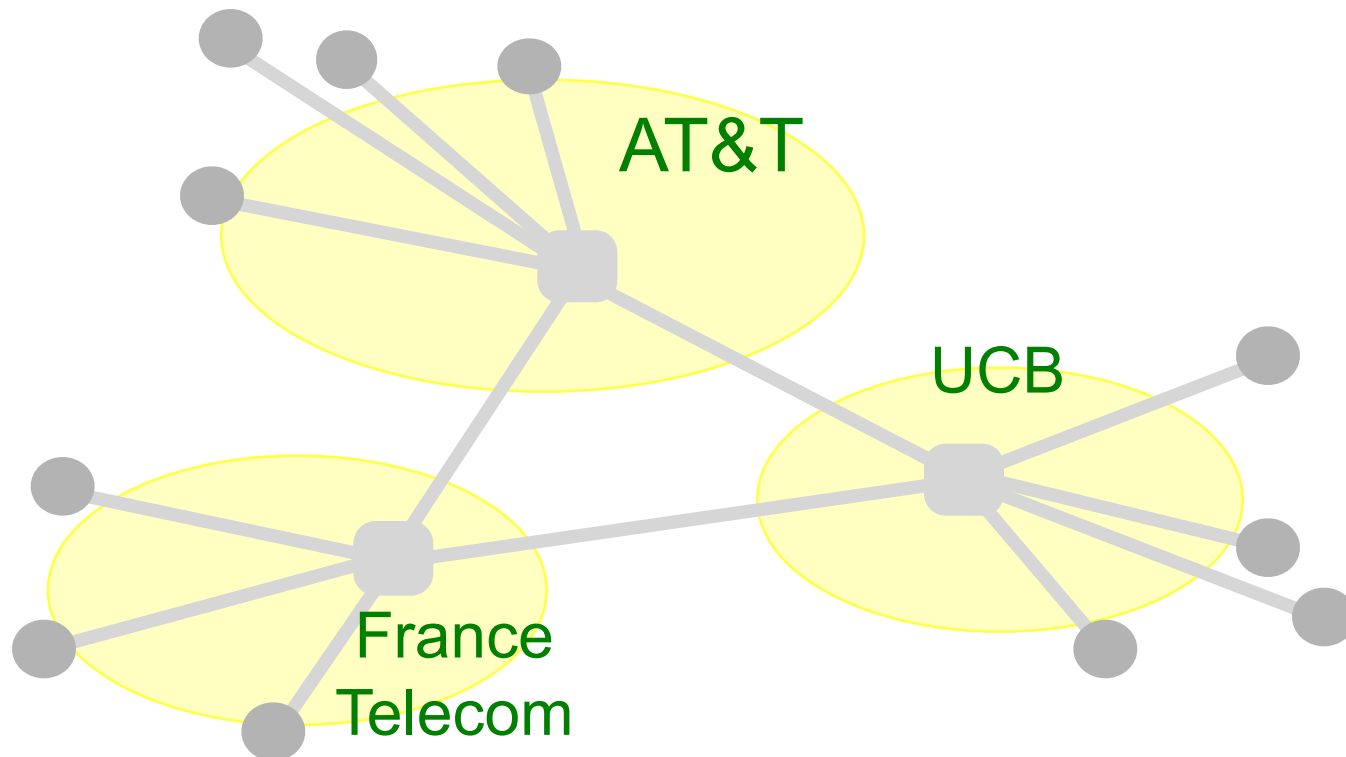


## #2 As a design challenge....

- The Internet has had to address many issues....
- Most of these were dealt with (by computer science) for the first time in the Internet....

# A federated system

The Internet interconnects different networks (>18,000 ISPs)



How do you interconnect competing and distrustful entities?

# Tremendous scale

- 2.92 Billion users (41% of world population)
- 1 Trillion unique URLs (in 2008)
- 294 Billion emails sent per day
- 1.75 Billion smartphones
- 1.24 Billion Facebook users
- 100 hours of video uploaded to YouTube every minute
- Switches that move 300Terabits/second ( $10^{14}$ )
- Links that carry 100Gigabits/second

*We use the phrase “Internet Scale” to refer to such systems*

# Enormous diversity and dynamic range

- **Communication latency**: microseconds to seconds ( $10^6$ )
- **Bandwidth**: 1Kbits/second to 100 Gigabits/second ( $10^7$ )
- **Packet loss**: 0 – 90%
- **Technology**: optical, wireless, satellite, copper,...
- **Endpoint devices**: sensors, cell phones, datacenters,...
- **Applications**: skype, live video, gaming, remote medicine,...
- **Users**: the governing, governed, operators, selfish, malicious, naïve, savvy, embarrassed, paranoid,...

# Constant Evolution

## 1970s:

- 56kilobits/second “backbone” links
- <100 computers, a handful of sites in the US
- Telnet and file transfer are the “killer” applications

## Today

- 100+Gigabits/second backbone links
- 5B+ devices, all over the globe
- 20M Facebook apps installed per day

*Designing the Internet is not designing for a fixed target!*

# Asynchronous Operation

- Fundamental constraint: **speed of light**
- Consider:
  - How many cycles does your 3GHz CPU in Berkeley execute before it can possibly get a response from a message it sends to a server in NY?
    - Berkeley to New York: 4,125 km
    - Traveling at 300,000 km/s: 13.75 milliseconds
    - Then back to Berkeley:  $2 \times 13.75 = 27.5$  milliseconds
    - $3,000,000,000 \text{ cycles/sec} \times 0.0275 = 84,000,000 \text{ cycles!}$
- Thus, communication feedback is always *dated*



# Prone to Failure

- To send a message, **all** components along a path must function correctly
  - software, modem, wireless access point, firewall, links, network interface cards, switches,...
  - Including **human operators**
- Consider: 50 components, that work correctly 99% of time → 39.5% chance communication will fail
- Driven by numerical and geographic scale
  - Scale → lots of components
  - Asynchrony → takes a long time to hear (bad) news

# Design Challenges

- Federation
- Scale
- Diversity and dynamic range
- Constant evolution
- Asynchronous
  - Made worse by geographic scale
- Failures
  - Made worse by numeric scale

# #3 Introduced a new design paradigm

- Completely different from the phone network
- Inventors had to overcome strong technical and commercial resistance to realize their dreams
  - Motivation not for personal gain, but societal benefit!
- A true success story of “thinking differently”
  - Their strong vision kept the design on track
  - Brilliant in conception, sometimes weak in execution
- Has lasted for over 40 years, almost unchanged
  - While mired in details, ***leave room for awe***

# The Internet Design Paradigm...

- ...has changed computer science
- It taught us that how you deal with failure is the key to scaling systems

# #4 Networking undergoing revolution

- Industry has been closed, stagnant, and feudal
  - Proprietary hardware and software
  - Slow moving standards
  - Dominated by powerful vendor
- **But we are on the verge of a revolution!**
  - Commodity hardware making inroads
  - Open source software starting to appear
  - Developing intellectual (and practical) framework of applying systems principles of abstraction and modularity
  - The rise of x86 forwarding

# Summary: The Internet....

- As an artifact, has transformed our lives
- As a design, addressed many new challenges
- As a paradigm, changed computer science
- And is on the cusp of a revolution....

**Why are networking classes so bad?**

# Reason #1: Static Architecture

- The basic Internet architecture has not changed since its invention almost 40 years ago
  - Even IPv6 is very similar to IP
- Can't test new architectures in lab or testbed
  - So we only understand what we currently have
- **We are teaching history, not principles**
  - You will learn “first tries” not “fundamental answers”
  - As if we taught MS-DOS in an operating system course



# Reason 2: No Intellectual Framework

- Internet inventors defined a brilliant paradigm
  - Since then, community has focused on protocols to realize this paradigm
- Research community has failed to provide a general framework for understanding protocols
- We therefore just teach a big bag of protocols
  - And let you try to make sense of it yourself

# Quote from John Day

*“There is a tendency in our field to believe that everything we currently use is a paragon of engineering, rather than a snapshot of our understanding at the time. We build great myths of spin about how what we have done is the only way to do it to the point that our universities now teach the flaws to students (and professors and textbook authors) who don’t know better.”*

# I will try to overcome these problems

- Focus on “fundamental questions”
  - And will present alternative designs in a few lectures
- You will have to learn the current design
  - But I will point out where it falls short
- You will end up with a mixture of the “big picture” and “current design details”

# Fundamental questions

- How can you deliver packets from source to destination? (*routing*)
- How do you build reliable transport on top of an unreliable network?
- How can you federate a set of competing ISPs?
- ....

# First half of course: Basics (14 L)

- General overview (~2 more lectures)
  - Packet switching, layering, history of Internet
- Idealized view of network (~4 lectures)
  - Focus on fundamental conceptual questions
  - Ignore all real-world unpleasantness
- Making this vision real (~7 lectures)
  - IP, TCP, DNS, Web
  - Emphasize concepts, but deal with realities

# Second half of course: Topics (12 L)

- Congestion control (~3 lectures)
- Advanced topics in routing (~3 lectures)
- Datacenter networks (~2 lectures)
- Middleboxes (~1 lecture)
- Network security and other rants (~1 lecture)
- Software-defined networking (~1 lecture)
- Alternate architectures (~1 lecture)
- Life lessons (~1 lecture)

# **Class Logistics**

# Class is Designed for Attendees

- I cannot force you to attend
- And I will make my slides available online
  - Typically a few minutes before class
- But I will not webcast the class
  - Profanity, blasphemy, are one reason
  - But class interaction is more important reason
- Your participation is vital to the class' success
  - More about this later...



# Teaching Assistants

## Professorial Understudy:

- Kay Ousterhout (G)

## Section TAs

- Luise Valentin (U): Head TA
- Ethan Jackson (G)
- Brian Kim (U)
- Ross Teixeira (U)
- Qiyin Wu (U)

## Project TAs:

- Alice Sheng (U)
- Ankur Dave (G)
- Aisha Mushtaq (G)
- Wenting Zheng (G)

## Reader:

- Steven Chen (U)

*See the course website for TA office hours and sections*

# Class Workload

- Four projects (due in 1 to 3 weeks)
- Six homeworks (due in one week)
  - Submission through web forms
- Exams:
  - Midterm: Oct 13<sup>th</sup> ***in class***
  - Final: Tuesday, December 13<sup>th</sup>, 8:00am-11:00am
  - Closed book, open crib sheet
    - Midterm: one crib sheet
    - Final: two crib sheets

# Grading

- Course graded to mean of ~B-

6 Homeworks	6% (6 x 1% each)
4 Projects	48% (4 x 12% each)
Midterm exam	25%
Final exam	30%

- Will drop the lowest 9% of your grade....

# Participation Requirement

- Must speak up in class, or see me in office hours
  - At least once, or else you flunk. Period.
- Participation must be technical in nature
  - It doesn't count to ask me what the final will cover
- See class web page to record your participation
  - Name, date, question asked/answered...

# When Can You Cooperate?

- Homeworks:
  - Can work jointly, each is worth only 1%
  - Cheating on these only hurts yourself
- Projects:
  - Project 1: Must do solo
  - Projects 2, 3, 4: Can work in pairs

# Four projects (mostly in Python)

- Project 1: **Chat** (1 week)
  - Will do first part in section
  - Goal: do something useful
- Project 2: **Routing** (2.5 weeks)
  - In a simple simulator
  - Goal: show that details matter
- Project 3: **Network Measurement** (1.5 weeks)
  - Goal: show how the real Internet behaves
- Project 4: **Build a WAN Optimizer** (3 weeks)
  - Goal: to make your life miserable

**Project TA handles all project-related questions!**

# Policy on late submissions, re-grade requests, cheating

- Detailed description is on the class website
- Summary version:
  - You may submit assignments late or request re-grades but to a point, and it will cost you
  - Project grading will be announced for each project
  - ***Your responsibility to keep your code private!***
  - When in doubt about the policy, ask us!
- **Zero-tolerance policy on cheating....**

# Textbook

- J. Kurose and K. Ross, *Computer Networking: A Top-Down Approach*, 7th Edition, 2016.
  - 5<sup>th</sup> , 6<sup>th</sup> Editions ok, but translate the reading assignments
- **You will not be tested on material we didn't cover in lecture or section**
  - Use as a reference and a source of examples



# Class Communications

- <http://inst.eecs.berkeley.edu/~cs168/fa16/>
  - CS168.io
  - Assignments, lecture slides, announcements
- Use instructional account for assignments
  - You should have those by now
- Use Piazza for all intra-class communication
  - You should all be signed up by now

# Policy on rescheduling exams

- Will accommodate dire circumstances
  - Need documentation of circumstances
- Off-schedule exam will be different from the normal exam:
  - Somewhat harder
  - With an oral exam portion
- And there will be only one alternate time

**Who am I and how do I teach?**

# Instructor: Scott Shenker

- Office Hours:
  - Thursday right after class (3:30-4:30)
    - Walking to my office, and in my office
  - Except for today, when they are at 5:30
- Two sons (in college and grad school)
- Favorites:
  - Book: Straight Man (Richard Russo)
  - Movie: Galaxy Quest
  - Food: Chocolate
  - Time to answer Piazza posts: 3am

# My Academic Background

- Trained as a theoretical physicist
  - Have never taken a CS course
  - Can't program my way out of a paper bag....
- Career:
  - ScB, PhD, Postdoc in Theoretical Physics
  - 14 years at Xerox PARC
  - Founded Center for Internet Research at ICSI in 1998
  - Joined Berkeley faculty ~2004

# Research Interests

- Have worked on a variety of topics:
  - Physics, economics, performance analysis, operating systems, security, distributed systems, data analytics,...
  - Diversity reflects my learning and teaching style
  - Competitive advantage: ignorance!
- For last ~30 years, main focus has been networking and Internet architecture

# My teaching style not for everyone...

- Next few slides provide a small taste of my flaws
- With comments from past class evaluations...

# I won't remember your name

- Prosopagnosia (as described by Oliver Sacks)
- In my case, it isn't recognizing faces, but attaching names to faces
- Don't take it personally....
  - Can't attach names to faces for 50% of the faculty



# But I do care....

- “He cares a ton...”
- I won’t recognize your face, I won’t know your name, but I care deeply about your success...
- If you put in half the work I put into this course, you’ll do just fine.....

# I don't think visually

- “For the love of god, use more pictures and diagrams.”
- “Uses blackboard terribly. Very poor diagrams when using it. and not legible also.”
- I'm not going to turn into a blackboard virtuoso or animation wizard
- Ask TAs for pictures
- Will try to use other visual means
  - Watch for our re-enactment of routing.....

# When you look bored, I speed up

- “Pace gets faster if no one asks questions.”
- If you are bored, feel free to sleep (*at your peril*)
- If you are lost, ask me a question!
  - Or just yell “HELP!”

# I hate details

- “Moves very quickly during difficult topics and slowly during basic topics.”
- Will try to go over examples in more depth
- Sections will go over examples in even more depth

# Can't always engage class

- “He asks questions but no one answers”
- Will try various approaches to get you to talk
- **But, I don't ask questions to get answers.....**

# I ask questions so you can think!

- The pause after I ask a question is the only time you get to think
  - When I ask a question, I don't care if you answer it
  - But please, ***think about the question!***
- The best way to understand networking is to *first* try to solve the design issues yourself
  - Then the current solution will make a lot more sense
- **Internet not principled design, mostly ad hoc**
  - Can't "follow the logic", have to try designing it yourself

**What do I expect from you?**

# I Expect Four Things

- Ask questions when you don't understand
  - Or when you want to understand better
- Answer (or think about) questions when I ask them
  - Even if you aren't sure of the answer
- **Turn off your @\$&#\$@ electronics!**
- **Be quiet during class**



# Why do I care about this?

- My goal is to teach you this material
  - To make you better computer scientists
  - To make you better people
- And I work my ass off trying to do so...
  - But this only works if you are listening
  - If you are reading email, checking facebook, playing games, talking, etc.....
  - ...then you are wasting my effort and your time
- If you aren't willing to pay attention for 80 minutes
  - Then please stay home.....

**Is CS168 the right class for you?**

# Four Questions to Ask Yourself

- Want to understand the “why” of networking?
  - Not just looking for definitions and techniques
- Willing to engage with the material?
  - Must be willing to think on your own
- Willing to actively participate in class?
  - Class is useless if you don't pay attention and engage
- Ready for some fun?
  - Must laugh at my bad jokes...

# For Next Time....

- Make sure you are on piazza
- Discussion sections will start on August 31
- My office hours: right after class on Thursdays
  - But today, 5:30-6:30 (need to leave right after class)