

Lecture 1:

Course Introduction

CSE 123: Computer Networks
Stefan Savage





Personnel

- Instructor: **Stefan Savage**
 - ◆ Office hours: **Thursday 4pm-5pm**
 - ◆ EBU3B 3106
- TAs:
 - ◆ **Shreeja Kumar**
 - ◆ **Vikas Lokesh**
 - ◆ Office hours: TBA

About me...



- I work at the intersection of computer security, networking and operating

- **Research**

- ◆ I'm director of the Center for Networked Systems (CNS) at UCSD and UCB.
- ◆ Lots of work on wireless networks

- **Policy**

- ◆ National Research Council
- ◆ Institute for Information Systems Security
- ◆ National Science Foundation

- **Industry**

- ◆ Asta Networks
- ◆ Netsift (UCSD and Cisco)



Center for Networked Systems (CNS) on
at UCSD and UCB.
Network management,



Policy Research group
Advisory group
Advisory Committee



Industry
(Netsift is a security
company) ->





Lecture 1 Overview

- Class overview
 - ◆ Expected outcomes
 - ◆ Structure of the course
 - ◆ Policies and procedures
- A brief overview of Computer Networking
 - ◆ High-level concepts
 - ◆ An end-to-end example



Prereqs

- CSE120
 - ◆ I've been instructed to enforce prerequisites
 - ◆ If there is room after the add deadline (i.e., no one is on the waitlist with prerequisites filled) then I will consider exceptions
 - » Sorry, but I can't predict what enrollment is going to look like (we are currently overfull **and** have a waitlist of 34)
- Programming experience
 - ◆ We will be assigning programming projects in C/C++
 - ◆ This course will not teach you C. The TAs will help, but you need to learn it on your own if you don't already know it.



Expected Outcomes

- This course *will* teach you the *fundamentals* of computer networks:
 - ◆ Layering, signaling, framing, MAC, switching, routing, naming, Internetworking, congestion control, router design, etc.
 - ◆ At the end of this course you should completely understand what's actually happening when you view a Web page
- This course *will not* teach you signals and coding
 - ◆ Take an EE course to learn about modulation, encoding, etc. on different hardware technologies
- Similarly, we *will not* cover Internet apps/services
 - ◆ CSE124 covers application layer protocols, Web, etc. (great class!)
 - ◆ You *will be able to* pick this up on your own with Google



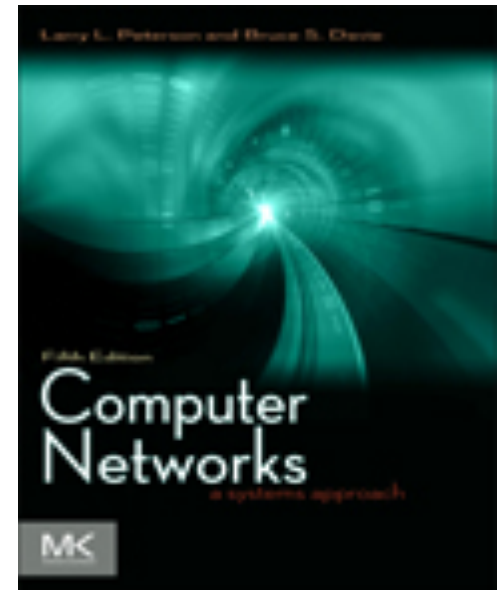
CSE 123 Class Overview

- Course material taught through class lectures, textbook readings, and discussion sections
- Course assignments are
 - ◆ Homework questions (based on lecture)
 - ◆ Two substantial programming projects
- Discussion sections are a forum for asking questions
 - ◆ Help you get started on the projects
 - ◆ Lecture material and homework
 - ◆ **Additional networking topics**
 - ◆ No discussion section this Monday
- Discussion board ([Piazza.com](https://piazza.com))
 - ◆ The place to ask questions about lecture, hw, projects, etc.

Textbook



- Peterson and Davie, *Computer Networks: A systems Approach*, Morgan Kaufmann, 5th Edition, ISBN 978-0-12-385059-1



- Available online for **free** from UCSD networks:
<https://roger.ucsd.edu/record=b8932307~S9>



Homeworks

- There will be 3-4 homeworks throughout the quarter
 - ◆ Reinforce lecture material...no better practice
- Cheating
 - ◆ You *should* discuss homework problems with others
 - » You can learn a lot from each other
 - » But there is a distinction between collaboration and cheating
 - ◆ Cheating is copying from other student's homeworks or solution sets, searching for answers on the Web, *anything* which misrepresents your knowledge of the material or undermines my ability to fairly assess your abilities
 - ◆ If you're not sure if something is allowed, then ask
 - ◆ All suspected violations will be reported to the Academic Integrity Office

Projects



- There will be two programming projects
- The projects must be completed in C/C++
 - ◆ We will provide skeleton code for you to use
 - ◆ Your job is to fill in the interesting/hard parts
 - ◆ The TAs will be available to help with coding
- The projects are **INDIVIDUAL** assignments
 - ◆ All code must be your own
 - ◆ OK to discuss design ideas, NOT OK to share/look at code

Espresso Prize



Computer Labs



- You are welcome to use any Linux machine in the labs in the basement of the CSE/EBU3B building
 - ◆ Linux running on Intel machines
- You can also use your home machine
 - ◆ The project source will work on Windows/OS X (with caveats)
 - ◆ Graders will test on ieng6 machines
 - ◆ **Be sure to test your projects there as well**

Exams



- Midterm
 - ◆ 10/28
 - ◆ Covers first half of class
- Final
 - ◆ Dec 9th (8am-11am I think)
 - ◆ Covers second half of class + selected material from first part
 - » I will be explicit about the material covered
- No makeup exams
 - ◆ Unless dire circumstances
- Closed book with crib sheet
 - ◆ You can bring one double-sided 8.5x11" page of notes to each exam to assist you in answering the questions
 - ◆ Not a substitute for thinking



Grading

- Homeworks: 20%
 - ◆ Think of these collectively as a take-home midterm
- Midterm: 15%
- Final: 25%
- Projects: 40%
 - ◆ Divided evenly among the projects

How *Not* To Pass CSE 123



- Do not come to lecture / discussion
 - ◆ It's nice out, class is early, the slides are online, and the material is in the book anyway
 - ◆ Lecture material is the basis for exams and directly relates to the projects
 - ◆ Besides, the professor thinks he's funny and needs his ego stroked
- Do not do the homework
 - ◆ It's only 20% of the grade
 - ◆ Excellent practice for the exams, and some homework problems are exercises for helping with the project
 - ◆ 20% is actually a significant fraction of your grade (easily the difference between an A and a C)



How *Not* To Pass (2)

- Do not ask questions in lecture, office hours, or email
 - ◆ Professor is scary, I don't want to embarrass myself
 - ◆ Asking questions is the best way to clarify lecture material at the time it is being presented
 - ◆ Office hours and email will help with homeworks, projects
- Wait until the last couple of days to start a project
 - ◆ We'll have to do the crunch anyways, why do it early?
 - ◆ The projects cannot be done in the last couple of days
 - ◆ Repeat: The projects cannot be done in the last couple of days

Class Web Page



<http://www.cs.ucsd.edu/classes/fa16/cse123-a/>

- Serves many roles...
 - ◆ Course syllabus and schedule (updated as quarter progresses)
 - » Lecture slides
 - ◆ Announcements
 - ◆ Homework handouts
 - ◆ Project information

Not up currently online due to IT problem, but will be up shortly

Questions



- Before we start the material, any questions about the class structure, contents, etc.?



This Class in One Slide

- **Protocols & Layering**
 - ◆ Manage complexity by decomposing the tasks
 - ◆ Standardizing syntax and semantics to support interoperability
- **Naming**
 - ◆ Agreeing on how to describe a host, application, network, etc.
- **Switching & Routing**
 - ◆ Deciding how to get from here to there
 - ◆ Forwarding messages across multiple physical components
- **Resource Allocation**
 - ◆ Figuring out how to share finite bandwidth, memory, etc.



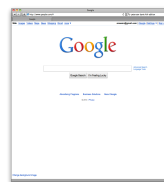
A “Simple” Task

- Send information from one computer to another



Host

- Endpoints are called **hosts**
 - ◆ Could be computer, iPod, cell phone, etc.
- The plumbing is called a **link**
 - ◆ We don't care what the physical technology is: Ethernet, wireless, cellular, etc.



Link



Host

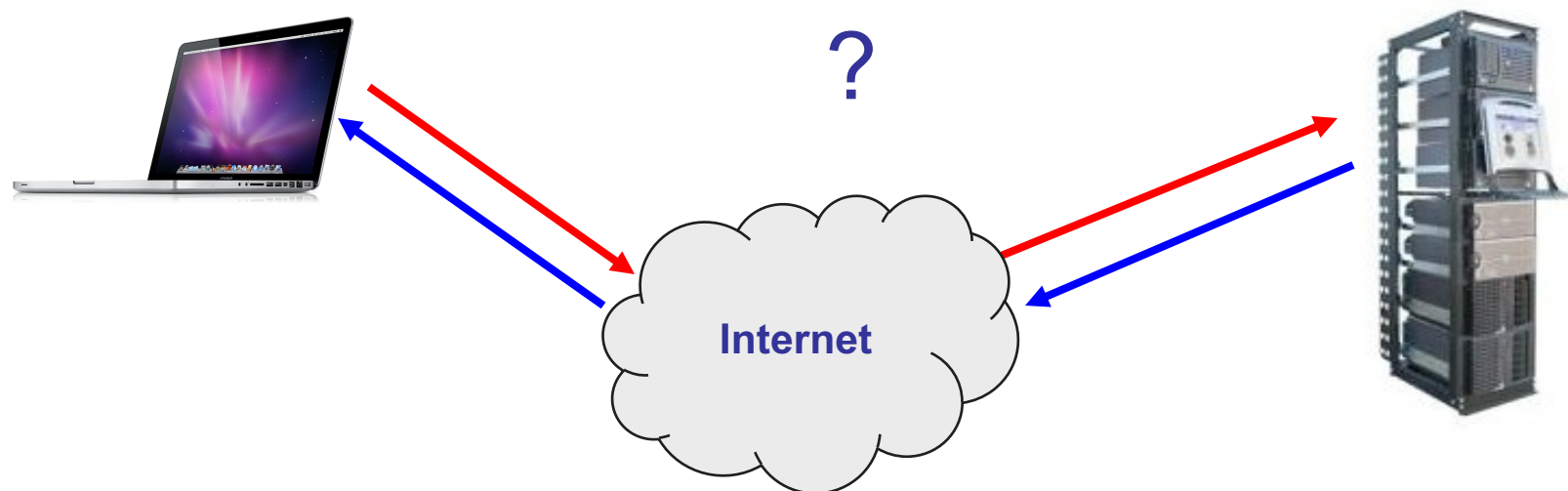
Actually Quite Complicated



- **ROUGHLY**, what happens when I click on a Web page from UCSD?

My computer

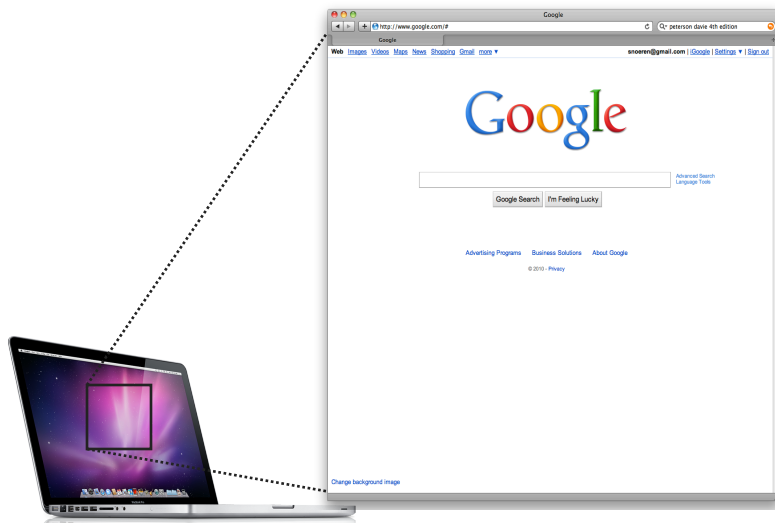
www.google.com





Web request (HTTP)

- Turn click into HTTP request



GET http://www.google.com/ HTTP/1.1
Host: www.google.com
Connection:keep-alive
...



Name resolution (DNS)

- Where is `www.google.com`?

My computer
(132.239.9.64)



What's the address for `www.google.com`



Local DNS server
(132.239.51.18)



Oh, you can find it at 66.102.7.104





Data transport (TCP)

- Break message into packets (TCP segments)
- Should be delivered reliably & in-order

```
GET http://www.google.com HTTP/1.1
Host: www.google.com
Connection:keep-alive
...
```



“and let me know when they got there”



Global Network Addressing

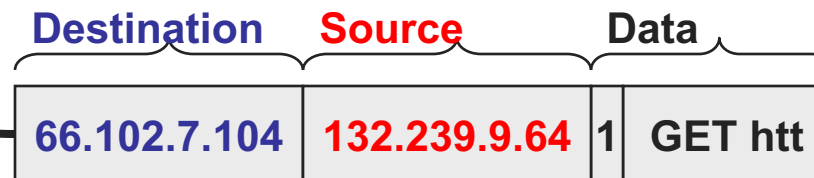


- Address each packet so it can traverse network and arrive at host

My computer
(132.239.9.64)



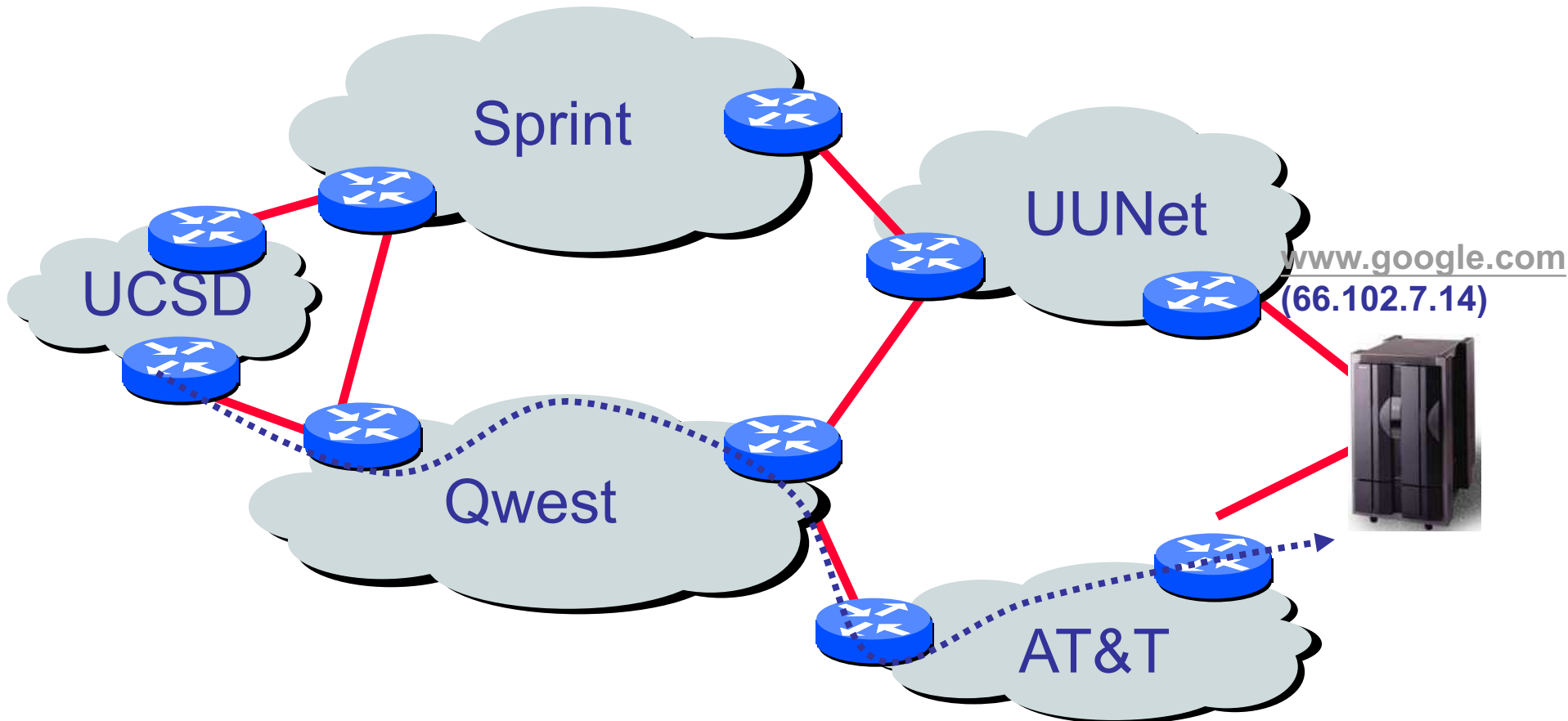
www.google.com
(66.102.7.104)





Network Routing

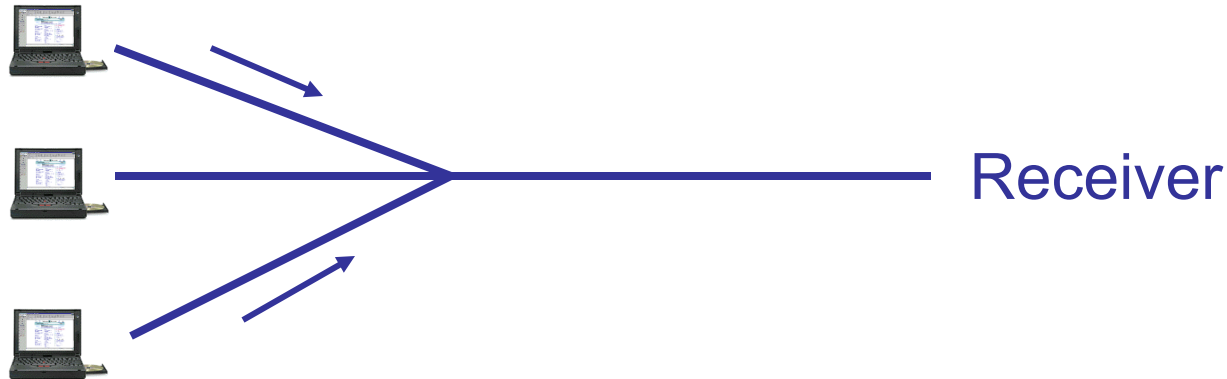
- Each router forwards packet towards destination



Link management (Ethernet)

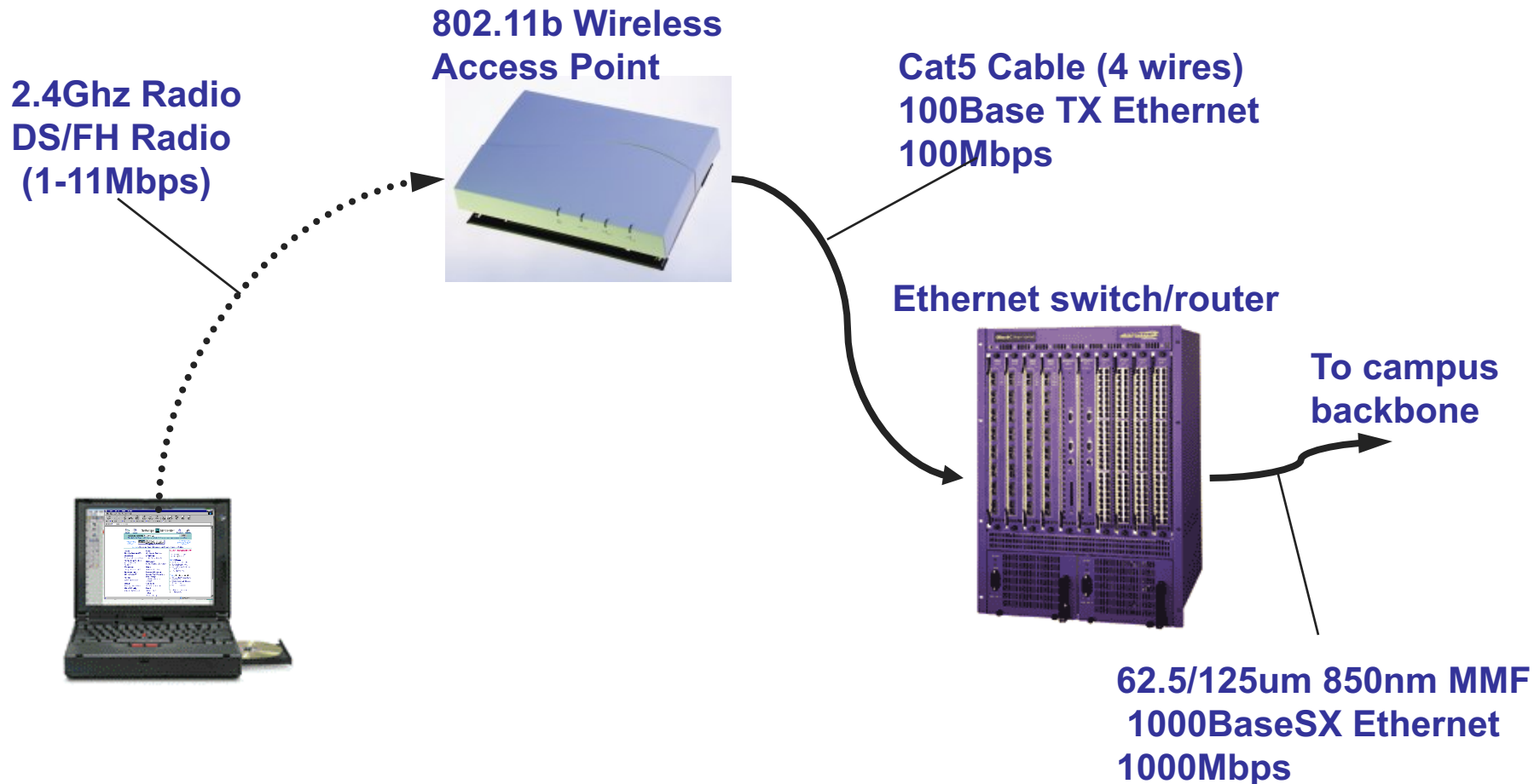


- Break message into frames
- Media Access Control (MAC)
 - ♦ Can I send now? Can I send now?



- Send frame

Physical layer





For Next Class...

- Browse the course web
 - ◆ <http://www.cs.ucsd.edu/classes/fa16/cse123-a/>
- Read Chapter 1 and start Chapter 2 (up to 2.2)
- Next class: links and signaling
- Drop now or plan to stick it out!