Recap

• Please make an effort to come to every class.
• Please do the work yourself and get permissions for other sources. Also, acknowledge them.
• Please check if you have the background by doing PA1 all by yourself.
• This course will expect:
  – Good work ethics
  – Independence
  – Respect for others
• This course is about:
  – Introducing common problems that arise when building a distributed system
  – Discussing algorithms, architectures, and abstractions that solve those problems
  – Practicing how to adapt those algorithms and concepts

Today and Next

• A brief overview of the Internet
• Two things
  – The design philosophy of the Internet ("The Design Philosophy of the DARPA Internet Protocols" by David Clark); today
  – Transport & application layers: next lecture
• Obviously can’t replace a networking course; this should be just a recap for you.
• Why teach these?
  – If there’s no network, there’s no distributed system.
  – Not just that: the design of the Internet is a great example of designing a solid distributed system.

What Is the Internet?

• 1969

• 1977

• Now

• A network of networks
• The fundamental goal of the original designers: interconnecting different networks by designing common protocols
**Detour: What is a Protocol?**
- Example: making an appointment

- Well... I think we need a better way...

**Returning back: What is the Internet?**
- A network of networks
- The fundamental goal of the original designers: *interconnecting* different networks by designing common protocols

**CSE 486/586 Administrivia**
- PA 1 is out. Please try it yourself.
- Please use Piazza; all announcements will go there.
  - Signup link: http://piazza.com/buffalo/spring2020/cse486586
  - Anonymous/private posting: generally questions are beneficial to the whole class; please consider posting it publicly first. All announcements will be posted there.
- Use good coding styles.
  - Use the Android code style guideline (Google it).
- After-class questions
  - Will answer them outside. There’s a class right after this one.

**Building the Internet**
- Why care?
  - Now: you might be just doing what’s given to you.
  - Later: you will likely define what you want to do and do it.
- Internet as a case study of a distributed system
  - Put a designer’s hat on for a moment.
- Questions to think about:
  - Why? i.e., why do we want to connect computers?
  - What is the ideal outcome? i.e., what do we want?
  - How do we do that?
**Why and What**

- **Why**
  - “The whole can be greater than the sum of its parts”

- **What**
  - Internet communication must continue despite loss of networks or gateways.
  - The Internet must support multiple types of communications service.
  - The Internet architecture must accommodate a variety of networks.
  - The Internet architecture must permit distributed management of its resources.
  - The Internet architecture must be cost effective.
  - The Internet architecture must permit host attachment with a low level of effort.
  - The resources used in the Internet architecture must be accountable.

**How?**

- Let’s first consider a single network, where every machine is wired together.
- What do we need to send data from machine A to machine B?
  - Hint: think about sending a letter from one place to another.
- **We need**
  - Addressing
  - Routing
- **However, if you think about the Internet, there are many types of networks based on various physical media.**
  - Coax, radio, satellite, etc.
  - The original designers wanted to send data across all these networks.

**How to Interconnect?**

- **Question:** how to send data from one machine to another that can be in different types of networks?
  - How to transfer data from one technology to another
  - How to ultimately deliver data from one machine to another

  - **How would you do it?** (E.g., you have radio signals coming in on one side, and the other side is a wire.)
  - A potential solution: signal translators

  - **Problem?**
    - Every new technology needs translators for all existing technologies.

**Layering: A Modular Approach**

- **Sub-divide the problem**
  - Each layer relies on services from layer below
  - Each layer exports services to layer above
  - Each layer is designed to solve a specific, narrow set of problems

- **Interface between layers defines interaction**
  - Hides implementation details
  - Layers can change without disturbing other layers

- **“The” computer science approach**
  - ISA, OS, networking...

**Summary**

- **The Internet**
  - A network of networks
  - A case study as a distributed system

- **Protocol**
  - An agreement between multiple parties
  - Syntax & semantics

- **Design a system**
  - Why, what, and how

- **The Internet**
  - Connecting by layering
Acknowledgements

- These slides contain material developed and copyrighted by
  - Indranil Gupta at UIUC
  - Mike Freedman and Jen Rexford at Princeton