CS 43: Computer Networks

01: Course Administration & Introduction

September 8, 2020

Slides adapted from Kurose & Ross, Kevin Webb

Today

- What is this course about?
- Course Administration
  - Structure & Grading
  - Academic Honesty
  - How does this class work?
- Introduction
  - What does it take to transmit a packet over the Internet?

What This Class is about

How networks (focus on the Internet) work

Mobile phone

www.google.com

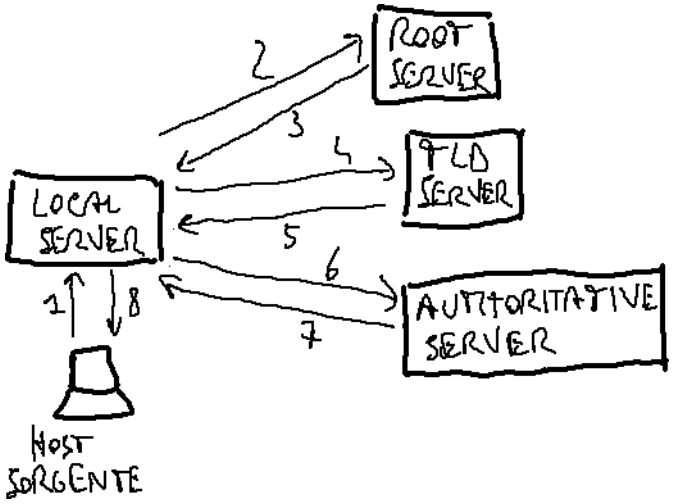
Google Server

What This Class is about

How applications that use networks work:

HTTP 404 Error Message

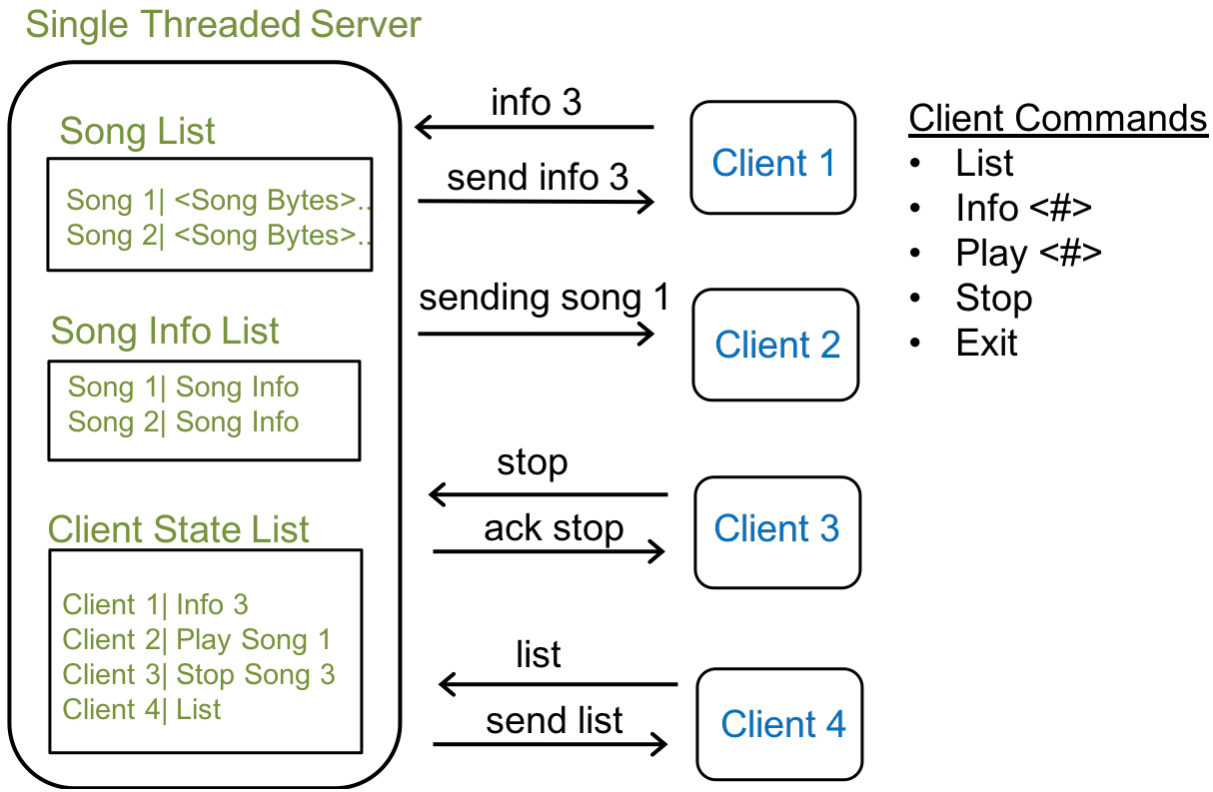
Email



DNS

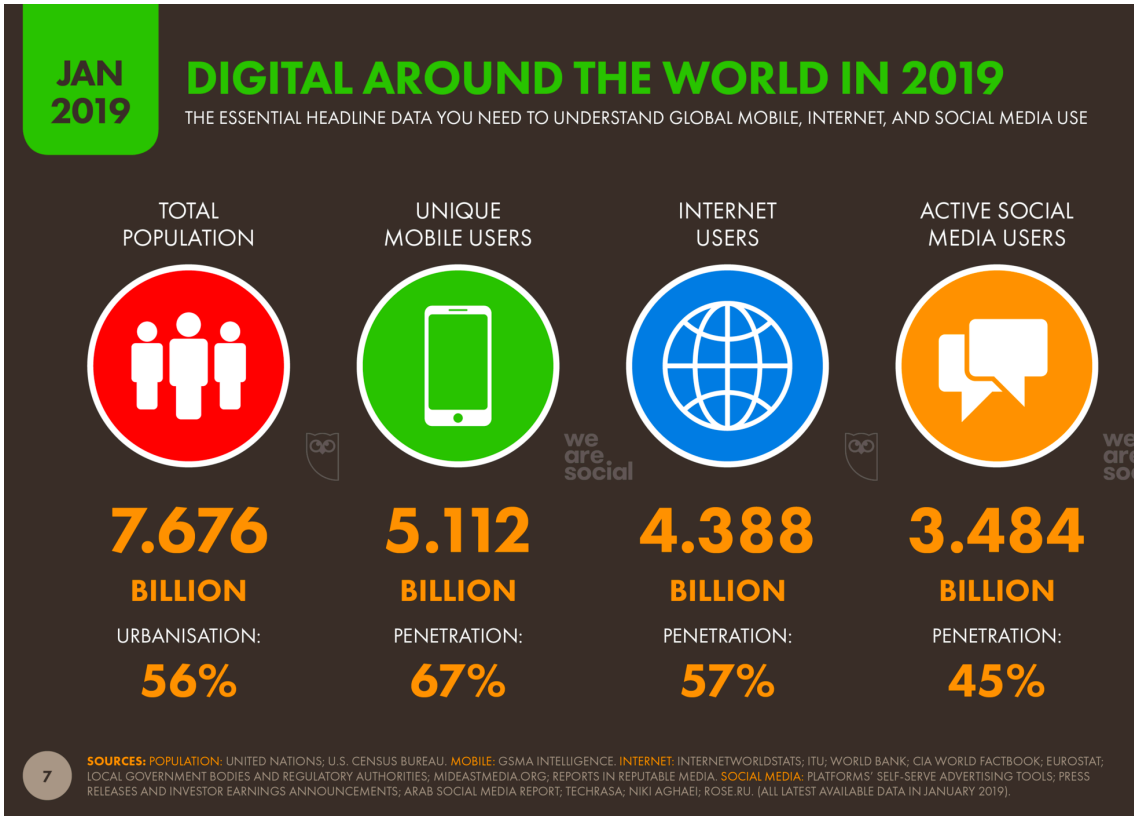
What This Class is about

How to write programs that communicate over networks



What This Class is about

How different protocols, policies, and mechanisms interact.



https://datareportal.com/reports/digital-2019-global-digital-overview

What This Class is about

How different protocols, policies, and mechanisms interact.

Type of usage:	Change vs. pre-COVID-19 typical day
Gaming	115%
VPN	49%
Video	36%
Downloads	39%
Web	27%
Social	-12%
Voice minutes of use	25%

https://thenewstack.io/the-network-impact-of-the-global-covid-19-pandemic/

Courtesy: Verizon

We're here at the beginning..

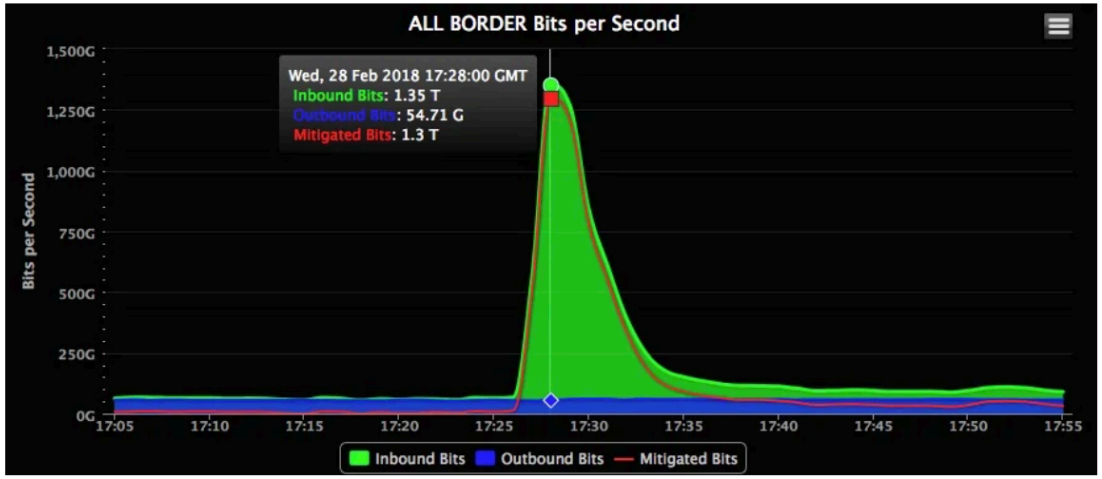
Type of usage:	Change vs. pre-COVID-19 typical day
Gaming	115%
VPN	49%
Video	36%
Downloads	39%
Web	27%
Social	-12%
Voice minutes of use	25%

https://thenewstack.io/the-network-impact-of-the-global-covid-19-pandemic/

Courtesy: Verizon

What This Class is about

How different protocols, policies, and mechanisms interact.



Real-time traffic from the DDoS attack. AKAMAI

Github, 2019: One of the largest DDoS Attacks

What This Class is about

How different protocols, policies, and mechanisms interact.

Enlarge / Netflix took an active role in fighting for net neutrality rules in 2014.

Image Courtesy: Ars Technica

Net Neutrality: ISPs vs Content Providers

https://arstechnica.com/tech-policy/2017/07/facebook-alphabet-amazon-and-netflix-called-to-testify-on-net-neutrality/

What This Class is about

How different protocols, policies, and mechanisms interact.

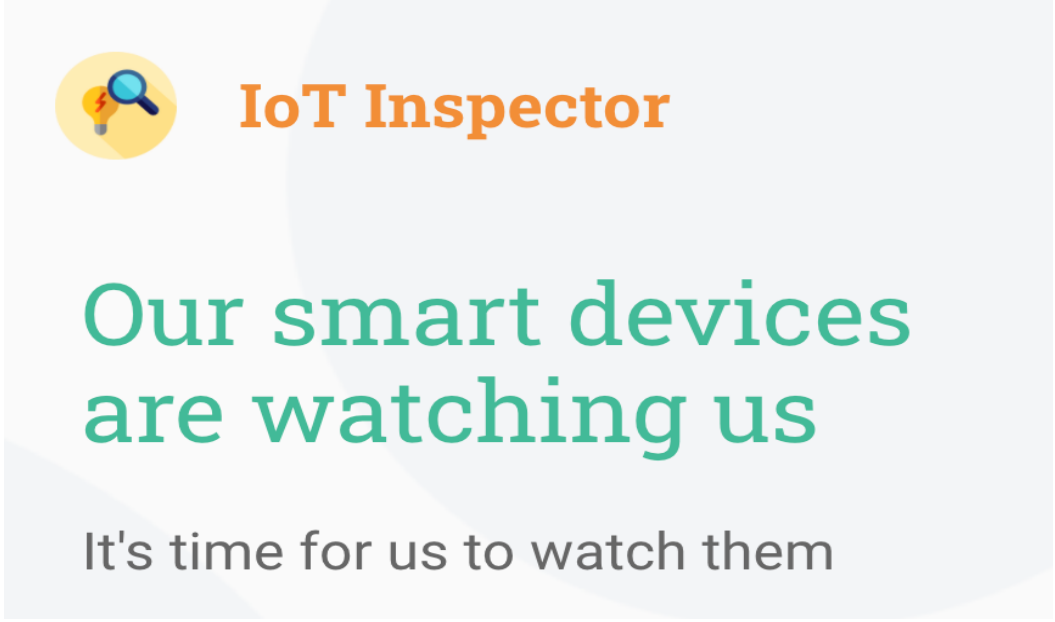


Image Courtesy: https://iotinspector.org/

Network Privacy

https://www.nytimes.com/2020/01/07/opinion/location-tracking-privacy.html

Instructor: Vasanta Chaganti

Please call me Vasanta or Prof. Chaganti

Research interests: Network performance and privacy

- measure the performance of network protocols
- what does your network data reveal about you?

Office Hours
On Slack

- Wednesdays: 3.00 – 4.30 PM
- Fridays: 10 – 11.30 AM
- By Appointment

The Internet is Exciting!

- Rapid growth and success.
  - 1977: 111 machines on Internet
  - 1981: 213
  - 1983: 562
  - 1986: 5000
  - 1989: 10,000
  - 1992: 1,000,000
  - 2001: 150 – 175 million
  - 2002: > 200 million
  - 2018: ~ 3 billion (>1B are phones/tablets)

Early Communication

The Internet is Exciting!

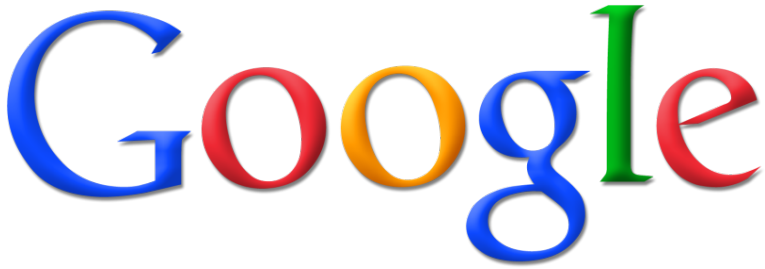
- Rapid growth and success.
- We're here at the beginning.
- Communication is empowering.

Star Trek (Late 60s)

Video calls

We're here at the beginning..

- Most of the growth happened in our lifetime.
- Still TONS of untapped potential.



Founded 1998

Founded 2004

We're here at the beginning..

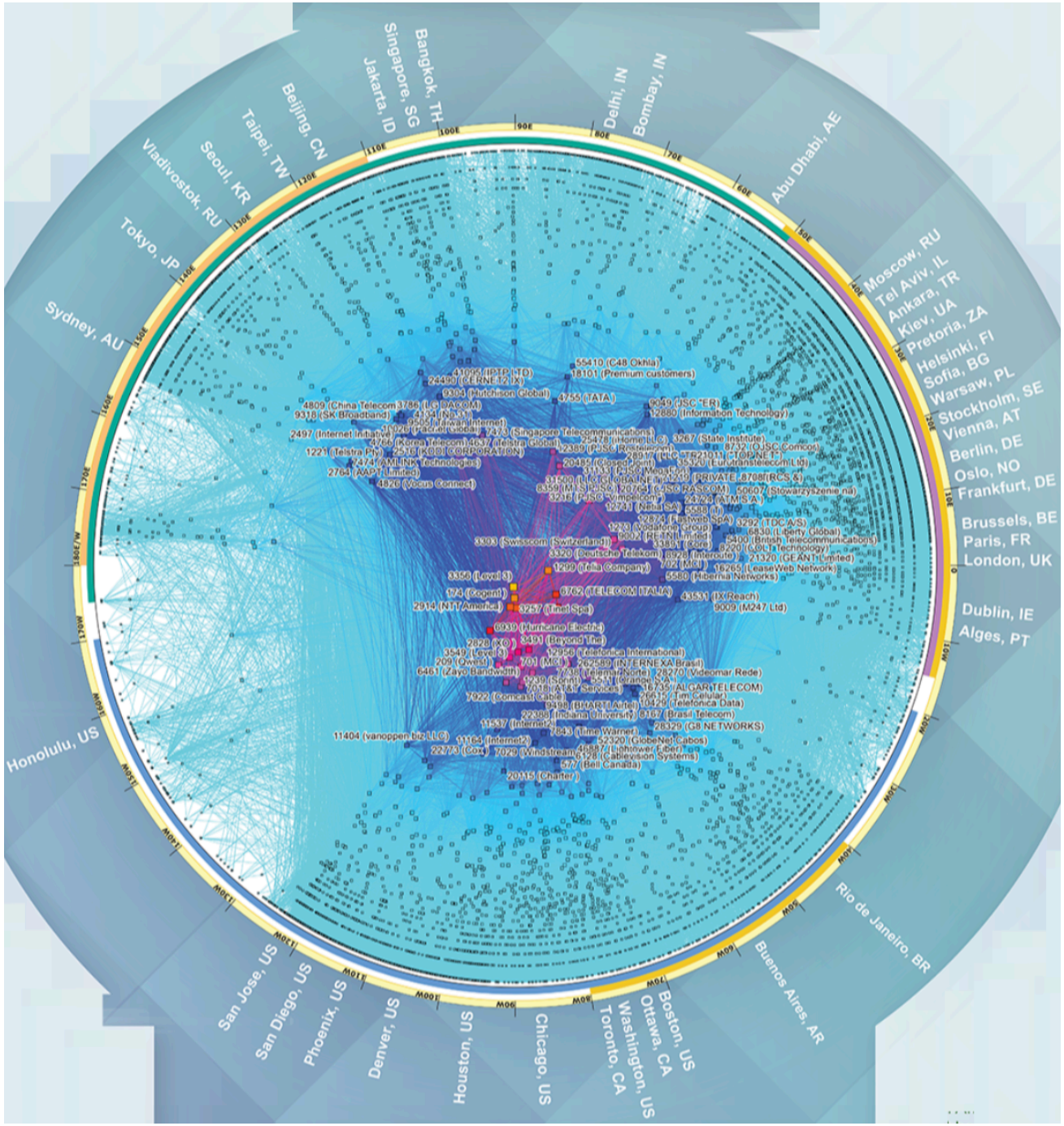
- Most of the growth happened <u>in our lifetime.</u>
- Still TONS of untapped potential.

Tweet-a-watt:
monitor energy use

Internet
refrigerator

sensorized,
bed
mattress

Web-enabled toaster +
weather forecaster



Internet traffic volume across the globe released by Cooperative Association for Internet Data Analysis (CAIDA)

Why should you care?

When was the last time you went 24 hours without going online?

Why should you care?

- To know how the Internet works
  - What may be wrong with your networks
  - When was the last time you went 24 hours without going online?

- Network programmers get respect
  - In high demand, get paid well

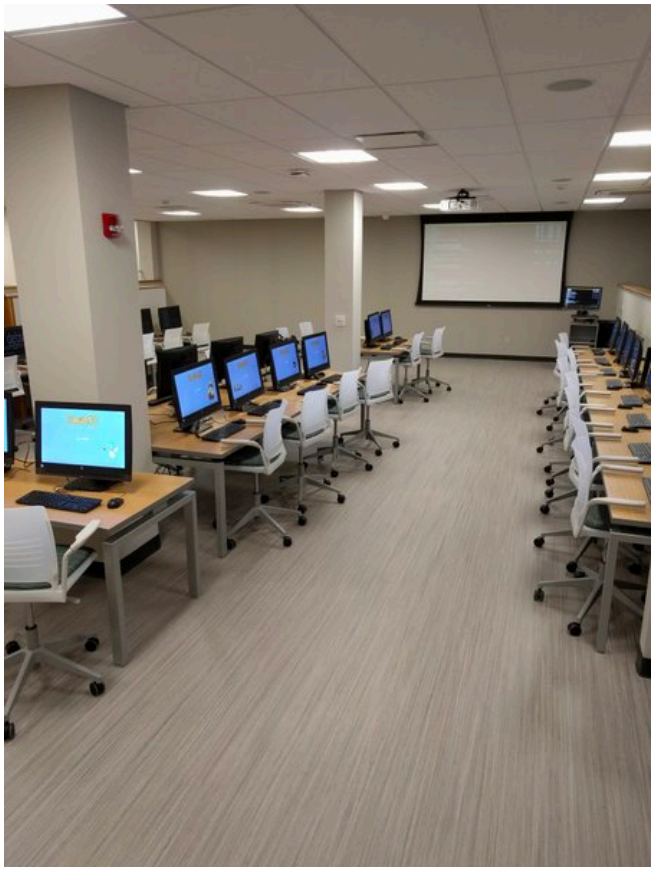
Pull back the curtain on the Internet

Dorothy and Toto pulling back the curtain in Wizard of Oz

Resources: Labs

- Github Enterprise: https://github.swarthmore.edu

- Lab sections:
  - Friday 3:00 - 4:30 PM
  - Friday 4:30 - 6:00PM



- slides on course website
- piazza: class/lab recordings
- slack: lab/office hours

Resources: Piazza!

- Piazza Q&A Forum
  https://piazza.com/swarthmore/fall2019/cs43/home

- All announcements will be on Piazza
  - Weekly in-class worksheets
  - Anonymized Grade Listing

- Use Piazza! (you will get participation points)
  - Your classmates benefit from your questions
  - Your classmates can answer your questions
  - We will check the forum frequently
  - Post publicly unless you have code in your question.

How does this class work?

This class is designed a bit differently:

- Class will be centered around you
- Requires your participation

- Before class:
  - Do the readings
  - Watch videos (if present for that week)
  - Complete the reading quiz (short google form)

How does this class work?

Class Discussions:

- Break-out sessions in groups of 4
- 2 – 3 Discussion questions (come prepared)
- Each discussion ~15 minutes
- Record group observations in google drive.

- Groups
  - Assigned for first week
  - Choose your own later on Piazza.

Grading

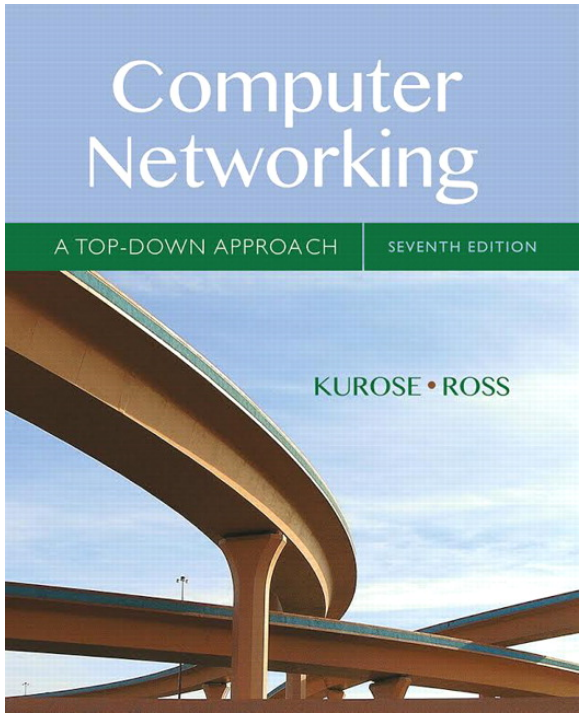
- 40% : Lab assignments (8%, 8%, 8%, 8%, 8%)
- 10% : Final presentation
- 45% : Homeworks
- 10% : Class participation, Reading quizzes & Lab attendance

Grading

- 40% : Lab assignments (8%, 8%, 8%, 8%, 8%)
- 10% : Final presentation
- 45% : Homeworks
- 10% : Class participation, Reading quizzes & Lab attendance

* I will drop your three lowest scores.

Textbook



- Computer Networking: A Top-Down Approach (7th Edition)

Policies

Genie (as William F. Buckley Jr)“ There are a few,..provisos, a, a couple of quid pro quos.” - in Aladdin

- Lab Lateness
  - 5 days of extra time for the semester to be used at the granularity of **2 late days per lab**
  - Email AFTER you are done!
  - No Email: Grade whatever is present at the deadline.

Policies

Genie (as William F. Buckley Jr)“ There are a few,..provisos, a, a couple of quid pro quos.” - in Aladdin

- Homework Lateness
  - 5 days of extra time for the semester to be used at the granularity of **1 late days per HW**
  - Email AFTER you are done!
  - No Email: Grade whatever is present at the deadline.

Policies: academic dishonesty

- Collaboration
  - **You may discuss approaches, not solutions**
  - You must submit your own work

- Cheating
  - We take this very seriously. It can have a negative impact on your course grade, your GPA and your record at Swarthmore and beyond.
  - **Don't do it!**

Policies: academic dishonesty

- Few examples of cheating on labs
  - Screen sharing with folks not in your lab partnership
  - “Let me read my code out to you, or share the exact API for a particular function”
  - Share in words the content in your code: “I first used strncpy to copy the string up to n bytes, and then appended a null character at the end”

Policies: academic dishonesty

- Examples of how not to cheat:
  - Behave as though you are a CS ninja
  - “What approaches did you try so far?”, “Looks like you have gotten more of the string than you need to, use man pages to look at other string functions”
  - Don’t know how to help your friend? Ask them to post to Piazza to the class or send it privately to me.

Lab/HW Schedule

- Labs due every two weeks:
  - Labs on Friday
  - Due Thursday via Github
- Homeworks due every two weeks: alternates with labs
  - Out on Friday
  - Due Thursday via Gradesource

Administrative Questions?

- All of this info on class website

- Feel free to ask on Piazza discussion board

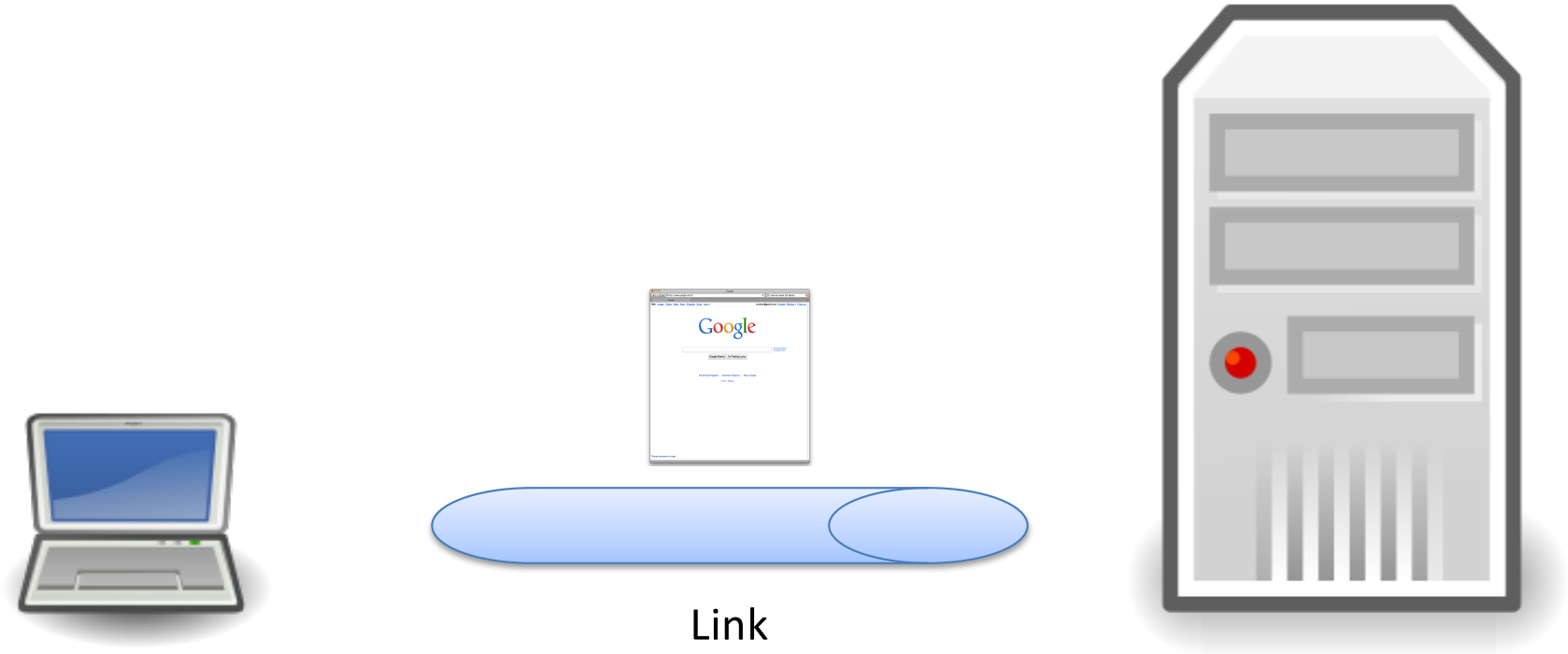
What is the goal of a network?

- Allow devices communicate with one another and coordinate their actions to work together.

- Piece of cake, right?

A "Simple" Task

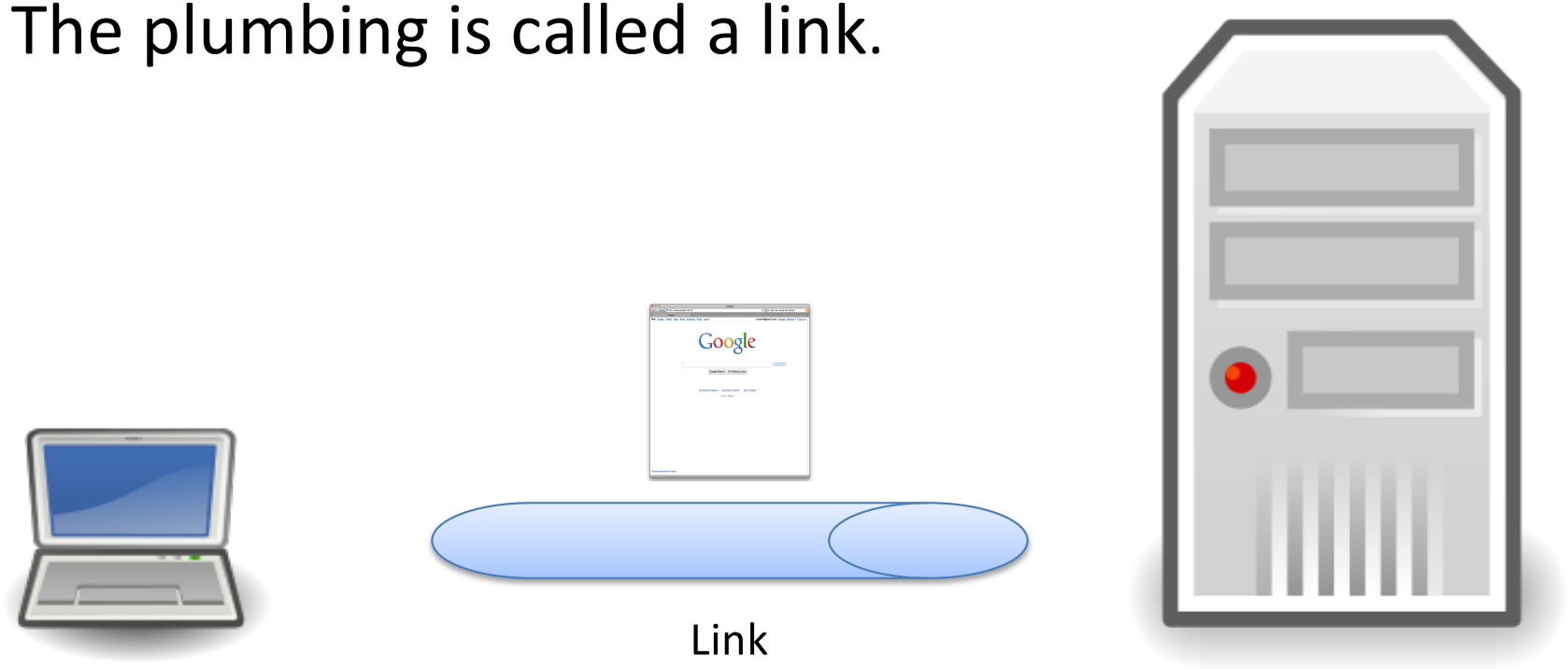
Send information from one computer to another



A "Simple" Task

Send information from one computer to another

- hosts: endpoints of a network
- The plumbing is called a link.



Host
(PC)

Link

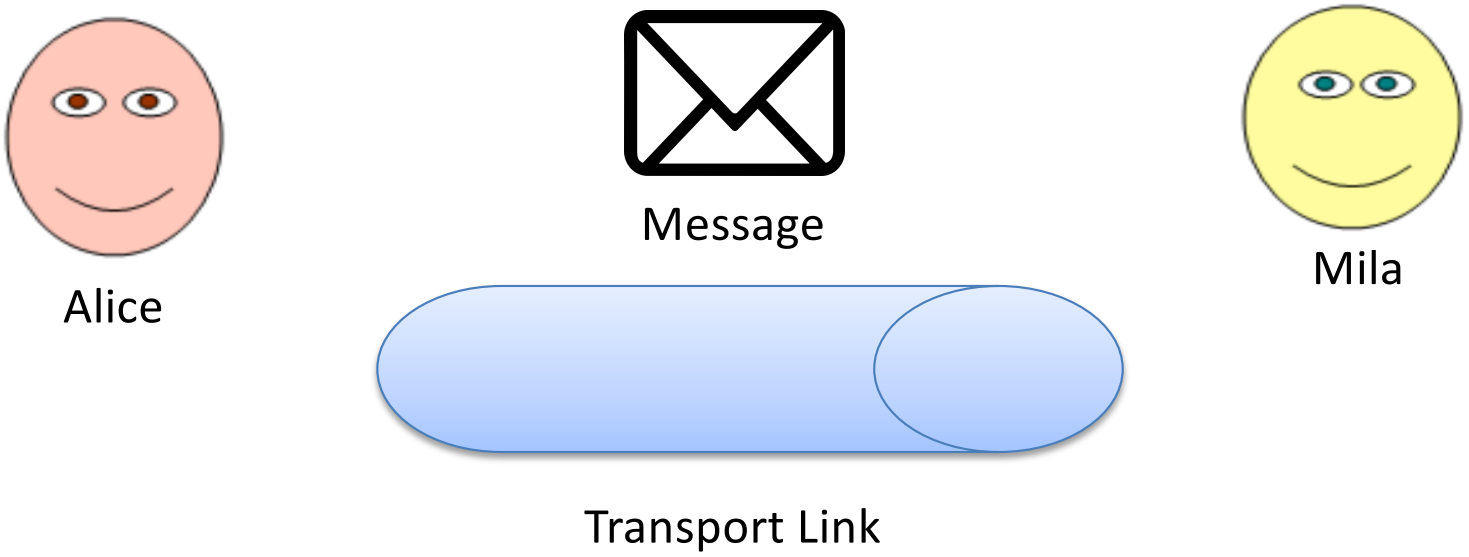
Host
(Server)

A "Simple" Task: Sending a message from host to destination

But first... let's try the postal system, something we are all (still!) familiar with and address a couple of key challenges..

A "Simple" analogous task: Post-it Note

Alice and Mila are Swatties starting out their semester and are roommates. Alice wants to give Mila a reminder to get milk.



A "Simple" analogous task: Post-it Note

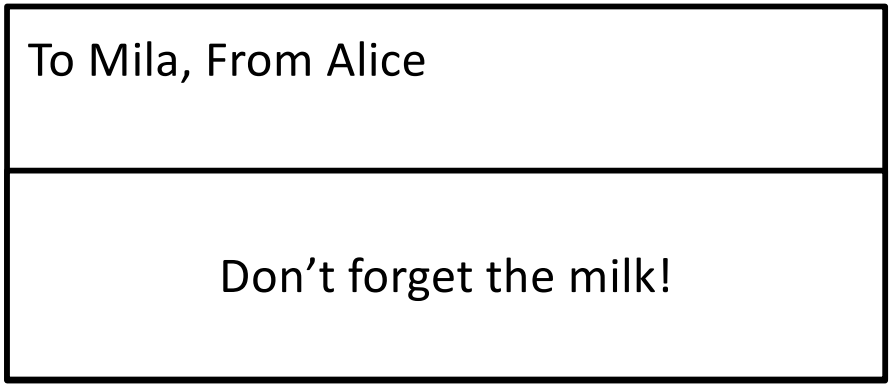
Alice and Mila are roommates, Alice wants to give Mila a reminder to get milk. Figure out some key tasks:

1. **Structure of the message:**
   - Construct the message that Alice posts to Mila.
2. **Organizing a drop-off point.**
   - Who chooses the drop-off point?
3. **Write a protocol to write a note /post—it to your housemate**

A "Simple" analogous task: Post-it Note

Alice and Mila are roommates, Alice wants to give Mila a reminder to get milk.

1. **Structure of the message: (Alice to Mila)**

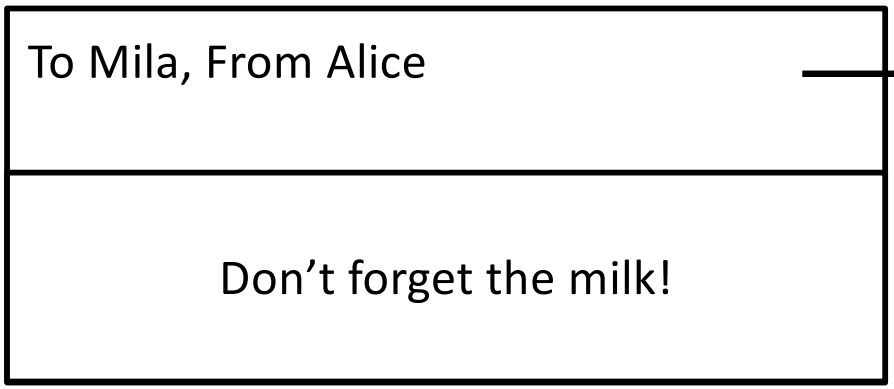


Irrespective of the source and destination, the format of the message stays the same.

A "Simple" analogous task: Post-it Note

Alice and Mila are roommates, Alice wants to give Mila a reminder to get milk.

1. **Structure of the message: (Alice to Mila)**



Header: contains sender/receiver information.

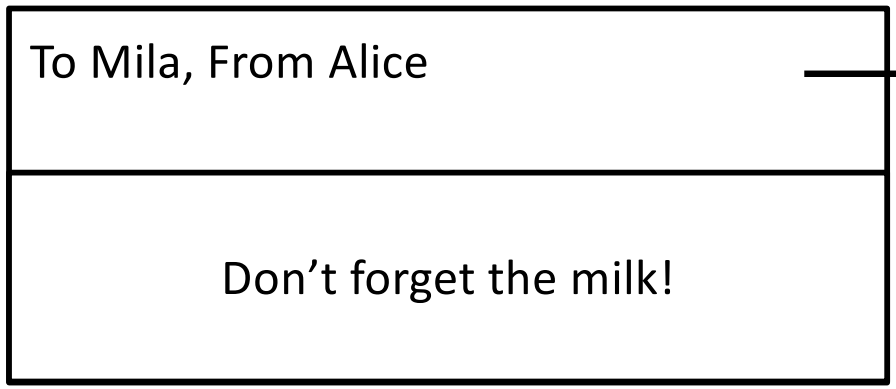
- metadata about the message
- what other metadata might you add?

Irrespective of the source and destination, the format of the message stays the same.

A "Simple" analogous task: Post-it Note

Alice and Mila are roommates, Alice wants to give Mila a reminder to get milk.

1. **Structure of the message: (Alice to Mila)**

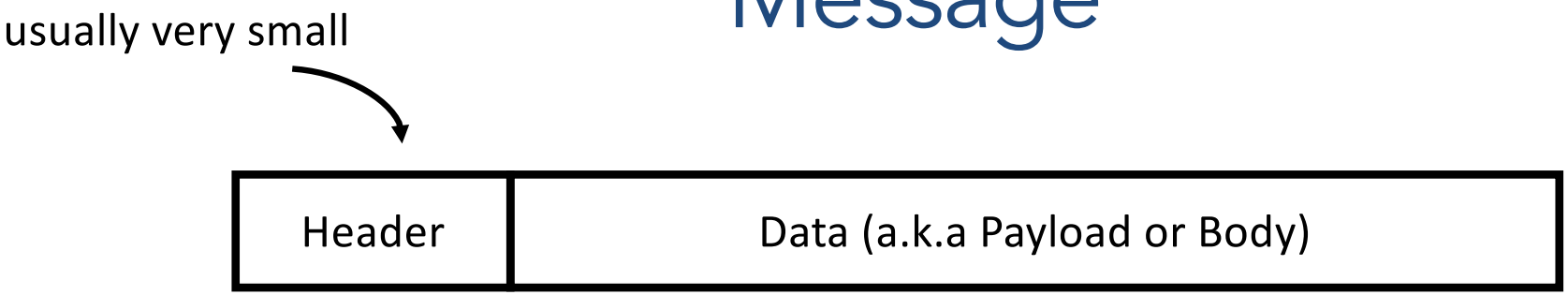


Header: contains sender/receiver information + additional state

- timestamp
- urgent! (priority)
- ordering of messages (1 of 10..)
- error control..

Irrespective of the source and destination, the format of the message stays the same.

Message



- Message: Header + Data
- Data: what sender wants the receiver to know
- Header: information to support protocol
  - Source and destination addresses
  - State of protocol operation
  - Error control (to check integrity of received data)

A "Simple" analogous task: Post-it Note

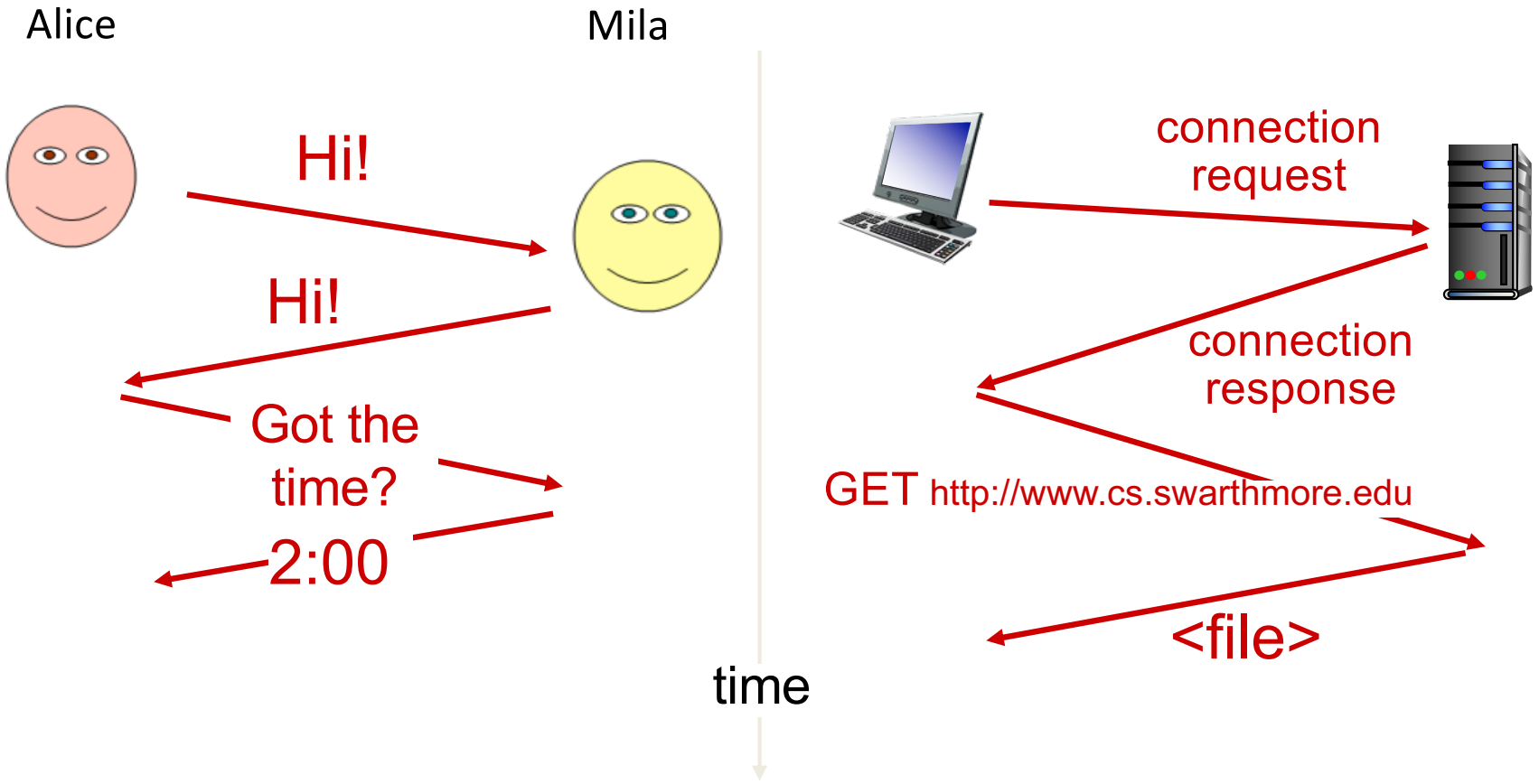
Alice and Mila are roommates, Alice wants to give Mila a reminder to get milk.

2. **Organizing a drop-off point.**
    - Who decides?
    - Generally by mutual consensus – previously agreed upon location.

Everyone agrees to place messages on refrigerator to relay messages to housemates

What is a protocol?

Protocol: message format + transfer procedure



Human Protocol

Network Protocols (defined in RFCs)

What is a protocol?

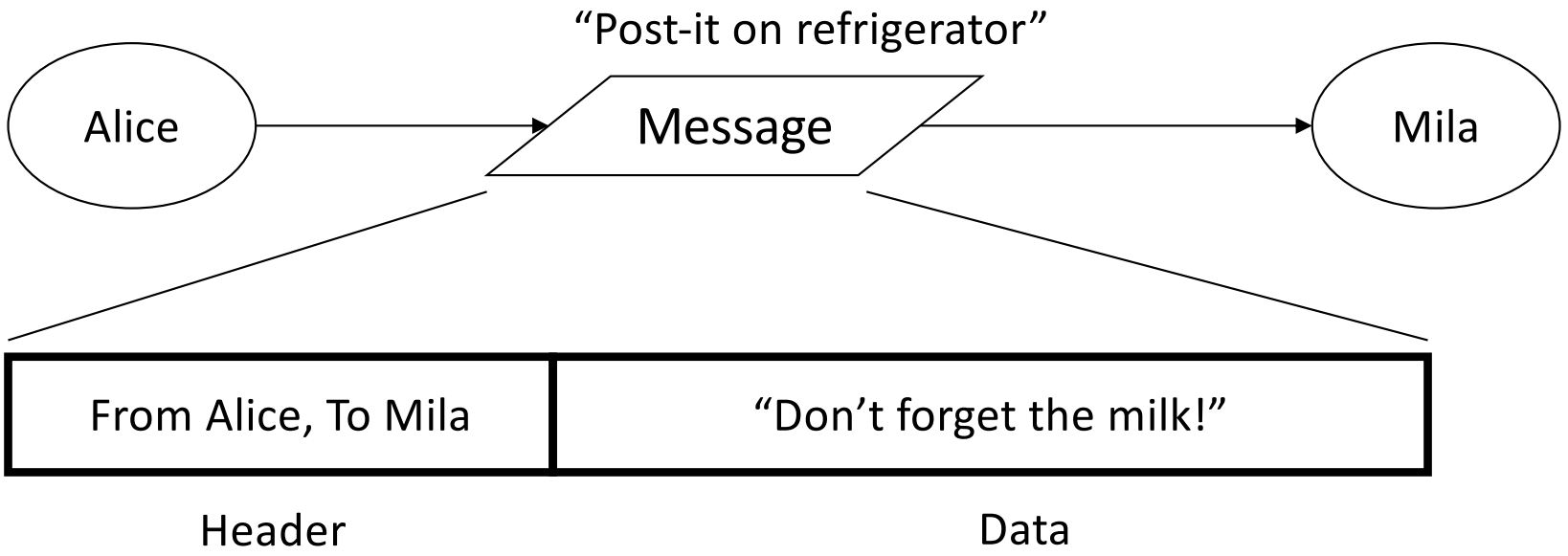
Goal: get message from sender to receiver

Protocol: message format + transfer procedure

- Expectations of operation
  - first you do x, then I do y, then you do z, ...

- Multiparty! so no central control
  - sender and receiver are separate processes

A "Simple" analogous task: Post-it Note



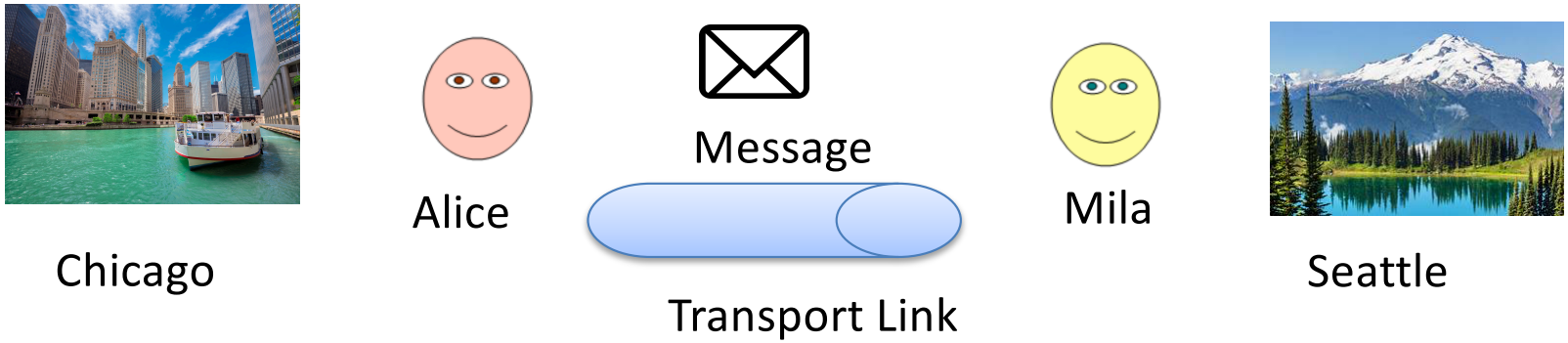
Write a protocol to write a note /post—it to your housemate

Protocol: message format + transfer procedure

- Message format: (from, to), message contents
- Transfer procedure: post on refrigerator

A "Simple" analogous task: Postal Mail

Alice moves to Chicago and Mila to Seattle for summer internships. Alice would like to send Mila a birthday card.



A "Simple" analogous task: Postal Mail

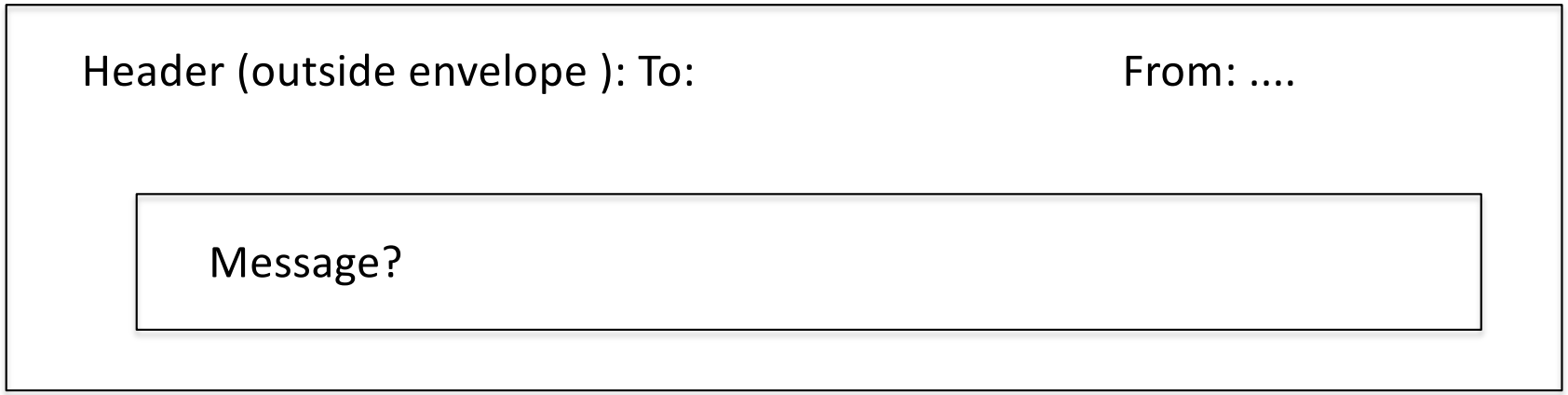
Alice would like to send Mila a birthday card.

1. **Construct the message and header. Have the header and message portions changed from the previous scenario?**
2. **List the message format and transfer procedure of the "mail sending protocol" that Alice uses.**
   - Who chooses the drop-off point?
   - Is this the only protocol in use?
3. **Message transportation and delivery**
   - Whose job is it to:
     - choose the carrier?
     - plan the route?
     - deliver the message?
     - ensure the message is not lost?

A "Simple" analogous task: Postal Mail

Alice would like to send Mila a birthday card.

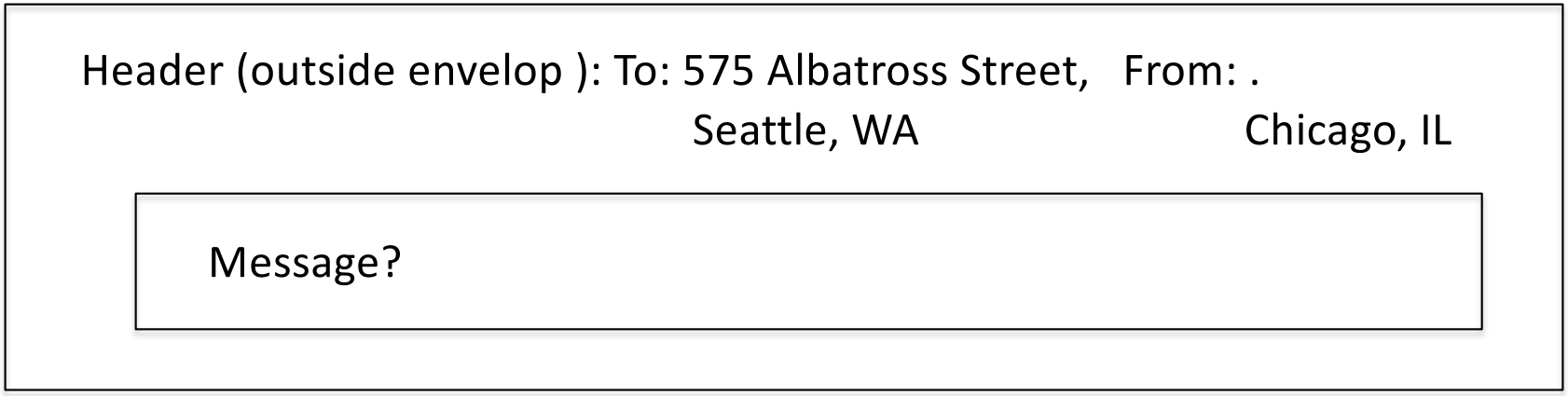
1. **<u>Construct the message and the header. Have the header and message portions changed from the previous scenario?</u>**



A "Simple" analogous task: Postal Mail

Alice would like to send Mila a birthday card.

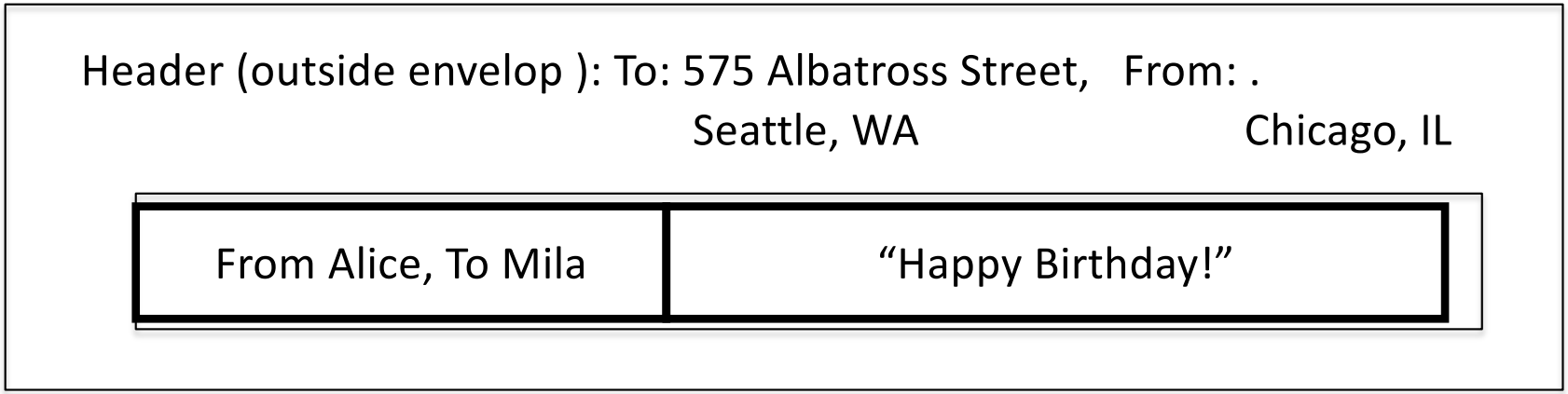
Header portion



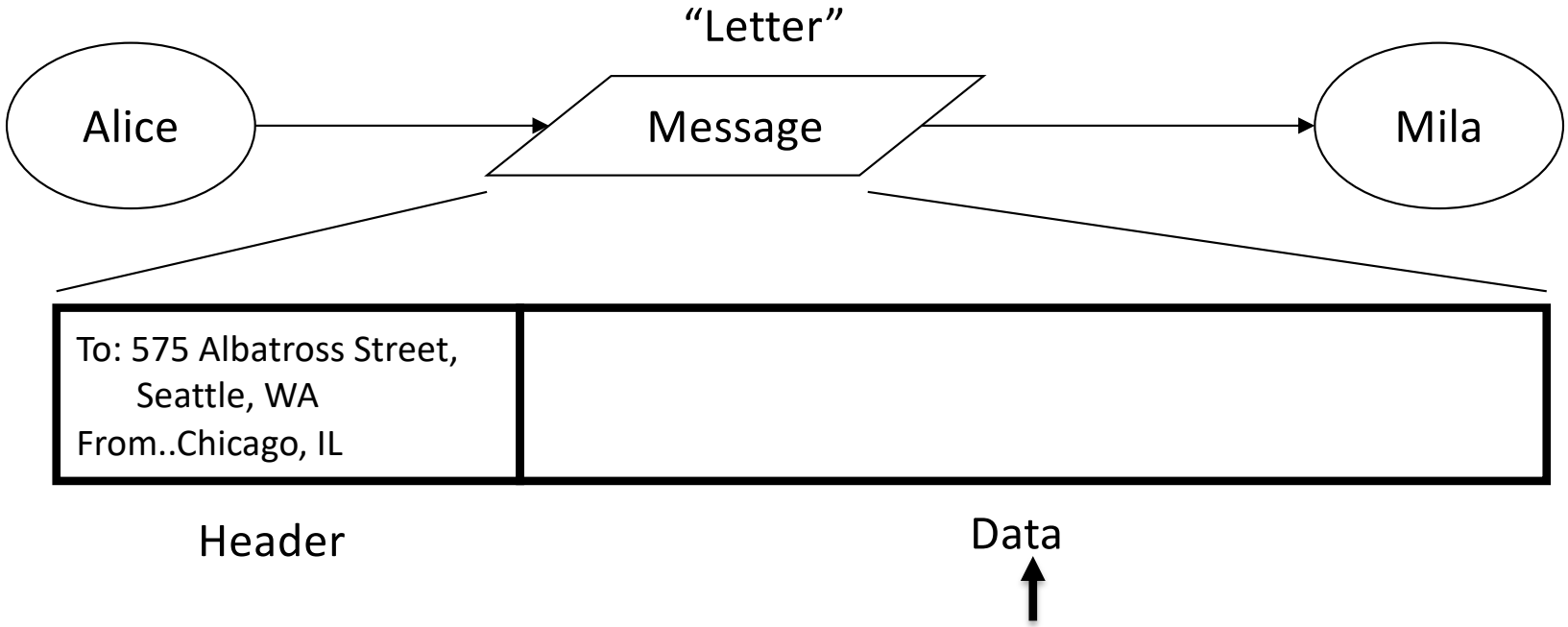
A "Simple" analogous task: Postal Mail

Alice would like to send Mila a birthday card.

Message portion

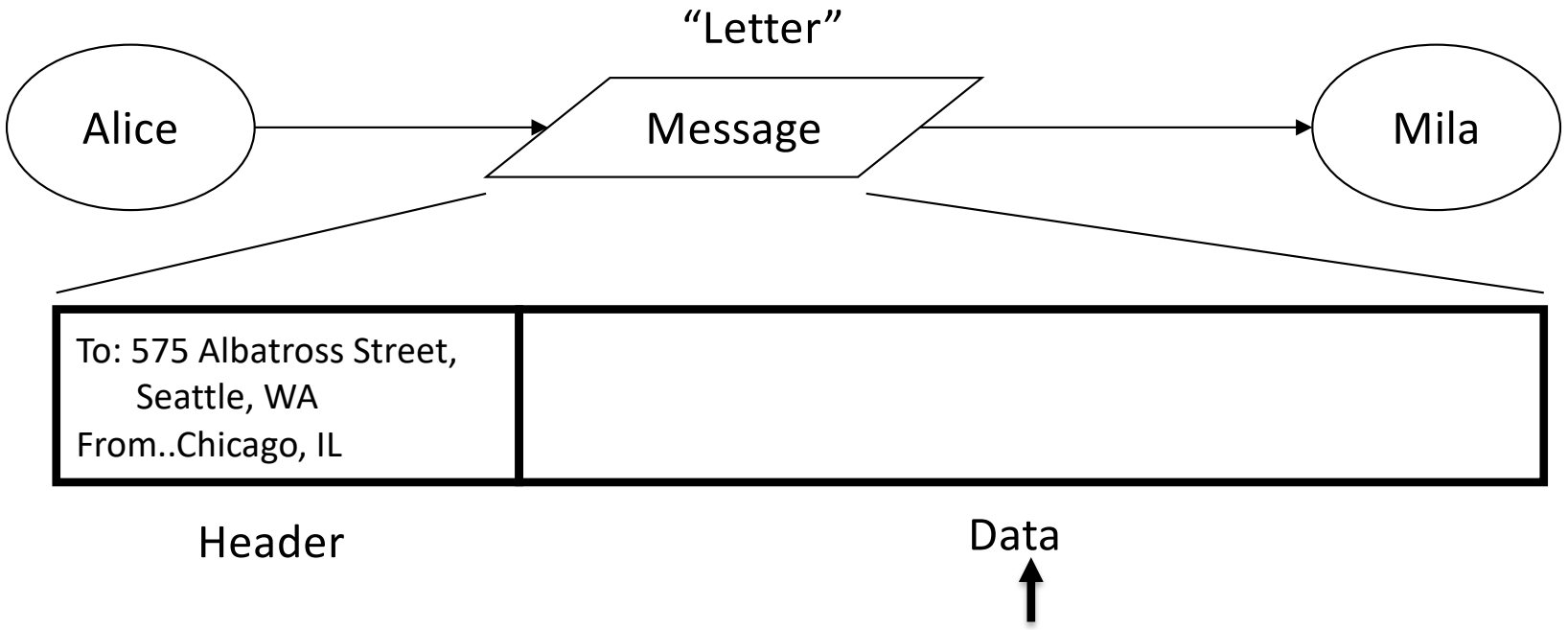


A "Simple" analogous task: Postal Mail



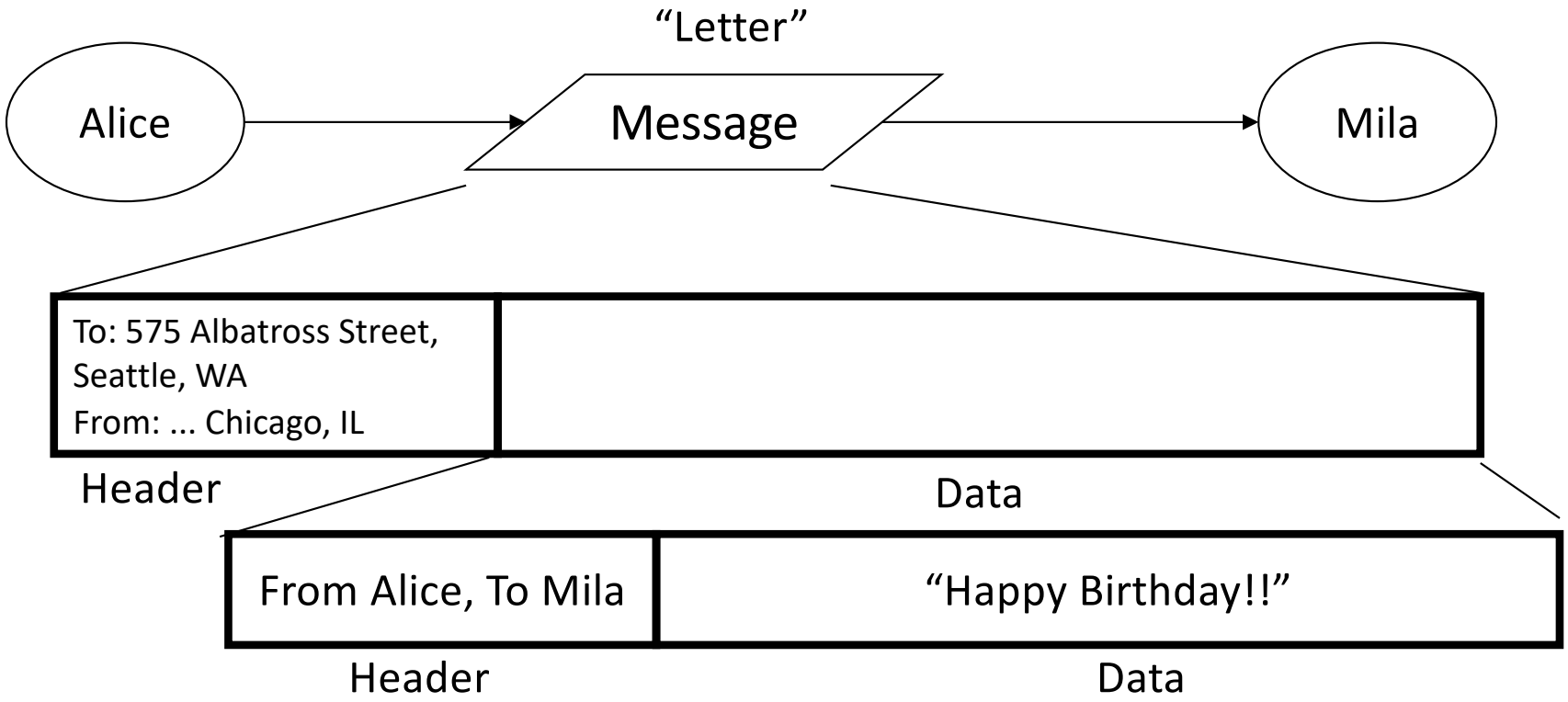
The post office does NOT care about what's in here, and shouldn't be looking at it…

A "Simple" analogous task: Postal Mail



- **Mail Sending Protocol**
  - Message format: (from, to), message contents
  - Transfer procedure: post mail in mailbox (agreed upon convention)

A "Simple" analogous task: Postal Mail: other protocols in use?



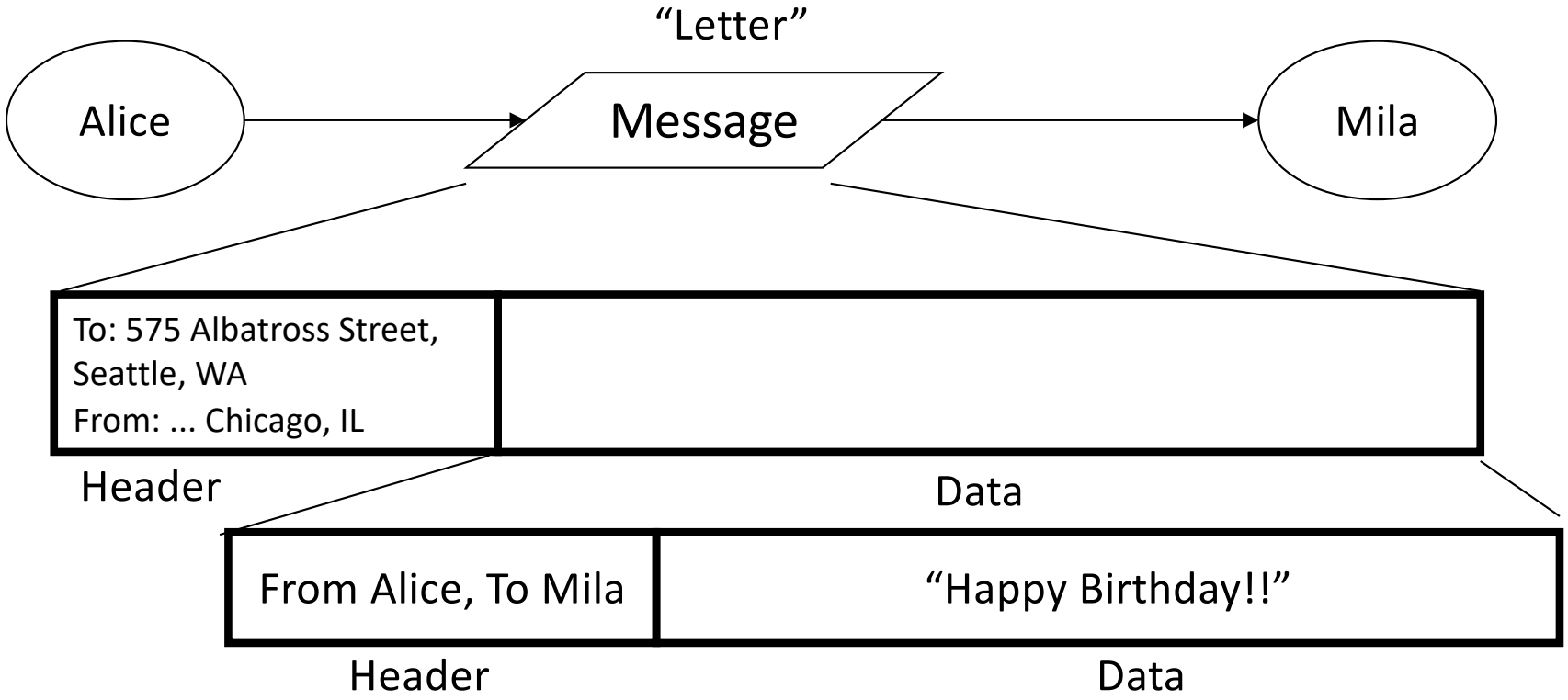
Mail Protocol

- Message format: (from, to), message contents
- Transfer procedure: post mail in mailbox (agreed upon convention)

Card Protocol (within the mail protocol!)

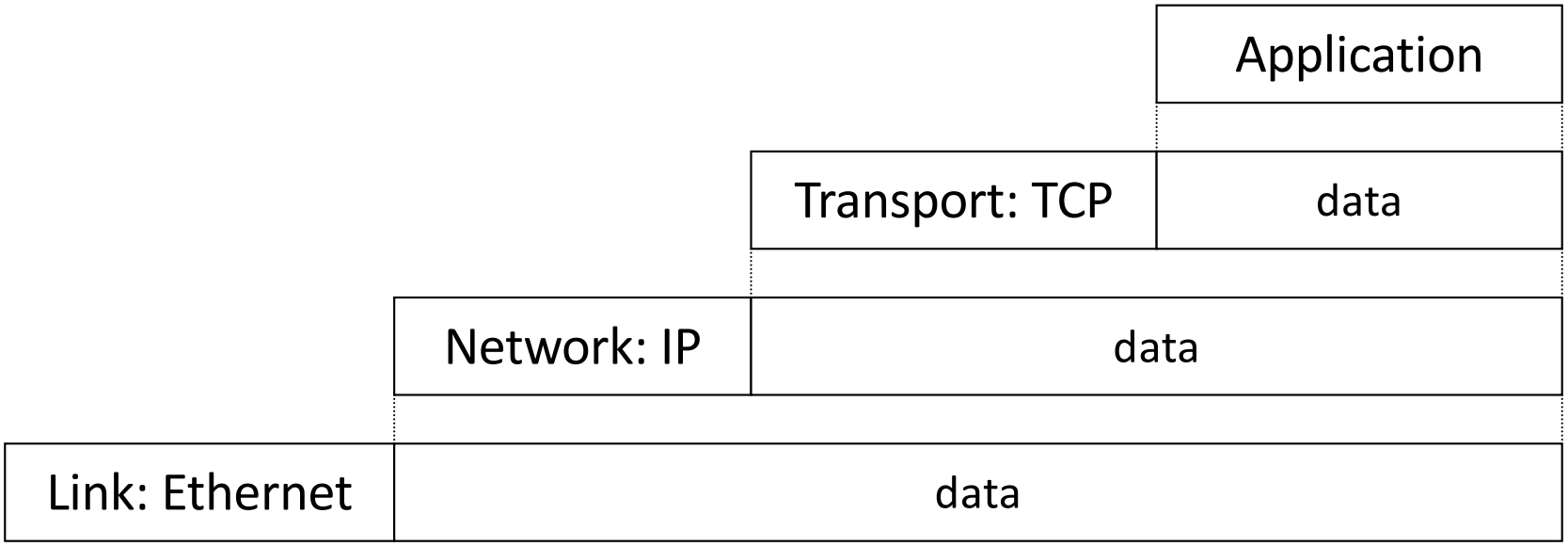
- Message format: (from, to), message contents

Message Encapsulation



- Card protocol: (message + header) treated as payload
- Put it in another protocol: append an additional header

Message Encapsulation



- Higher layer within lower layer
- Each layer has different concerns, provides abstract services to those above

A "Simple" analogous task: Postal Mail

- **Message transportation and delivery**
- Who's job is it to:
  1. provide the sender and receiver addresses?
  2. choose the carrier?
  3. plan the route?
  4. deliver the message?
  5. ensure the message is not lost?

A "Simple" analogous task: Postal Mail

- **Message transportation and delivery**

- Who's job is it to: Alice decides as the "end host"(1, 2)

1. provide the sender and receiver addresses?
2. choose the carrier?
3. plan the route?
4. transport vehicles?
5. ensure the message is not lost? (reliability)

Postal Department decides as the service that provides message transfer (3, 4)

Reliability? Open question – stay tuned!

Layering: Separation of Functions

Letter: written/sent by Alice, received/read by Mila
Postal System: Mail delivery of letter in envelope

- Alice and Mila
  - Don’t have to know about delivery
  - However, aid postal system by providing addresses
- Postal System
  - Only has to know addresses and how to deliver
  - Doesn’t care about “data”: Alice, Mila, letter

Abstraction!

- Hides the complex details of a process
- Use abstract representation of relevant properties make reasoning simpler
- Ex: Alice and Mila knowledge of postal system:
  - Letters with addresses go in, come out other side

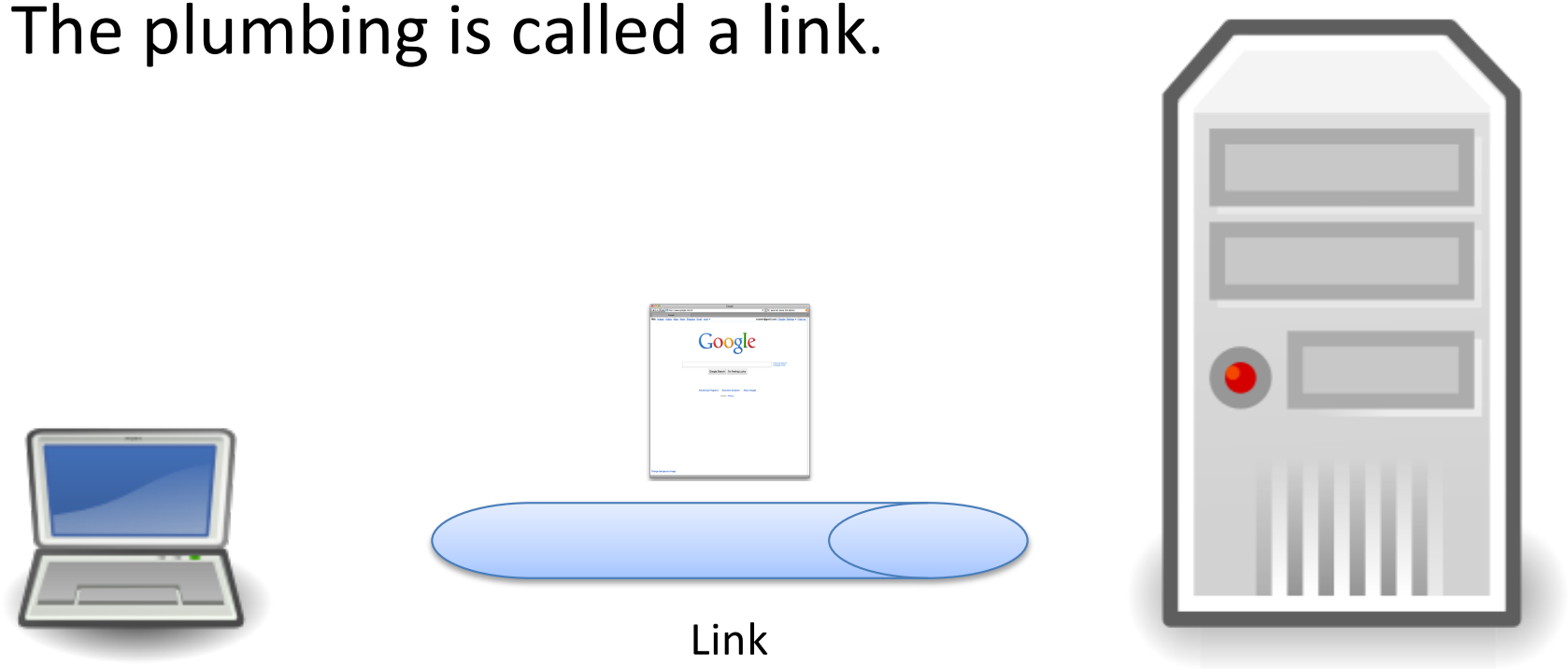
A "Simple" analogous task: Postal Mail

- Many more considerations..
  - Who decides the the sender and receiver addresses? Does someone maintain a mapping peoples' names to addresses?
  - Can Mila always be guaranteed of this delivery date? What factors influence delivery ?
  - What if the mail gets lost – who's responsibility is it? Alice, Mila or someone else?
  - What about security? privacy?

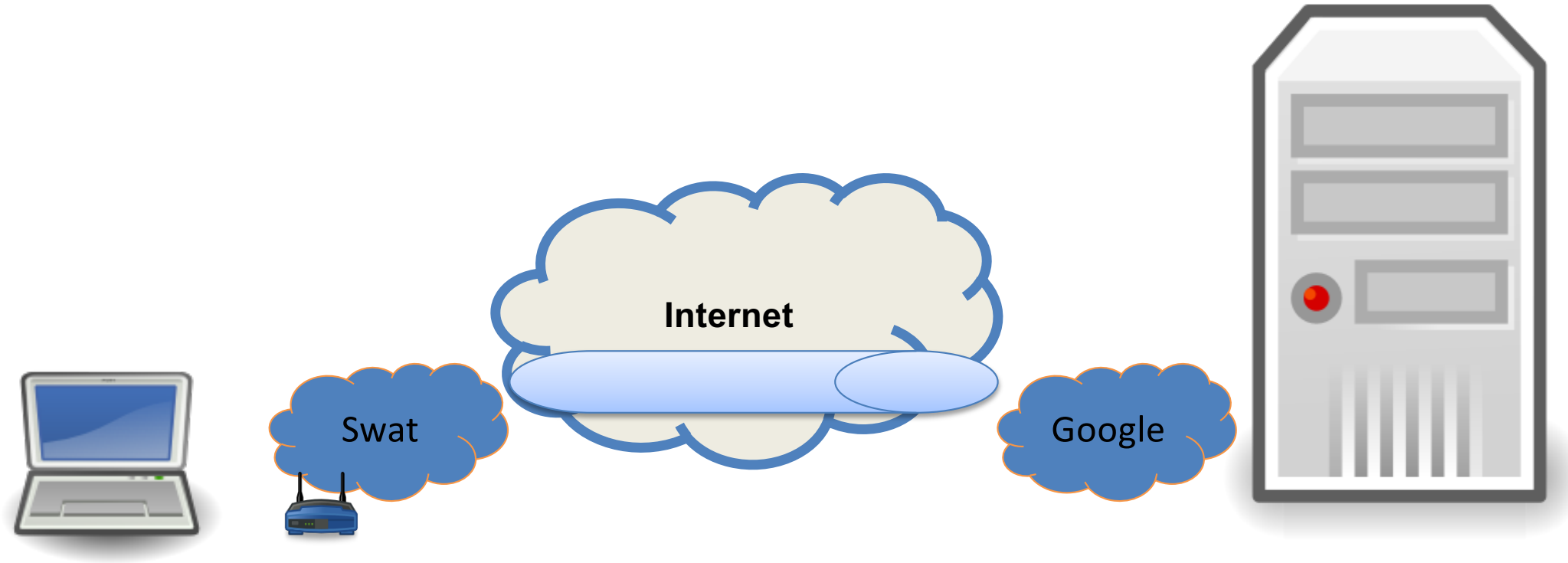
A "Simple" Task

Send information from one computer to another

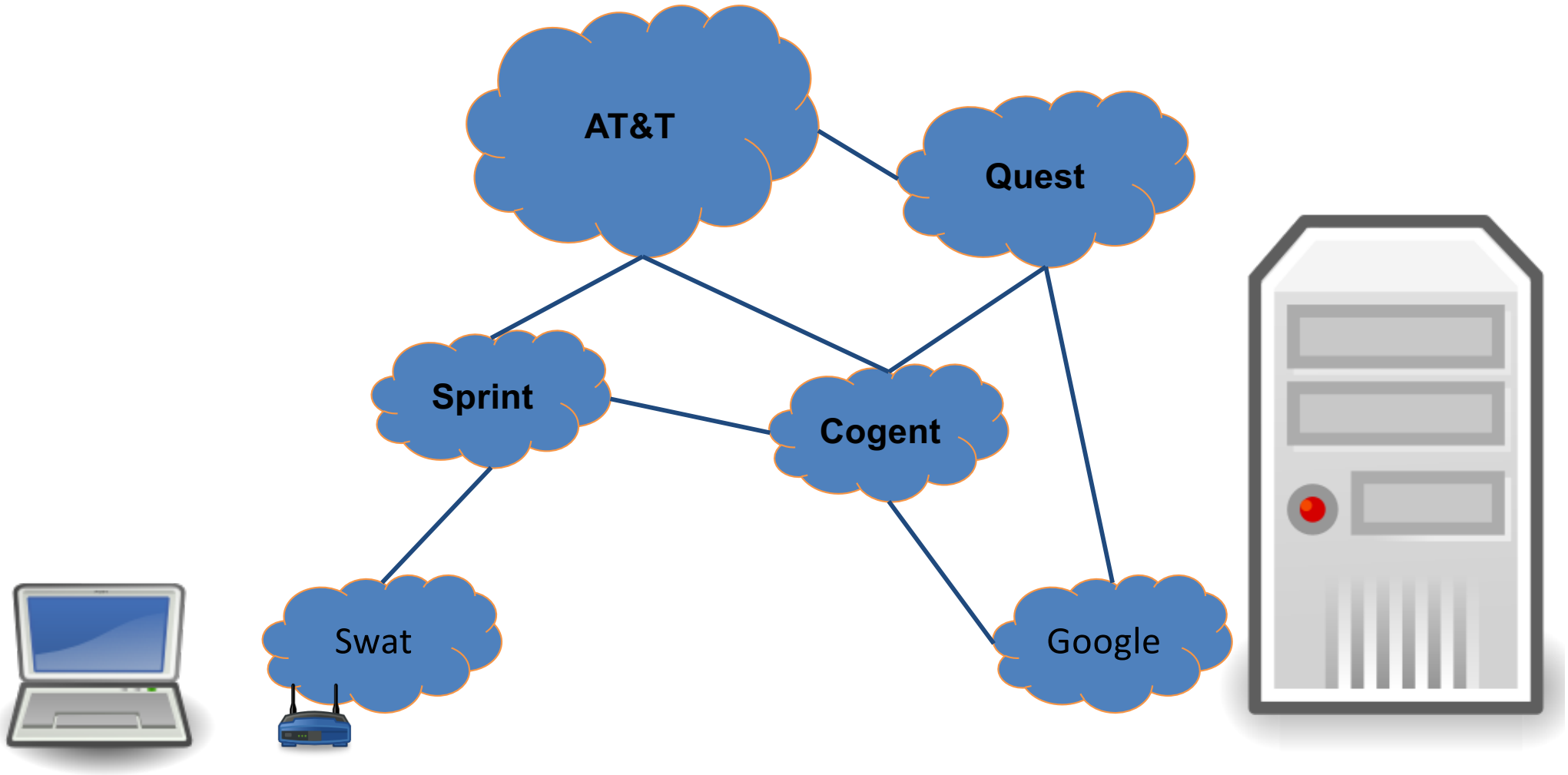
- hosts: endpoints of a network
- The plumbing is called a link.



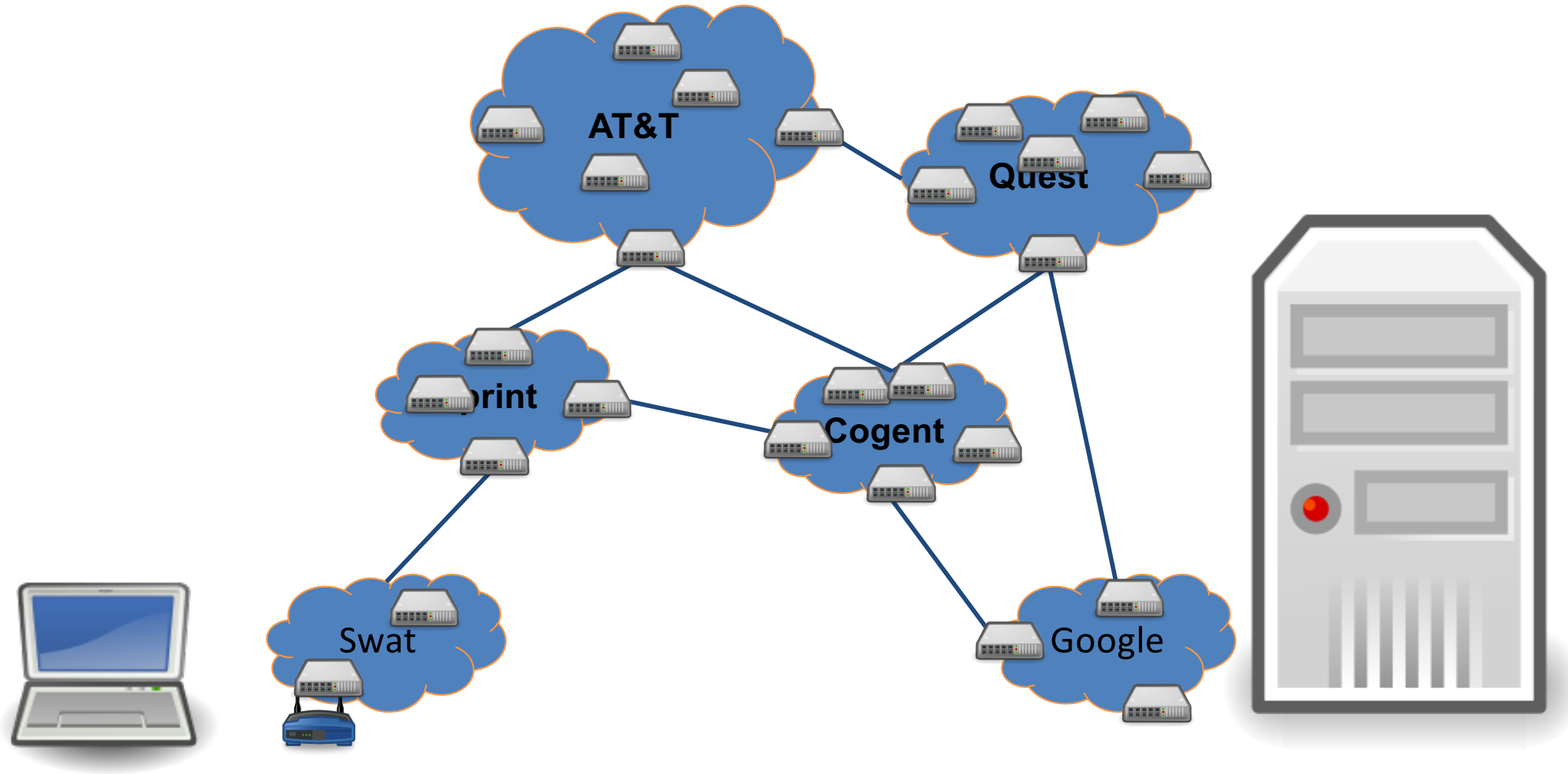
Not Really So Simple…



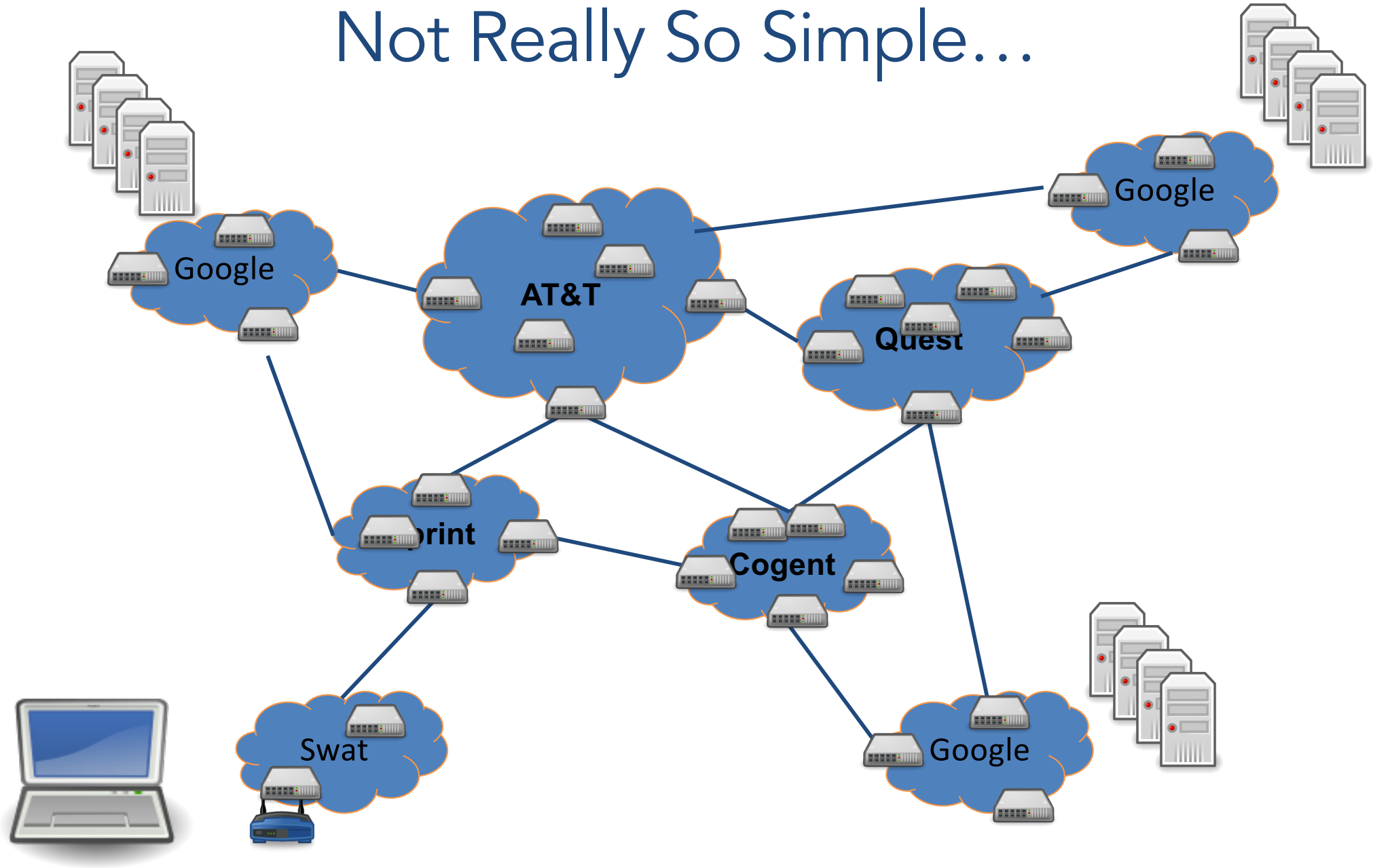
Not Really So Simple…



Not Really So Simple…



Not Really So Simple…



We only need…

- Manage complexity and scale up
  - Layering abstraction: divide responsibility
  - Protocols: standardize behavior for interoperability

We only need…

- Manage complexity and scale up
- Naming and addressing
  - Agreeing on how to describe/express a host, application, network, etc.

We only need…

- Manage complexity and scale up
- Naming and addressing
- Moving data to the destination
  - Routing: deciding how to get it there
  - Forwarding: copying data across devices/links

We only need…

- Manage complexity and scale up
- Naming and addressing
- Moving data to the destination
- Reliability and fault tolerance
  - How can we guarantee that the data arrives?
  - How do we handle link or device failures?

We only need…

- Manage complexity and scale up
- Naming and addressing
- Moving data to the destination
- Reliability and fault tolerance
- Resource allocation, Security, Privacy..

We only need…

- Manage complexity and scale up
- Naming and addressing
- Moving data to the destination
- Reliability and fault tolerance
- Resource allocation, Security, Privacy..

(Lots of others too.)

Next Class

- Layering & division of responsibilities
- OSI Model
- End-to-end argument
- HTTP! An Application Layer Protocol

TODO List

- Reading: Protocols
  - Sections 1.1, 1.5
- Sign up on Piazza!
- Please let me know:
  - Your preferred name/pronouns, if different than roster information
  - Academic accommodations