Announcements

Final Project : Deadlines

- Wed (12/7): Project draft to Learn@UW dropbox by 5pm
  - Whatever you have completed TODAY
  - No partner changes after TODAY (email us if problems)
- Due December 12 – In-class Demos
  - Final version due at 5pm
  - Demo with TA for grading – last week

Upcoming Office/Lab hours

- Wednesday: 11-12 Instructor in Office, 12-2 Victor in Lab
- Thursday: 2-4 Instructor in Office, 4-6 Thea in Lab

More Announcements

Extra Credit: Fill out survey for College Board

- Screenshot to Dropbox HW11-Extra (1/2 hw grade)

Final Exam: 15% of final grade (others 10%)

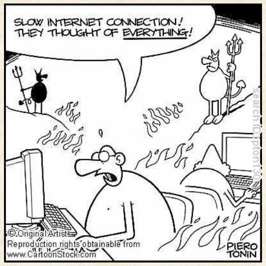
- 12/21 (Wed) from 2:45 - 4:45 pm in Noland 132
- Part 3 of course: 10% - 1 hour
- Cumulative portion: 5%
  - On-line AP test (take: 100% or don't take: 0%)
  - Must take on laptop or in lab with TA supervision
  - After this Monday
  - Due: Exam day (Dec 21st)
  - Bring laptop to Final exam if didn't take beforehand

UNIVERSITY of WISCONSIN-MADISON
Computer Sciences Department

CS 202: Introduction to Computation

Professor Andrea Arpaci-Dusseau

How does a computer… send messages over the Internet?


SLOW INTERNET CONNECTION! THEY THOUGHT OF EVERYTHING!
PIERO TONIN
search ID: pton130
©Original Artist
Reproduction rights obtainable from
www.CartoonStock.com

University of Wisconsin–Madison

http://www.wisc.edu/

WISCONSIN
UNIVERSITY OF WISCONSIN–MADISON

GREAT PEOPLE SCHOLARSHIP GIVE

UW-Madison H1N1 Influenza Updates

Research and Graduate Studies Discussion

Today's news »

UW-Madison students foster community development on island in Uganda Nov. 12

UW-Madison computer team heads to world finals for ninth straight year Nov. 11

For mothers of children with autism, the

Visiting campus
Tours | Campus map | Events calendar | Athletics | Arts | Science | UW Hospital | More »

Admissions
Apply | Undergraduate | Graduate | Professional | Orientation (SOAR) | More »

Student life
Housing | Financial aid | Health | Safety | Recreation/sports | Wisconsin Union | Getting involved | Diversity | More »

Academics
Academic calendar | Catalogs | Course Guide | Schedule of Classes | Schools/colleges | Departments | Majors | Libraries | More »

International
International studies | Study abroad | International students | Languages | More »

Research
Centers/programs | Administration | More »

Outreach
Continuing education | Teacher resources | Business/industry resources | More »

Information for:
Faculty/staff | Alumni | Parents | Business/industry

A campus overpass gets steady foot traffic

Search... UW People

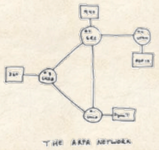
MY UW MADISON
- My UW
- WiscMail
- WiscCal
- My WebSpace
- Learn@UW

About UW-Madison
Employment
Facts
Slide show
Administration
Making a gift
More »

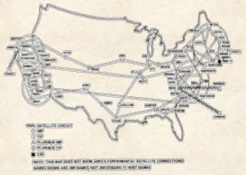
Connect with us on...
Facebook
Twitter
YouTube
iTunes U
RSS

Done

Brief History of Networking



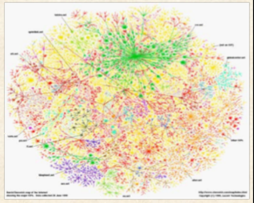
1970



1980

"Internet" developed

1989: World Wide Web



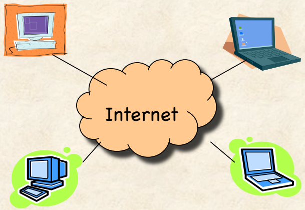
2000

Caveat: Internet ≠ Web

Internet:

Collection of computers connected on network

Communicate with TCP/IP protocol

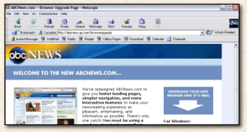


Web (WWW):

Hyperlinked content (web pages) stored on servers

Request and serve pages using HTTP protocol

Built *on top of* the internet


ABCNews.com -- Browser Upgrade Page - Netscape

http://abcnews.go.com/browserupgrade

abcNEWS

WELCOME TO THE NEW ABCNEWS.COM...

We've redesigned ABCNews.com to give you **faster loading pages, simpler navigation, and more interactive features** to make your newsreading experience as pleasant, entertaining, and informative as possible. There's only one catch: **You must be using a**

DOWNLOAD YOUR NEW BROWSER HERE (IT'S FREE):

For Windows:

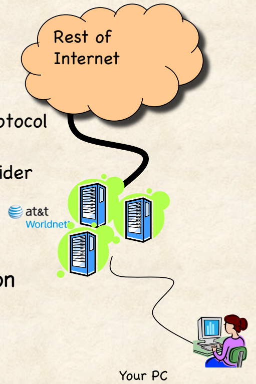
Modern Internet

Simple to connect and use

Need:

- Device capable of speaking right protocol (TCP/IP)
- IP "address" given by Internet provider
- Connection to provider's servers (via modem, DSL, wireless, etc.)

Able to access any other machine on Internet!



Your PC
IP Address:
128.156.16.201

Today's Challenges

Interesting example of large, heterogeneous system

Challenge 1:

How to find someone when no one knows everybody?

Challenge 2:

How to build reliability on top of unreliable protocols?

Challenge 3:

How to cooperate when not in your selfish best interest?

First Challenge

Challenge 1:

How to find someone when no one knows everybody?

Scenario: Imagine you're royal leader with 10,000 peasants in your kingdom

- You need to deliver message to all of them
- "We are being attacked! Come fight for me!"

How might you do that?

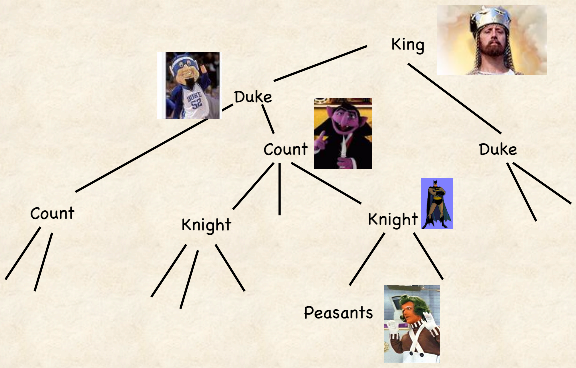
Candy Distribution

How long does it take to distribute a piece of candy to everyone in the room?

First, try a slow approach…

How long did this take?

Solution: Hierarchy



King

Duke

Count

Duke

Count

Knight

Knight

Peasants

Deliver orders by broadcasting message down tree

Candy Distribution #2

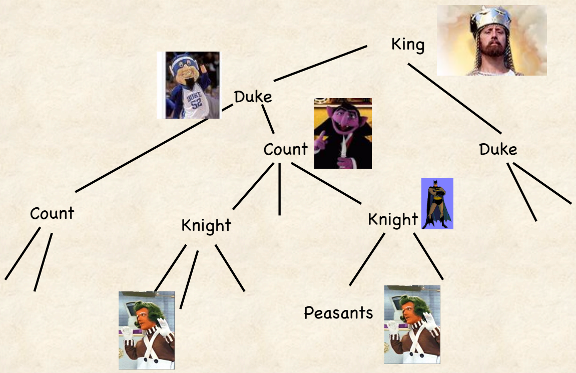
Use a hierarchical approach to distribute a piece of candy to every person in the room

Who should be in charge of each each sub-tree?

What information should each sub-root know?

How long does this take?

Solution: Hierarchy



King

Duke

Count

Duke

Count

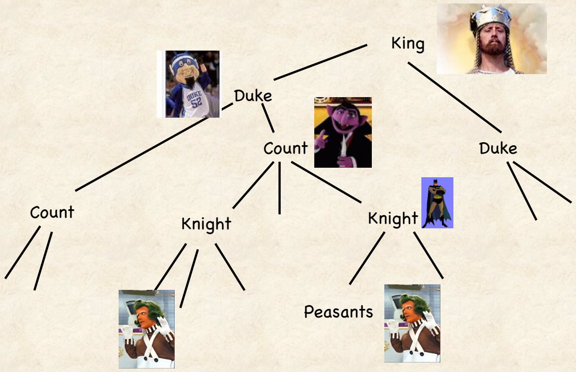
Knight

Knight

Peasants

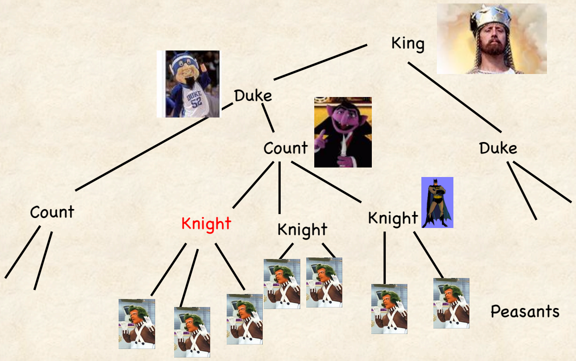
How can peasant send msg to nearby peasant?

Solution: Hierarchy



How can peasant send msg to faraway peasant?

Solution: Hierarchy

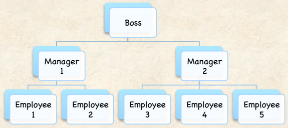


What if knight leaves army? What must happen?

Summary of Hierarchies

Advantages

- Manages complexity
- Decentralized
  - No single entity knows or controls everything
- Boss doesn't know all employees
  - Manager filters info; only propagate relevant info up
- Managers/employees don't know about other branches



Disadvantages

- Can be tricky to handle new entities leaving/ entering system
- Can be bad if too much info is passed up to boss (overworked if micromanage)

How to find machine?

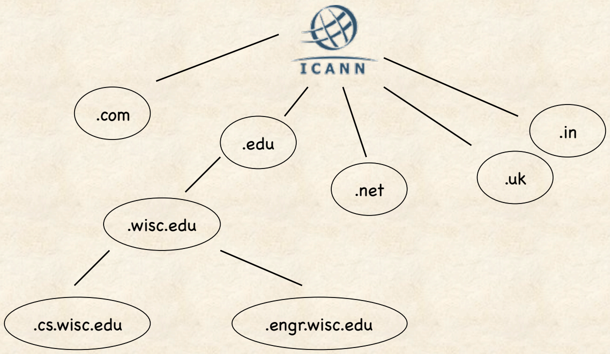
Scenario: Want to send message to www.cs.wisc.edu

- Must translate www.cs.wisc.edu to IP address to send messages
- How are we going to find IP address????

Can't have one server that knows all IP addresses

Use hierarchy!

Hierarchy in Networks: Domain Name System (DNS)


ICANN

.com

.edu

.net

.uk

.in

.wisc.edu

.cs.wisc.edu

.engr.wisc.edu

How to find machine?

Ask appropriate DNS server in hierarchy

- Contact DNS server managing "edu" namespace
- Contact DNS server managing wisc.edu
- Get IP address: 128.105.7.31

Use IP address for routing messages thru physical network hierarchy based upon address

Challenge 2

How to send message reliably when medium might not deliver your message?

Is there some unreliable communication device you use everyday?

What problems do you experience w/ cellphones?

How do you deal with them?

Reliability Problems and Solutions

Can't quite understand message?

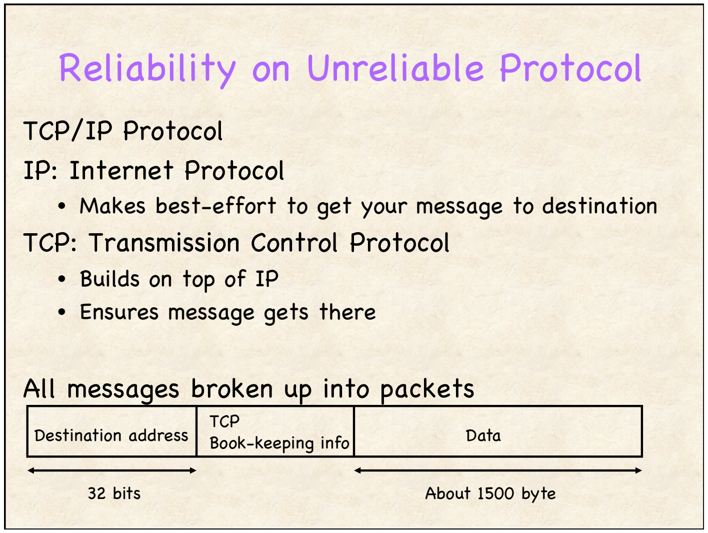
- "Could you say that again?"
- Retransmit message

Don't hear anything?

- "Can you hear me???" <wait> "Can you hear me???"
- Timeout and retry

Sick of listening to other person repeating themselves?

- "I heard you. Go on."
- Acknowledgements



Reliability on Unreliable Protocol

TCP/IP Protocol

IP: Internet Protocol

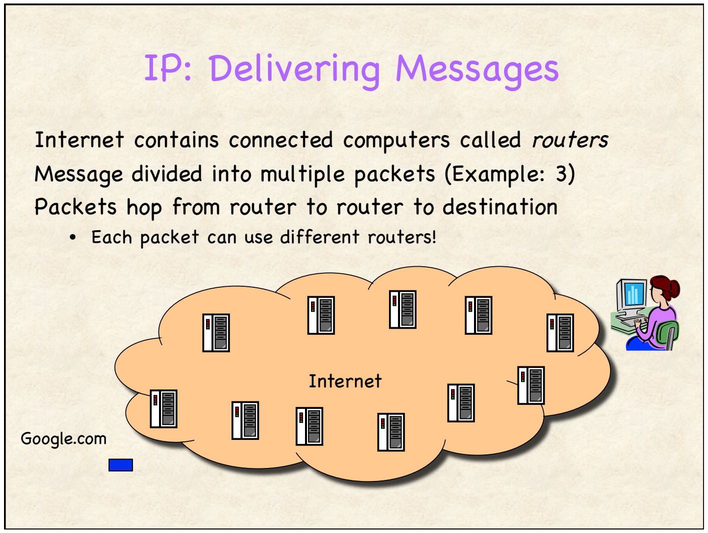
- Makes best-effort to get your message to destination

TCP: Transmission Control Protocol

- Builds on top of IP
- Ensures message gets there

All messages broken up into packets

Destination address	TCP Book-keeping info	Data
32 bits		About 1500 byte



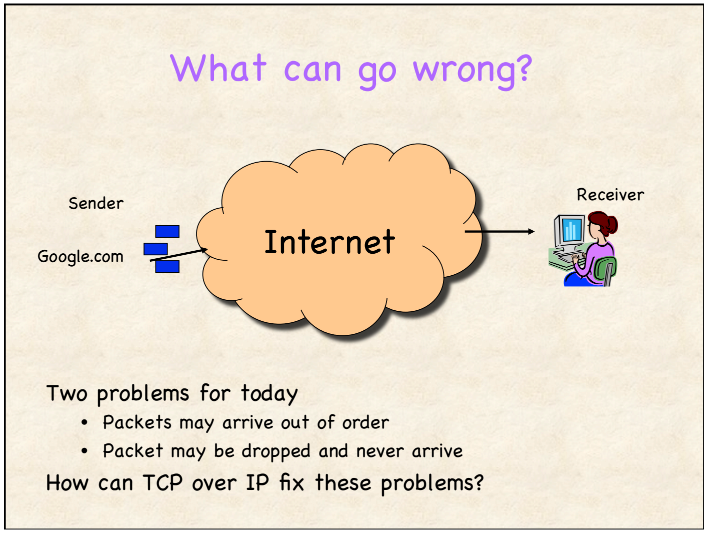
IP: Delivering Messages

Internet contains connected computers called *routers*

Message divided into multiple packets (Example: 3)

Packets hop from router to router to destination

- Each packet can use different routers!

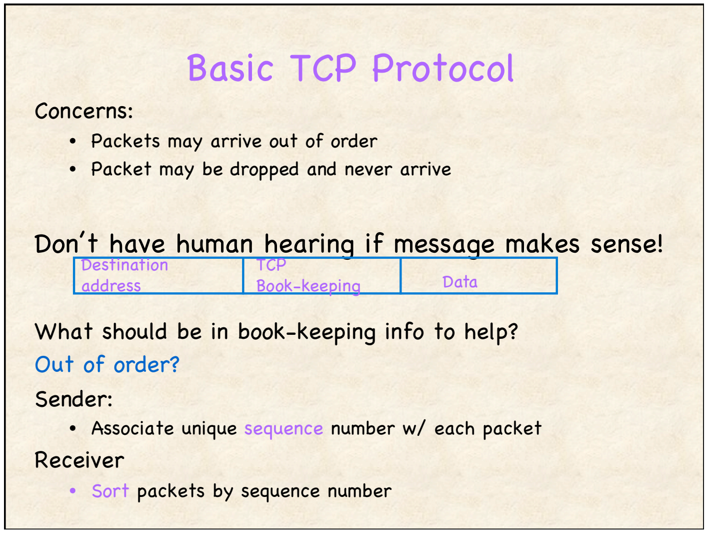


What can go wrong?

Two problems for today

- Packets may arrive out of order
- Packet may be dropped and never arrive

How can TCP over IP fix these problems?



Basic TCP Protocol

Concerns:

- Packets may arrive out of order
- Packet may be dropped and never arrive

Don't have human hearing if message makes sense!

Destination address	TCP Book-keeping	Data

What should be in book-keeping info to help?

Out of order?

Sender:

- Associate unique sequence number w/ each packet

Receiver

- Sort packets by sequence number

Demo: Reconstructing Message

Student A sends message to student B

- Message divided into multiple "packets"

Student A must add information to packets so receiver can reconstruct

Students in between are "routers"

- Different packets take different routes, take different amounts of time

What does receiving student need to do to reconstruct message?

Basic TCP Protocol

Concerns:

- Packets may arrive out of order
- Packet may be dropped and never arrive

Don't have human hearing if message makes sense!

Destination address	TCP Book-keeping	Data

What should be in book-keeping info to help?

Packet dropped?

Receiver

- Acknowledge packets that arrived (by sequence number)

Sender

- Resends packet if no acknowledgment in some time-out interval

Demo: Lost Packets

One bad router in our network

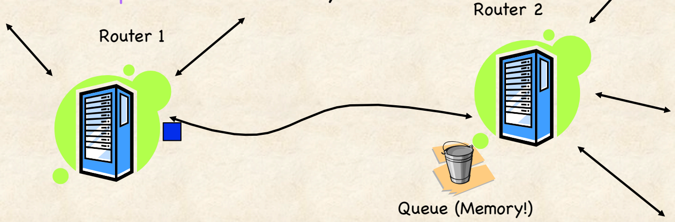
- Drops packets that are sent through it

What must receiver do now to reconstruct packet?

Why should the receiver send "acks" of what it has received instead of requests for missing packets ("negative ack -- nacks")?

Challenge 3

How to cooperate when not in your selfish best interest?

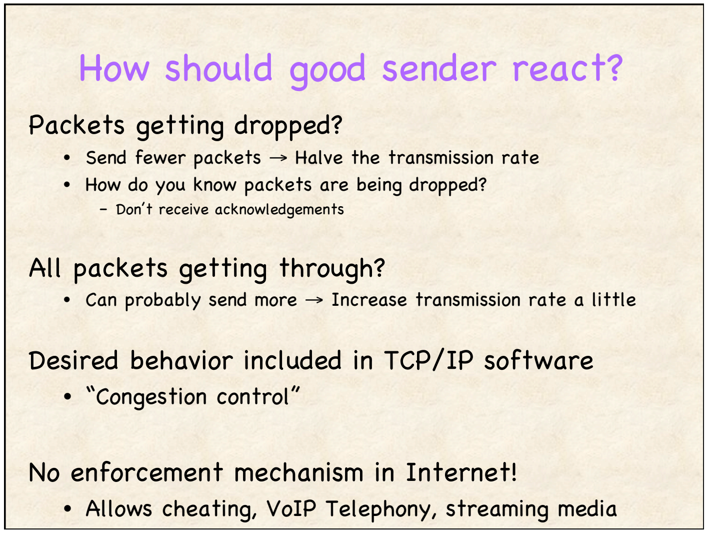


Routers contain fixed amount of memory

- Queue full → packets dropped

Goal is to avoid stressing network and dropping packets

Not just one person's fault that network is overloaded!



How should good sender react?

Packets getting dropped?

- Send fewer packets → Halve the transmission rate
- How do you know packets are being dropped?
  - Don't receive acknowledgements

All packets getting through?

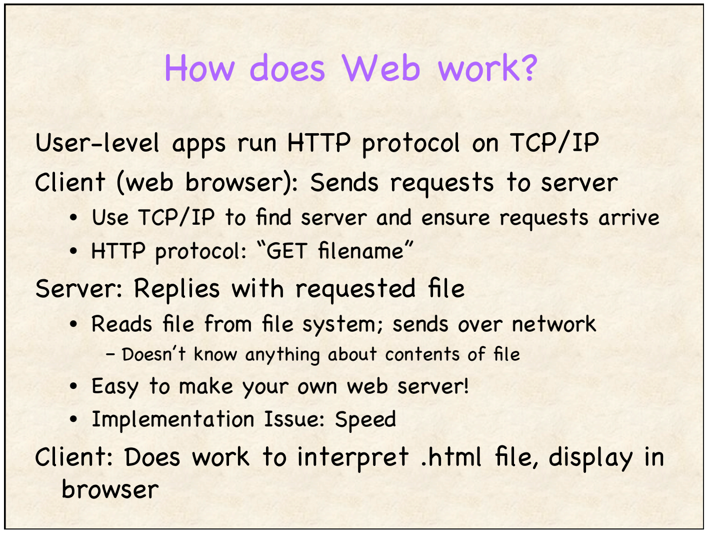
- Can probably send more → Increase transmission rate a little

Desired behavior included in TCP/IP software

- "Congestion control"

No enforcement mechanism in Internet!

- Allows cheating, VoIP Telephony, streaming media



How does Web work?

User-level apps run HTTP protocol on TCP/IP

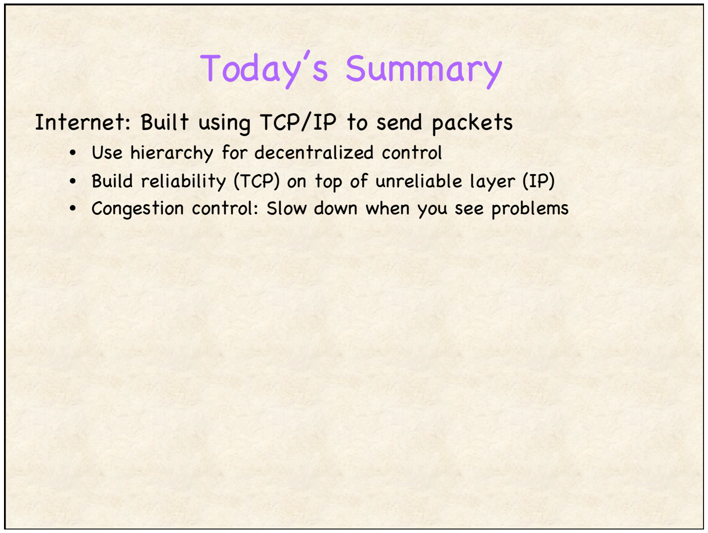
Client (web browser): Sends requests to server

- Use TCP/IP to find server and ensure requests arrive
- HTTP protocol: "GET filename"

Server: Replies with requested file

- Reads file from file system; sends over network
  - Doesn't know anything about contents of file
- Easy to make your own web server!
- Implementation Issue: Speed

Client: Does work to interpret .html file, display in browser



Today's Summary

Internet: Built using TCP/IP to send packets

- Use hierarchy for decentralized control
- Build reliability (TCP) on top of unreliable layer (IP)
- Congestion control: Slow down when you see problems