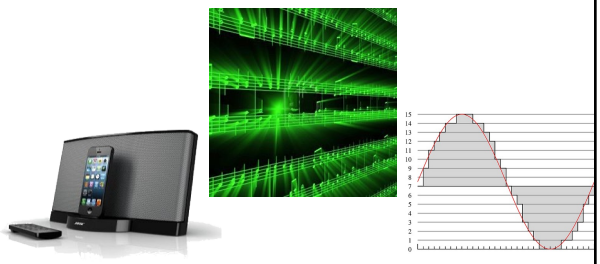
Penn Engineering

ESE



Lecture #19 – Networking

ESE 150 – Digital Audio Basics

Based on slides © 2009--2022 DeHon
Additional Material © 2014 Farmer

1

1

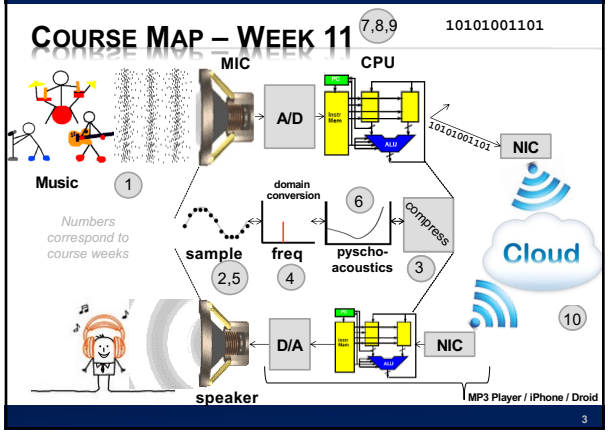
Lecture Topics

- **Where are we on course map?**
- **Networks**
  - Communicating Between Machines
  - Bandwidth Requirements
  - Technology Costs
  - Network Layering (Part 2)
    - Transport
    - Network (get started)
    - More…on Monday

2

2

Course Map – Week 11



3

3

What We'll Cover Today…

- **Established can**
  - represent things (sound, computations, images, movies, 3D objects…) as bits
  - Store and reconstruct from bits
- **If we can send bits between machines…**
  - Communicate (from MP3 player to Cell Phone)
  - Transport (from scanner and 3D printer to a transporter?)

4

4

Communicating between Machines

Fundamentals of Networks

5

5

Networked Systems

- **Today**
  - We expect our computers to be networked
    - Google, wikipedia, Email, IM, …
  - Can work stand alone
    - Airplane mode?
  - But, are crippled when not connected
  - Phone isn't a phone unless its networked

6

6

Minimal Setup

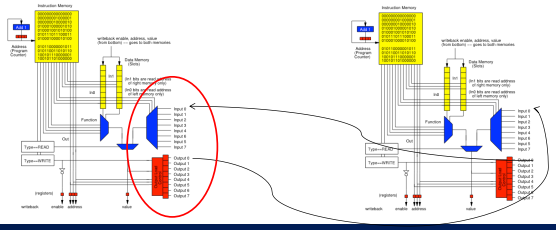
- **Have two computers**
  - think raw processors for the moment
- **Want them to communicate**
  - Send an mp3 file from A to B

7

7

Physical Connection

- **Place an I/O datapath in each computer**
- **String wire between computer's IO ports**
  - E.g. one wire from A→B, another B→A

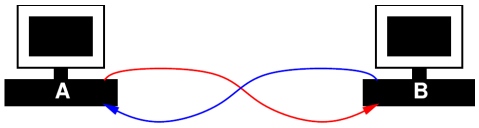


8

8

Physical Connection

- **Place an I/O datapath in each computer**
- **String wire between computer's IO peripheral**
  - E.g. one wire from A→B, another B→A

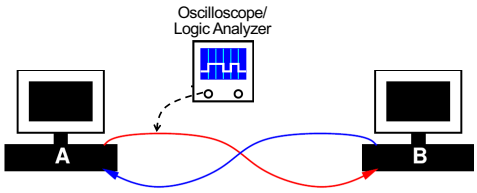


9

9

Signaling

- **Communicate with Voltage pulses**
  - A pulls line low (0)
  - B senses low (0) line
- **Data encoded as series of pulses/voltages on line**

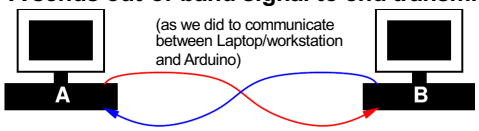


10

10

Communication Basic Steps

1. **Start program on B to receive data (file)**
2. **Start program on A to send data (file)**
3. **B waits for valid symbols**
4. **A sends data**
5. **B receives**
6. **A sends out-of-band signal to end transmission**



11

11

Preclass 1

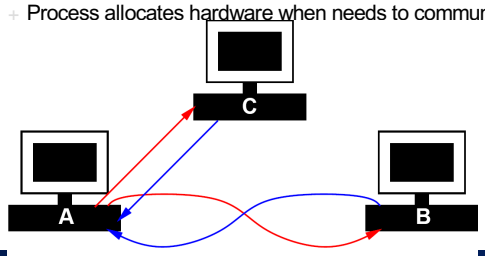
- **How many computers does your laptop communicate with?**
  - E-mail
  - Weather
  - Canvas, Piazza
  - Source code repositories (svn, git, …)
  - eniac
  - Web servers
    - Seas, news, facebook, youtube, wikipedia, google, ….
  - Spotify, iTunes, Windows Update

12

12

Multiple Tasks – *Multiple Wires?*

- **Back to wired connections**
- **E.g. download song and browse**
  - Could have a separate interface/wire for each application
  - Process allocates hardware when needs to communicate

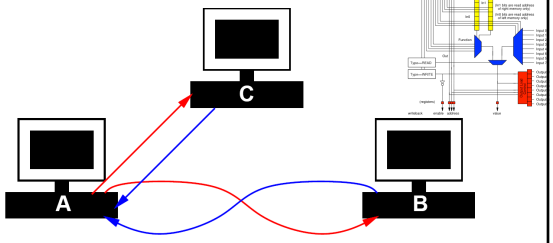


13

13

Connect to Multiple Machines

- **Add interface/wire for every machine want to talk to**
  - Talk to machine through its dedicated wire



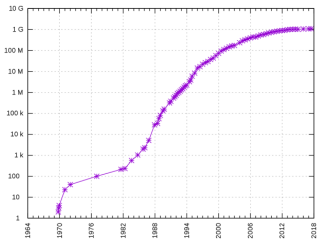
14

14

Scalability

[Source: Kopiesperre CC Share-alike 3.0 https://wikivisually.com/wiki/File:Internet_Hosts_Count_log.svg]

- **Do we like where this is going?**
- **Hosts on Internet**



- **How many things are connected to Internet?**
  - Estimate 30--50 Billion connected devices!
  - And growing …https://techjury.net/blog/how-many-iot-devices-are-there

15

15

How many connections?

- **Conclusion:** Single wire (or radio) per host or application we want to communicate with is not going to scale
  - need to look scalability of a network solution
- Do we have capacity to share wires?

16

16

Bandwidth Requirements and Costs

17

17

Wires

- **How fast can I send data over a wire?**
- **Consider a Category-5 Ethernet cable**
  - Bandwidth (bits/s)
    - 1Gbit/s – 1000Base-T (Gigabit ethernet)
  - Latency or transit time (distance/time)
    - 0.64 c [c=speed of light = 3×10^8 m/s]
    - 0.192 m/ns or roughly 5ns/m

[image: http://en.wikipedia.org/wiki/File:Cat_5.jpg]

18

18

Comparison: Audio (Preclass 3)

- **Real-Time stereo (2-channel) MP3**
  - 128Kbits/s
  - How many can share 1Gbit/s link?
- How long to download 3 minute song at full rate?
- How long for first bit to travel across 4000km wire at 0.6 × speed-of-light?

19

19

Comparison: Video (Preclass 3)

- **HDTV compressed**
  - Around 36Mbits/s
  - How many can share 1 Gbit/s link?

20

20

Costs (Preclass 4)

- **Cat 5e per foot ~ $0.20/foot**
  - Say $0.60/m
  - Raw wire
    - Ignoring handling to run
    - Ignoring rent/lease/buy land to run
  - Philly → San Francisco: ~4,000km
  - Wire cost?

21

21

Implications?

- **Today's wire bandwidth exceeds the throughput needs of any real-time single-stream data**
  - Can afford to share the wire
- **Wires are not cheap**
  - Cannot afford not to share the wire

22

22

Simulation 0

- **Do data stream (pipeline) simulation as warmup**
- **Think bucket brigade**
  - Everyone picking up from one side and handing to next
  - One item (packet) at a time
  - All working concurrently
  - Multiple things working way through the pipeline/brigade at a time

Graphic: Christoph Roser https://www.allaboutlean.com/bucket-brigade-1/bucket-brigade-2

23

23

Penn Engineering ESE

Part 2

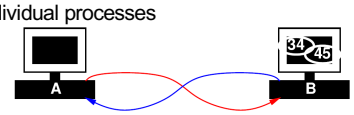
Sharing (Virtualizing) Connections

24

24

Sharing Link

- **Idea: Tag data with target**
  - "this is for process 34"
  - "this is for process 45"
- **Have transport layer deal with…**
  - Mixing data from separate streams
  - Separating data out into individual streams
  - Delivering to individual processes

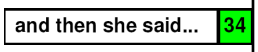


34: and then she said...
45: 80004010 00001200

25

25

Packet

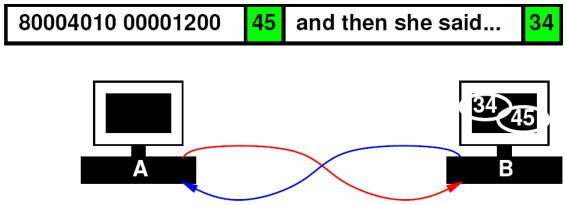


- **Begin to form a packet**
  - Header: says where to go
  - Payload: the data to send
- **Header:**
  - Added, consumed by network handling in routing
- **Payload:**
  - Only thing seen by the application processes

26

26

Packets



27

27

Transport Layer

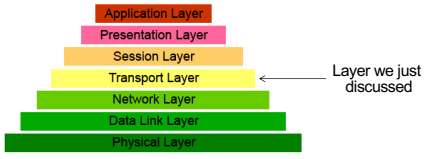
- **Call this the "Transport" Layer**
  - responsible for delivering data to the individual application process on the computer

28

28

OSI Model of a Network

The Seven Layers of OSI



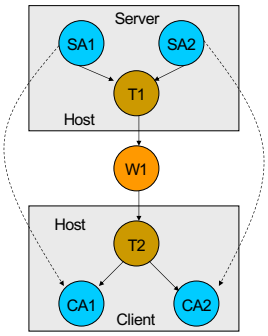
- **OSI – Open Systems Interconnection Reference Model**
  - Developed in 1980's; maintained by ISO
  - Abstract different functions of a network into layers
    - Each layer only knows about layer above and below (at the interface level)
  - Think of it like this: your "Application" doesn't know if its on a wired or wireless network (*physical layer*)…but it knows it needs a network!

29

29

Simulation 1

- **Send 4 verses or digits from each**
  - from song-server-app, even-server-app
  - to song-listener-app, even-consumer
- All go through one wire W1
- T1 – Transport tagging
- T2 – Transport sorting



32

32

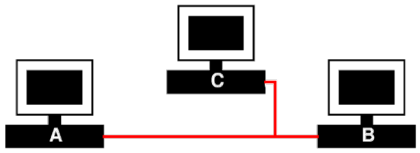
Virtualize Physical Wires

33

33

Start Simple

- **Add more computers to same pair of wires**



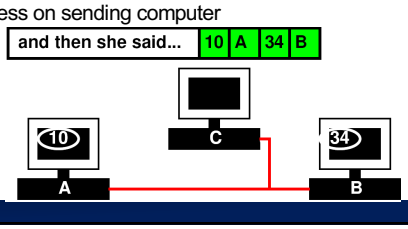
- All computers on wire see all the data on the wire
  - *How do computers know who the message is for?*

34

34

Extended Packet

- **Extend our packet header:**
  - Destination computer
  - Process on destination computer
  - Sending computer
  - Process on sending computer

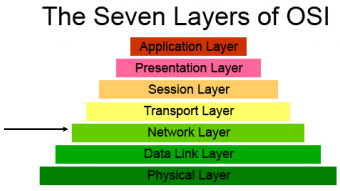


35

35

Network Layer

- **responsible for end-to-end (source to destination) packet delivery**



36

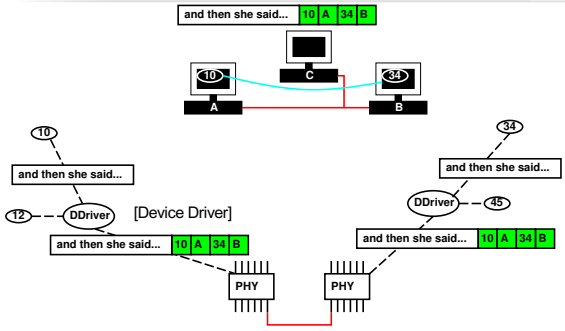
36

Virtualization Effect

- **Each pair of processes on different computers**
  - Has the view of a point-to-point connection
  - Each process, thinks it "owns the network" and has a dedicated connection to the other node

37

37



38

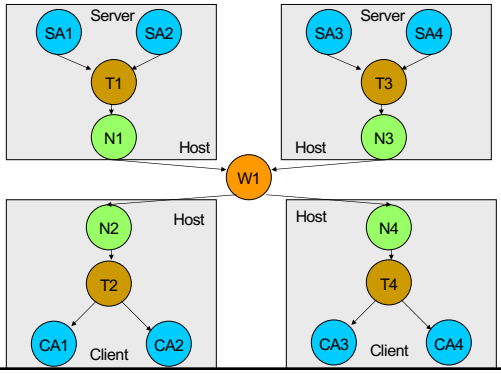
38

Simulation 2

- **Send 4 verses or digits from each**
  - from song server serving 2 verses
  - And digit-server serving 2 fundamental constants (or number sequence)
  - To two physical clients

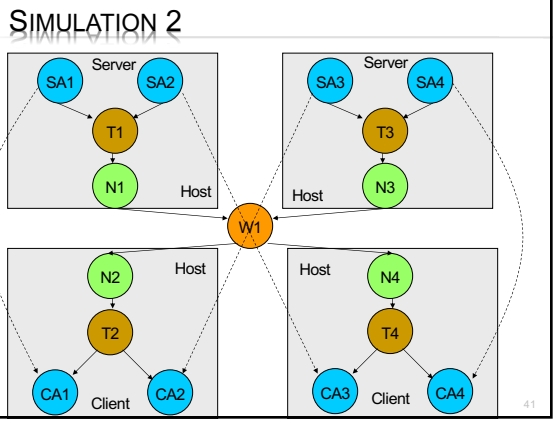
39

Simulation 2



40

Simulation 2



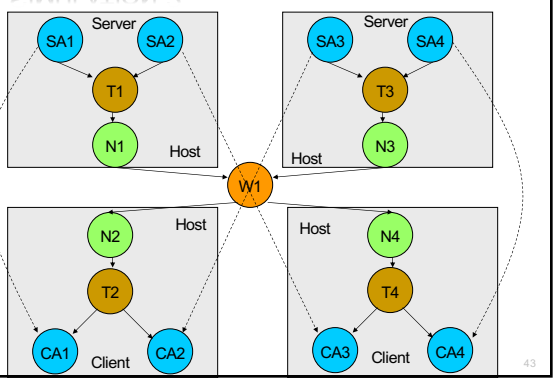
41

Simulation 2

- **N1, N3**
  - Add network-layer source/destination packet headers
- **W1 – Wire**
  - Duplicate packets to both destinations
  - Simulate shared wire
- **N2, N4**
  - Look at network-layer source/destination header
  - Discard packets not destined for this computer

42

Simulation 2



43

More to Come (Next Time)

- **Routing** – machines not directly connected
- **Routing Delays**
- **Data Ordering**
- **(Un)Reliability**
  - Data corruption
  - Packet Loss
  - Data Duplication
- **TCP/IP**

44

Big Ideas

- **Sharing – Network interface, wires**
  - Previously gates, processor, memory
- **Virtualization – datastream abstracts physical point-to-point link**
- **Layering**
  - Abstracts media and implementation
  - Decomposes functionality

45

45

Next Week In Lab

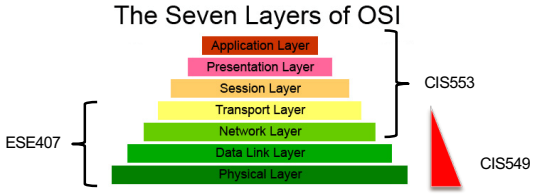
- **Lab 10:**
  - Look at naming, addressing, network diagnostics, …
  - Including a packet sniffer!
    - …see all the bits on the network you aren't supposed to see!
    - Get an appreciation for what is going on, on the lower network layers

46

46

Learn More @ Penn

- **Courses**
  - ESE407 – Intro Networks and Protocols
  - CIS553 – Networked Systems
  - CIS549 – Wireless Mobile Communications



47

47

Remember

- **Feedback**
- **Lab 9 tonight**
- **Lab 10 out**

48

48