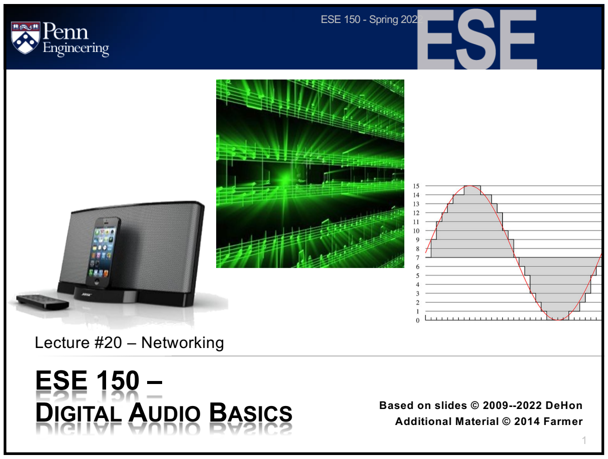
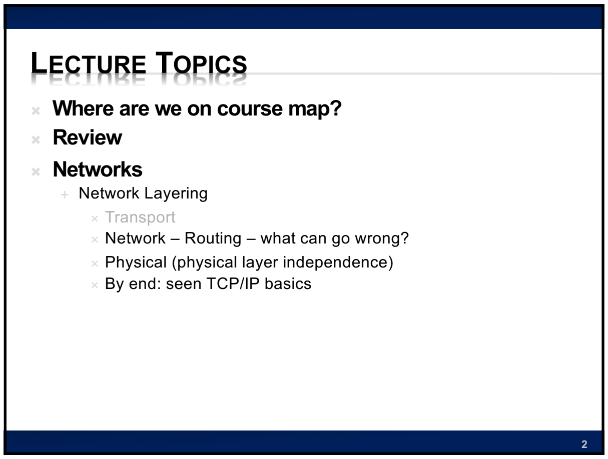


Lecture #20 – Networking

ESE 150 – Digital Audio Basics

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

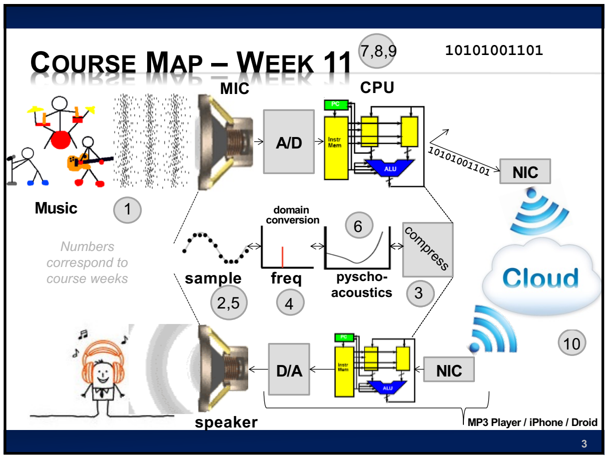
1



Lecture Topics

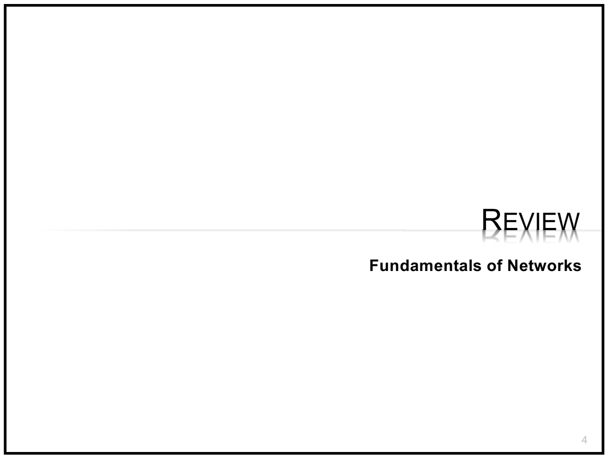
- Where are we on course map?
- Review
- Networks
  - Network Layering
    - Transport
    - Network – Routing – what can go wrong?
    - Physical (physical layer independence)
    - By end: seen TCP/IP basics

2



Course Map – Week 11

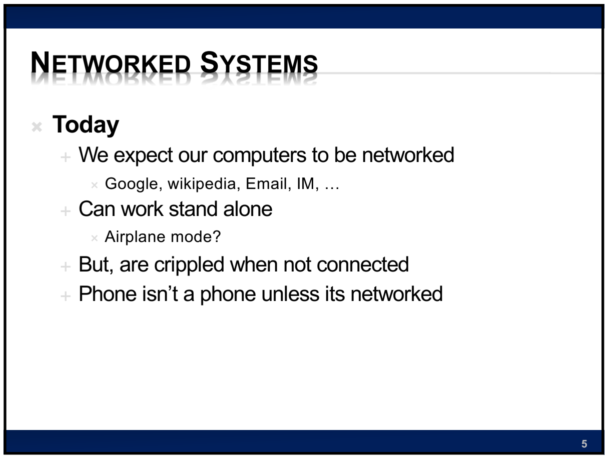
3



Review

Fundamentals of Networks

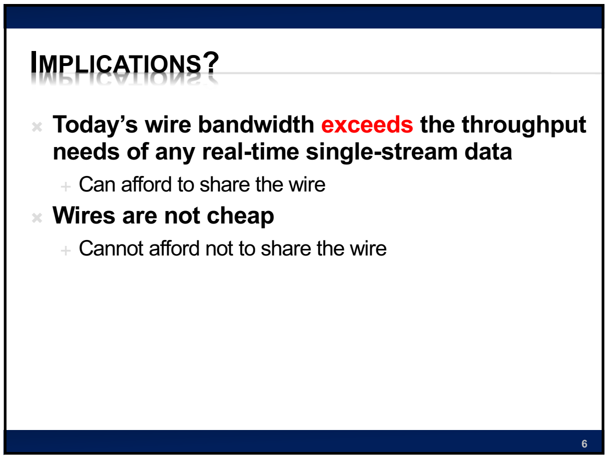
4



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

5



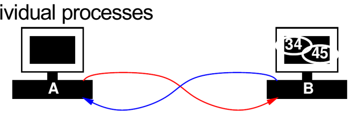
Implications?

- 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

6

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

7

Packet

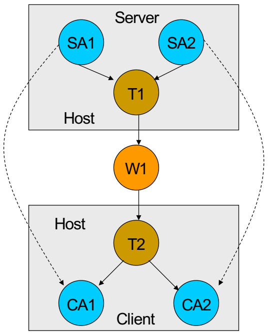
and then she said...	34

- **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

8

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

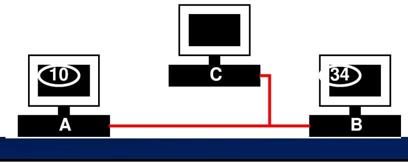


9

Multiple Computers -- Extended Packet

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

and then she said...	10	A	34	B

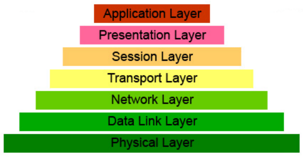


10

Network Layer

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

The Seven Layers of OSI

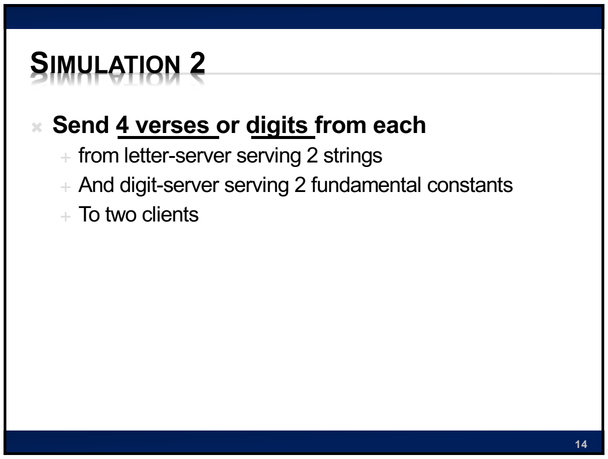


11

Abstractions for Sharing

- **Virtualized Processor**
  - Share single *processor* among multiple tasks
  - Make it look like process (program) has its own processor
- **Virtualized Communication between programs**
  - Share *wires* and processors
  - Make it look like a dedicated point-to-point link between processes (programs)

13

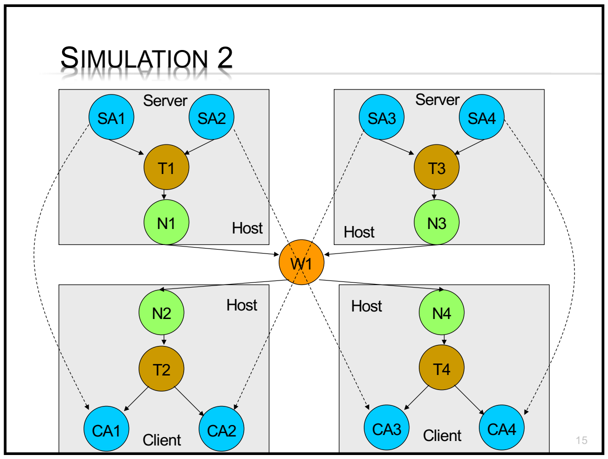


Simulation 2

- **Send 4 verses or digits from each**
  - from letter-server serving 2 strings
  - And digit-server serving 2 fundamental constants
  - To two clients

14

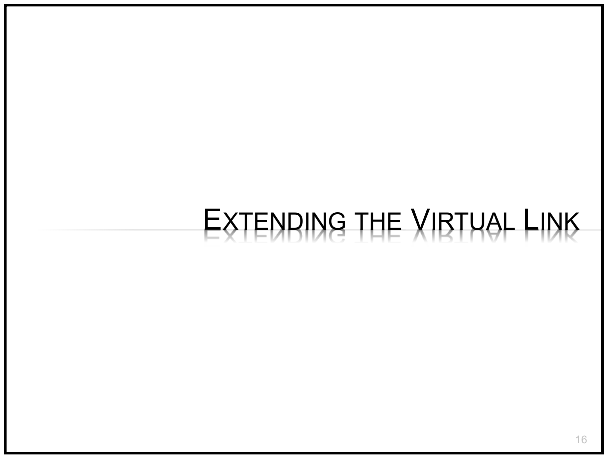
14



Simulation 2

15

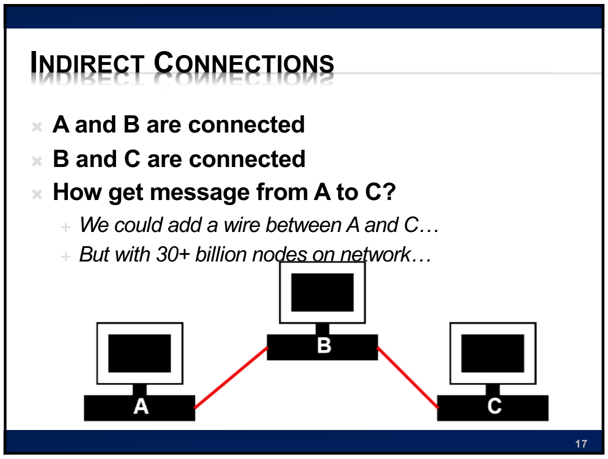
15



Extending the Virtual Link

16

16

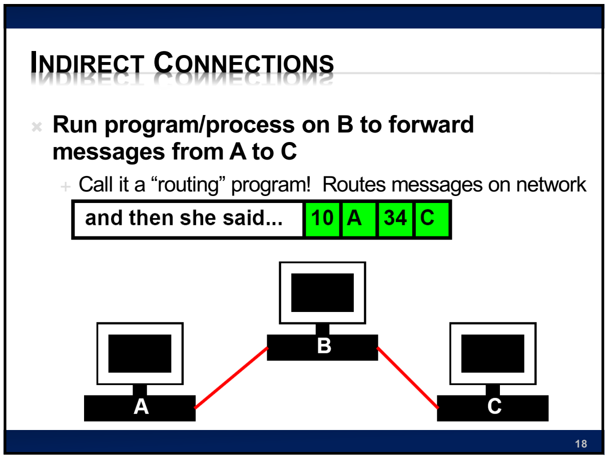


Indirect Connections

- **A and B are connected**
- **B and C are connected**
- **How get message from A to C?**
  - *We could add a wire between A and C…*
  - *But with 30+ billion nodes on network…*

17

17



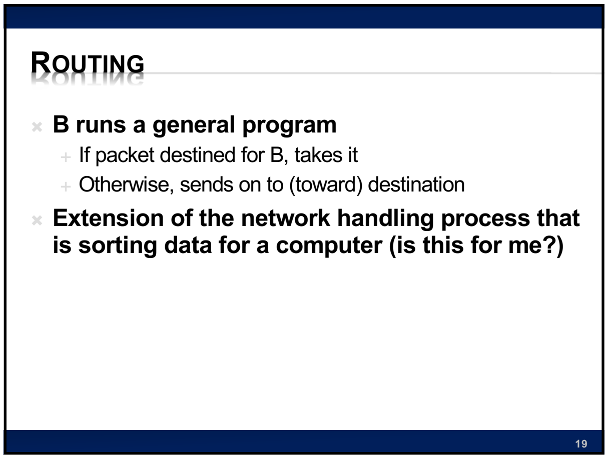
Indirect Connections

- **Run program/process on B to forward messages from A to C**
  - Call it a "routing" program! Routes messages on network

and then she said…	10	A	34	C

18

18

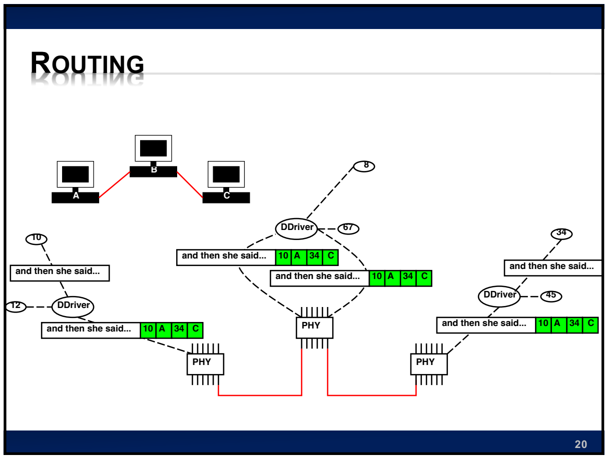


Routing

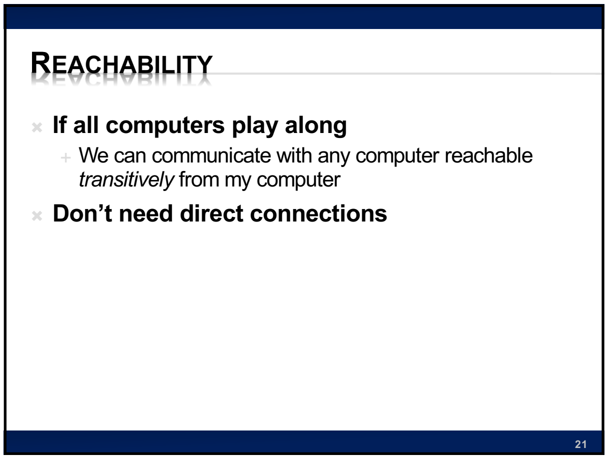
- **B runs a general program**
  - If packet destined for B, takes it
  - Otherwise, sends on to (toward) destination
- **Extension of the network handling process that is sorting data for a computer (is this for me?)**

19

19

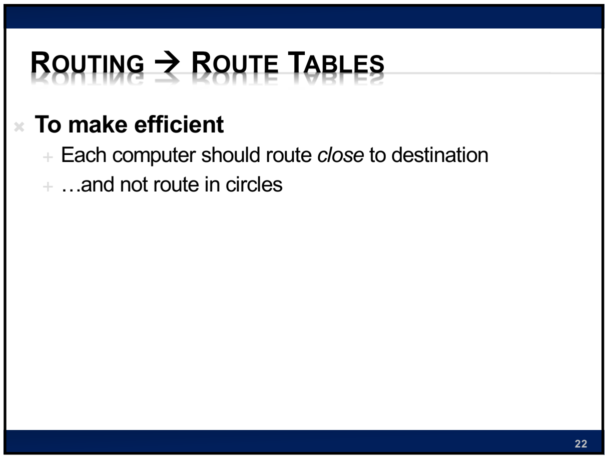


Routing



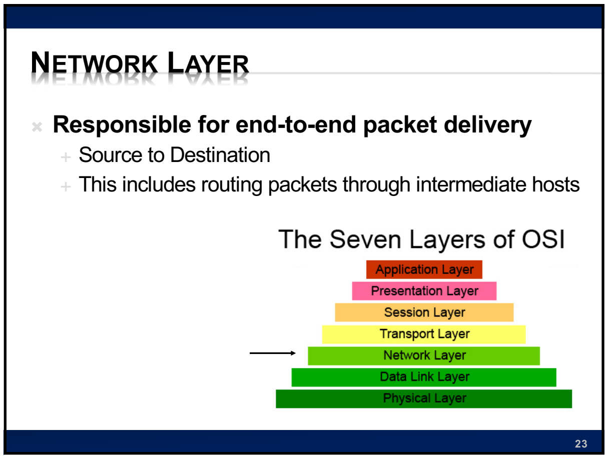
Reachability

- **If all computers play along**
  - We can communicate with any computer reachable *transitively* from my computer
- **Don't need direct connections**



Routing → Route Tables

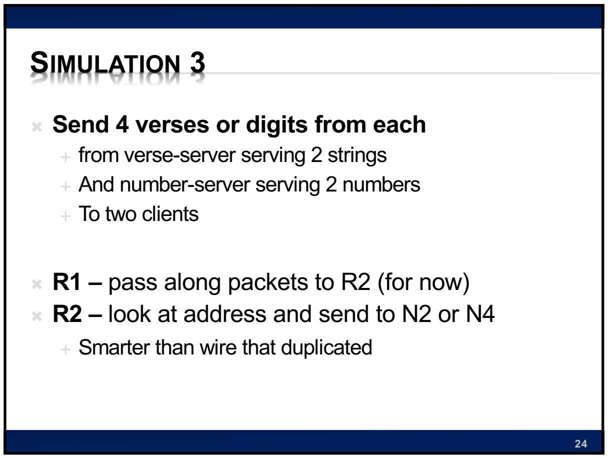
- **To make efficient**
  - Each computer should route *close* to destination
  - …and not route in circles



Network Layer

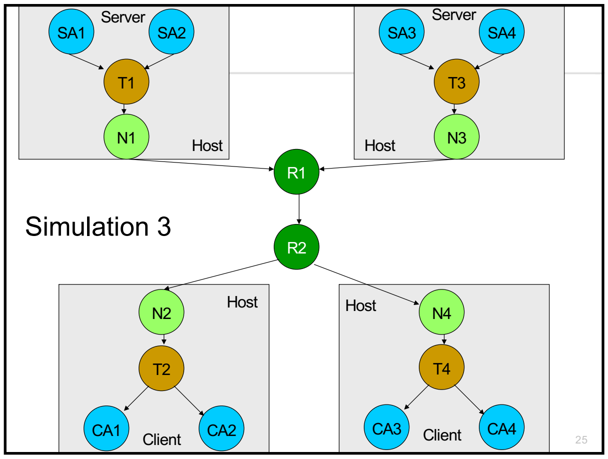
- **Responsible for end-to-end packet delivery**
  - Source to Destination
  - This includes routing packets through intermediate hosts

The Seven Layers of OSI

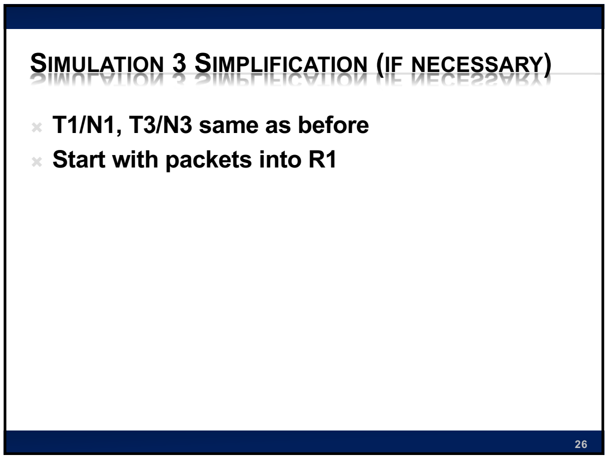


Simulation 3

- **Send 4 verses or digits from each**
  - from verse-server serving 2 strings
  - And number-server serving 2 numbers
  - To two clients
- **R1 –** pass along packets to R2 (for now)
- **R2 –** look at address and send to N2 or N4
  - Smarter than wire that duplicated



Simulation 3

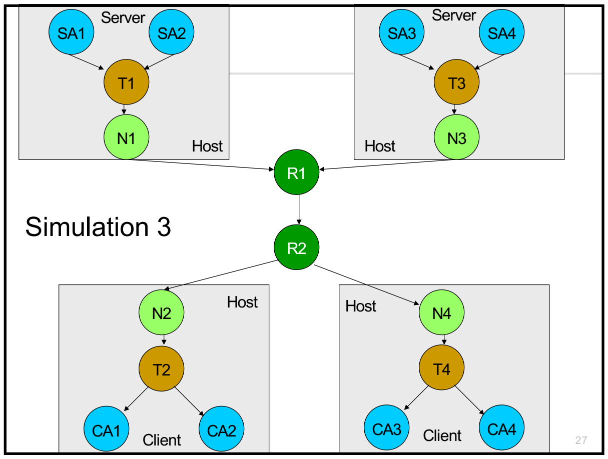


Simulation 3 Simplification (if necessary)

- T1/N1, T3/N3 same as before
- Start with packets into R1

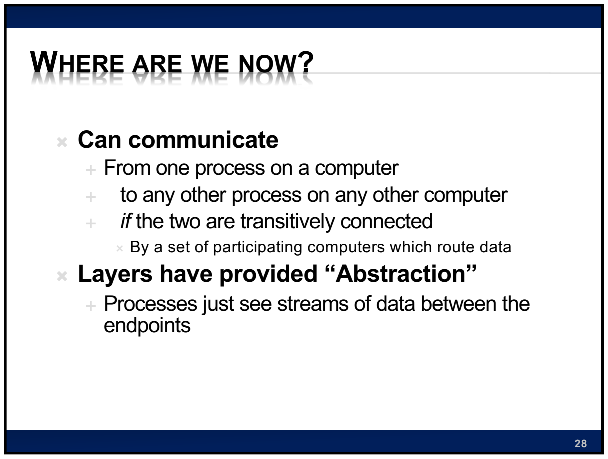
26

26



Simulation 3

27

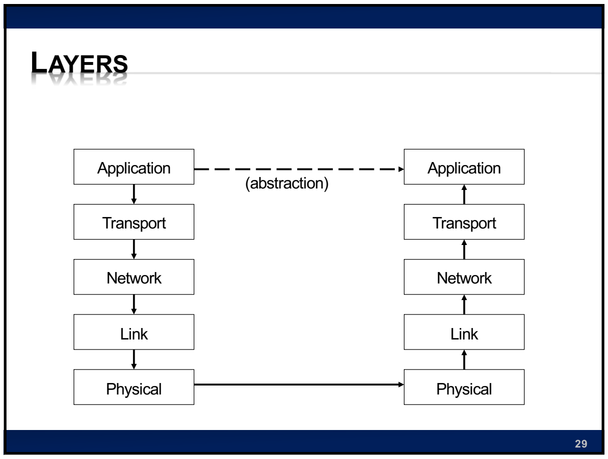


Where are we now?

- Can communicate
  - From one process on a computer
  - to any other process on any other computer
  - *if* the two are transitively connected
    - By a set of participating computers which route data
- Layers have provided "Abstraction"
  - Processes just see streams of data between the endpoints

28

28



Layers

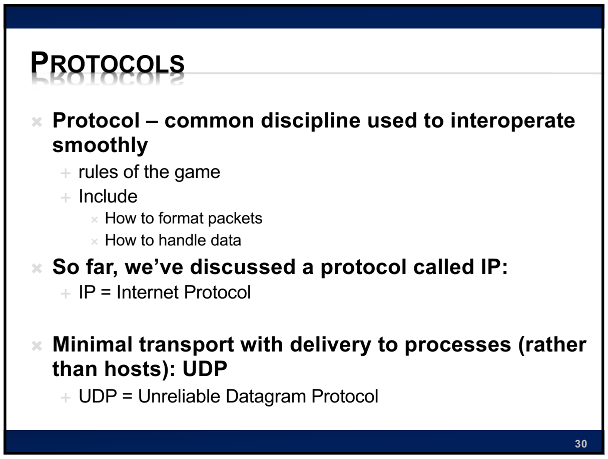
Application
Transport
Network
Link
Physical

(abstraction)

Application
Transport
Network
Link
Physical

29

29

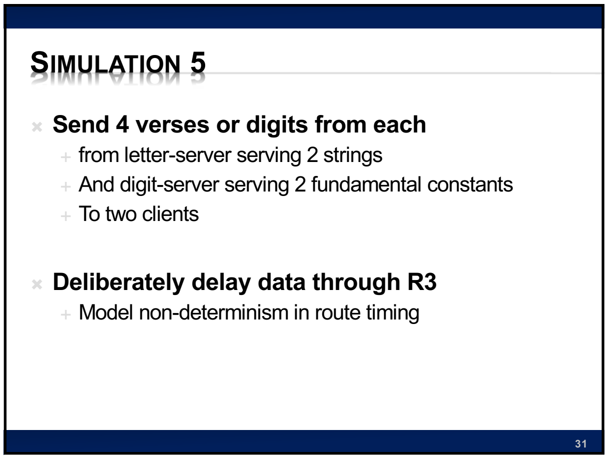


Protocols

- Protocol – common discipline used to interoperate smoothly
  - rules of the game
  - Include
    - How to format packets
    - How to handle data
- So far, we've discussed a protocol called IP:
  - IP = Internet Protocol
- Minimal transport with delivery to processes (rather than hosts): UDP
  - UDP = Unreliable Datagram Protocol

30

30

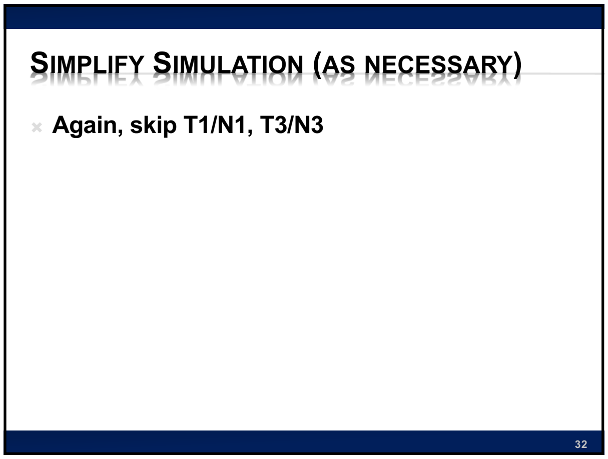


Simulation 5

- Send 4 verses or digits from each
  - from letter-server serving 2 strings
  - And digit-server serving 2 fundamental constants
  - To two clients
- Deliberately delay data through R3
  - Model non-determinism in route timing

31

31

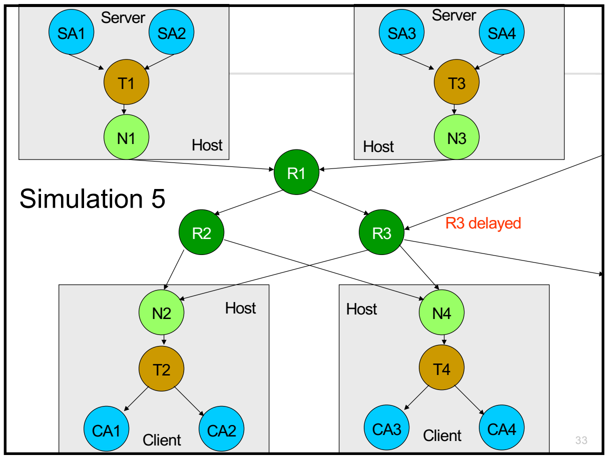


Simplify Simulation (as necessary)

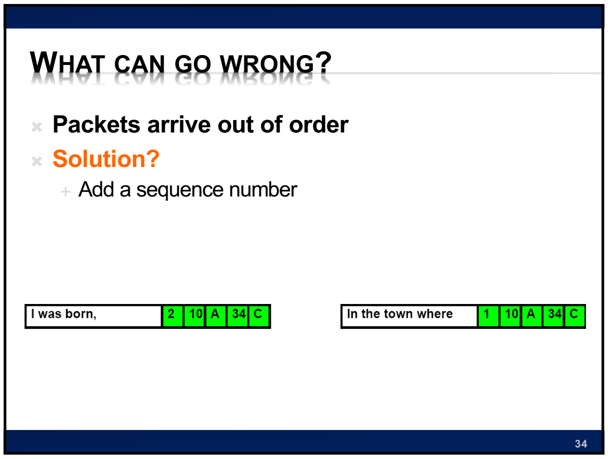
- Again, skip T1/N1, T3/N3

32

32



33



What can go wrong?

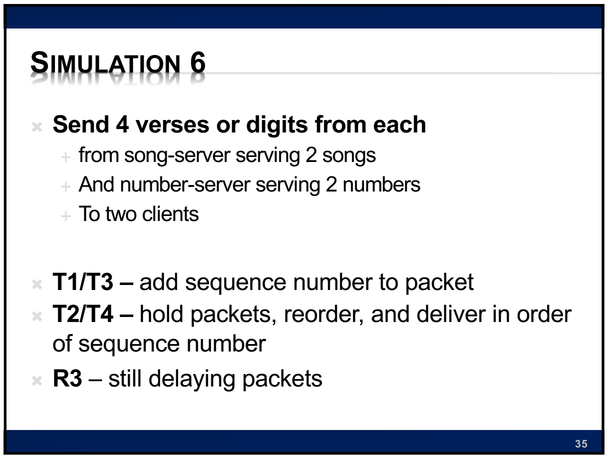
- Packets arrive out of order
- Solution?
  - Add a sequence number

I was born, | 2 | 10 | A | 34 | C

In the town where | 1 | 10 | A | 34 | C

34

34

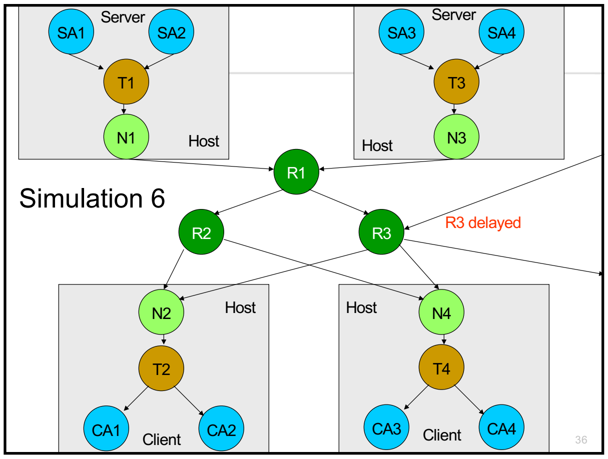


Simulation 6

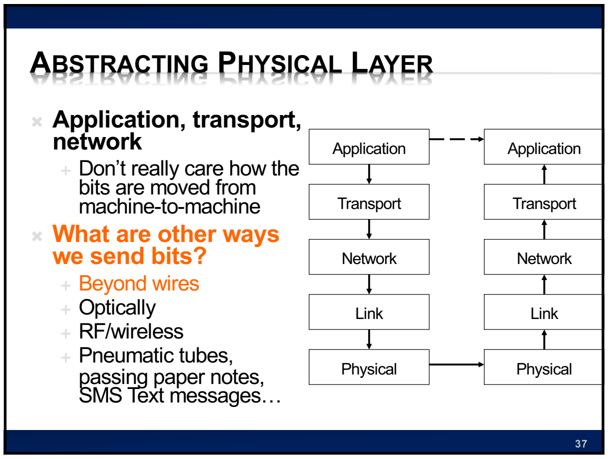
- Send 4 verses or digits from each
  - from song-server serving 2 songs
  - And number-server serving 2 numbers
  - To two clients
- **T1/T3 –** add sequence number to packet
- **T2/T4 –** hold packets, reorder, and deliver in order of sequence number
- **R3** – still delaying packets

35

35



36

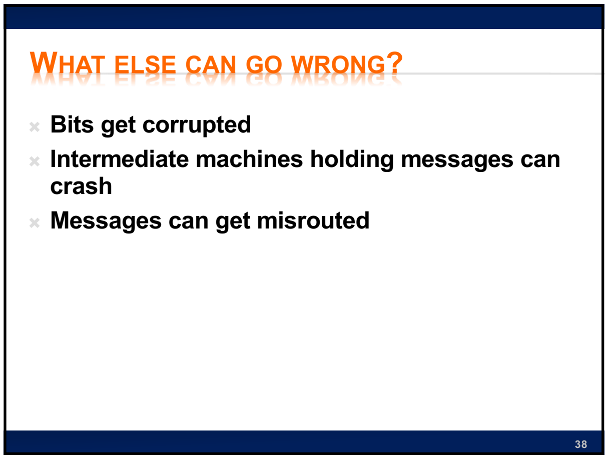


Abstracting Physical Layer

- Application, transport, network
  - Don't really care how the bits are moved from machine-to-machine
- What are other ways we send bits?
  - Beyond wires
  - Optically
  - RF/wireless
  - Pneumatic tubes, passing paper notes, SMS Text messages…

37

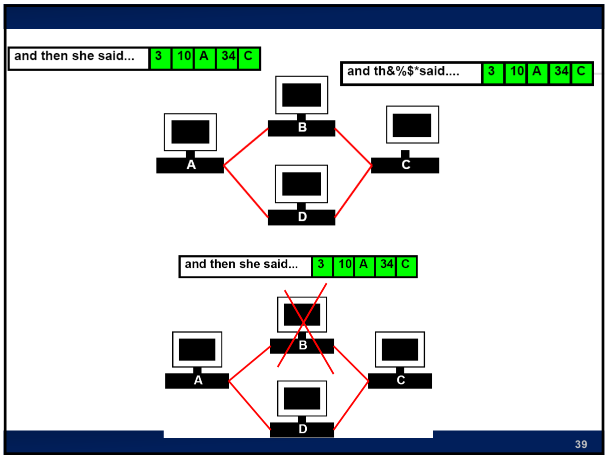
37



What else can go wrong?

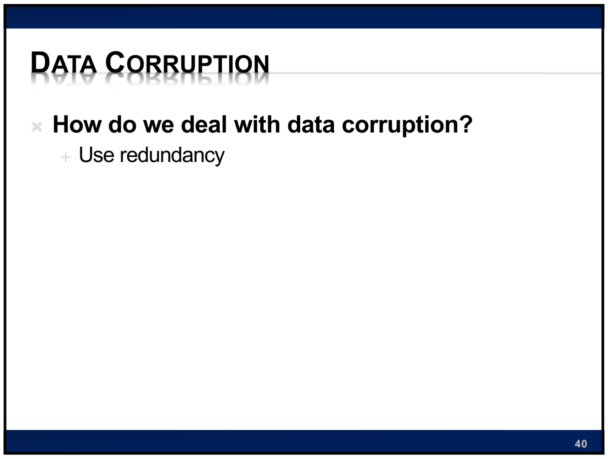
- Bits get corrupted
- Intermediate machines holding messages can crash
- Messages can get misrouted

38



and then she said... 3 10 A 34 C
and th&%$*said.... 3 10 A 34 C
A
B
C
D
and then she said... 3 10 A 34 C
A
B
C
D

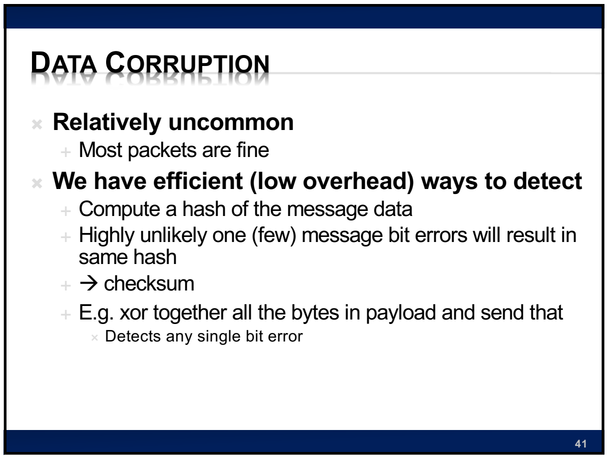
39



Data Corruption

- How do we deal with data corruption?
  - Use redundancy

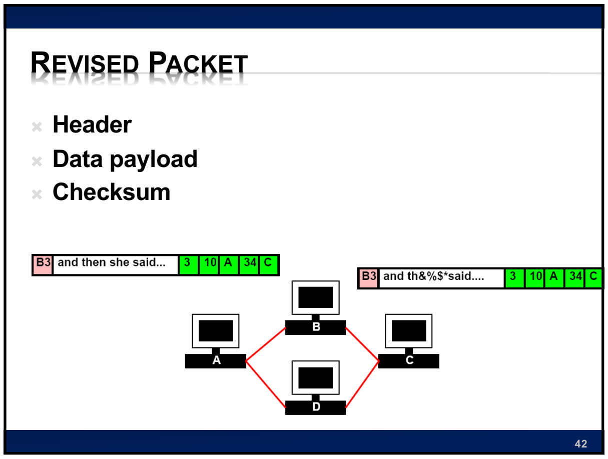
40



Data Corruption

- Relatively uncommon
  - Most packets are fine
- We have efficient (low overhead) ways to detect
  - Compute a hash of the message data
  - Highly unlikely one (few) message bit errors will result in same hash
  - → checksum
  - E.g. xor together all the bytes in payload and send that
    - Detects any single bit error

41

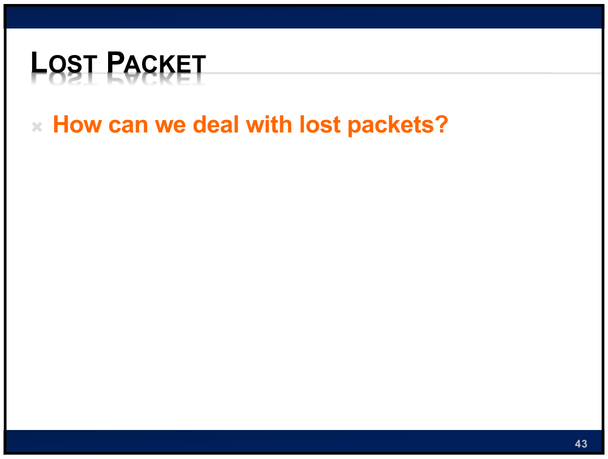


Revised Packet

- Header
- Data payload
- Checksum

B3 and then she said... 3 10 A 34 C
B3 and th&%$*said.... 3 10 A 34 C
A
B
C
D

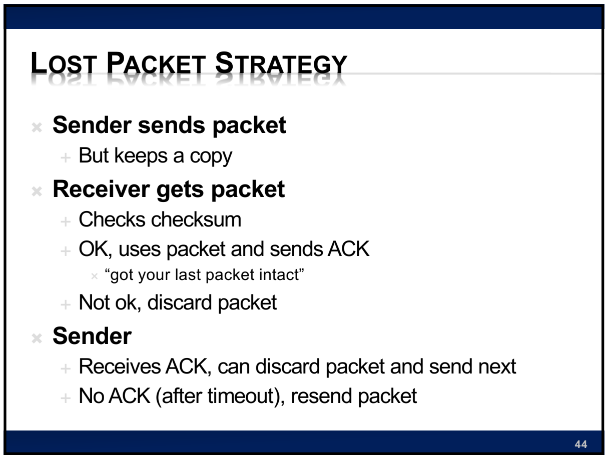
42



Lost Packet

- How can we deal with lost packets?

43

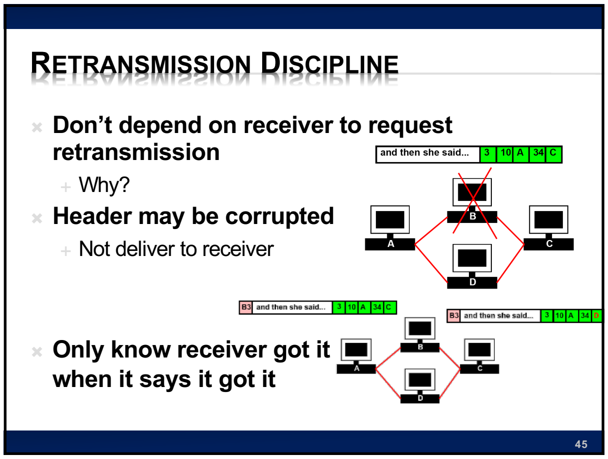


Lost Packet Strategy

- Sender sends packet
  - But keeps a copy
- Receiver gets packet
  - Checks checksum
  - OK, uses packet and sends ACK
    - "got your last packet intact"
  - Not ok, discard packet
- Sender
  - Receives ACK, can discard packet and send next
  - No ACK (after timeout), resend packet

44

44



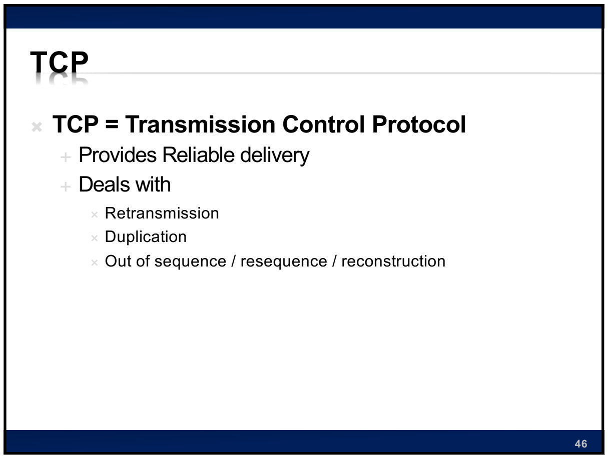
Retransmission Discipline

- Don't depend on receiver to request retransmission
  - Why?
- Header may be corrupted
  - Not deliver to receiver
- Only know receiver got it when it says it got it

and then she said… 3 10 A 34 C
A
B
C
D
B3 and then she said… 3 10 A 34 C
B3 and then she said… 3 10 A 34 D
A
B
C
D

45

45

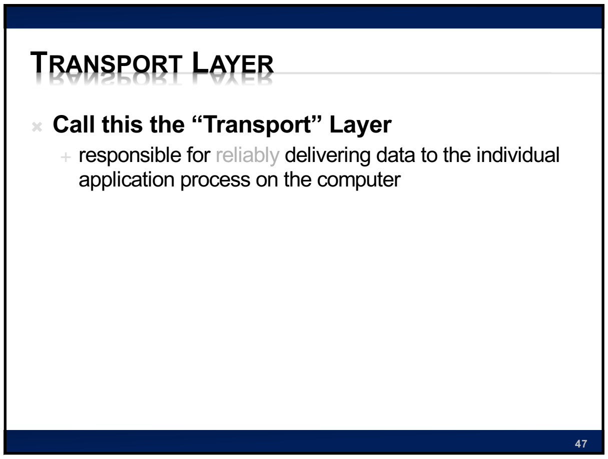


TCP

- TCP = Transmission Control Protocol
  - Provides Reliable delivery
  - Deals with
    - Retransmission
    - Duplication
    - Out of sequence / resequence / reconstruction

46

46

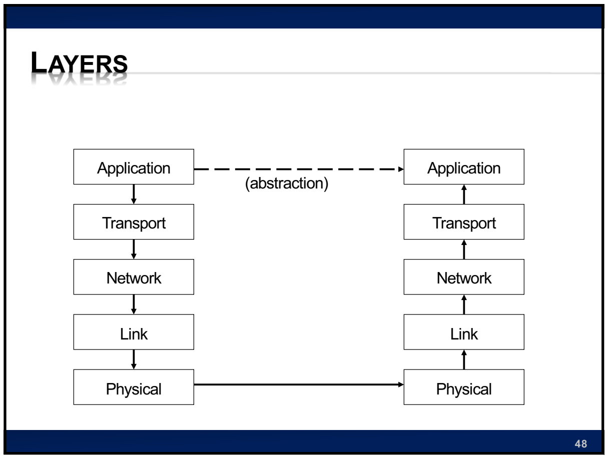


Transport Layer

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

47

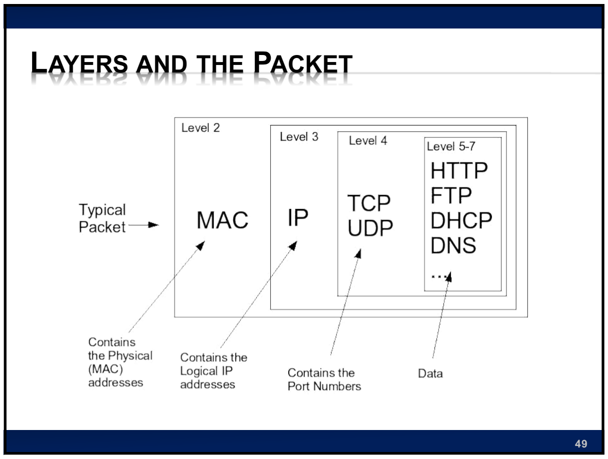
47



Layers

48

48

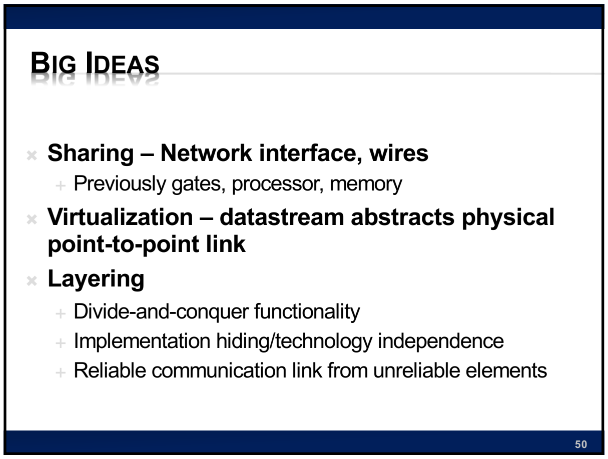


Layers and the Packet

Typical Packet
Level 2
Level 3
Level 4
Level 5-7
MAC
IP
TCP
UDP
HTTP
FTP
DHCP
DNS
…
Contains the Physical (MAC) addresses
Contains the Logical IP addresses
Contains the Port Numbers
Data

49

49

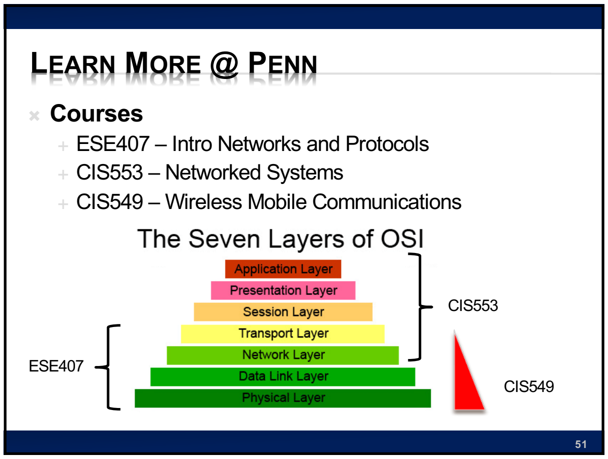


Big Ideas

- **Sharing – Network interface, wires**
  - Previously gates, processor, memory
- **Virtualization – datastream abstracts physical point-to-point link**
- **Layering**
  - Divide-and-conquer functionality
  - Implementation hiding/technology independence
  - Reliable communication link from unreliable elements

50

50



Learn More @ Penn

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

The Seven Layers of OSI

Application Layer
Presentation Layer
Session Layer
Transport Layer
Network Layer
Data Link Layer
Physical Layer

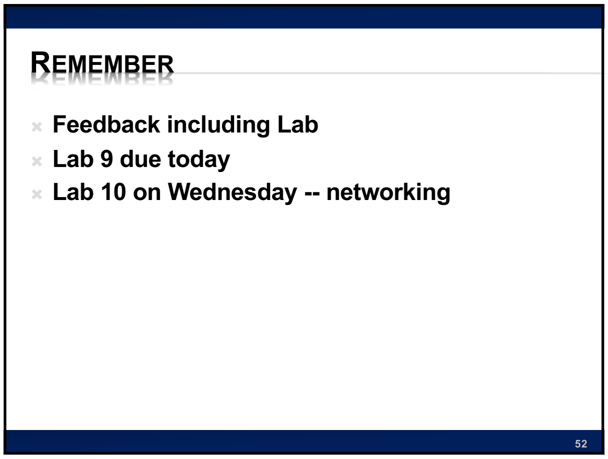
CIS553

ESE407

CIS549

51

51



Remember

- **Feedback including Lab**
- **Lab 9 due today**
- **Lab 10 on Wednesday -- networking**

52

52