15-441 Computer Networking

Lecture 1 – Introduction

Peter Steenkiste

Fall 2013

www.cs.cmu.edu/~prs/15-441-F13

1

Today's Lecture

- Administrivia
- Whirlwind tour of networking
- Protocol stacks

2

Teaching Staff

- Instructors.
  - Peter Steenkiste
  - Eric Andersen
- Teaching assistants.
  - Matt Mukerjee
  - David Naylor
  - Ben Wasserman
- staff-441@cs.cmu.edu → course staff
  - Please use this instead of emailing just one of us!
- Office hours and contact information on the web site.

3

Course Goals

- Become familiar with the principles and practice of data networking
  - Routing, transport protocols, naming, ...
  - Design of networks and services
- Learn how to write applications that use the network
  - A web server
  - Optimizing application performance
- Get some understanding about network internals in a hands-on way
  - TCP-style reliable congestion control

4

Course Format

- ~30 lectures
  - Cover the “principles and practice”
  - Complete readings before lecture
- 4 homework assignments
  - “Paper”: Do you understand and can you apply the material?
  - “Lab”: Illustrate networking concepts
  - Teach networking concepts/tools
  - Preparation for midterm and final
- 3 programming projects
  - How to use and build networks / networked applications
  - Application-layer programming; include key ideas from kernel
  - Larger, open-ended group projects. *Start early!*
- Midterm and final
  - Emphasis on understanding of course material

5

Recitation Sections

- Key 441 objective: system programming - C
- Different from what you’ve done before!
  - Often designed to run indefinitely. Handle all errors!
  - Must be secure
  - Interfaces specified by documented protocols
  - Concurrency involved (inter and intra-machine)
  - Must have good test methods
- Recitations address this
  - “A system hackers’ view of software engineering”
  - Help develop practical needed in projects (and beyond)
- Recitations are only one hours long!

6

Projects

- 3 projects: implementing application and network level functionality
  - First project: web server solo
  - Later projects: teams of two; hope to do something involving video
- Plan to use google group for communication
- Must use version control – git:
  - Creating andrew directories for everyone

7

Sounds Great! How Do I Get In?

- Currently about 80 people are enrolled – there is no waiting list.
  - If you do not plan to take the course, please drop it ASAP
  - While nobody is waiting for your slot, it helps in preparing projects, handouts, ...

8

Administrative Stuff

- Watch the course web page
  - http://www.cs.cmu.edu/~prs/15-441-F13/
  - Handouts, readings, ..
  - Always check here first
- Office hours posted on web page
  - Make an appointment if you have a conflict
- Course secretary
  - Kathy McNiff, Gates 9213
  - Pick up graded assignments, …

9

Grading

- Roughly equal weight in projects and testing
  - 45% for Projects I, II and III
  - 15% for Midterm exam
  - 25% for Final exam
  - 15% for Homework
- You **MUST** demonstrate competence in both projects and tests to pass the course
  - Fail either and you fail the class!

10

Policy on Collaboration

- Working together is important
  - Discuss course material in general terms
  - Work together on program debugging, ..
  - Final submission must be your own work
    - Homeworks, midterm, final
- Projects: Solo (P1) + Teams of two (P2,P3)
  - Collaboration, group project skills
  - Both students should understand the entire project
- Web page has details, e.g., university policy, etc.
- Things we don't want to have to say: We run projects through cheat-checkers against all previously and concurrently handed in versions…

11

Late Work and Regrading

- Late work will receive a 15% penalty/day
  - No assignment can be more than 2 days late
  - Only exceptions are documented illness and family emergencies
- Requests for regrading must be submitted in writing to course secretary within 2 weeks.
  - Do not contact us by e-mail
  - Office hours are fine for discussion but not for regrading
  - Regrading will be done by original grader
- No assignments with a "short fuse"
  - Homeworks: ~1-2 weeks - Projects: ~4 weeks
  - Start on time!
  - Every year some students discover that a 4 week project cannot be completed in a week

12

The Slides

- The slides are a resource that is shared by the many instructors of 15-441
  - Also some sharing with 18-345
- They include contributions from Peter Steenkiste, Srini Seshan, Dave Andersen, Hui Zhang, Eric Anderson, and others

13

Today's Lecture

- Administrivia
- Whirlwind tour of networking
- Protocol stacks

14

This Week

- Intro – what is the course all about?
- Protocol stacks and layering
- Recitations start this week: Git version control
- Next recitation: review of sockets
  - 213 review, but not quite
- On to the good stuff…Whirlwind tour of networking
  - Course outline:
    - Low-level (physical, link, circuits, etc.)
    - Internet core concepts (addressing, routing, DNS)
    - Advanced topics

15

What is the Objective of Networking?

- Enable communication between applications on different computers
  - Web, peer-to-peer, video streaming, distributed processing, video and audio conferencing, …
- Over very diverse infrastructures
  - The "Internet", WiFi and cellular, data center networks, corporate networks, dedicated private networks, …
- Must understand application needs/demands (Lecture 3)
  - Traffic data rate
  - Traffic pattern (bursty or constant bit rate)
  - Traffic target (multipoint or single destination, mobile or fixed)
  - Delay sensitivity
  - Loss sensitivity

16

What Is a Network?

- Collection of nodes and links that connect them
- This is vague. Why? Consider different networks:
  - Internet
  - Telephone
  - TV delivery
  - Your house
  - Others – sensor nets, cellular, WiFi, …
- The course focuses on the Internet, but the Internet is very diverse, and ..
- Almost everything runs over the Internet!

17

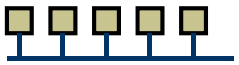
Basic Building Block: Links

- Electrical questions
  - Voltage, frequency, …
  - Or maybe it is optical or wireless?
  - Touched on very lightly in this course: build some intuition
- Link-layer issues: How to send data over a link?
  - When to talk – can either side talk at once?
  - What to say – low-level format of the bits and packets?
- Okay… what about more nodes?

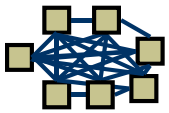
19

Basic Building Block: Links

- … But what if we want more hosts?



One wire



Wires for everybody!

- Scalability?!
  - 10^x nodes, for x: $1 \rightarrow \infty$

20

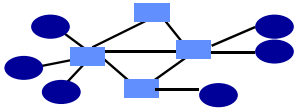
Local Area Networks (LANs)

- Benefits of being "local":
  - Lower cost
    - Efficiency (usually) less pressing
  - Short distance = low latency
  - One management domain
- Today's technology: Ethernet++
- But not all Ethernets are the same
  - Today's ethernet versus traditional ethernet
  - Wired versus wireless
  - Campus network versus data center
- And then there is cellular …

21

Multiplexing

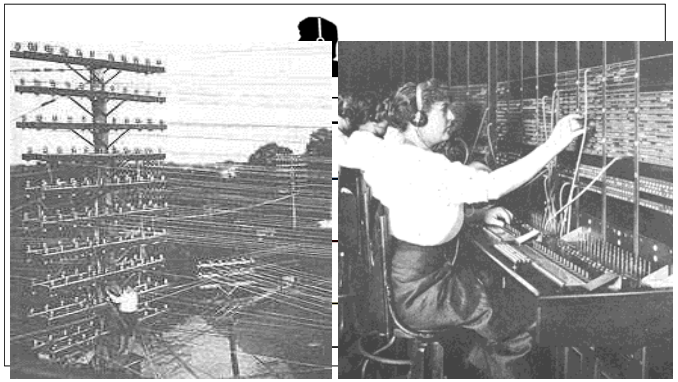
- Need to share network resources



- How? Switched network
  - Party “A” gets resources sometimes
  - Party “B” gets them sometimes
- Interior nodes act as “Switches”
- What mechanisms to share resources?

22

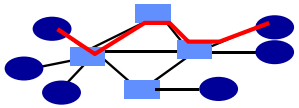
Back in the Old Days…



23

Circuit Switching

- Source first establishes a connection (circuit) to the destination
  - Each switch along the way stores info about connection (and possibly allocates resources)
- Source sends the data over the circuit
  - No need to include the destination address with the data since the switches know the path
- The connection is explicitly torn down
- Example: telephone network (analog)



24

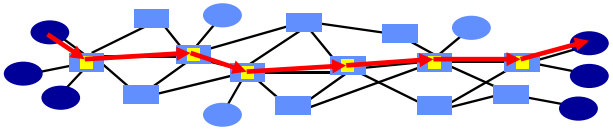
Circuit Switching Discussion

- Circuits have some very attractive properties.
  - Fast and simple data transfer, once the circuit has been established
  - Predictable performance since the circuit provides isolation from other users
  - E.g. guaranteed bandwidth
- But it also has some shortcomings.
  - How about bursty traffic
    - Do you need a permanent circuit to Facebook?
    - Circuit will be idle for significant periods of time
  - How about users with different bandwidth needs
    - Do they have to use multiple circuits?
- Alternative: packet switching.

25

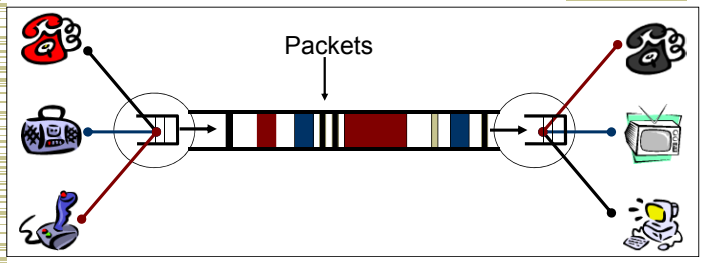
Packet Switching (our emphasis)

- Source sends information as self-contained packets that have an address.
  - Source may have to break up single message in multiple
- Each packet travels independently to the destination host.
  - Switches use the address in the packet to determine how to forward the packets
  - Store and forward
- Analogy: a letter in surface mail.



26

Packet Switching – Statistical Multiplexing



- Switches arbitrate between inputs
- Can send from *any* input that's ready
  - Links are never idle when there is traffic to send
  - (Efficiency!)

27

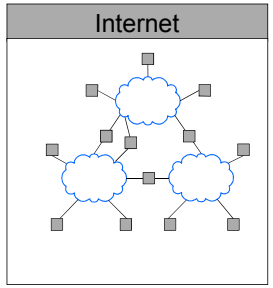
Networks Juggle Many Goals

- Efficiency – resource use; cost
- The "ilities":
  - Evolvability
  - Managability
  - Security (securability, if you must)
  - Ease of:
    - Deployment
    - Management
    - Creating useful applications
  - Scalability

29

Internet

- An inter-net: a network of networks.
  - Networks are connected using routers that support communication in a hierarchical fashion
  - Often need other special devices at the boundaries for security, accounting, ..
- The Internet: the interconnected set of networks of the Internet Service Providers (ISPs)
  - About 20,000 ISPs make up the Internet
  - Many more "edge" networks



31

Challenges of the Internet

- Heterogeneity
  - Address formats
  - Performance – bandwidth/latency
  - Packet size
  - Loss rate/pattern/handling
  - Routing
  - Diverse network technologies → satellite links, cellular links, carrier pigeons

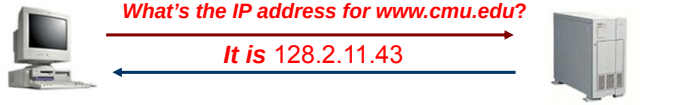
32

Challenges of the Internet

- Scale
  - 100,000,000s of hosts
  - 18,000+ administrative domains,
  - Thousands of applications
- Adversarial environment
- How do we get all these diverse networks to work together?
  - In order to inter-operate, all participating networks have to follow a common set of rules

33

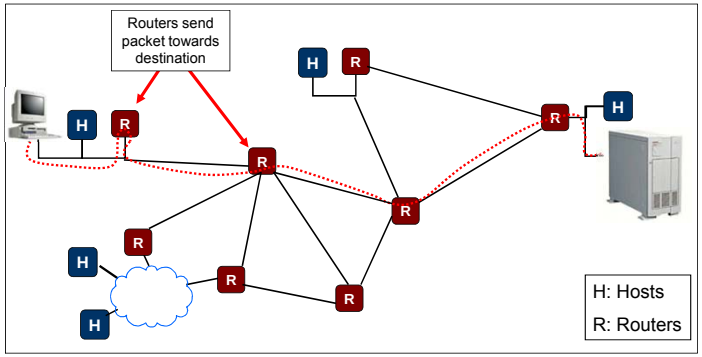
Naming



Translates human readable names to logical endpoints

36

Routing



37

Network Service Model

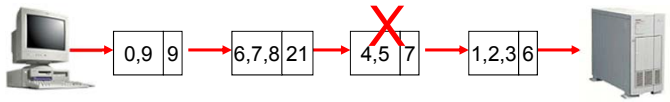
- What is the *service model?*
  - Internet: *best-effort* – packets can get lost, etc.
- What if you want more?
  - Performance guarantees (QoS)
  - Reliability
    - Corruption
    - Lost packets
  - Flow and congestion control
  - Fragmentation
  - In-order delivery
  - Etc…

38

What if the Data gets Corrupted?

Problem: Data Corruption

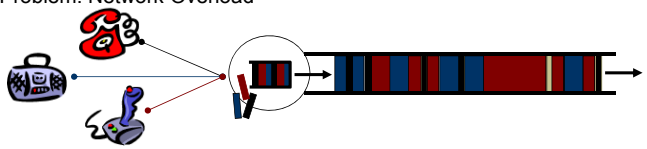
Solution: Add a *checksum*



39

What if Network is Overloaded?

Problem: Network Overload



Solution: Buffering and Congestion Control

- Short bursts: buffer
- What if buffer overflows?
  - Packets dropped
  - Sender adjusts rate until load = resources → "congestion control"

40

What if the Data gets Lost?

Problem: Lost Data

Solution: Timeout and Retransmit

41

Course Content

- Intro, applications, protocol stacks, …
- Key network components: physical layers, datalink, IP, transport protocols
- Miscellaneous topics: security, wireless and mobility, peer-peer, the Web, video, some research, etc.
- Changes relative to previous years:
  - Reduce details in the key network components
  - Add material on actual apps and deployments

Networks [including end points] Implement Many Functions

- Link
- Multiplexing
- Routing
- Addressing/naming (locating peers)
- Reliability
- Flow control
- Etc.…

How do we implement all this? (without making a mess)

Meeting Application Demands

- Sometimes interior of the network can do it
  - E.g., Quality of Service
    - Benefits of circuit switching in packet-switched net
    - Hard in the Internet, easy in restricted contexts
- OR hosts can do it
  - E.g., end-to-end *Transport protocols*
    - TCP performs end-to-end retransmission of lost packets to give the illusion of a reliable underlying network.

Next Lecture

- How to determine split of functionality
  - Across protocol layers (more in a few minutes)
  - Across network nodes
- Read two papers on the motivations for the Internet architecture (web site):
  - "The design philosophy of the DARPA Internet Protocols", Dave Clark, SIGCOMM 88
  - "End-to-end arguments in system design", Saltzer, Reed, and Clark, ACM Transactions on Computer Systems, November 1984

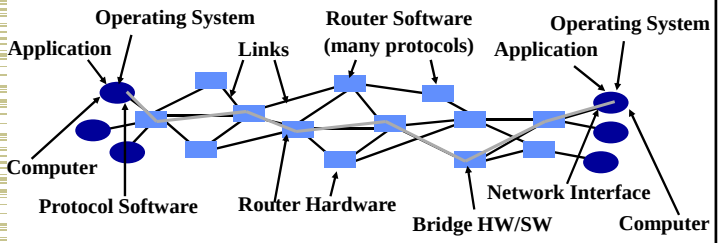
Today's Lecture

- Administrivia
- Whirlwind tour of networking
- Protocol stacks

48

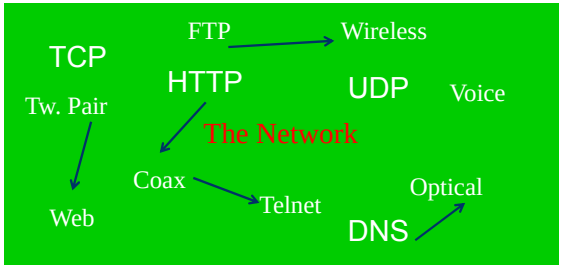
How to Design a Network?

- Has many users
- Offers diverse services
- Mixes very diverse technologies
- Components built by many companies
- Diverse ownership
- Can evolve over time



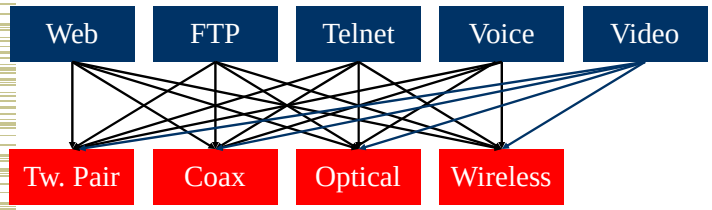
49

Solution #1



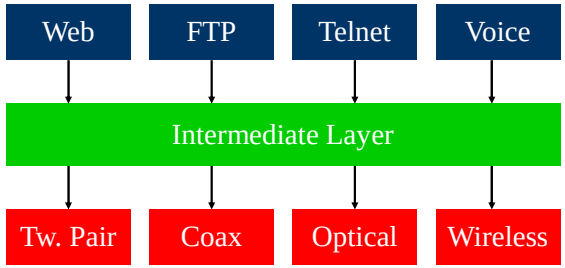
50

Solution #2?



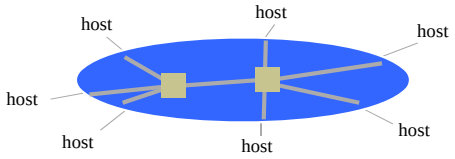
51

Solution #3



52

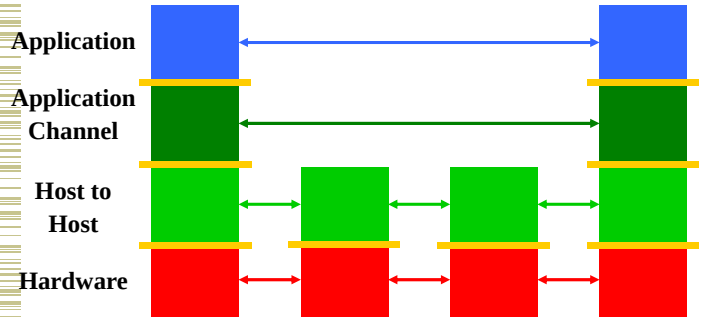
Types of Protocols



- Core network: responsible for transferring data between a sending and receiving host.
- End-to-end protocols: present a network service to applications and users.
  - May add value to the core network protocols
- Driven by differences in constraints: scalability, power, management, speed, etc.

53

Protocol and Service Levels

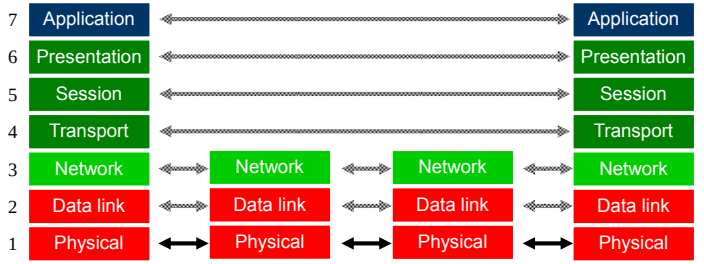


Layering: modular approach to network functionality

54

A Layer Network Model

The Open Systems Interconnection (OSI) Model



55

OSI Model: 7 Protocol Layers

- Physical: how to transmit bits
- Data link: how to transmit frames
- Network: how to route packets
- Transport: how to send packets end2end
- Session: how to tie flows together
- Presentation: byte ordering, security
- Application: everything else

- TCP/IP has been amazingly successful, and it is not based on a rigid OSI model. The OSI model has been very successful at shaping thought

56