CS/ECE 438, CSE 425 Communication Networks

Nikita Borisov
ECE Department, UIUC

Course Information

- Instructor
  - Prof. Nikita Borisov Office Hours: TBA
    460 CSL, 244-5385
    nikita@uiuc.edu
- TA
  - Steven (Zixia) Huang, zhuang21@uiuc..edu
  - Office hours TBA
- Webpage
  - http://www.cs.uiuc.edu/class/fa07/cs438/
- Newsgroup
  - class.cs438 on news.cs.uiuc.edu

Acknowledgments

- Slides are adapted from Prof. Kravets
- Some material contributed by Profs. Luo, Lumetta, Hajek, Vaidya
- Some material from Larry Peterson & James Kurose & Keith Ross

Prerequisites

- C Programming (CS241)
  - Pre-req for ECE students is ECE290, but ECE391/398SSL or C experience *highly recommended*
- Probability and Statistics (MATH 461,463 or ECE 413)

Textbook

- *Computer Networks: A Systems Approach*, by Peterson & Davie, 4th Edition
- Other editions
  - 3rd ed nearly identical, just changed page numbers
  - 2nd ed contains 90% of the material
  - 1st ed is significantly different, not recommended

Recommended Text

- *UNIX Network Programming, Volume 1*, by Stevens
- There are 3 editions
  - Second & third edition more up-to-date
  - First edition (1990) contains more background on general UNIX programming

Grading Policy

- Homework 14%
  - 7 homework assignments
- Mid-term Exam 20%
- Programming Projects 36%
  - 3 Programming projects
  - 2% off per hour late
- Final Exam 30%

Exams

- Midterm
  - Will be scheduled for mid-October
- Final
  - Not scheduled by the university
  - Please fill out survey card and give it back by next class

Homework and Projects

- Homeworks:
  - Due Wednesdays at 2:00 in class.
  - General extension to Fridays at 2:00pm (hard deadline).
  - No questions to TA or on newsgroup after class on Wednesday.
- Projects:
  - Due Fridays at 9:00pm.

Academic Honesty

- Your work in this class **must** be your own.
- Penalties for excessive collaboration and cheating are severe
- Sharing strategies and small code fragments (5-10 lines) OK
- Sharing homework answers and large sections of code forbidden
  - Don't post these to newsgroup!
- If in doubt, ask the professor

Course Objectives

- At the end of the semester, you should be able to:
  - Identify the problems that arise in networked communication
  - Explain the advantages and disadvantages of existing solutions to these problems in the context of different networking regimes
  - Understand the implications of a given solution for performance in various networking regimes
  - Evaluate novel approaches to these problems

Programming Objectives

- At the end of the semester, you should be able to
  - Identify and describe the purpose of each component of the TCP/IP protocol suite
  - Develop solid client-server applications using TCP/IP
  - Understand the impact of trends in network hardware on network software issues

What is a Network?

- Something that connects two or more computers
- Tools that help computers communicate
  - .. and people communicate

Pre-computer communication networks

- Telephone
- "Snail" mail
- Voice
- Telephone is closest to Internet, but each shares some problems

Networking issues

- Physical medium use
- Encoding
- Addressing & routing
- Sharing
- Performance

Physical medium use

- Electrical signals over copper wires
- Optical signals over glass fibers
- Radio signals

Encoding

- Bit encoding
- Framing
- Data formats

Addressing & Routing

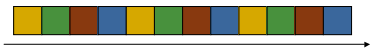
- How do you write an address on a letter?
  - Name
  - Street address
  - City
  - State
- How do letters get routed?

Addressing & Routing

- How do phone calls get addressed? routed?
  - Numerical addresses
  - Circuit switching
- How do voice conversations get addressed? routed?
  - Broadcast routing
  - Local addressing

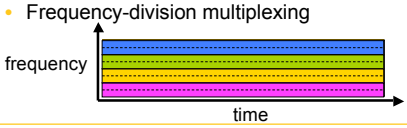
Sharing

- Time-Division Multiplexing
  - Synchronous



  - Ad-hoc

- Frequency-division multiplexing



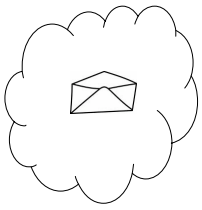
Performance

- Get more bits across faster
- Performance issues
  - Faster pipes
  - Sharing
  - Processing
  - ...

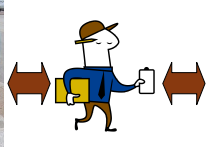
Functionality & Abstraction

- How does the mail system work?
- Elements:
  - Envelopes
  - Stamps
  - Mail boxes
  - Mail carriers
  - Delivery trucks
  - Sorting centers
  - Planes, trains, automobiles, ...

Mail system functionality



Mail carrier's view

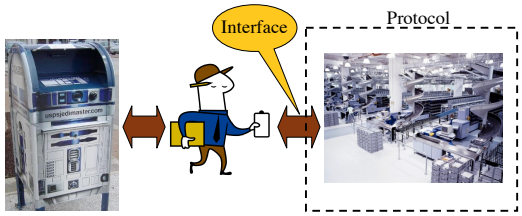


Protocols and Interfaces

- Interface
  - Specification of how to use an underlying system
- Protocol
  - Implementation of a particular interface
  - Sometimes also used to describe just the interface

Layered Architecture

- Decompose complex problem into layers
- Design explicit interfaces between layers



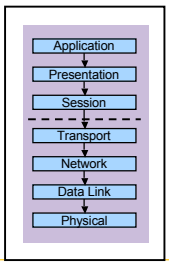
Layering

- Advantages:
  - Simplifies each component part
  - Allows flexibility for implementing each part
- Disadvantages:
  - Extra overhead imposed by interfaces
  - Processing & communication

OSI Architecture

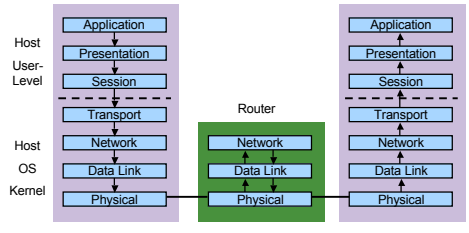
- Open Systems Interconnect (OSI) Architecture
  - International Standards Organization (ISO)
  - International Telecommunications Union (ITU, formerly CCITT)
  - "X dot" series: X.25, X.400, X.500
  - Primarily a reference model

OSI Protocol Stack



- Application: Application specific protocols
- Presentation: Format of exchanged data
- Session: Name space for connection mgmt
- Transport: Process-to-process channel
- Network: Host-to-host packet delivery
- Data Link: Framing of data bits
- Physical: Transmission of raw bits

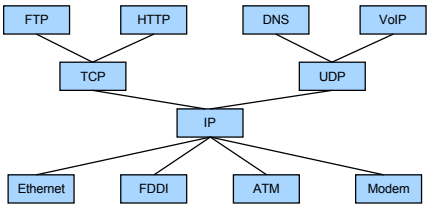
OSI Protocol Stack



Internet Architecture

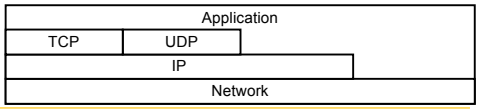
- Internet Architecture (TCP/IP)
  - Developed with ARPANET and NSFNET
  - Internet Engineering Task Force (IETF)
    - Culture: implement, then standardize
    - OSI culture: standardize, then implement
  - Popular with release of Berkeley Software Distribution (BSD) Unix; i.e., frees software
  - Standard suggestions debated publicly through "requests for comments" (RFC's)
    - We reject kings, presidents, and voting. We believe in rough consensus and running code. – David Clark

Internet Architecture – Hourglass Design



Internet Architecture

- Features:
  - No strict layering
  - Hourglass shape – IP is the focal point



Course Contents

- Overview
- UNIX Network Programming
- Direct Link Networks
- Multiple Access
- Packet Switched Networks
- Internetworking
- Reliable Transport
- Congestion Control, QoS & Fair Sharing
- Performance Analysis and Queueing Theory
- Application Protocols