Teaching Networking
For Fun And Profit

(Curricula and Labs)

Douglas Comer
Computer Science Department
Purdue University

www.cs.purdue.edu/people/comer
Undergraduate Courses (Breadth)

- Understand vocabulary and concepts
- Be able to state purpose and function of fundamental hardware and software components
- Know the role of protocols
- Be able to write programs that use computer networks
- Typical courses
  - Overview (everything from wires to applications)
  - Intro to Internetworking (TCP/IP)
  - Network Programming (sockets & middleware)
  - Intro to Network System Design (implementation of packet processing systems)
Graduate Course (Complete Mastery)

- Understand the design and implementation of protocols
- Be able to build correct and efficient system components
- Know how to architect large-scale networks
- Be able to discuss tradeoffs and limitations
- Typical courses
  - Internetworking (all the details)
  - Network System Design (engineering tradeoffs, network processors)
  - Seminars on latest topics (e.g., optical networks, wireless, web technologies, etc.)
Labs

• Motivation
  – Absolutely essential: students learn by doing
  – Reinforce concepts presented in class
  – Give students concrete understanding of details
  – Keep courses tied to reality

• Equipment
  – Doesn’t have to be the latest/fastest
  – Any lab is better than no lab (but simulation isn’t sufficient)
  – Show me your equipment and I’ll show you experiments
Equipment In Comer’s Labs

• Undergraduate Lab
  – 20 workstations with extra NICs that students can connect to private hub/switch

• Graduate Lab
  – 24 front-end workstations on gigabit ethernet
  – 85 (downloadable) back-end systems
  – 22 network processors
  – Miscellaneous equipment for experiments
    * VLAN switches and hubs
    * IP routers
    * Load balancer

†Thanks to: Intel, IBM, Cisco, Agere, AT&T, and others.
Front-End Systems In The Xinu Lab
Back-End Systems In The Xinu Lab
Example Undergraduate Lab Exercises

• Network programming
  – Build client and server using simplified API
  – Build a concurrent web server that supports server-side scripting using sockets

• Measurement
  – Compare throughput of 10/100 Mbps networks under load
  – Compare throughput of hub and switch

• Protocols
  – Capture and reassemble IP fragments
  – Trace a TCP connection and extract data
Example Timeline For Undergrad Overview Course And Lab

Course

- Signals & Media (LANs, WANs and Local-loops)
- Networking
- Internetworking (TCP/IP)
- Applications & other

Lab

- Net. Programming (Simplified API)
- Packet Analysis (Header decoding)
- Net. Programming (Socket API)
- Final Project (Concurrent Web server)
Example Graduate Lab Exercises (Team Projects)

- Design and implement a software-based IP router
  - Required protocols: IP, ICMP, UDP, and ARP
  - Student’s choice of multicast, NAT, SNMP...
- Design and implement a high speed IPsec box using network processors
- Design and implement a voice service over IP


Awards For Undergrad Network Curricula

- **Bronze**
  - One course focused on irrelevant mathematics

- **Silver**
  - One course about real networks and internets

- **Gold**
  - One course about real networks and internets with a hands-on lab

- **Platinum**
  - Two or more courses about real networks, each with a hands-on lab
Questions?