Undergraduate Curriculum

SIGCOMM 2002 Education Workshop

Russell J Clark
College of Computing
Georgia Institute of Technology
rjc@cc.gatech.edu
GT’s Current Undergrad Offerings

• 3251 Introduction to Networking
  Internet “Best” Practices - 120 students, 2/year

• 4251 Networking II
  More Depth, +Physical Layer - 40 students, 2/year

• 4255 Network Management
  FCAPS, SNMP - 30 students, 1/year

• 4260 Telecommunication Systems
  Telco Architecture and History - 30 students, 2/year

• 4270 Data Communications Laboratory
  Hands On Labs - 24 students, 2/year
3251 - Intro to Networking

- A survey course
- Current “best” practices of the Internet.
- Teach core networking concepts through the examples of current Internet protocols.
- Not a lab course. 60% written, 40% sockets programming
- Currently using Kurose and Ross.
Key Themes - My View

• What are protocols?
  why needed, standards, specifications, evolution

• Layering Abstraction
  service interface, encapsulation, modular design

• Performance Issues
  protocol overhead, bandwidth vs propagation delay, effect of error rate

• Network Programming Issues
  API's (e.g. sockets), data representation, reliability, security

• Security Issues
  cleartext messages, address spoofing, DOS
Key Themes - Students’ View

- **Applications** - HTTP, SMTP, POP, DNS
  Contrast of design, signaling. Demo using Telnet.

- **Transport** - TCP, UDP
  Connection management, ARQ, Flow and Congestion Control

- **Network** - IP, IPv6
  Routing protocols, addressing, NAT, firewalls, IP vs MAC addressing, IPv6 transition

- **Datalink** - CSMA/CD Ethernet
  Emphasis on probabilistic nature and timing issues, evolution to 100Mb, Gb and wireless.
Student Projects

Successes

- Network discovery: traceroute, ping, arp
- Socket implementations
  - TCP and UDP
  - Both client and server to my specification.
  - Just one end, must interoperate with mine.
  - Student designs protocol and implements.

Experiences

- commercial simulation packages
- performance competitions, bakeoffs
Challenges and Opportunities

• Top Down vs Bottom Up vs Neither
• Theory vs hands-on?
• How much programming and when?
• How to teach performance analysis issues?
• Lighter weight simulation projects.
• Dealing with larger class sizes