Computer Communication Networks
CSE 561 Spring 2002

David Wetherall and Neil Spring
{djw,nspring}@cs.washington.edu

This Lecture

1. Administrative
2. Projects (discussion)
3. Reviews (discussion)
4. Concepts

Administrative Stuff

- Everything you need is off the course web page
  - www.cs.washington.edu/561
- Your immediate TODO list:
  - Join the mailing list cse561@cs.washington.edu
  - Partner and find a project

The networks we study

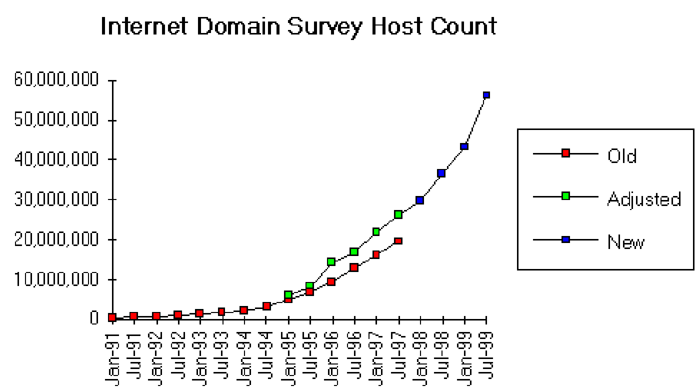
- Q: Why build networks?
  - Communication at a distance
  - Resource sharing
- Q: What do we mean in this course by network?
  - Economic networks, regulatory networks, …
  - Telephone, Cable TV, Bank tellers, computer clusters
- We are interested in networks that are: distributed, large scale, and heterogeneous. (Read, we study the Internet.)
- The above characteristics entail fundamental problems.

The meaning of "Distributed"

- There are distributed and parallel networks:
  - Cash machines versus a parallel computer
  - Both support concurrent computation
- What is the essential difference?
  - Tolerance of failed components
  - Decentralized operation

"A distributed system is a system in which I can't do my work because some computer has failed that I've never even heard of." – Lamport

The meaning of "Large-scale"



Source: Internet Software Consortium (http://www.isc.org/)

The meaning of "Heterogeneous"

- Telephone network
  - Designed for telephone calls
- Internet
  - Web, email, Quake, e-commerce, audio/video, …
  - But evolution was at work: Web/email a "surprise"
- Computer networks
  - Carry digital information and support a rich variety of distributed applications

Goals of this Course

1. To understand those fundamental issues and concepts that underlie the design of the Internet protocols and architecture.
2. To understand the state of network research and how to evaluate and perform networking research.

- This is a systems course, not queuing theory, signals, or hardware design.
- We mostly focus on networks, rather than applications that run on top of them (distributed systems).

Multiplexing

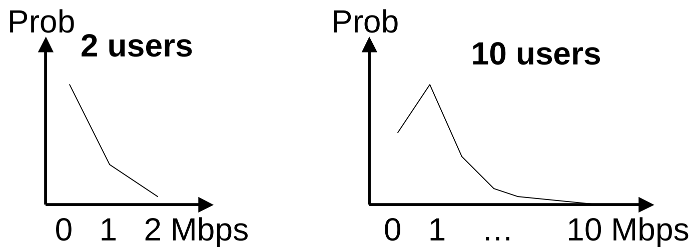
- Networks are shared among users
  - This is an important benefit of building them
  - Fundamental design issues concern effective sharing of distributed resources (effective = cost, control, secure, reliable, …)
- Problem: How to multiplex (share) a resource amongst multiple users, especially sharing a link?
- Well, we could statically partition the link:
  - Frequency Division Multiplexing (FDM)
  - (Synchronous) Time Division Multiplexing (TDM, STDM)

Statistical Multiplexing

- Static partitioning schemes are not suited to data communications because peak >> average rate (bursty).
- If we share on demand we can support more users
  - Based on the statistics of their transmissions
  - Occasionally we might be oversubscribed
- Statistical multiplexing is heavily used in data networks

Example

- One user sends at 1 Mbps and is idle 90% of the time.
  - 10 Mbps channel; 10 users if statically allocated



- What are the likely loads if we share on demand?

Example continued

- For 10 users, Prob(need 10 Mbps) = 10^{-10}
- So keep adding users …
- For 35 users, Prob(>10 active users) = 0.17%, which is acceptably low

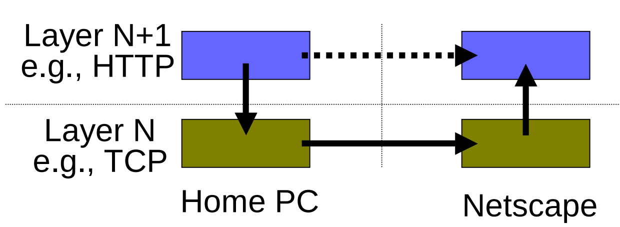
- We can support three times as many users!
- But: there is an important caveat here …

Protocols and Layering

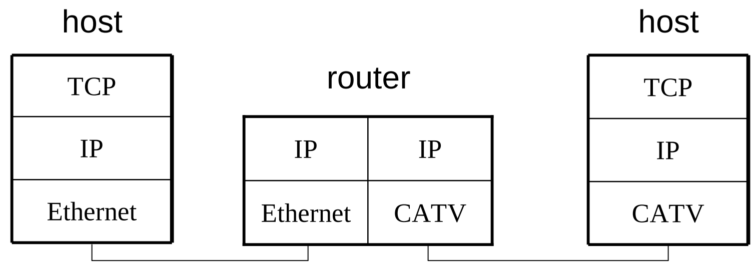
- We need abstractions to support interconnection and handle complexity
- A protocol is an agreement dictating the form and function of data exchanged between parties to effect communication
- Two parts:
  - Syntax: where the bits go
  - Semantics: what they mean, what to do with them
- Examples:
  - IP, the Internet protocol
  - TCP and HTTP, for the Web

Layering and Protocol Stacks

- Layering is how we combine protocols
  - Higher level protocols build on services provided by lower levels
  - Peer layers communicate with each other



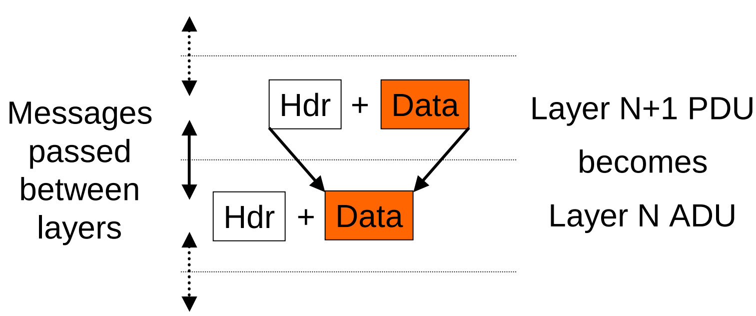
Example – Layering at work



- We can connect different systems

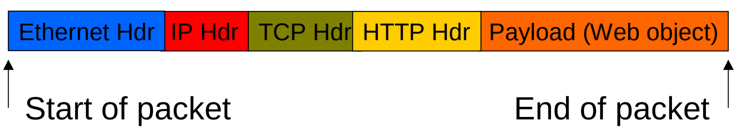
Layering Mechanics

- Encapsulation and decapsulation



A Packet on the Wire

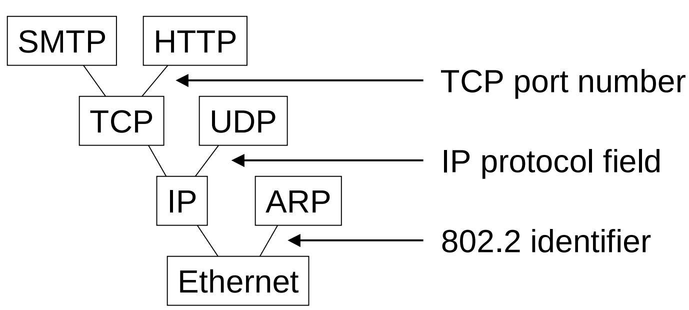
- Starts looking like an onion!



- This isn't entirely accurate
  - ignores segmentation and reassembly, Ethernet trailers, etc.
- But you can see that layering adds overhead

More Layering Mechanics

- Multiplexing and demultiplexing in a protocol graph



Internet Protocol Framework

Model	Protocols
Application	Many (HTTP,SMTP)
Transport	TCP / UDP
Network	IP
Link	Many (Ethernet, …)

OSI "Seven Layer" Reference Model

- Seven Layers:

Seven Layers	Their functions:
Application	Your call
Presentation	Encode/decode messages
Session	Manage connections
Transport	Reliability, congestion control
Network	Routing
Link	Framing, multiple access
Physical	Symbol coding, modulation

Key Concepts

- Distributed resource sharing
- Statistical multiplexing
- Protocols and layering