

Announcements

- Handouts

- class syllabus
- programming assignment #1 (includes computer account)
- department newsletter

- Enrollment

- there are 35 people in the class, and 22 on the wait list
- due to the size projects enrollment will not be increased
 - priority to fill drops will be given to **senior CS undergrads**
- this class will be offered again in the spring

- Required Background

- must have 311 and 330 (412 or 430 would be helpful)
 - if you have not passed 311 & 330 you will be dropped
- strong working knowledge of C or C++ (take your pick)
- willingness to work in a group environment

Announcements (cont.)

- **Required Work**

- will require about the same amount of effort as 412
 - 412 a (slightly) harder project to debug
 - 417 project is (by design) more ambiguous
- will need to write project proposals plus the code

- **Materials**

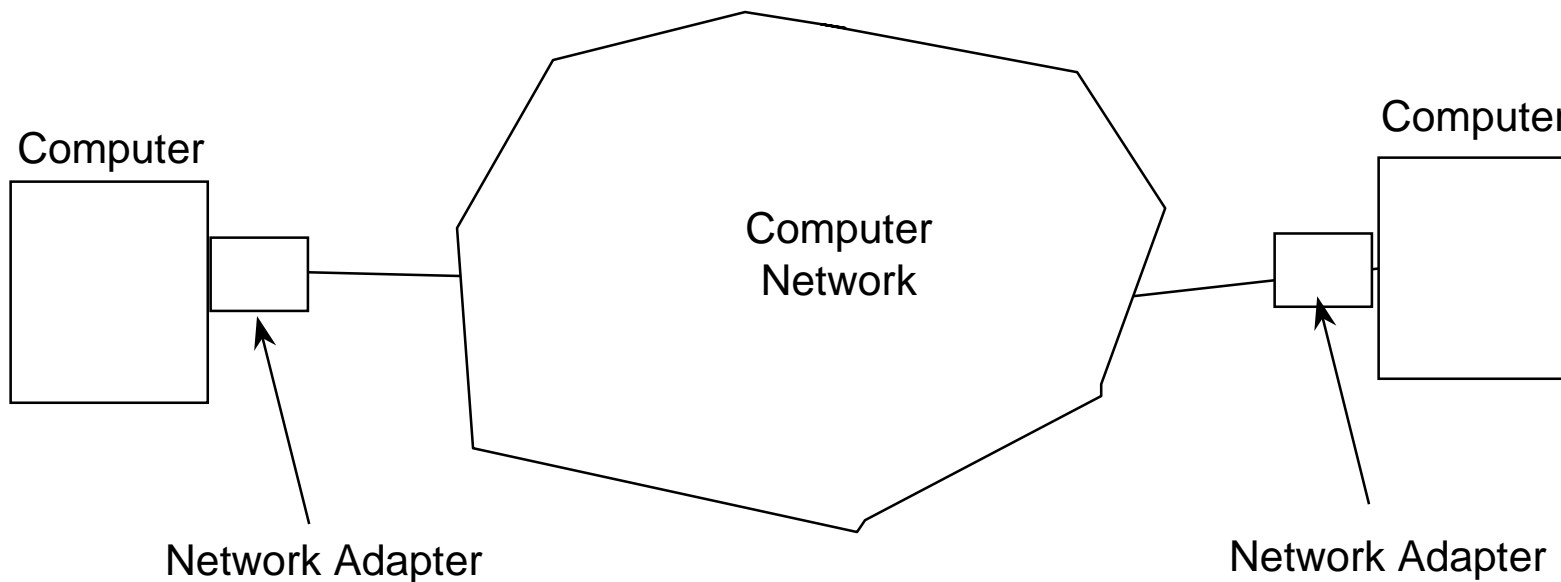
- Tanenbaum, “Computer Networks”, **3rd Edition**
- Nichols, Buttlar, and Farrell, “Pthreads Programming”
- Handouts from Web page

- **Reading (for this week)**

- Chapter 1

Networks

- Communication between semi-autonomous computers
- Attached to host system by an adapter



Many Types of Networks

- Physical Media

- copper wires (Ethernet, RS232-C, V.32, etc.)
- fiber optics (ATM, FDDI)
- air (IR, Radio, micro-wave)

- Speeds (link not aggregate)

- low
 - modems (few k bits/sec)
 - pagers
- medium
 - Ethernet (10-1000 Mbps)
 - Token Ring (16 Mbps)
- high
 - ATM (155-655+ Mbps)
 - Myrinet (600-1200 Mbps)
 - SONET (OC-48 - 2488 Mbps)

Network Topologies

- How are the communicating objects connected
- Fully connected - link between all sites
- Partially connected
 - links between subset of sites
 - can be an arbitrary graph
- Hierarchical networks
 - network topology looks like a tree
 - internal nodes route messages between different sub-trees
 - if an internal node fails, children can not communicate with each other
 - star network - hierarchical network with single internal node

A Network is not an Island

- Reason for networks is to share information
 - must be able to communicate in a common language
 - called protocols
 - The nice thing about protocols is that there are so many of them!
- Protocols
 - must be unambiguous and followed exactly
 - rule of thumb for good protocol implementations
 - be rigorous in what you generate
 - be liberal in what you accept
 - there are many different aspects to protocols
 - electrical through web services

Layering

- Layers provide information hiding
 - doesn't matter what lower level layers use as long as higher layers speak the same protocol.

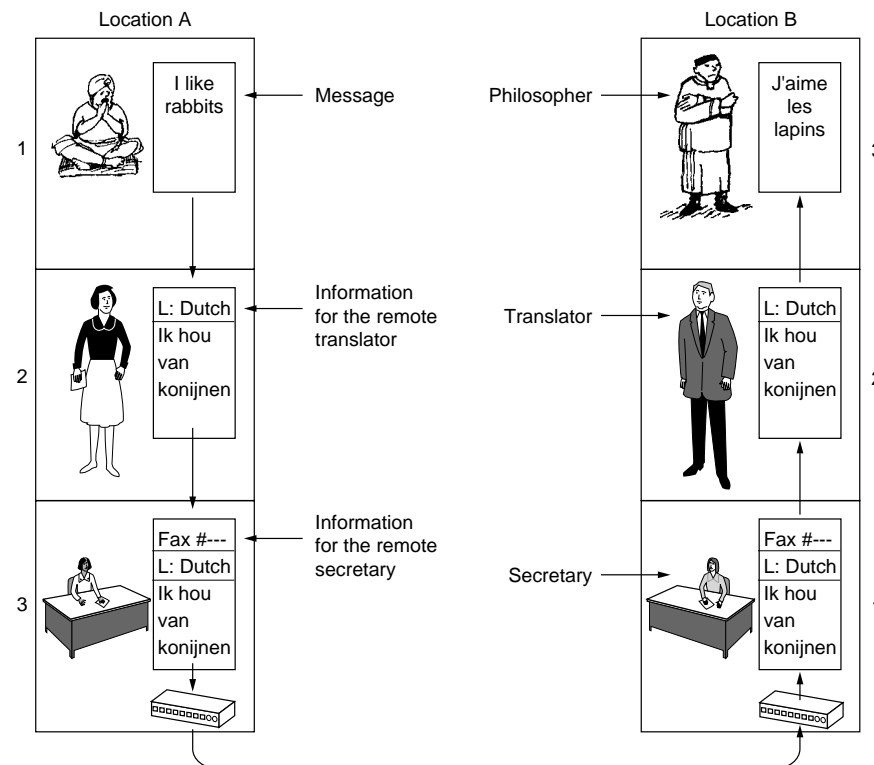


figure copyright , 1996, Andrew S. Tanenbaum

Design Issues In Layers

- Rules for data transmission (Protocol)
 - full vs. half duplex
 - error control (detection, correction, etc.)
 - flow control (rate matching, overuse of shared resources)
 - message order (do things arrive in the same order as sent?)
- Abstractions for communications
 - end points for communication
 - switches, nodes, processes, threads in a process
 - how are these end points named (addresses)?
 - service providers and service users
- Service Primitives
 - operations performed by a layer
 - events and their actions
 - request, indication, response, confirm

Protocols are divided into layers

- ISO - seven layer reference model
 - Application
 - Presentation
 - Session
 - Transport
 - Network
 - Link
 - Physical
- TCP/IP - four layer model
 - link
 - network
 - transport/session/presentation
 - application
- Old Saying: If you know what you are doing, four layers is enough; if you don't seven won't help.