* WEEK 1 NOTES CS372
* CS 372 Lecture #1
* Preliminaries
  + Networking is complicated
    - Enormous body of knowledge
    - Years of training
      * Specialization in
      * Hardware
      * Algorithms
      * Applications
      * Administration
      * Protocols
      * Encryption
      * Security
      * Wireless
      * Etc.
  + Why is networking so complex?
    - Variety of hardware
    - Variety of software
    - Variety of protocols / standards
    - Terminology can be confusing
    - Acronyms
      * Industry re‐defines or changes terminology from academia
      * New terms invented continuously
    - Wireless / mobility issues add even more complexity
  + Why is networking so complex?
    - There are many standards that have to be able to communicate with each other
    - Outside the most isolated local‐area network, every communication must include information about which standards are being used.
  + Handling the complexity
    - Concentrate on abstractions / concepts to unravel complexity
    - Use a few example technologies to illustrate the concepts
    - Use some hands‐on lab experience to reinforce the concepts.
  + Networks and the Internet
    - **Network**: system for connecting computers using a single transmission technology
    - **Internet**: set of networks connected by routers that are configured to communicate among a variety of network transmission technologies
      * A network of networks
  + The Internet
    - Millions of connected computing server devices:
      * hosts= end systems wireless running network apps
    - Communication links:
      * fiber, copper, radio, satellite
      * transmission rate = bandwidth
    - Packet switches: forward packets (chunks of data)
      * routers and switches
  + Networks and the Internet
  + We will analyze networks and the Internet in terms of
    - Transmission technologies
      * Data transmission ‐ media, data encoding
      * Packet transmission ‐ data exchange over a network
    - Communication protocols – reliable transmission
    - Internetworking ‐ universal service over a network of networks
    - Internet applications ‐ programs that use the Internet
  + In the real world …
  + Networks are an important part of everyday activities
    - Business
    - Home
    - Government
    - Education
  + Globally, the Internet is growing exponentially
    - Started out as a research project with a few dozen university sites
      * Support came from military
    - Today, billions of computers and millions of networks world‐wide
  + Growth of the internet
    - Ethernet made local networking possible
      * Ethernet is a networking standard
      * Several other standards exist
    - TCP/IP protocol made internetworking possible
      * Transmission Control Protocol / Internet Protocol
  + Growth of the internet
    - Fundamental changes from centralized to distributed computing
    - Incorporated features for reliability, robustness, and scalability
      * Multiple links
      * Distributed routing / messaging
    - Exponential growth
      * Counting Internet Hosts
        + Aug 1981 213
        + …
        + Feb 1986 2308
        + …
        + Jan 1991 376000
        + …
        + Sep 2000 360,985,492
        + …
        + Sep 2006 1.1 billion
        + Sep 2008 1.6 billion
        + Sep 2010 1.97 billion
        + Sep 2012 2.41 billion
      * Percent of Earth’s population 2006-2012
        + 16.7 %
        + 23.5 %
        + 28.7 %
        + 34.3 %
  + **CHARTS TO SHOW GEOGRAPHIC INTERNET USAGE, WORLD PENETRATION RATES**
  + Economic impact
    - Phenomenal industry growth around:
      * Computer hardware / software
      * Networking hardware / software
      * Network management / security
    - Companies must integrate planning, implementation, management and upgrades
    - Job creation
  + Network standards
    - Research started in 1967 to develop an Interface Message Processor (IMP)
    - Group started a repository for comments by other researchers … Request for Comments (RFC) ‐ 1969
      * http://www.rfc‐editor.org grew from this. It is the place to find all of the latest adopted standards for networking.
      * First “open‐source” community
    - Internet Engineering Task Force (IETF)
      * http://www.ietf.org
      * Non‐profit organization, produces technical documents that influence the way people design, use, and manage the Internet.
  + Preliminaries: Metrics (measurements)
    - Speed (distance/time) is measured in electronic units:
      * K = 103, M = 106, G = 109, etc.
      * E.G., network speed of 8 Mbps means 8,000,000 bits per second
    - Size in bits, Bytes is measured in binary units
      * Commonly used: K = 210, M = 220, G = 230, etc.
      * In this course, use: Ki = 210, Mi = 220, Gi = 230
        + E.G., disk size of 200 GiB means
        + 200 x 230 Bytes = 214,748,364,800 Bytes = 1,717,986,918,400 bits
    - Bytes and bits (abbreviations)
      * Use lower‐case b for bits
      * Use upper‐case B for Bytes
      * Example: 1 Mib = 128 KiB
  + Etymology
  + Common variable names in RFC
    - foo
    - bar
    - foobar
    - http://www.faqs.org/rfcs/rfc3092.html
  + **Summary Lecture #1**
    - Definitions:
      * network, internet
    - Networking complexity
    - Growth of the Internet
      * Economic impacts
    - RFC and IETF (networking standards)
    - Metrics (speed and size)
* CS 372 Lecture #2
  + Overview of Networking:
    - Network protocols
    - Service models
    - Network structure
    - Network edge ("fringe", "border")
      * end systems, links
      * applications
      * Note: Many of the lecture slides are based on presentations that accompany: Computer Networking: A Top Down Approach, 6th edition, by Jim Kurose & Keith Ross, Addison-Wesley, 2013.
  + What is a protocol?
    - human protocols:
      * “What time is it?”
      * “The chair recognizes …”
      * introductions
      * … specific messages sent
      * … specific actions taken when messages received, etc.
    - network protocols:
      * machines rather than humans
      * all communication activity in Internet is governed by protocols
    - protocols define
      * **format** and **order** of messages sent and received among network entities
      * actions taken on message transmission and receipt
  + **Example computer network protocol**
    - Has to follow a certain order in a certain format
  + A closer look at network structure:
    - **network edge**:
      * hosts and applications
      * clients and servers
    - **network core**:
      * interconnected routers
        + receive messages in the form of a packet
        + create a network of networks
      * network of networks
    - **physical media**:
      * communication links
        + Cables
        + Wireless
  + The network edge: service models
    - **end systems (hosts):**
      * individual computers that initiate requests or provide services
        + e.g. personal computers, cellphones
        + e.g. server computers
    - **client/server model**
      * client host requests/receives service from a server that is “always on”
        + e.g. Web browser (client), Web server, email client/server
    - **peer‐to‐peer model (P2P)**
      * minimal (or no) use of dedicated servers
        + e.g. Skype, BitTorrent
  + The network edge: connection‐oriented service
    - Goal: data transfer between end systems
    - TCP service [RFC 793]
      * **TCP ‐ Transmission Control Protocol** 
        + Internet’s connection‐oriented service
      * *handshake*: prepare for transfer
        + Hello, hello back (human protocol)
        + set up “state” in two communicating hosts
      * *reliable, in‐order, byte‐stream* data transfer
        + acknowledgements and retransmissions
      * *flow control*:
        + sender won’t overwhelm receiver
      * *congestion control*:
        + senders “slow down sending rate” when network is congested
  + The network edge: connectionless service
    - Goal: data transfer between end systems
      * same as connection‐oriented
    - UDP service [RFC 768]:
      * **UDP – User Data Protocol** 
        + Internet’s connectionless service
        + “light‐weight”, fast
      * no handshake
      * "unreliable" (best effort) data transfer
      * no flow control
      * no congestion control
  + The network edge:
    - Applications that use TCP:
      * HTTP (Web)
      * FTP (file transfer)
      * Telnet (remote login)
      * SMTP (email)
    - Applications that use UDP:
      * streaming media
      * Teleconferencing
      * DNS (Domain Name Service)
      * Internet telephony
  + **Summary Lecture #2**
    - Definitions:
      * protocol, network edge, network core
    - Network service models
      * Client/Server, Peer‐to‐Peer (P2P)
    - Network edge service types
      * Connection‐oriented, connectionless

CS 372 Lecture #3 (Part 1)

* + Overview of Networking:
    - Network core
      * circuit switching
        + frequency‐division multiplexing
        + time‐division multiplexing
      * packet switching
      * statistical multiplexing
    - Utilization
      * Note: Many of the lecture slides are based on presentations that accompany
      * Computer Networking: A Top Down Approach, 6th edition, by Jim Kurose & Keith Ross, Addison-Wesley, 2013.
  + The network core:
    - mesh of interconnected routers
    - the fundamental question: how is data transferred through the network?
      * circuit switching: dedicated circuit per call: telephone net
      * packet‐switching: data sent through net in discrete “chunks” (packets) on shared media
  + The network core: Circuit Switching
    - End to end resources reserved for “call”
      * link bandwidth switch capacity
      * dedicated resources: no sharing
        + **RESOURCE HOG**
      * circuit-like (guaranteed) performance
      * call setup required
    - For our purposes bandwidth means transmission rate which is usually expressed in bits per second
  + The network core: Circuit Switching
    - network resources (e.g., bandwidth) divided into “pieces”
      * pieces allocated to calls
      * resource piece idle if not used by owning call (no sharing)
      * Consumers are charged on a per‐minute basis
      * 2 ways of dividing the link bandwidth into “pieces”
        + frequency division multiplexing (FDM)
        + time division multiplexing (TDM)
  + Circuit Switching: FDM and TDM
  + Example:
    - Frequency Division Multiplexing (FDM)
      * 4 users
      * frequency v time
    - Time Division Multiplexing. (TDM)
      * frequency v time
  + Numerical example
  + How long does it take to send a file of 80 KiB from host A to host B over a circuit‐switched network?
    - The link’s transmission rate = 1.5 Mbps
    - Each link uses TDM with 24 slots/sec
    - 500 ms to establish end‐to‐end circuit
      * Figure it out … (watch the K, Ki and s, ms)
        + Solution:
        + 80 KiB = (80 x 210 Bytes) x (8 bits per Byte) = 640 Kib = 655.36 Kb
        + Bandwidth of one circuit = (1.5 Mbps)/24 = 62.5 Kbps
        + Time to send: (655.36 Kb)/(62.5 Kbps) + 0.5s
        + = ~10.5 s + 0.5 s = ~ 11 s
    - Discussion question: What would be different if we use FDM instead of TDM?
  + The network core: Packet Switching
    - all streams share network resources
    - each packet uses full link bandwidth
    - resources used as needed

* + - **Resource contention**:
      * aggregate resource demand can exceed amount available
      * congestion: packets queue, wait for link
        + too much traffic comes through the gateway

solutions? Queue

* + - ~~Bandwidth division into “pieces”~~
    - ~~Dedicated allocation~~
    - ~~Resource reservation~~
  + The network core: Packet Switching
    - Data transmitted in small, independent pieces
      * Source divides outgoing messages into packets
      * Destination recovers original data
    - Each packet travels independently (no circuit required)
      * Includes enough information for delivery
      * May follow different paths
      * Can be retransmitted if lost
  + The network core: Packet Switching
    - Functions of packet‐switching networks
    - Source host (edge): Packet construction
      * encode/package data at source
    - Routers (core): Packet transmission
      * send packet from source to destination
    - Destination (edge): Packet interpretation
      * unpack/decode data from packet at destination
      * acknowledge receipt
  + The network core: Packet Switching
    - Two key functions
      * **routing**: determines source- packets destination route taken by packets
      * **forwarding**: move packets from router’s input to appropriate router output
  + The network core: Packet Switching
    - Other functions
      * Queuing
      * Route discovery
      * Traffic/congestion control
      * Retransmitting lost packets
      * Determining type of data
        + messages
        + service requests/responses
        + files
        + audio/video – etc.
      * etc.
  + **Summary Lecture #3 (End of Part 1)**
    - Definitions:
      * network core
      * circuit‐switching, packet‐switching
      * multiplexing
    - Network core
      * composition (interconnected routers)
    - functions
      * FDM, TDM
* CS 372 Lecture #3 (Part 2)
  + Overview of Networking:
    - Network core
      * circuit switching
        + frequency‐division multiplexing
        + time‐division multiplexing
    - packet switching
      * statistical multiplexing
    - Utilization
  + Note: Many of the lecture slides are based on presentations that accompany
  + Computer Networking: A Top Down Approach, 6th edition, by Jim Kurose & Keith Ross, Addison-Wesley, 2013.
  + The network core: Packet Switching
    - Statistical multiplexing
      * DIAGRAM
  + The network core: Packet Switching
    - Statistical multiplexing
      * Sequence of A & B packets does not have fixed pattern
      * Transmission medium is shared on demand.
        + if a host is idle, its bandwidth is available to others.
      * Compare:
        + in TDM, each host gets same slot (periodically)
        + in FDM, each host gets same bandwidth (continuously)
        + with either TDM or FDM

a limited number of hosts get exclusive use of one slot or one channel

if a host is idle, its bandwidth is wasted.

* + The network core:
    - circuit switching vs packet switching
      * Circuit Switching
        + A reserves 1 circuit
        + B reserves 1 circuit but has none to send
        + Utilization = only 50% = 1 Mb/s B reserves 1
      * Packet Switching
        + A uses full link since B has none to send
        + Utilization = 100% = 2 Mb/s
        + Note: Packet switching has some additional overhead
  + Utilization (average % of bandwidth used)
    - Circuit switching transmits at a constant rate.
    - Packet switching allows more users to use network! (better utilization)
      * Suppose each user is active 10% of time
      * circuit‐switching:
        + with 10 users, each gets 0.1 Mbps when active
      * packet‐switching:
        + with 35 users, probability that more than 10 are active is less than .0004.
        + performance is nearly the same as 10 users with circuit‐switching

35 with packet is as efficient as 10 with circuit

* + - Discussion question: how did we get value 0.0004?
  + The network core: Circuit Switching vs Packet switching
    - Sometimes all (or most) users want to be active
    - Packet switching is fine for data that is not time‐critical
    - However …
      * *Excessive congestion causes*:
        + packet delay and loss
        + "jitter"
      * packet construction creates additional overhead
      * protocols needed for reliable data transfer, congestion control
      * bandwidth guarantees are needed for audio/video apps
  + Q: How to provide circuit‐like behavior (constant rate or appearance of constant rate)?
    - still a research problem
  + **Summary Lecture #3**
    - **Definitions:**
      * **network core**
      * **circuit‐switching, packet‐switching**
      * **multiplexing**
      * **utilization**
    - **Network core**
      * **composition (interconnected routers)**
      * **functions**

**FDM, TDM**

* + - **Statistical multiplexing**
* CS 372 Lecture #4
  + Overview of Networking:
    - access networks
    - internet structure
    - network performance
      * throughput
      * nodal delay
    - Note: Many of the lecture slides are based on presentations that accompany
    - Computer Networking: A Top Down Approach, 6th edition, by Jim Kurose & Keith Ross, Addison-Wesley, 2013.
  + Access networks
    - Access Networks are managed by Internet
    - Service Providers (ISP)
    - Connection to ISP edge routers via
      * Telephone lines
      * Cable
        + copper coaxial
        + fiber
      * Wireless
        + stationary
        + mobile

More later on transmission media

* + Access networks
    - **Modems** connect end systems to edge routers via access network media
    - Legacy
      * Dial‐up
      * DSL (digital subscriber line)
    - Cable
      * residential access nets
      * institutional access networks
      * (school, company)
    - Wireless
      * WiFi, WiMax
        + More later on routers and modems
  + Internet structure: network of networks
    - Hierarchical Structure of the Internet core.
      * Tier 1 includes commercial ISPs (AT&T, Sprint, etc.) and content provider ISPs (Google, etc.)
      * Tier 1 ISPs connect to each other, regional ISPs, and access networks via Internet Exchange Points (IXPs), or sometimes bypass the IXPs to connect more directly.
  + Network performance metrics
    - Throughput :
      * Rate (bits/sec) at which bits are actually being transferred between sender/receiver
        + **instantaneous**: rate at given point in time
        + **average**: rate over longer period of time
    - End‐to‐end delay (nodal delay):
      * Total time from initiating “send” (from source) to completed “receive” (at destination)
  + Four sources of packet delay
    - nodal processing:
      * Put packet together
      * check bit errors
      * determine output link
    - queueing delay
      * time waiting at output link for transmission
      * depends on congestion level of router
    - 3. Transmission delay:
      * R=link bandwidth (speed in meters)
        + R = transmission length
      * L=packet length (in bits) –
      * transmission delay = L/R
    - 4. Propagation delay:
      * d = length of physical link (in bits per second, i.e. “bps”)
      * s = propagation speed in medium (~2.5 x 108 m/sec) – speed of light
      * propagation delay = d/s
    - **Note: R and s are very different quantities!**
  + Nodal delay
    - dnodal = dproc + dqueue + dtrans + dprop
      * dproc = processing delay
        + typically a few microsecs (depends on hardware)
      * dqueue = queuing delay
        + depends on congestion
      * dtrans = transmission delay
        + = L/R, significant for low‐speed links (depends on hardware)
      * dprop = propagation delay
        + a few microsecs to hundreds of msecs (depends on distance)
  + The network core:
    - Packet‐switching: store‐and‐forward
      * It takes L/R seconds to transmit (push out) packet of L bits on to link at R bps
      * Entire packet must arrive at router before it can be transmitted on next link:
        + **store and forward**
      * delay = 3 x L/R (assuming zero propagation delay)
    - Example:
      * L = 12000 bits
      * R = 1.5 Mbps
      * The packet is transmitted
      * 3 times, so delay =
        + 3 x (12000/1500000) = 0.024 sec
  + Measuring delay with traceroute
    - Sends 3 packets to each router in path to destination.
    - Each router replies, and sender calculates total round-trip delay. Example: gaia.cs.umass.edu to www.eurecom.fr

traceroute www.eurecom.fr

* + Packet loss
    - Queue (buffer) has finite capacity
    - If packet arrives at a full queue, it is dropped (lost)
    - Lost packet may possibly be retransmitted by the previous node, by the source, or not at all
  + **Summary Lecture #4**
    - **Access networks**
    - **Structure of the Internet**
    - **Definitions:**
      * **Throughput**
      * **Nodal delay**
        + **processing, queuing, transmission, propagation**
      * **Store‐and‐forward**
    - **See the Java applet animations on the textbook website**