Benessa Defend

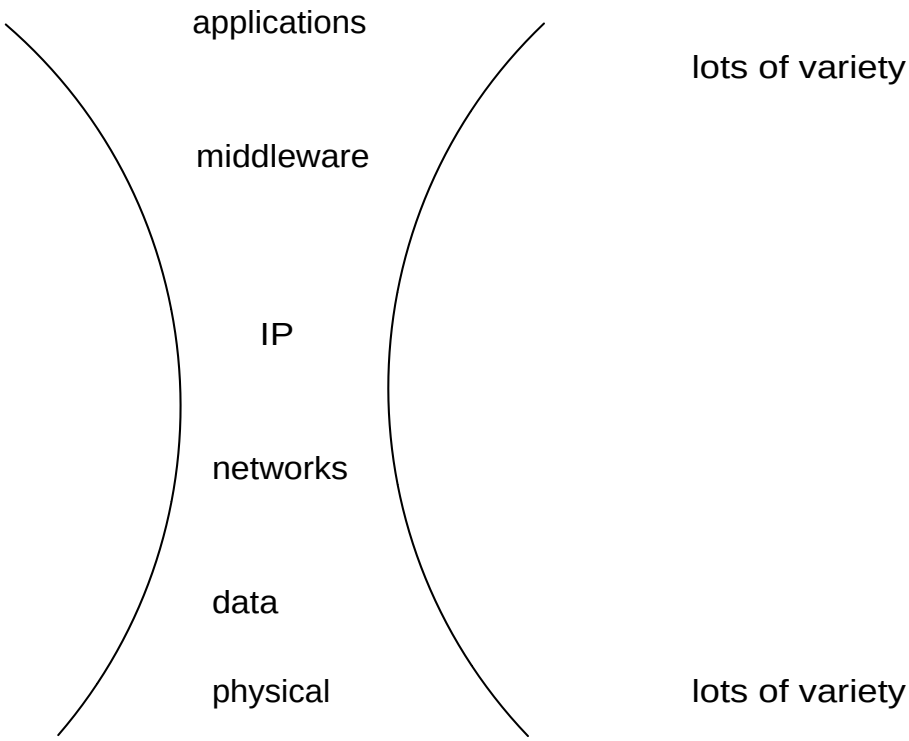
Notes

6/3/04 - composition of a network

the world wide web is an application that runs on top of the internet
new applications are created to run on top of the internet

layers:
- applications
- middleware layer
- application layer
- inter network
- network
- data
- physical (ex. Wires, optical)

similar to object-oriented programming b/c data is abstracted

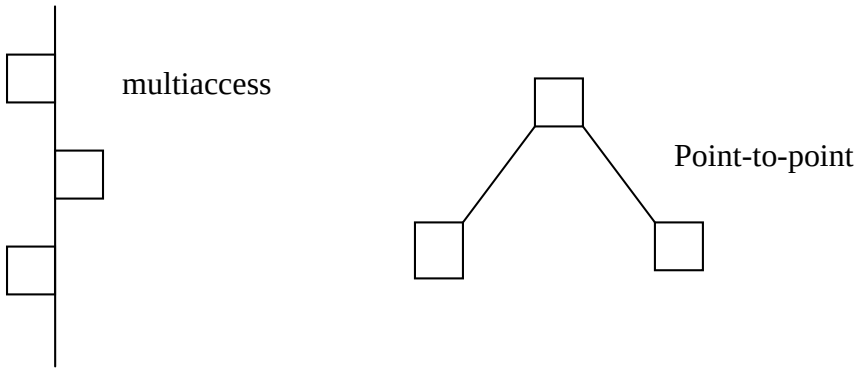


applications only have to work with IP and they will automatically work with the physical network

6/7/04 – paper discussion with grad students

peer to peer networks- have no client or server, both computers act as clients and servers, both are equal

multi-access- shared medium, must figure out who sent what where



WAN- wide area network, less reliable than LAN, ex. Telephone lines

MAN- metro area network, ex. Campus-wide

Internetwork- connects different networks together

Bandwidth (throughput) – data transmitted per time unit

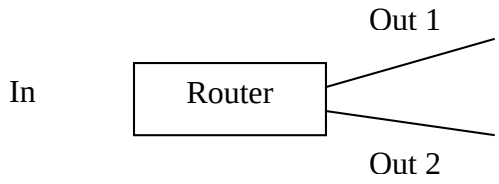
Ex. The amount of water that can flow through a hose- higher Bandwith means the hose has a larger opening

Latency (delay) – time to send a message from point A to point B

Reliability – bandwidth and latency may not be reliable

Layering – break up a task and solve each piece separately

Router – sends packet closer to destination



network layer- may need to break up message

packet headers – addresses

transportation layer – combines packets into messages, tells what process to go to, asks lost packets to be resent

end to end arguments – implement changes at the endpoints, not within the actual network. To implement the assurance of reliable communication (making sure a msg is received), implement changes at the end points. You want the network to remain simple- don't want to change TCP/IP, only change things at the end points.

6/8/04 – meeting with Monica

synchronous systems – every msg is sent/received in bounded time,
assume local computations have bounded time
assume clocks on computers have drift (aren't accurate)
assume you can make an assumption on drift value

asynchronous model – assume no bound on message delivery time
assume no bound on clock drift,
assume no bound on message processing time

hard to tell if a computer is crashed or just really slow

we need a model in the middle, a partial synchrony model with some assumptions

6/10/04 – Xinjie's presentation

Napster, Kazaa – unstructured P2P, files can be on any computer but
They aren't guaranteed to be on any computer

Namespace – all possible id's, legal names. To get id, hash file name

Circular namespace – biggest id + 1 = first id

Finger table – biggest finger is across half the circle (chord – like in Geometry), look in finger table first, hop directly through file System to the resource, store a resource close to where it Hashes

CAN – content addressable networks, keys and nodes mapped in d-dimensional space, each node responsible for keys mapped into its "cube", reduce the cost of one overlay hop by the node choosing the one closest to it

tapestry – cached local pointers, can send message to object (update, request), suffix-based, route by suffix- go to the node that shares the next digit

overlap hops can be large in the underlying network

Planet Lab notes

Slice name: wsu_monica

Nodes: planetlab2.iis.sinica.edu.tw
PlanetLab1.arizona-gigapop.net
Planetlab~1.cs.unibas.ch
Planetlab1.dcs.bbk.ac
Planetlab1.cse.msu.edu

Planetlab2.cse.msu.edu
Planetlab1.cs.wayne.edu
Planetlab2.cs.wayne.edu

To connect: ssh -l wsu_monica nodename
To transfer files: scp -r directory_name wsu_monica@nodename:path

7/8/04 - meeting with Monica
use fping instead of ping
fping < input file of host names to ping
easy to parse output
www.fping.com
write a program using fping in which one machine pings the other 7 machines in the PL slice - are times the same in both directions, what is the round-trip time?
try sending 5 probes, wait 2-5 minutes and send 5 more, continue for a few hours to determine the stability of the connection

7/13/04 - meeting with Monica
ping, traceroute- use ICMP "Internet Control Message Protocol"
raw socket - for sending other messages
PL uses safe raw sockets- let you program directly on IP w/o TCP or UDP
Scriptroute won't let you flood the server
TTL - time to live layer, # of hops allowed before protocol gives up
TTL decremented once at each hop
Traceroute - sends echo requests w/ ttl of 1, so msg hops 1 and sends back an error message. Thus you know the name of the first hop. It sends an echo request w/ ttl of 2 to determine the 2nd hop, etc. to determine the full path.