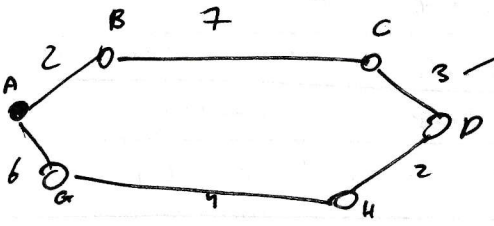
→ Shortest Path Routing (Non-adaptive)

* finds shortest path b/w 2 routers on the graph.
* Each node represent a router and each arc represent Comm. link.
* Metrics: no. of hops, distance, delay, etc.

e.g.



→ weight is based on metrics

* Djinstra algorithm.
* identify shortest-path tree for a dest. root node.
* Assume weights are non-negative (cuz this path w/ fewest hops)
* shortest path is one of lowest total weight
* algorithm:
  - set P = set of all nodes added to tree, initialize empty and root node. Iteratively add 1 node which has shortest distance to root node.
  - start w/ table. put lines at all to nodes = ∞.
  - put weights of neighbours for every iteration
  - Select the node with shortest distance (lowest weight)
  - put node in P and check neighbours of the selected node, repeat.
  - Note that this is the distance from curr. node to first node NOT curr. node to neighbour.

Flooding algorithm

* Every incoming packet is sent out on every outgoing line except the on on which it arrived.
* Inefficient: generates lots of duplicates; uses lots of bandwidth
* Selective flooding: send packets only on lines which are approx in right direction

- Private IP Addresses that cannot appear on Internet (reserved for private networks)
  - 10.0.0.0/8 ($2^{32} - 2^{8} = 2^{24}$ hosts)
  - 172.16.0.0/12 (2^{20} hosts)
  - 192.168.0.0/16 (2^{16} hosts)
- Network Address Translation (NAT)
  - Box maps one one external IP address to many internal IP addresses. Allows multiple devices on a private local network to share a single public IP address when accessing internet by translating private IP address to public.

→ IPV6

- larger address space: 128-bit address
- use hexadecimal color notation
- support more security, encryption and authentication.
- 40 bytes IPV6 header

→ Internet Control protocols

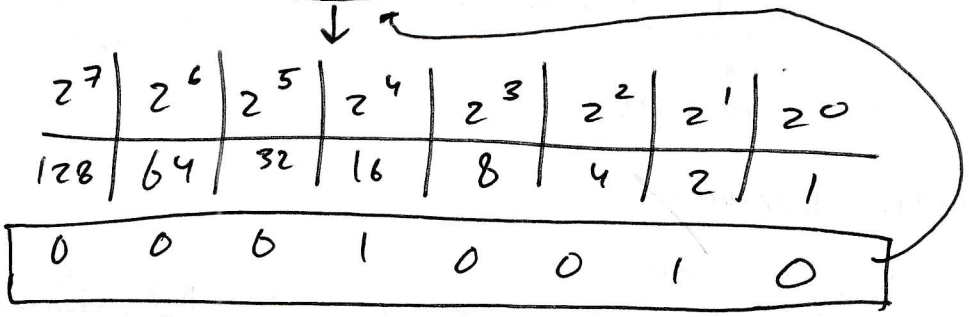
- IP works w/ help from other protocols
- ICMP (Internet Control message protocol): returns error info (e.g. traceroute, ping)
- ARP (Address Resolution protocol): finds MAC address of local IP address (Host queries an address and owner replies)
- DHCP (Dynamic Host Control Protocol): assigns a local IP address to a host (new host started by automatically config it.) (host sends req. to server, which grant a lease)

→ IpV4 addresses

✡ Subnetting: divide a block of addresses in IpV4

✡ ip v4 is 32-bit long, in dotted decimal notation.
4 segments, each up 8 bits.

e.g. 128 . 18 . 3 . 11



range: 0 - 255

✡ addresses are hierarchical and allocated in blocks

✡ prefix: determined by network portion, all hosts on a single network has the same network portion.

✡ network portion (24 bits) + host portion (8 bits)

All device in this network share the network portion

Chosen to a certain computer / device

✡ prefix: lowest address / bit-length for network portion

✡ Address allocation

* First come first serve, user who tells an Ip address first gets it. (flat)
* Allocate based on group. Split addresses based on group.
And group allocates each address in order to their users.

(first address in a network should have all 0's,
last — 1's)

e.g. 128.18.3.0 - 128.18.3.255

10000000 00010010 00000011 00000000 → 11111111

e.g. 128.18.3.0/27

host portion: 000 | 00000 first 0 (2^0-1)
000 | 11111 last 31 (2^5-1)

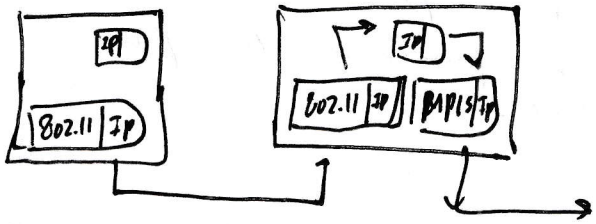
→ Internetworking

☆ joining multiple different networks into a single larger network

☆ Issues when connecting networks:

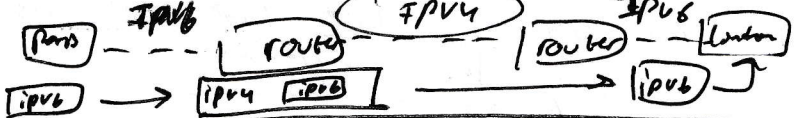
1.) Different network types and protocols
2.) Different motivations for network choices
3.) — tech at both hardware and software levels

} How to deal w/ different network requirements.

→ 

☆ Tunneling: dealing w/ when source and dest. are on the same network but there is another network in b/w (e.g. network is IPV6 but a subnetwork is IPV4)

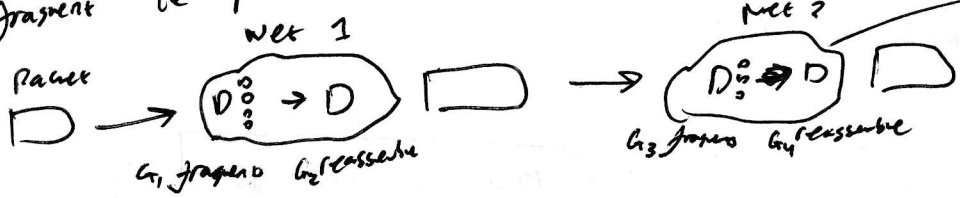
Source packets are encapsulated in packets travelling through connecting network.



Source and dest. must be on the same network for tunneling to work !!! (e.g. both must be IPV4/IPV6)

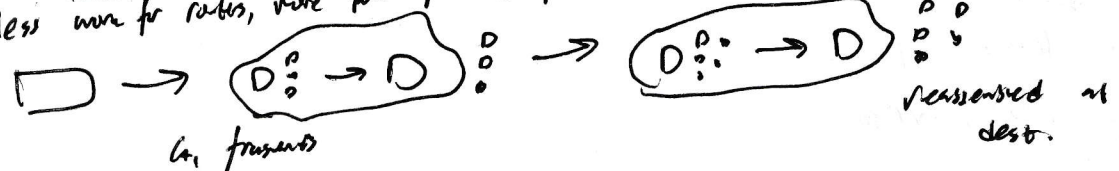
☆ Fragmentation

* All networks have max size for packets, max Transmission unit (MTU) to improve transmission efficiency and confidence (don't want to block channel or lose packets)
* Large packet size decreases overhead since division into smaller sizes would result in many smaller packets having to get headers.
* If packet is over limit, fragmentation divides packet into smaller packets and sent individually and independently to the channel.
* Transparent Fragmentation
  - invisible to the next network
  - fragment the packet and reassemble before exiting network



fragments based on network own requirements. Net 2 doesn't know it was been split by.

* Non Transparent Fragmentation
  - reassembled at destination
  - less work for routers, more work for computers (used in internet)



→ MAC Address
- Physical address for hardware (Physical interface)
- 48-bit no. encoded in frame, written in hexadecimal notation
  - 16 symbols (1 → 9, A → F)
  - (6 bytes)

→ Network Layer
- Connecting different networks (internetworking), routing
  - then appropriate paths
- Internet is a network of networks inter connected by IP
- IP provides best-effort service to route datagrams from source host to destination host
- Host may be on different networks
- Routing: Take the shortest path w/ shortest hops b/w source and dest.

→ Store-and-Forward Packet Switching
- Hosts generate packets and inject to network
- Router routes packets through network
  - treat packets as messages, receive/store them and then forward them based on address

→ Service provided to Transport layer
- Services independent of router technologies
- Network address to send packet (IP address)
- Types of services:
  - Connectionless: packets are injected into subnet individually and routed independently to dest. (Internet) (Chaos is hard)
  - Connection-oriented: packets travelling to dest. follow same route (Telecomm.) (Chaos is easy)

- Connectionless – post office model (Datagram subnet)
  - routing table is dynamic
  - receive – store – process – forward
  - routed individually based on dest. addresses
- Connection-oriented – tel. network model (Virtual Circuit Subnet)
  - define connection id used packet
  - virtual circuits are based on.
  - Send all packets using connection id.
  - To update, have to set up another network.
- Virtual circuit → can be whatever transmission media

→ Contention vs. Collision Free

☆ low loads : Collision free is less attractive due to Overhead

☆ higher loads : Contention method is less attractive due to higher no. of collisions.

→ Limited Connection Protocols

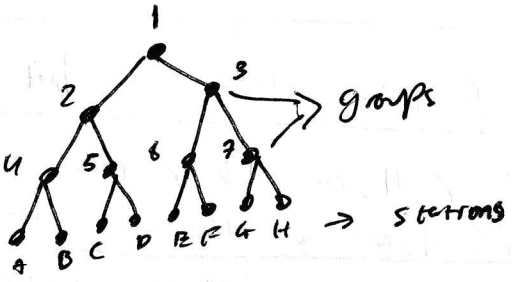
☆ Contention + Collision Free

☆ divide stations into groups, within which only a very small no. of stations are likely to transmit data

☆ Avoid waste due to idle periods and collisions

☆ Adaptive Tree Walk Protocol (LCP)

* organize stations in a tree and use tree to divide stations into groups.



* All stations can transmit. If collision occurs, binary division used to resolve contention.

* DFS under nodes w/ poll collisions

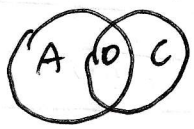
* start search at lower levels if > 1 station in a group wants to transmit.

* Go down the tree and if only 1 station in a group wants to send, that station will send first. (left to right)

* If it goes to final group w only 2 stations, use Contention or Collision free alg.

* Dynamic

→ Wireless LAN Protocols



☆ Stations have Coverage regions in wireless. (Can only hear or transmit in a certain range)

☆ Consider Wireless Coverage range of multiple stations. Interference affects signal reception.

☆ Stations must detect transmission around receiver, not just Carrier Sensing.

1.) Chanel is Shared; users Compete for resources

2.) stations generate new frames at a certain rate independent of others.

3.) Simultaneous transmission results in damaged frames (Collisions)

4.) Time
- Continuous transmission can begin at any time
- Slotted transmission, where timeline is divided into discrete slots, Stations can only send at slots.

5.) Carrier Sense
- Carrier sense: Detection of channel use prior to transmission (to check for potential collisions)
- no Carrier sense: No detection

→ MAC Protocols
- Contention: ALOHA, Carrier sense MA (Compete for channel) (random access: contention)
- Collision Free
- Limited Contention
- MACA / MACAW (for wireless LANs)

→ ALOHA (Contention)
- User transmit frames whenever they have data (no carrier sense)
- If there are collisions, retry after a random time (or no Ack arrived)
- No central control mechanism required
- Efficient under low load, high load is inefficient due to high traffic → high collisions

→ Slotted ALOHA (Contention) (frames wait for the next slot and send)
- Allow users to start sending only at the start of defined slot.
- Increase efficiency → reducing collisions
- Limitation: have a synchronized clock that every frame must follow!!!

→ Carrier Sense Multiple Access (CSMA) (Contention)
- Before transmission, require all senders to check channel to detect active transmission. Postpone transmission if it is used.
- Determine transmission rights dynamically (persistent: continue to try after listening)
- Protocols: • persistent and non-persistent CSMA
  • CSMA w/ Collision detection

→ Persistent and Non-Persistent CSMA
- 1-persistent CSMA: Continuous checking. Transmit 1 frame and check collisions. If yes, wait a random time and repeat.
- Non-persistent CSMA: If channel busy, wait random period and check again; If idle, start transmitting. If collision, " and repeat.
- p-persistent CSMA: If channel idle, transmit w/ prob p or defer to next slot w/ prob (1-p) and check again. If collision, wait for random time and repeat.

→ Data Link Protocols

✲ Point-to-Point Protocol (PPP)

* Delivery packets across links
* Framing: uses Flag byte w/ byte stuffing (0 111 1110)
* Payload contains IP packet from network layer.
* Default in unnumbered mode: Connectionless and unacknowledged service.
* Error detection using checksum.

Bytes: 1 | 1 | 1 | 1 or 2 | var | 2 or 4 | 1

Flag 0 111 1110	Address 11111111	Control 00000011	Protocol	Payload	Checksum	Flag 0 111 1110

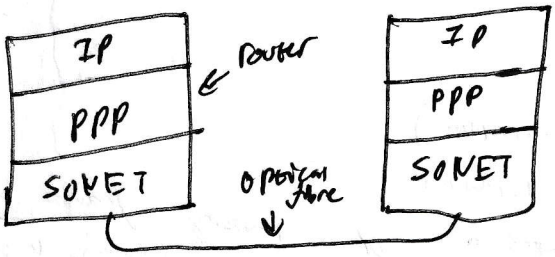
Control: ↑ Unnumbered mode

Payload: IP packet

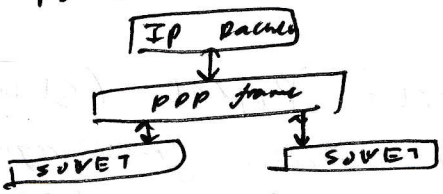
} frame in PPP

✲ Packet over SONET (Synchronous Optical Network)

* Carry IP packets over SONET optical fibre links
* Use PPP for framing
* SONET is fibre optic protocol in Physical layer, IP is protocol in Network layer
* SONET may split PPP frame into fixed length payload that is defined



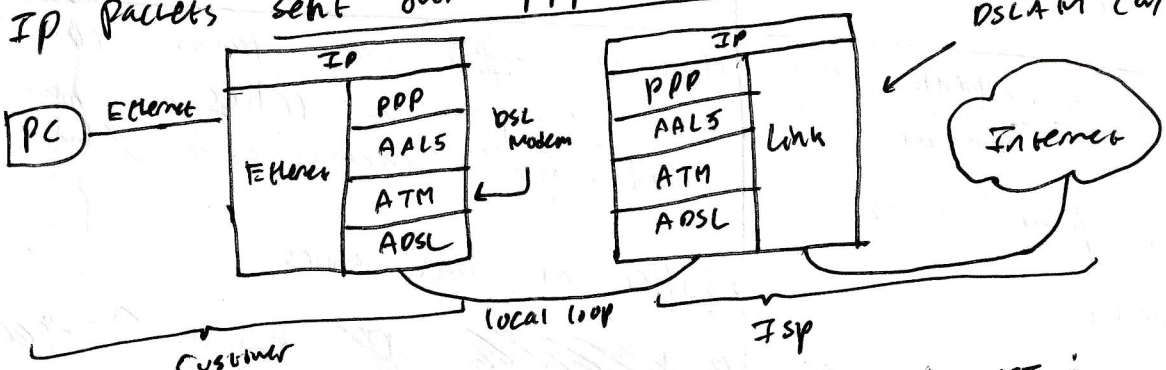
protocol stack

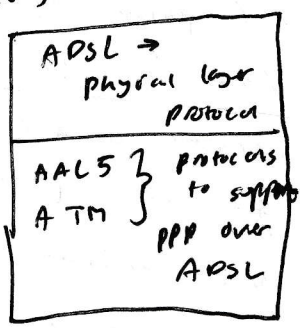


* frames are split and sent to the other side in regular intervals.

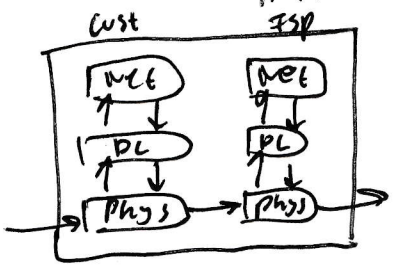
✲ Packet over ADSL

* Used for broadband Internet over telephone lines
* Runs from modem (Customer) to DSLAM (ISP)
* IP packets sent over PPP and AAL5, ATM



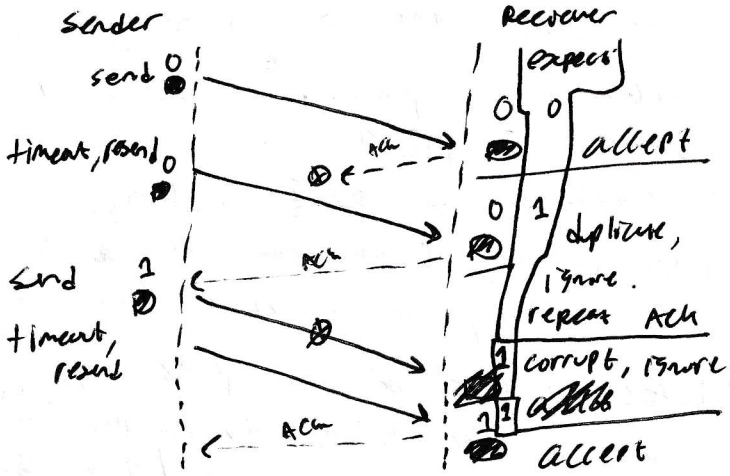


Again to intermediate steps in OSI:



✡ Noisy Channel Protocol

* frames can be lost. What if Ack is lost? Sender will be blocked.
* Timeout fn. to determine arrival or non-arrival of complete frames.
  To avoid being blocked by missing Ack.
* Reject duplicate frame? Requires seq. no. to avoid accepting the frame again
* Stop and Wait protocol
  - ARQ (Automatic Repeat reQuest)
  Ack, timeout, and seq. no.
  - range of seq. no. is 0 and 1,
  0 = prev frame, 1 = curr. frame (if buffer = 1)
  - expect var to see if it is duplicate or not.
  - timeout fn is set by us. • timeout fn. used to wait ~~for~~ until resend.



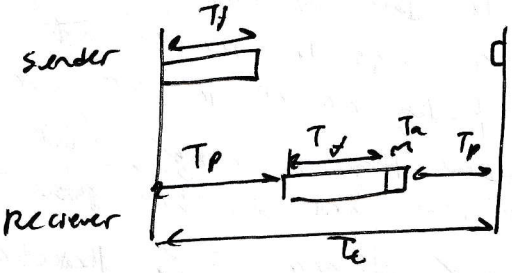
* Link Utilisation (LU) in Stop and Wait Protocols
  - measures efficiency of comm.

T_f = transmission delay, time needed to transmit frame of length L

T_p = propagation delay, wasted time

$T_a = 0$ (time for transmitting Ack)

$$T_t = T_f + 2T_p \qquad U = \frac{T_f}{T_t}$$



$$U = \frac{L}{L + 2T_pB}, \quad L = \text{frame length}, \quad B = \text{Bandwidth}$$

* Sliding Window Protocols
  - increase LU to decrease waiting time (wastage)
  - Have buffer in both sides:
  - Sending window: server maintains a set of sequence no. corresponding to frames allowed to send
  - Receiving window: Receiver ~~has~~ to frames allowed to accept
  - keep sending and increase proportion of time used for transmission, improve LU
  - Tradeoff: buffer frames before processing

→ Parity Bit (Detection)

- add 1 bit redundancy to make sure total no. of 1s is even or odd
- e.g. given 1000 1110, count no. of 1s
  - Sender: add parity bit → 1000 1110 0 (for even parity) → already even no. of 1s so add 0.
  - 1000 1110 1 (for odd parity) → need to add 1 since there are even 1s and need odd
  - Receiver: Count no. of 1s
- Hamming dist = 2, detection = 2 − 1 = 1 bit error
  - Correction = $\frac{2-1}{2} = 0.5$ bit error
  - Cannot locate any error, so can't correct errors.

→ Internet Checksum (16-bit word) (Detection)

- sum modulo 2^{16} and add any overflow of high order bits back to low-order bits
- e.g. 7, 11, 12, 0, 6, −36 ← make the sum negative so the receiver can just add everything and see if it adds up to 0.
- Add all of the binary codewords
  - ↓ Check if there is bit overflow → resolve by putting the 1 to the end
  - ↓ find one's-complement (flipping 0 to 1 and 1 to 0)
- Receiver repeats this process w/ the checksum to see if it results in all 1s or all 0s. If yes, no errors.
- Hamming dist = 2
- for internet, each codeword is 16-bits.
- fail if bits are flipped w/o changing the sum.

→ Cyclic Redundancy Check (Detection)

- Based on division. If there are any random errors, remainder should be different.
- long division based on generator polynomial G(x)

1.) get divisor from G(x). Divisor is the coeff of the polynomial.
(e.g. $x^4 + x + 1 = G(x)$, divisor = 10011) (use all terms x^4, x^3, x^2, x^1, x^0)

2.) Get dividend from augment data which is data to be sent plus 0s at the end, and the no. of 0s is equal to the highest power of G(x).

3.) Do long division using XOR, any 0s at the front can be crossed out and replaced w/ the same no. of the next bits in the augmented data.

4.) After everything, get final 4 bits in result and subtract it from augmented data (use XOR) and this is the data sent to the receiver after CRC.

5.) Receiver does long division on the sent data w/ the same divisor from the same G(x) and should get all 0's since remainder = 0.

Exclusive OR (XOR)

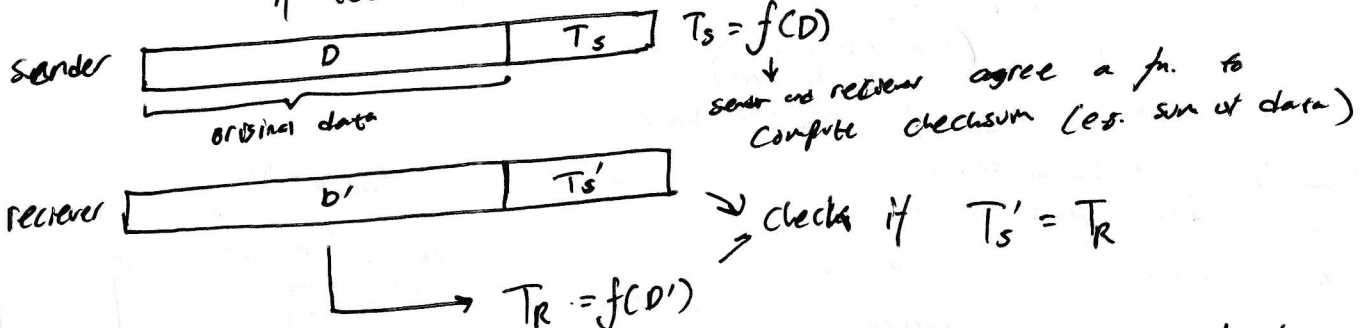
0	0	0
0	1	1
1	0	1
1	1	0

- 32 bits
- Hamming distance = 4

* Acknowledged Connection-oriented
* Source transmits independent frames to receiver host.
* Frames are numbered, counted, acknowledge w/ logical order enforced
* applications: unreliable links such as satellite channel

→ Framing

* Frame: groups raw bits into discrete units (group of bits)
* Framing: provide reliability over unreliable physical layer
* e.g. checksums: extra bits added to the end of data to help the other side if there are errors

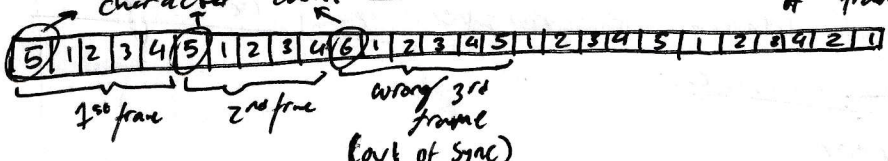


This is part of the trailer.

* let the other side know the start and end of a frame for error handling:
  * Char. (Byte) Count
  * Flag bytes w/ byte stuffing
  * Start and end flags w/ bit stuffing

} helping reciever side the start and end of a frame

* Character Count
* uses a field in the frame header to specify the length (no. of char) in the whole frame (header inc.) → tells you where the start of frame is.



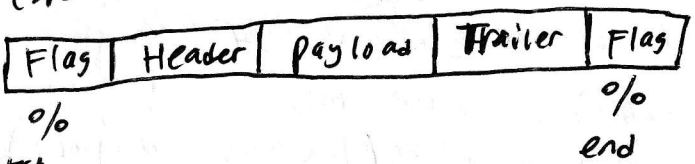
* To Convert to binary: 1 byte = 8 bits:

2^7	2^6	2^5	2^4	2^3	2^2	2^1	2^0
0	0	0	0	0	1	0	1

for 5

* use other method to find ending of frame

* Flag Bytes w/ Byte stuffing
* each frame starts and ends w/ a special byte - "flag byte" (special character: "o/o") → e.g.



what if o/o is in payload?
define "\" char after o/o to avoid this: "\ o/o" → stuffing byte

* for each esc "\" char, it only excuses the next byte. we need to use esc on esc symbols in payload as well.

e.g.

} make sure to avoid byte stuffing to avoid wasting bandwidth

* Modulation : Process of sending 0s and 1s (bits) using signals

- Baseband Transmission
  using signals from 0 to max freq. f
- Passband Transmission
  signals are shifted to occupy a higher range of freq.
- NRZ signal of bits } Baseband
  Amplitude shift keying } Passband
  Freq. —"—
  Phase —"—
  } Modulation methods
- Sampling Theorem : If the signal has highest freq at f_H (freq = f_H), then lowest sampling rate should be $2f_H$.
- Modulation is changing amplitude, freq, phase to represent 1s or 0s. symbol 1 symbol 2 $\log_2 2 = 1$ bit
- How many bits can I send based on N types/symbols (eg. 0 and 1 or 00, 01, 10, 11) = $\log_2 N$ ($\log_2 4 = 2$ bits)
- Symbol Rate
  Symbol = piece of signal or signal element (represent combination of bits (data elements))
  1 Symbol can represent multiple bits

Data Rate = $\log_2 N \times$ symbol rate → amount of data transmitted per unit of time across a medium !!!

Symbol rate = no. signal elements we can send per second (Baud rate)

- Max Data Rate of a Channel

Nyquist Theorem : Max data rate = $2B \log_2 V$ bits/sec
($2B$ → symbols/sec; $\log_2 V$ → bits/symbol)

Max data rate for noiseless channel, B = bandwidth, V = no. of signal levels (binary, ternary, etc)
(capacity for both noisy and noiseless)

Shannon Theorem : Max data rate = $B \log_2 (1 + S/N)$ bits/sec
(B → How fast signal can change; S/N → How many levels can be seen)

Max data rate for noisy channel, B = bandwidth, S = signal strength, N = noise

- High SNR : signal is understandable, low SNR = not understandable.
- Use both theorems to find the lower value which is the bottleneck (max channel cap.)
- If Nyquist is bottleneck, increase signal levels

- Channel sharing
* Users can transmit data at the same time (upload as download)
* Full duplex : transmission in both directions
* Half duplex : Both directions, but not at the same time
* Simplex : Only one fixed direction at all times
* Multiplexing (to increase cap. of channel)
- When multiple users want to access the medium
- Time division : Users can send according to a fixed schedule (slotted access to the full speed of channel)
- Freq division : Users can only use specific freq. to send their data (Continuous access of lower speed)

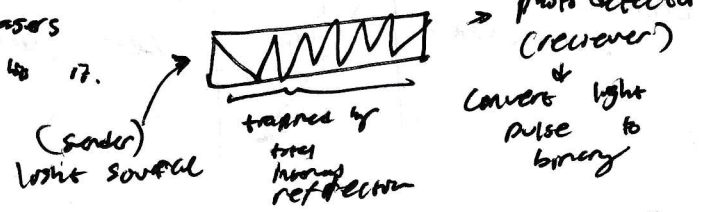
→ Architecture of Internet
- Users connect to local area network or Isp network
- service networks are connected to exchange points
- Backbone network support high speed of low distance transmission

→ Physical layer (data transmission over physical medium)
- Core abt bit level transmission
- 3 aspects : Mechanical, Electrical and Timing
- Both sides must agree to data rate and be in sync. → timing
- Mechanical : material, cable length (attenuation)
  Electrical : voltage levels, signal strength
- Bandwidth : rate of transmission in bits/second
  (how much data can be sent/recieved per sec)
- Delay / latency : time required for 1st bit to travel from comp A to comp B.
- Message latency : associate w/ sending a message over a link
- transmission delay : T-delay = $\frac{\text{Message length in bits}}{\text{Rate of transmission (bandwidth)}}$
  time needed to put message on the link
- Propagation delay : p-delay = $\frac{\text{length of channel}}{\text{speed of signals } (\frac{2}{3}c \text{ for wire})}$
  time needed to send message to the other side

$$\text{Latency} = \text{T-delay} + \text{p-delay}$$

$$\text{end-end transit time} = \frac{2 \times \text{distance}}{c}$$

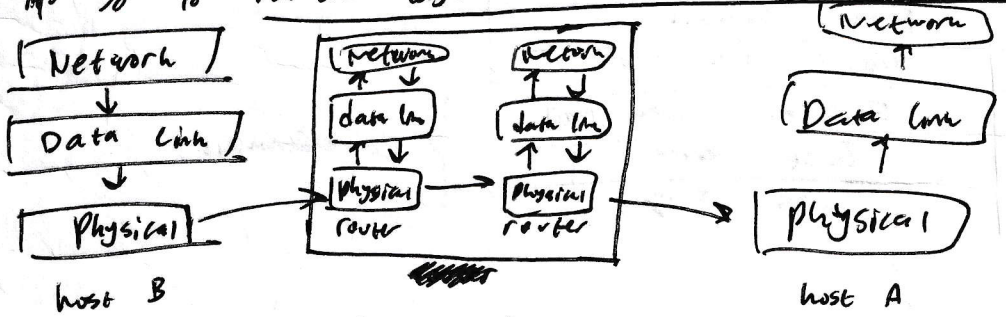
→ Transmission Media
- Wired : twisted pair, co-axial, fiber optics
  Wireless : EM waves and satellites
- Performance for physical media affected by physical properties
- Signal attenuation : loss in amplitude of a signal as it pass thugh a medium
  impacts how far and how much data a medium can carry.
- Twisted pair : 2 insulated copper wires twisted in helical form.
- twisting reduces interference ; cancelling out EM interference from external sources
- used by telephone line
- Bandwidth depends on distance, wire quality, density
  Cat 8 > cat 5 > cat 3
- Coaxial cable
- copper wire w/ insulation, mesh, sheath
- better shielding w/ twisted pair = higher speeds over greater distances
- Bandwidth approaches 1 GHz, used for TV/internet
- Fibre optics
- THz bandwidth and tiny signal loss
- fibre of glass
- high rates and long distances ; backbone lines b/w Isp facilities
- light source, transmission medium, detector
- signalling using LED's or semiconductor lasers
- detector generates electrical pulses when light hit it.



- protocol : set of rules that define details on how to provide service
  - how to add and what's inside header
- services are abstract, implemented using primitives
- service : set of primitives that a layer provides to a layer above it through interfaces btw layers
- analogy : services = interfaces, protocols = class implementation of interfaces

→ OSI (Open Systems Interconnection) (ref. model of a network)

- 7 layers, each performing a defined fn. chosen to minimize info flow across interfaces
- physical layer (L1) (bit)
  - sends raw data (0s and 1s) to the other side
  - Both sides agree to a transmission rate
  - transmission medium
- Data Link Layer (L2) (frame)
  - error detection / correction and flow control
  - provide reliable and efficient service for upper layer
  - can ask for retransmissions
- Network layer (L3) (packet)
  - How packets are sent from source to dst. across different networks
  - Handle intermediate steps btw sender and receiver
  - Info go to network layer and check IP and determines next router or pack



- intermediate steps to routers at physical, data link and network layers

- Transport Layer (L4) (segment)
  - doesn't care abt intermediate steps
  - How to send message from source to dst.
  - Conducts segmentation and re-assemble them
  - Transport and Network could split messages depending on protocol.
- session layer (L5) (SPDU)
  - control session, crash management (restart from certain point), e.g. fn. that match video and audio of data
- presentation layer (L6)
  - data representation and compression fn. (ensure data from app can be accepted)
- Application layer (L7)
  - web browsing, email, etc. * users can be humans or other software

→ TCP/IP (Transmission Control protocol / Internet protocol)

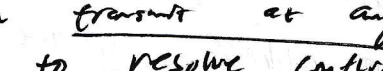
- 4 layers

	Application	HTTP, SMTP, RTP, DNS
	Transport	TCP, UDP
Network	Internet	IP, ICMP
	Link	DSL, SONET, 802.11, Ethernet

→ Network Topology

* Fully mesh: each device has a dedicated point-point link to every other device.
* n computers, each should connect to n-1 computers.
  total links: $n(n-1)$, bidirectional: $\frac{n(n-1)}{2}$
  Bad scalability since no. of lines increases quadratically w/ no. of comps.

* Bus
* runs a central cable as a backbone and all computers are connected to this backbone.
* only a single device on the network can transmit at any point in time. Requires negotiation mechanism to resolve conflicts.
* pro: easy to remove/add devices
* cons: single point of failure (cable)



* Star
* All devices attached to a central device (hub)
* hub manages the network
* Data from client sent to hub first, then it sends to dest.
* pro and can save w/ bus
* collision needs management!!!

* Ring
* Each device pass on data to the next device until it reaches dest.
* Transmission dir. set clockwise or anti-clockwise
* requires access control to resolve prop. issues
* bottleneck: device is slower and process info slower relative to others.

→ Network software: protocol Hierarchies (1)

* Network: stack of layers
* each layer offers services to layers above via interfaces
* Protocol: agreement b/w communicating parties on how communication is to proceed.
* API is an interface.
* L5 access service via L4/5 interface and L4 completes task. L4 completes it by getting service from L3 via L3/4 interface and so on.