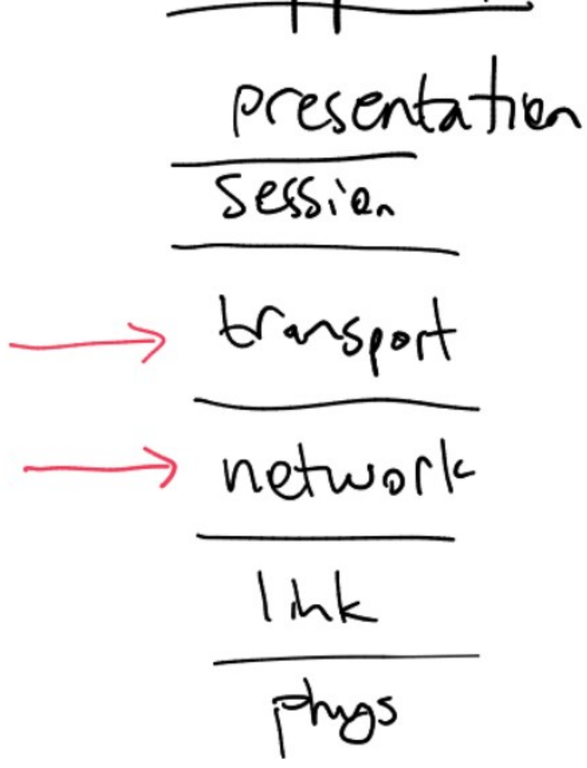
Lec 22: routing & transport

- Routing
  - source vs. path routing, BGP
- transport - layer
  - UDP & TCP



Reality: routing table entries correspond to subnetworks of different sizes.

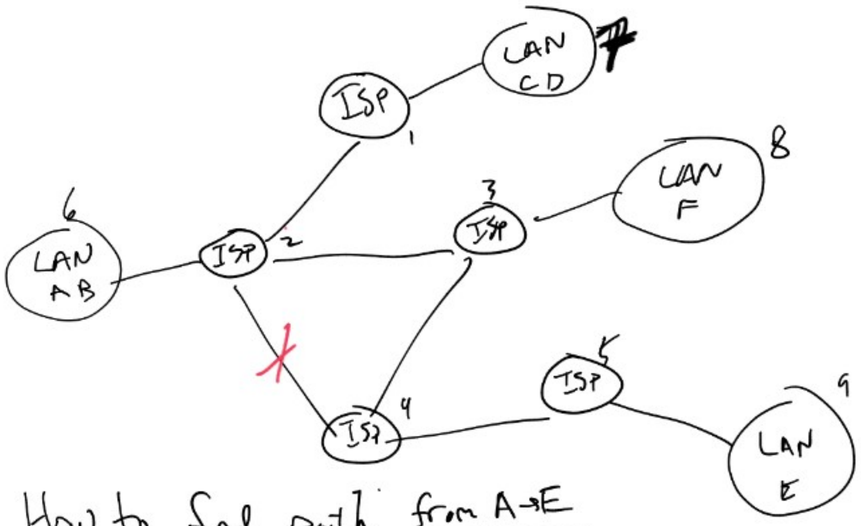
Subnetwork is identified by IP address + mask

e.g. 1.2.x.x

represented as 1.2.0.0 / 255.255.0.0

(IP; bits in first 2 digits set; clear)

to see if address is in a network:
(logical) and address w/ netmask, compare to network IP.



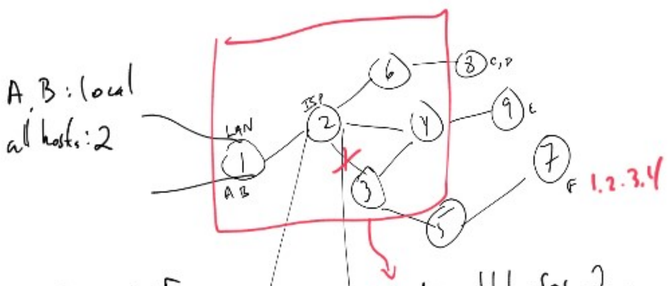
How to find path from A→E

- A computes shortest path through graph, writes path into packet, each ISP on path forwards to next.
- Eg: 6→2→4→5→9
  Alice is in 6, forward to 2, 2 looks @path, forward to 4, ...

Source routing: source of msg (A) finds route.

Pro:
- shortest path

Con:
- data can get dropped if topology changes
- each packet has a path in it: overhead.
- endpoints need to know entire graph



routing tbl for 2

if destined for	forward it to
A,B	1
C,D	6
E	4
F	3

Send from A → F

A → (gateway) 1 → 2

2 → 3

3 → 5

5 → 7

7 (gateway) → F

network IDs, not host IDs.

routing tbl for 3:

A,B,C,D: 2

E: 4

F: 5

routing tbl for 5

F: 7

anyone else: 3

1.2.3.4 / 255.255.255.255

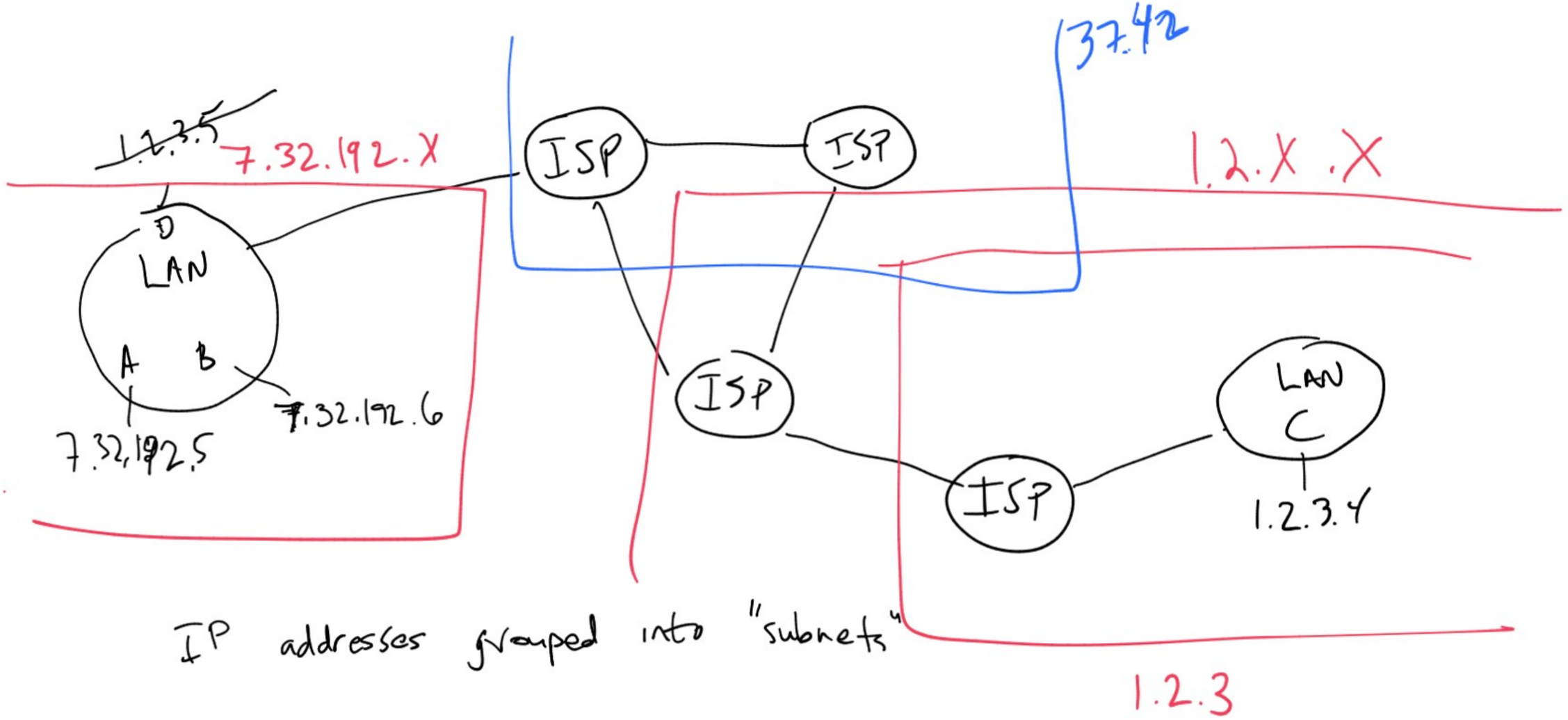
0.0.0.0 / 0.0.0.0

Path routing:
each node in path has table telling it how to forward packets. (based on dest addr).

pro:
- ISPs can dynamically adjust to respond to traffic.
- topology can be private.

Con:
- each ISP needs routing tbl: lot of storage?
- need to update routing tables when network changes.

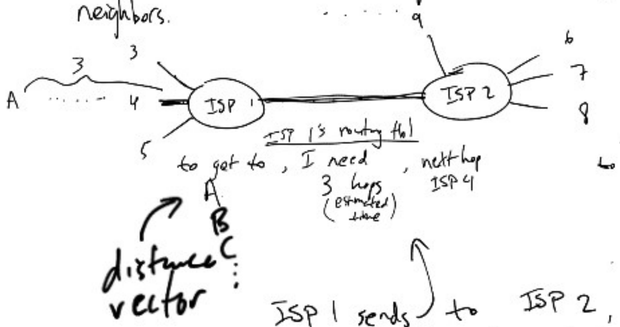
other concerns:
what about traffic on each link?



IP addresses grouped into "subnets"

How to build / maintain routing tables?

each ISP will periodically share routing table with neighbors.



ISP 1 sends to ISP 2, who notices it can forward packets to A using only 4 hops (better than before) by forwarding packets to ISP 1.

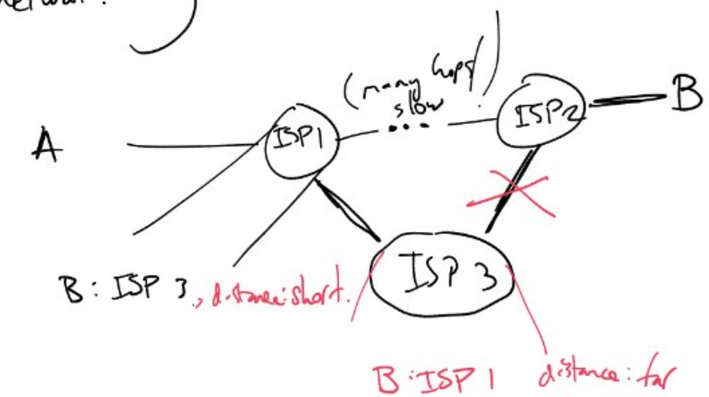
- Distributed solution to all-pairs shortest path algorithm.
- if topology changes, ISPs closest to change will notice first; eventually data will propagate to rest of network.

Suboptimal in mean time.

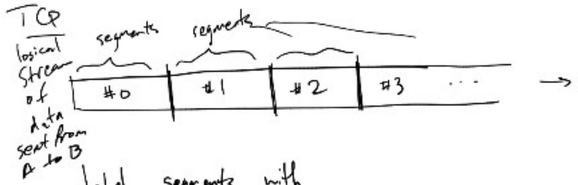
(could even loop!)

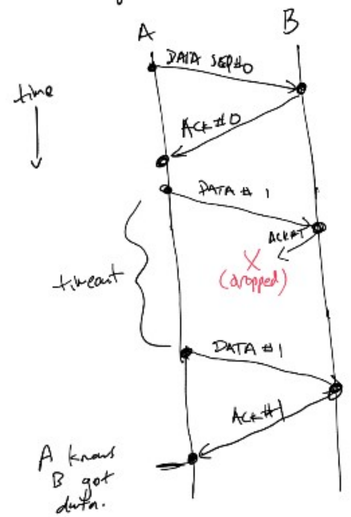
Border gateway protocol (BGP)

- maintain "distance vector".
  distance to each end network
- exchange with neighbors periodically.



- Transmission Control protocol (TCP)
  - multiplexing
  - "stream" abstraction.
    - two-way stream of data, each endpoint can write into stream, read from stream.
    - In-order, guaranteed delivery.



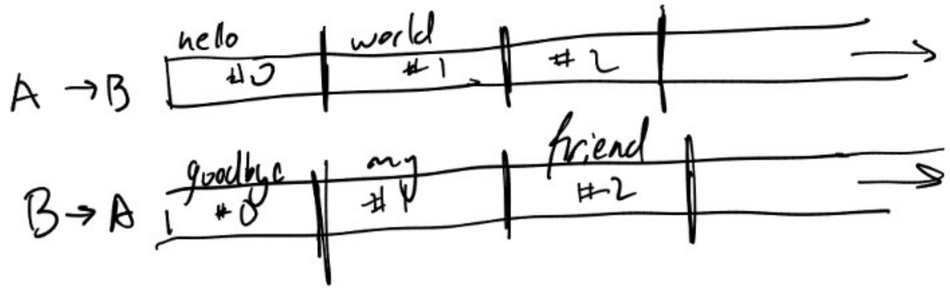


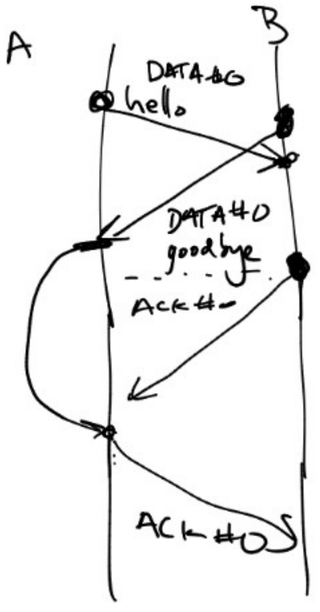
A send data, with sequence #

B reply with acknowledgement (ACK) and also containing seq. #.

- when A receives ACK, knows B received data.
- if it doesn't, resend data.

actually 2 logical streams in a TCP connection





Can combine DATA & ACK messages to send 1 packet instead of 2.

DATA / ACK

"piggybacking"