Transport Layer

- Service Models
- TCP vs UDP
- TCP Connections
- Flow Control and Sliding Window
- TCP Congestion Control
- Newer TCP Implementations

Service Models

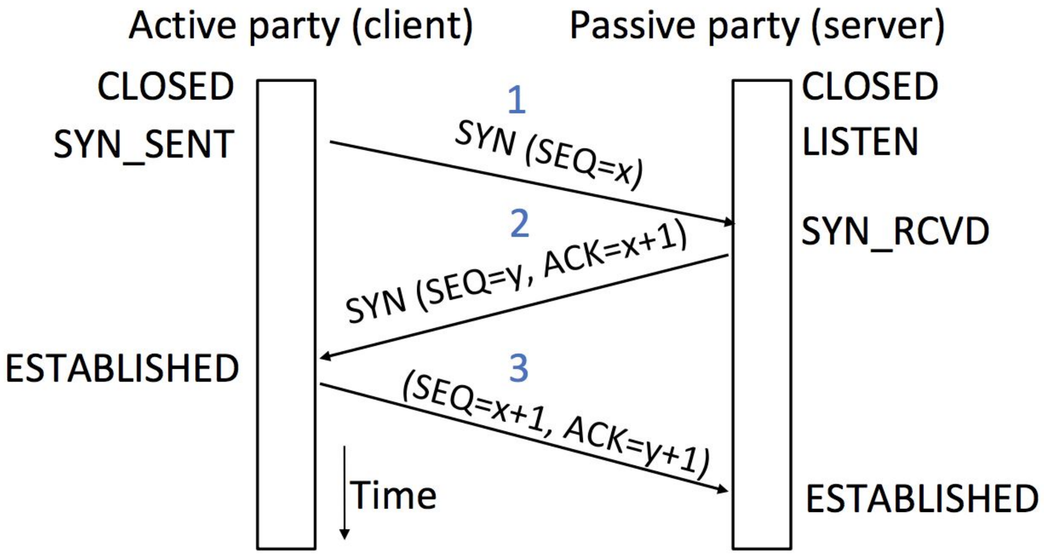
- Transport Layer Services
  - Datagrams (UDP): Unreliable Messages
  - Streams (TCP): Reliable Bytestreams
- Socket API: simple abstraction to use the network
  - Port: Identify different applications / application layer protocols on a host

TCP vs UDP

TCP (Streams)	UDP (Datagrams)
Connections	Datagrams
Bytes are delivered once, reliably, and in order	Messages may be lost, reordered, duplicated
Arbitrary length content	Limited message size
Flow control matches sender to receiver	Can send regardless of receiver state
Congestion control matches sender to network	Can send regardless of network state

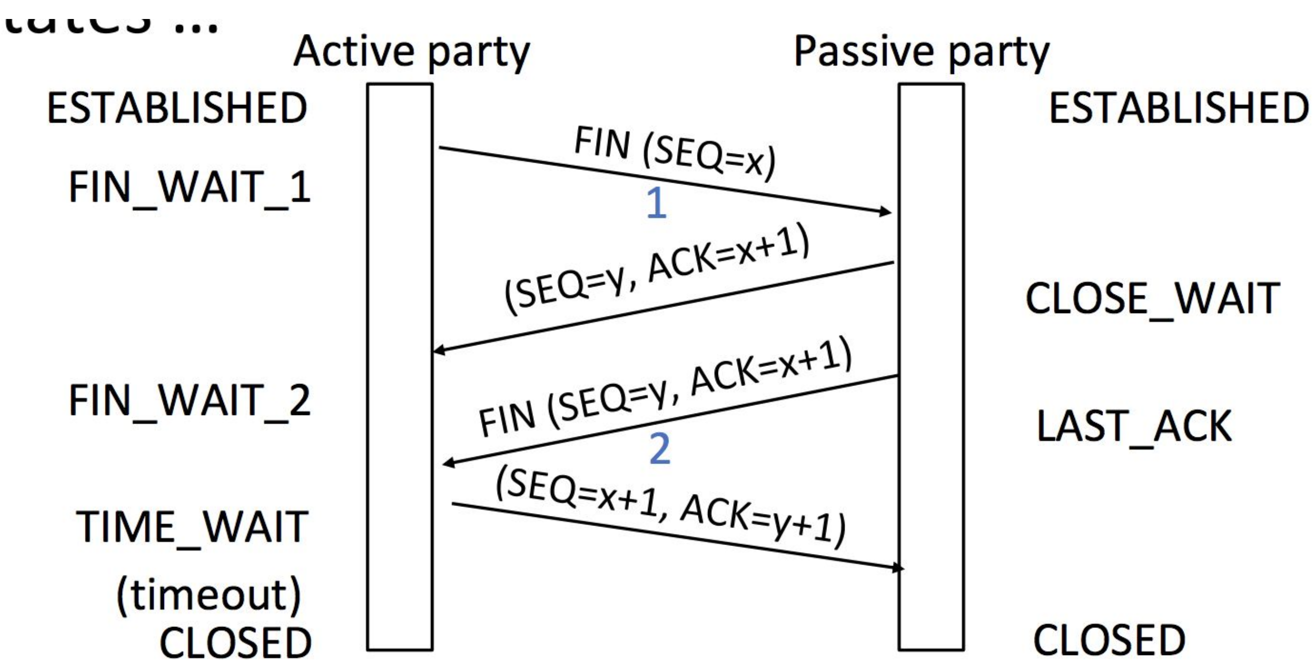
TCP Connection Establishment

Three-way handshake



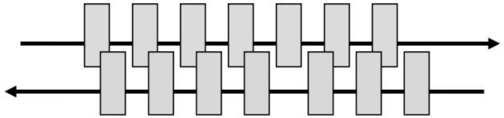
CSE 461 University of Washington

TCP Connection Release

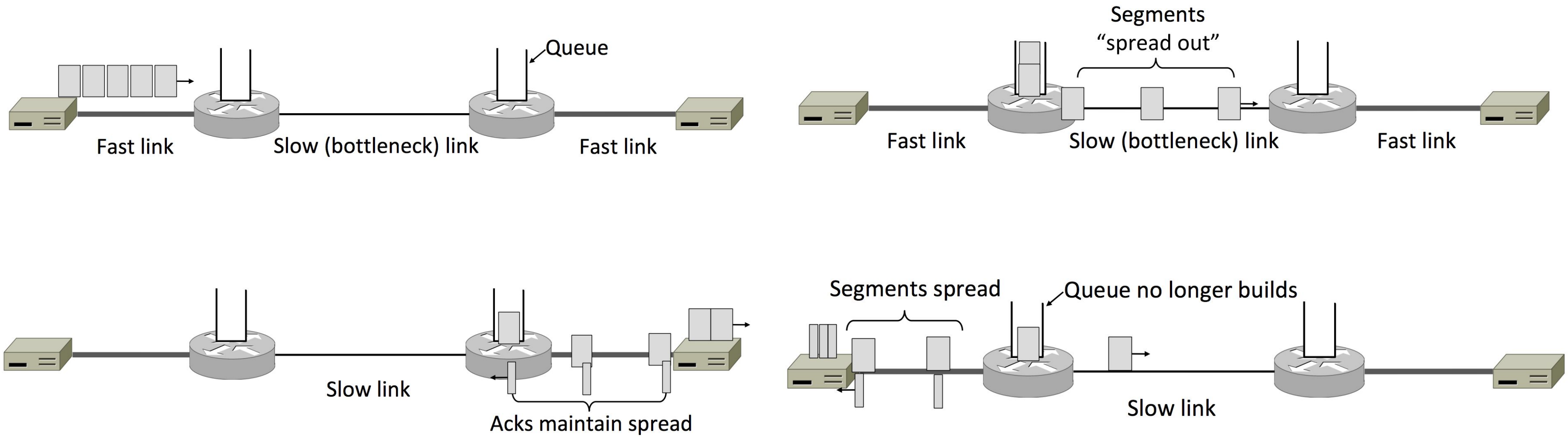


Flow Control - Sliding Window Protocol

- Instead of stop-and-wait, sends W packets per 1 RTT
  - To fill network path, W=2BD (make sure ans in packets!!!)
- **Shortest bandwidth determines rate**
- Receiver sends ACK upon receiving packets
  - Go-Back-N (similar to project 1 stage b): not efficient
  - **Selective Repeat**
    - Receiver passes data to app in order, and buffers out-of-order segments to reduce retransmissions
    - ACK conveys highest in-order segment
      - As well as hints about out-of-order segments
- **Selective Retransmission** on sender's side



Flow Control - ACK Clock



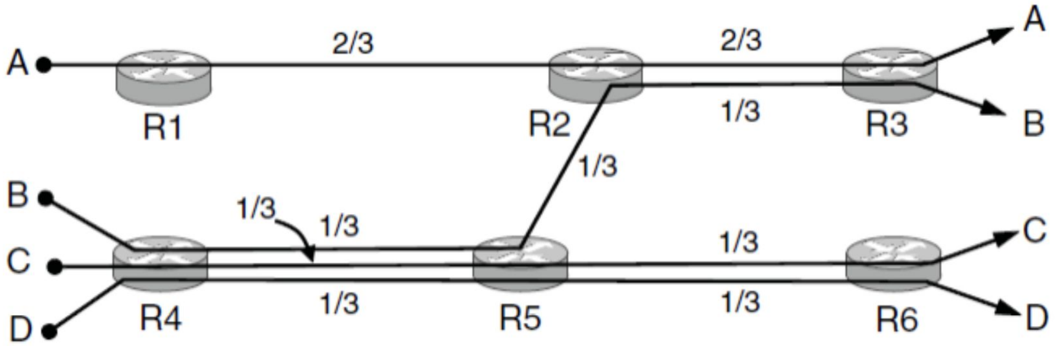
Flow Control - Sliding Window Protocol (2)

- Flow control on receiver's side
  - In order to avoid loss caused by user application not calling recv(), receiver tells sender its available buffer space (WIN)
  - Sender uses lower of the WIN and W as the effective window size

- How to set a **timeout** for retransmission on sender's side?
  - Adaptively determine timeout value based on smoothed estimate of RTT

Max-Min Fair Allocation

- Start with all flows at rate 0
- Increase the flows until there is a new bottleneck in the network
- Hold fixed the rate of the flows that are bottlenecked
- Go to step 2 for any remaining flows

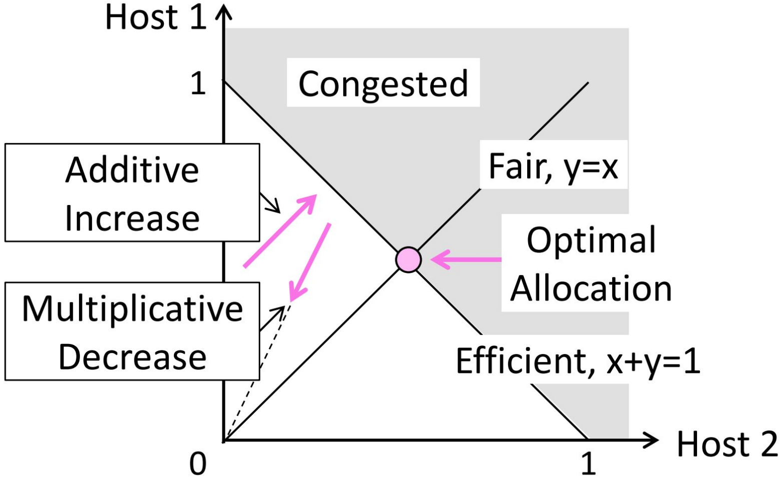


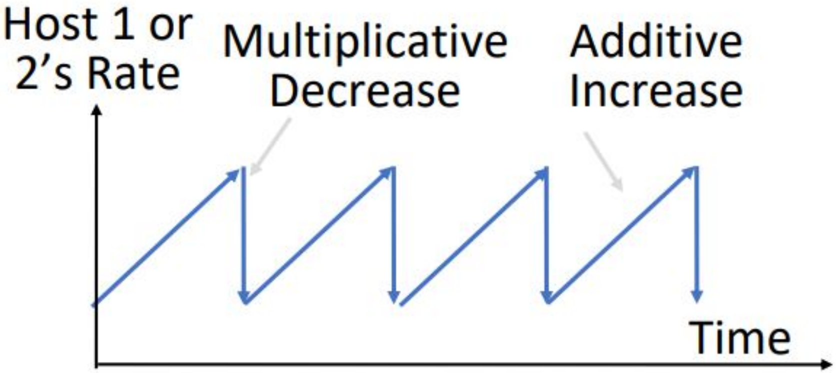
TCP Bandwidth Allocation

- Closed loop: use feedback to adjust rates
  - NOT open loop: reserve bandwidth before use
- Host driven: host sets/enforces allocations
  - NOT network driven
- Window based
  - NOT rate based
- Congestion signal
  - Packet loss, Packet delay, Router indication

AIMD!

AIMD - Additive Increase Multiplicative Decrease

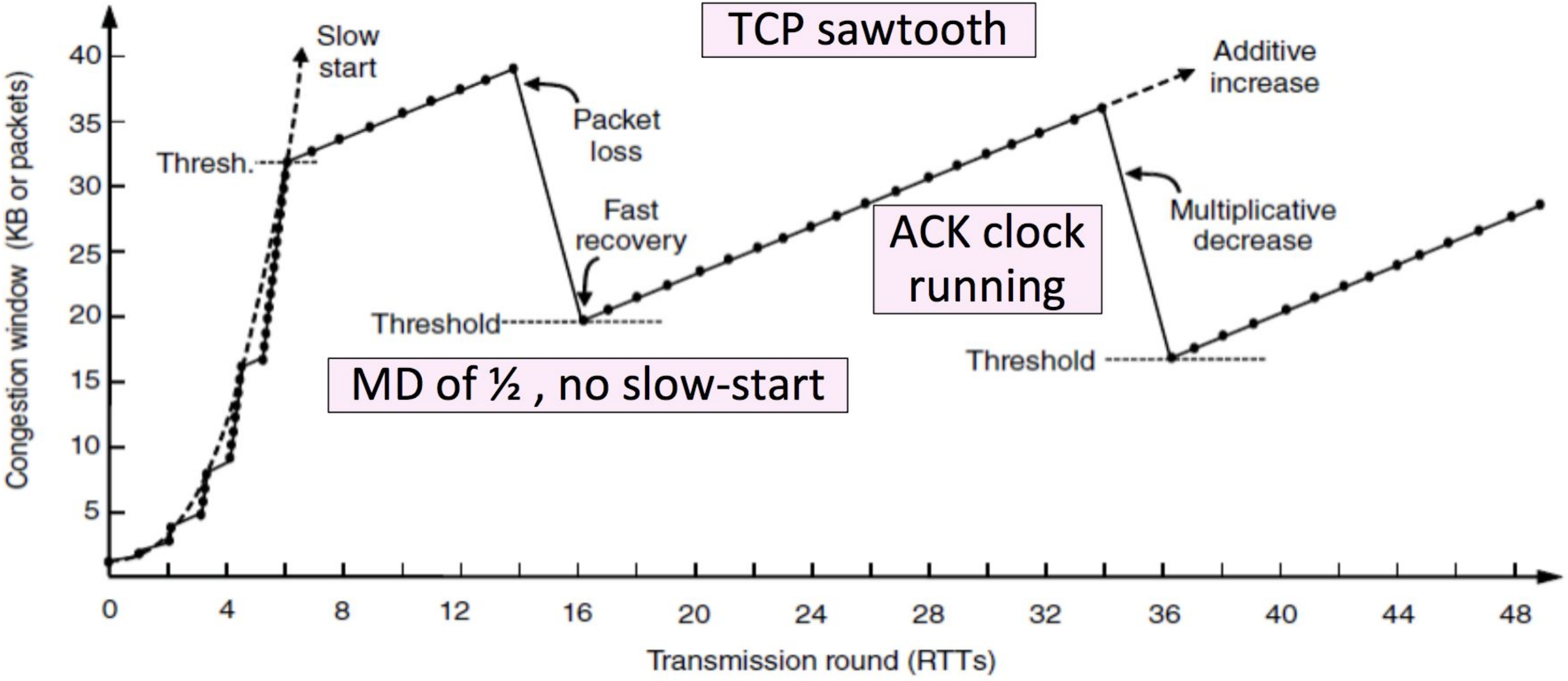




AIMD

- **Slow-Start** (used in AI)
  - Double cwnd until packet timeout
  - Restart and double until cwnd/2, then AI
- **Fast-Retransmit** (used in MD)
  - Three duplicate ACKs = packet loss
  - Don’t have to wait for TIMEOUT
- **Fast-Recovery** (used in MD)
  - MD after fast-retransmit
  - Then pretend further duplicate ACKs are the expected ACKs

TCP Reno



TCP sawtooth
Slow start
Additive increase
Thresh.
Packet loss
Fast recovery
Threshold
ACK clock running
Multiplicative decrease
Threshold
MD of ½ , no slow-start
Congestion window (KB or packets)
40
35
30
25
20
15
10
5
0
4
8
12
16
20
24
28
32
36
40
44
48
Transmission round (RTTs)

TCP CUBIC

- Problem with standard TCP?
  - Flows with lower RTT's "grow" faster than those with higher RTTs
  - Flows grow too "slowly" (linearly) after congestion

TCP BBR

- **Bufferbloat Problem**
  - performance can decrease when buffer size is increased
- **Model based** instead of loss based
  - Measure RTT, latency, bottleneck bandwidth
  - Use this to predict window size

Familiarize yourself with different protocols

- ICMP - Network Layer
- DHCP - Application layer
- OSPF - Network layer
- DCTCP - Internet layer
- Other protocols that are mentioned in lecture.

Internet protocol suite

Application layer

BGP · DHCP(v6) · DNS · FTP · HTTP · HTTPS · IMAP · IRC · LDAP · MGCP · MQTT · NNTP · NTP · OSPF · POP · PTP · ONC/RPC · RTP · RTSP · RIP · SIP · SMTP · SNMP · SSH · Telnet · TLS/SSL · XMPP · *more...*

Transport layer

TCP · UDP · DCCP · SCTP · RSVP · QUIC · *more...*

Internet layer

IP (IPv4 · IPv6) · ICMP(v6) · NDP · ECN · IGMP · IPsec · *more...*

Link layer

Tunnels · PPP · MAC

more...

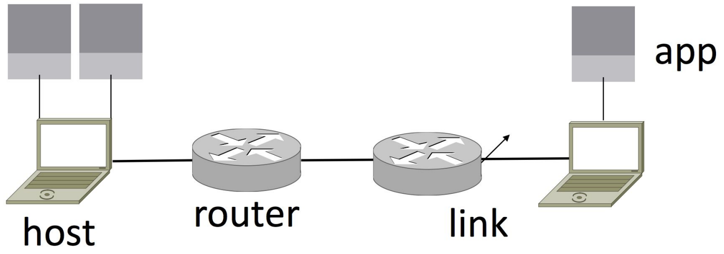
V · T · E

Network Components

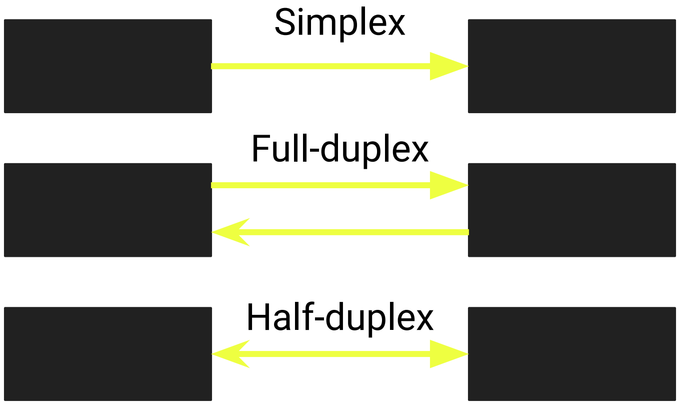
- Parts of a Network
- Types of Links
- Protocols and Layers
- Encapsulation
- Demultiplexing

Parts of a Network

- Parts of a Network



- Types of Links



Protocols and Layers

	Purpose	Protocols	Unit of Data
Application	Programs that use network service	HTTP, DNS	Message
Transport	Provides end-to-end data delivery	TCP, UDP	Segment
Network	Sends packets across multiple networks	IP	Packet
Link	Sends frames across a link	Ethernet, Cable	Frame
Physical	Transmit bits	__	Bit

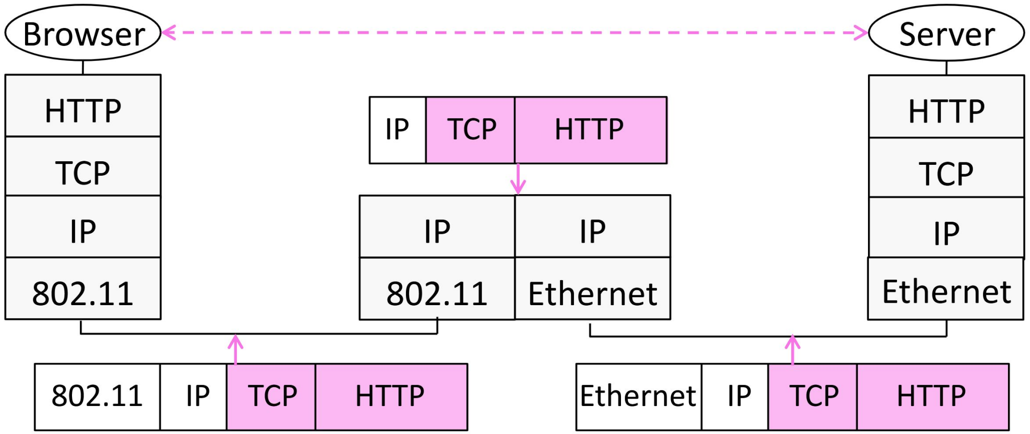
Protocols and Layers

ADVANTAGES

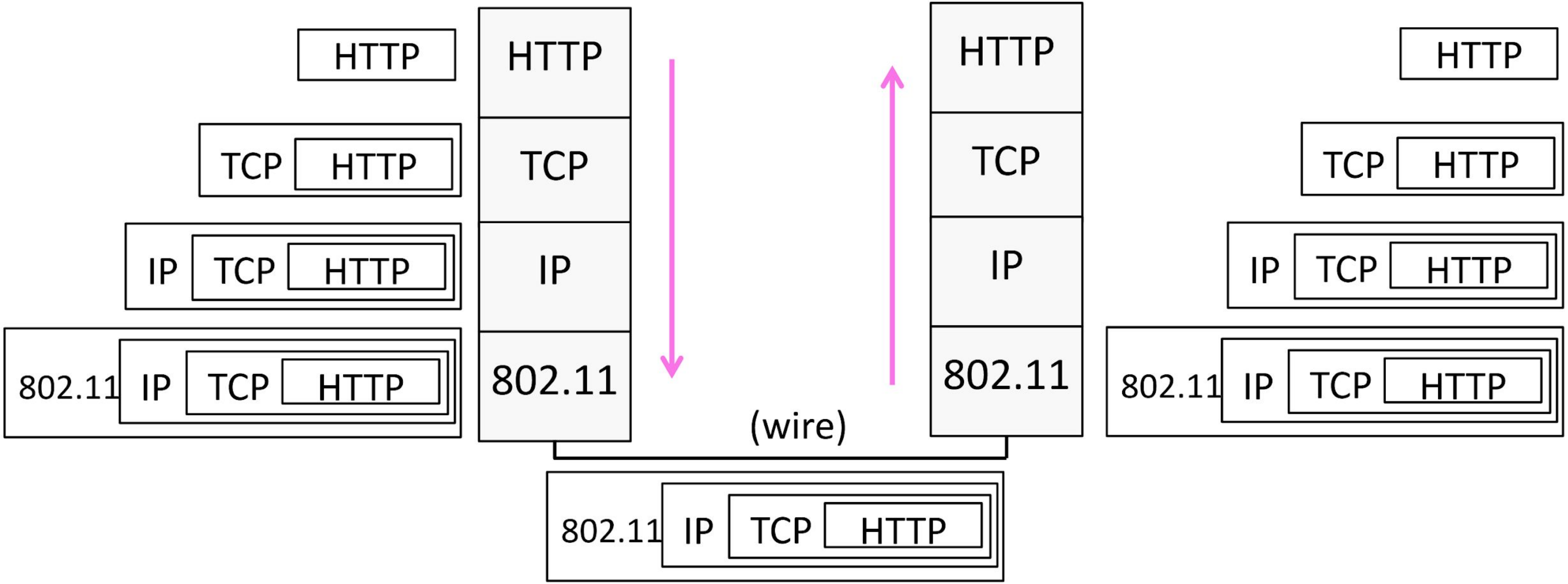
- Use information hiding to connect different systems
- Information reuse to build new protocols

DISADVANTAGES

- Adds overhead
- Hides information



Encapsulation



Motivation

- What does the network layer do?
  - Connect different networks (send packets over multiple networks)

- Why do we need the network layer?
  - Switches don’t scale to large networks
  - Switches don’t work across more than one link layer technology
  - Switches don’t give much traffic control

Network Service Models

Datagram Model

- Connectionless service
- Packets contain destination address
- Routers looks up address in its forwarding table to determine next hop
- Example: IP

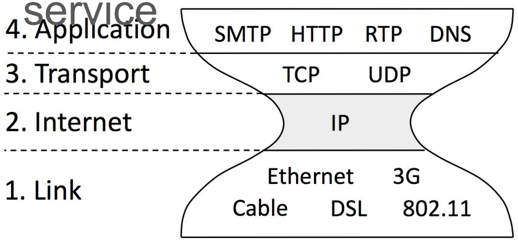
Virtual Circuits

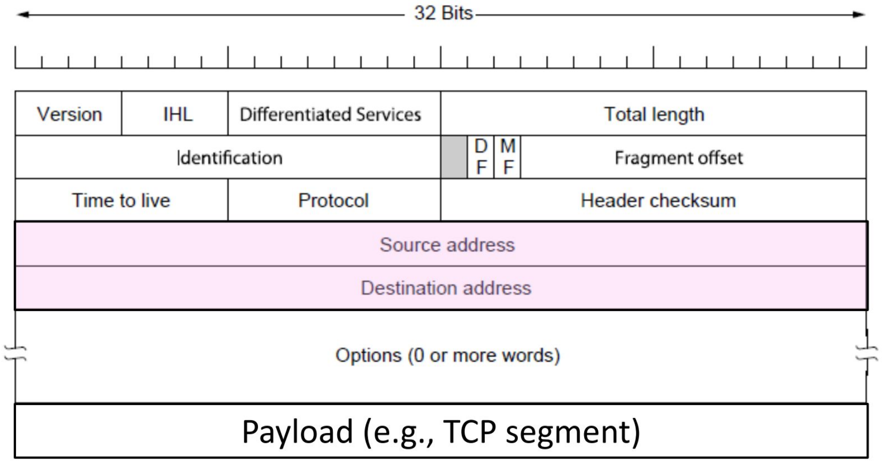
- Connection-oriented service
- Connection establishment → data transfer → connection teardown
- Packets contain label for circuit
- Router looks up circuit in forwarding table to determine next hop
- Example: MPLS

Both of them use **Store-and-Forward packet switching**

Internetworking - IP

- How do we connect different networks together?
- **IP - Internet Protocol**
- Lowest Common Denominator
  - Asks little of lower-layer networks
  - Gives little as a higher layer service



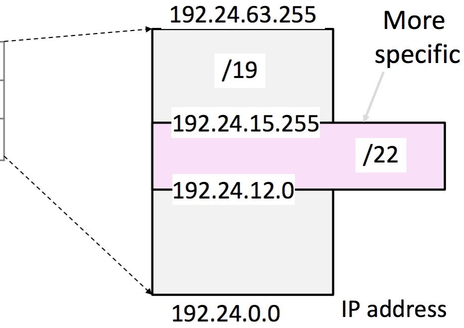


IP Addresses Prefix and Forwarding

- IP prefix   a.b.c.d/**L**
  - Represents addresses that have the same first L bits
  - e.g. 128.13.0.0/16 -> all 65536 addresses between 128.13.0.0 to 128.13.255.255
  - e.g. 18.31.0.0/32 -> 18.31.0.0 (only one address)
- **Longest Matching Prefix**
  - find the longest prefix that contains the destination address, i.e., the most specific entry

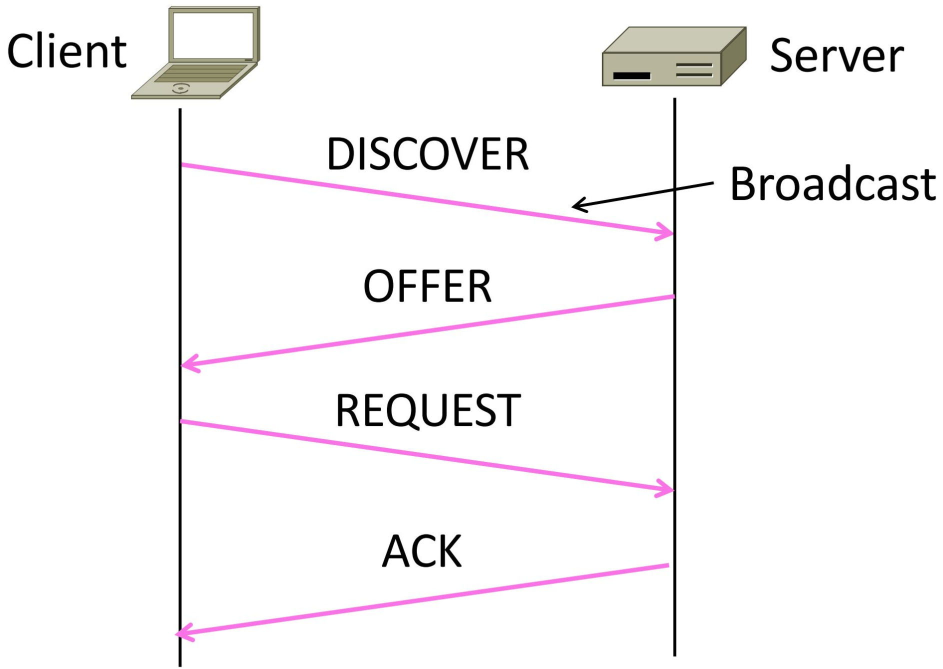
Prefix	Next Hop
192.24.0.0/19	D
192.24.12.0/22	B

192.24.6.0 → D
192.24.14.32 → B
192.24.54.0 → D



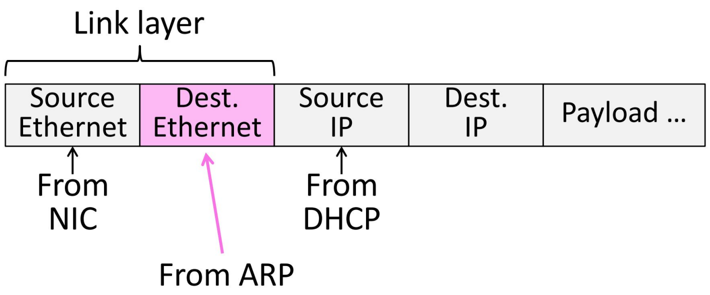
DHCP - Dynamic Host Configuration Protocol

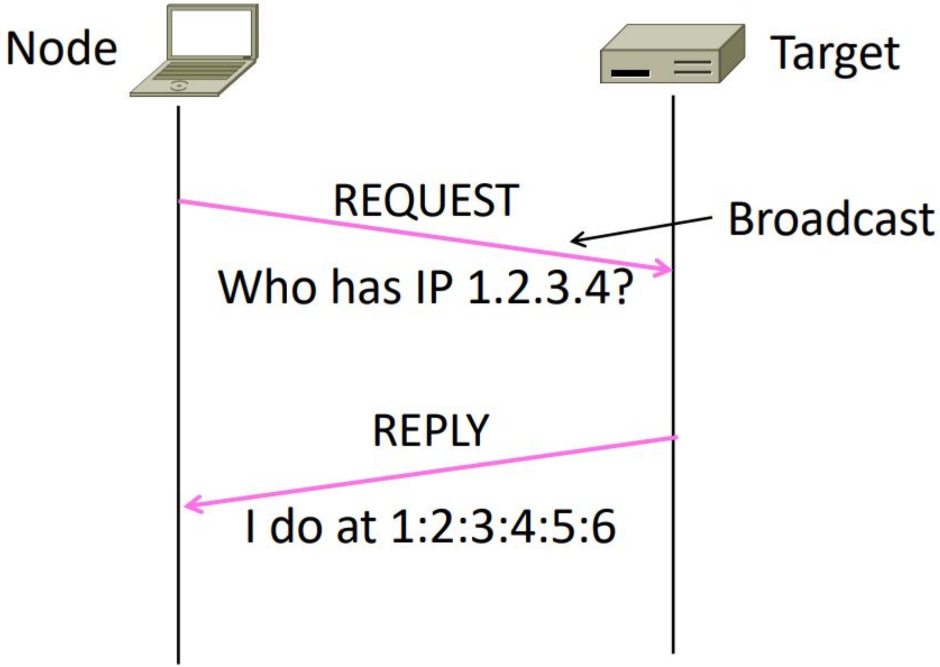
- Bootstrapping problem
- Leases IP address to nodes
- UDP
- Also setup other parameters:
    - DNS server
    - IP address of local router
    - Network prefix



ARP - Address Resolution Protocol

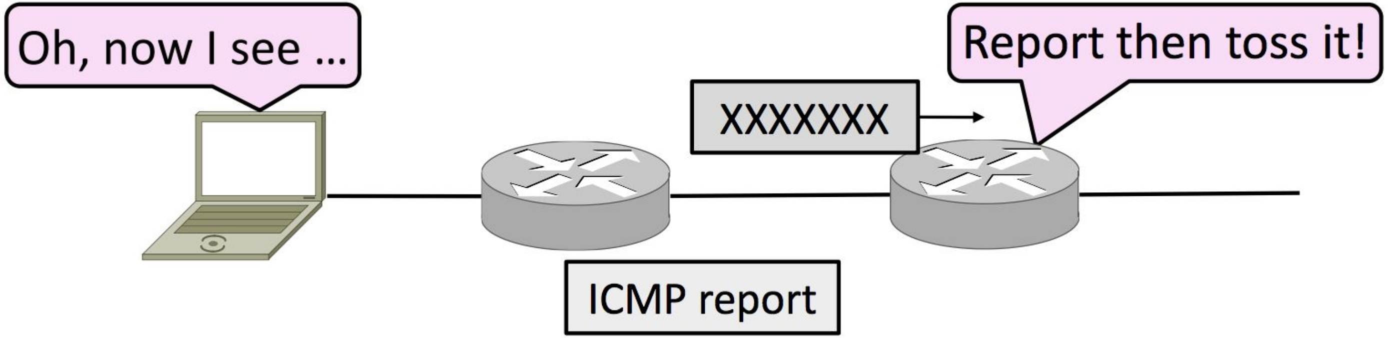
- MAC is needed to send a frame over the local link
- ARP to map an IP to MAC
- Sits on top of link layer





ICMP - Internet Control Message Protocol

- Provides error reporting and testing
- Companion protocol to IP
- Traceroute, Ping

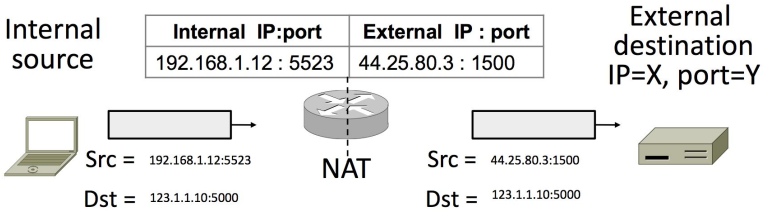


NAT - Network Address Translation

- One solution to **IPv4 address exhaustion**
- Map many private IP to one public IP, with different port number
- Pros: useful functionality (firewall), easy to deploy, etc.
- Cons: Connectivity has been broken!
- Many other cons...

What host thinks | What ISP thinks

Internal IP:port	External IP : port
192.168.1.12 : 5523	44.25.80.3 : 1500
192.168.1.13 : 1234	44.25.80.3 : 1501
192.168.2.20 : 1234	44.25.80.3 : 1502



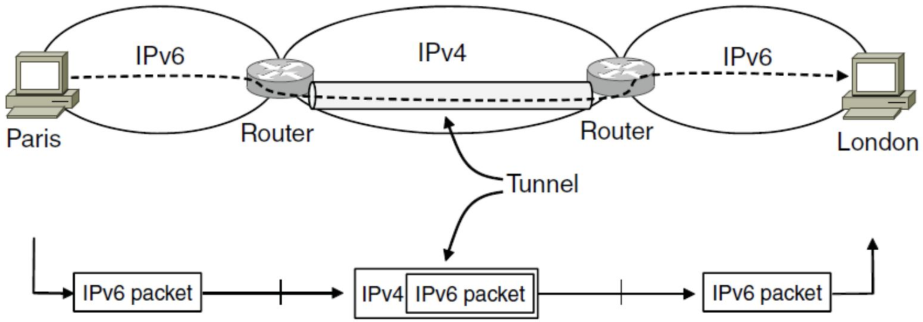
IPv6

- A much better solution to IPv4 address exhaustion
- Uses 128-bit addresses, with lots of other changes
- IPv6 version protocols: NDP -> ARP, SLAAC -> DHCP
- Problem: being incompatible with IPV4. Solution: Tunnelling

What's my IP

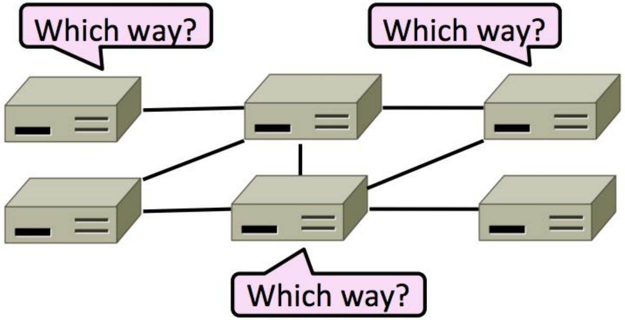
2601:602:8b00:5f0:30b3:2d19:3fe:db9e

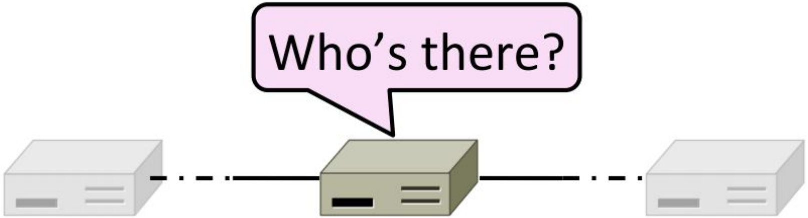
Your public IP address



Routing

- The process of deciding in which direction to send traffic
- Delivery models: unicast, broadcast, multicast, anycast
- Goals: correctness, efficient paths, fair paths, fast convergence, scalability
- Rules: decentralized, distributed setting

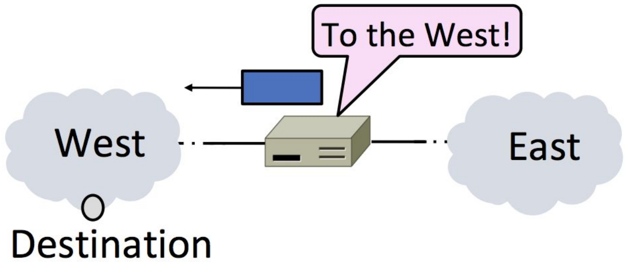




Techniques to Scale Routing

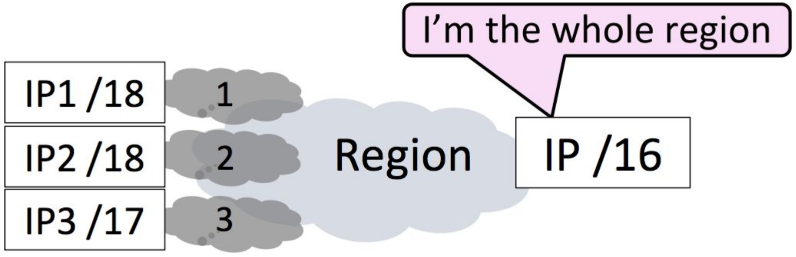
Hierarchical Routing

- Route first to the region, then to the IP prefix within the region



IP Prefix Aggregation and Subnets

- Adjusting the size of IP prefixes
  - Internally split one large prefix
  - Externally join multiple IP prefixes



Best Path Routing

Distance Vector Routing

Each node maintains a vector of distances (and next hops) to all destinations.

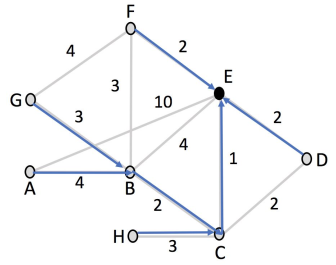
Sometimes doesn't perform very well: count-to-infinity scenario

Link State Routing (widely used)

Phase 1. **Topology Dissemination**: Nodes flood topology

Phase 2. **Route Computation**: running Dijkstra algorithm (or equivalent)

Algorithm details available in lecture slides

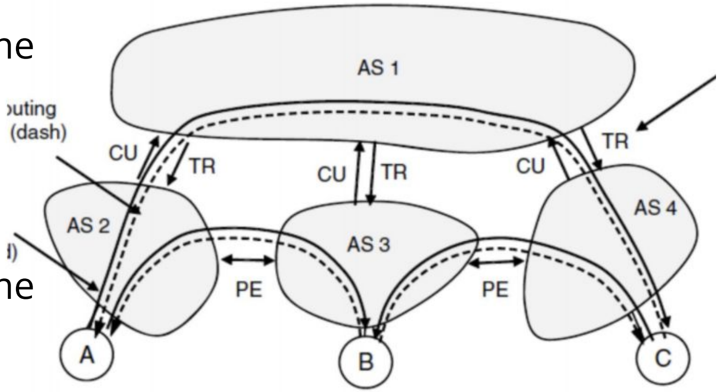


BGP - Border Gateway Protocol

- Internet-wide routing between ISPs (ASes)
  - Each has their own policy decisions
- Peer and Transit (Customer) relationship
- Border routers of ISPs announce BGP routes only to other parties who may use those paths.
- Border routers of ISPs select the best path of the ones they hear in any, non-shortest way

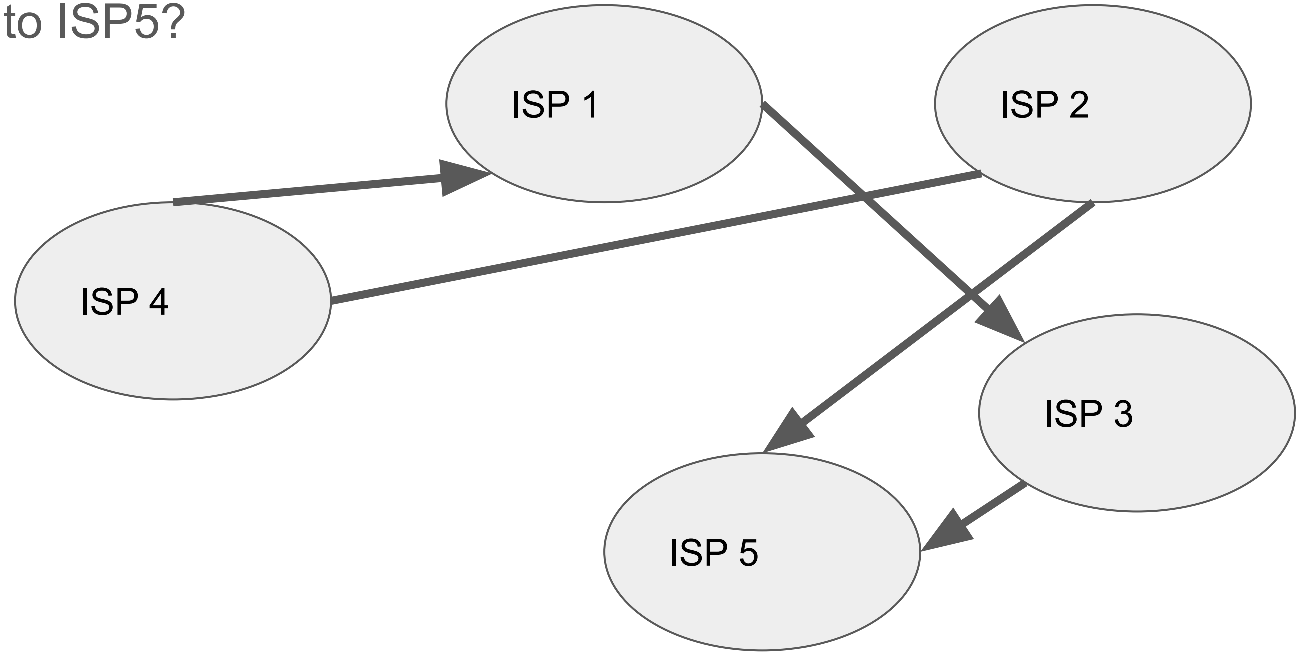
BGP example

- Transit (ISP & Customer)
  - ISP announce everything it can reach to its customer
    - AS1 to AS2: you can send packet to AS4 through me
  - Customer ISP only announce its customers to ISP
    - AS2 to AS1: you can send packet to A through me
- Peer (ISP 1 & ISP 2)
  - ISP 1 only announces its customer to ISP 2
    - AS2 to AS3: you can send packet to A through me



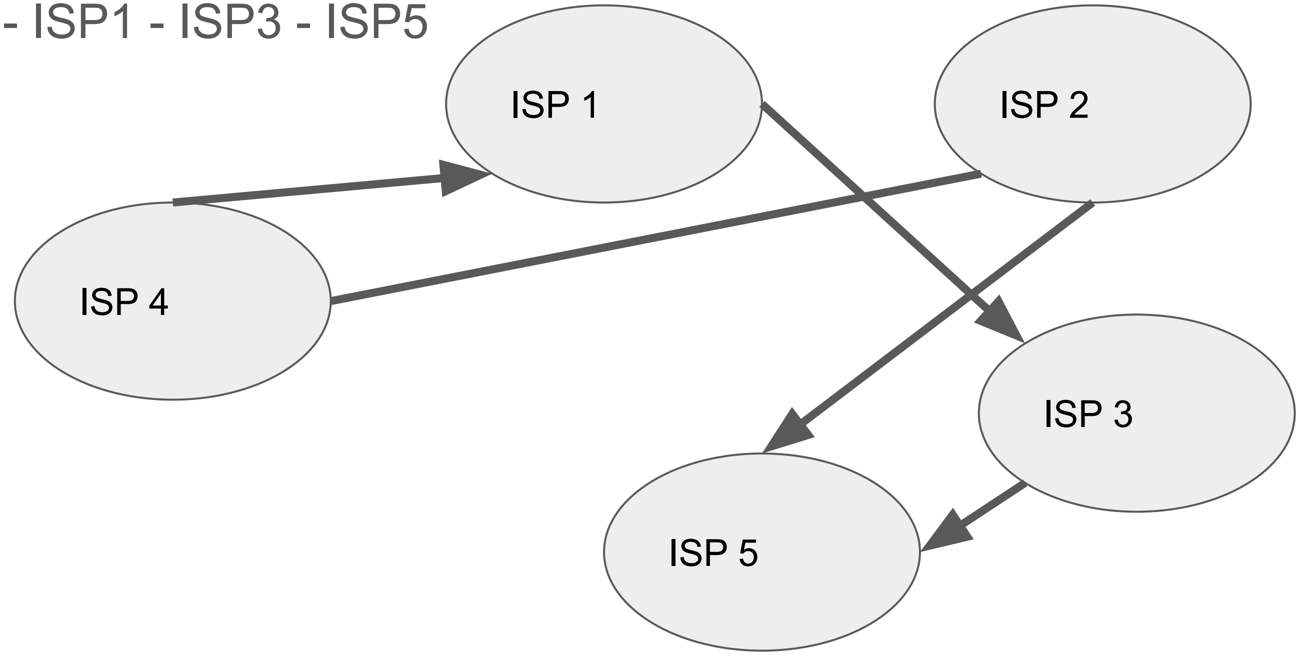
Customer → Provider

ISP4 to ISP5?



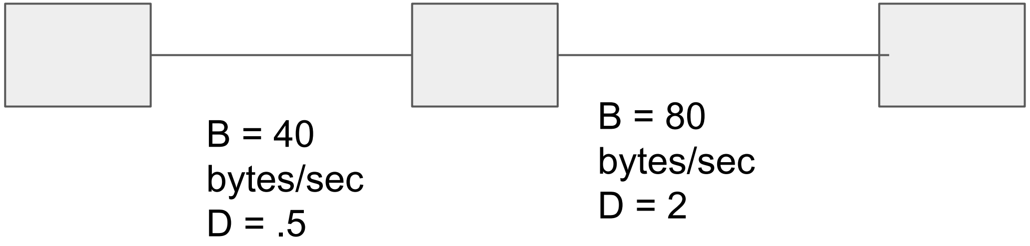
Customer → Provider

ISP4 - ISP1 - ISP3 - ISP5



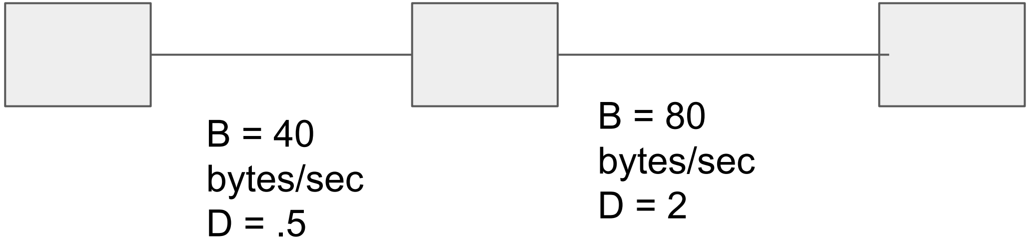
TCP window size?

Packet is 10 bytes



TCP window size?

Packet is 10 bytes



(40(2+.5)*2) / 10 = 20 packet window size