

Week 5 – Network Layer

COMP90007
Internet Technologies

Chien Aun Chan

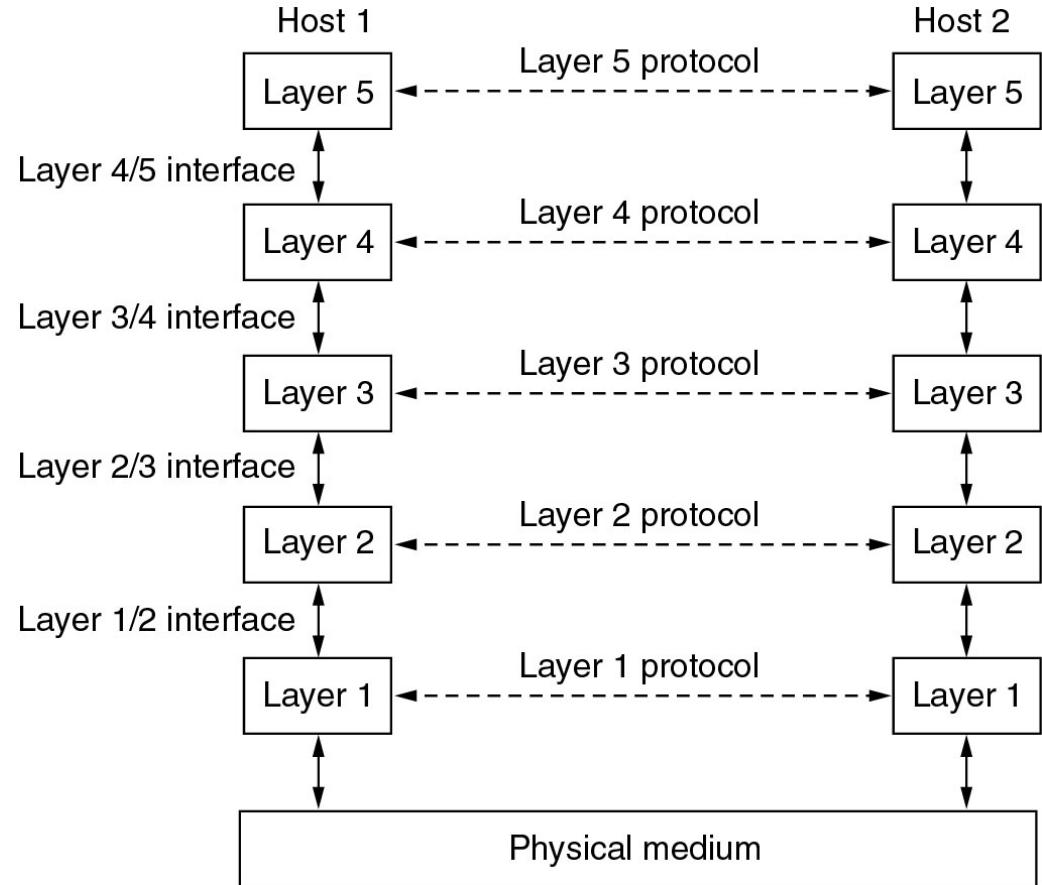
Network Software

Protocol Hierarchies

Connecting different networks
(internetworking)

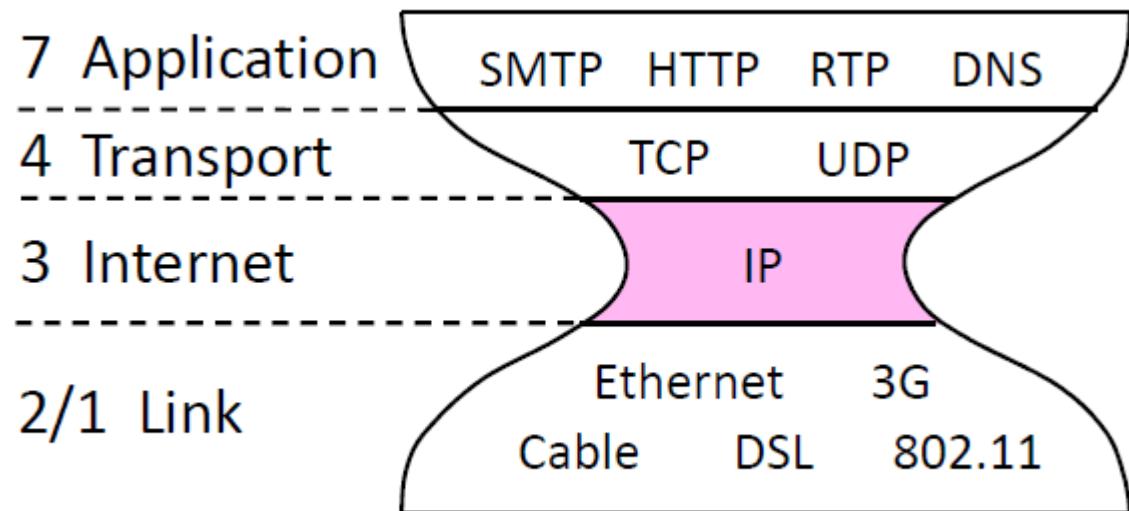
Framing, reliability and
flow control (direct conn.)

Different Cables, wireless
signalling digital to analogue



Outline

- Network Layer Design Issues
- Packet Forwarding
- Internetworking
- IPv4
- Routing Algorithms

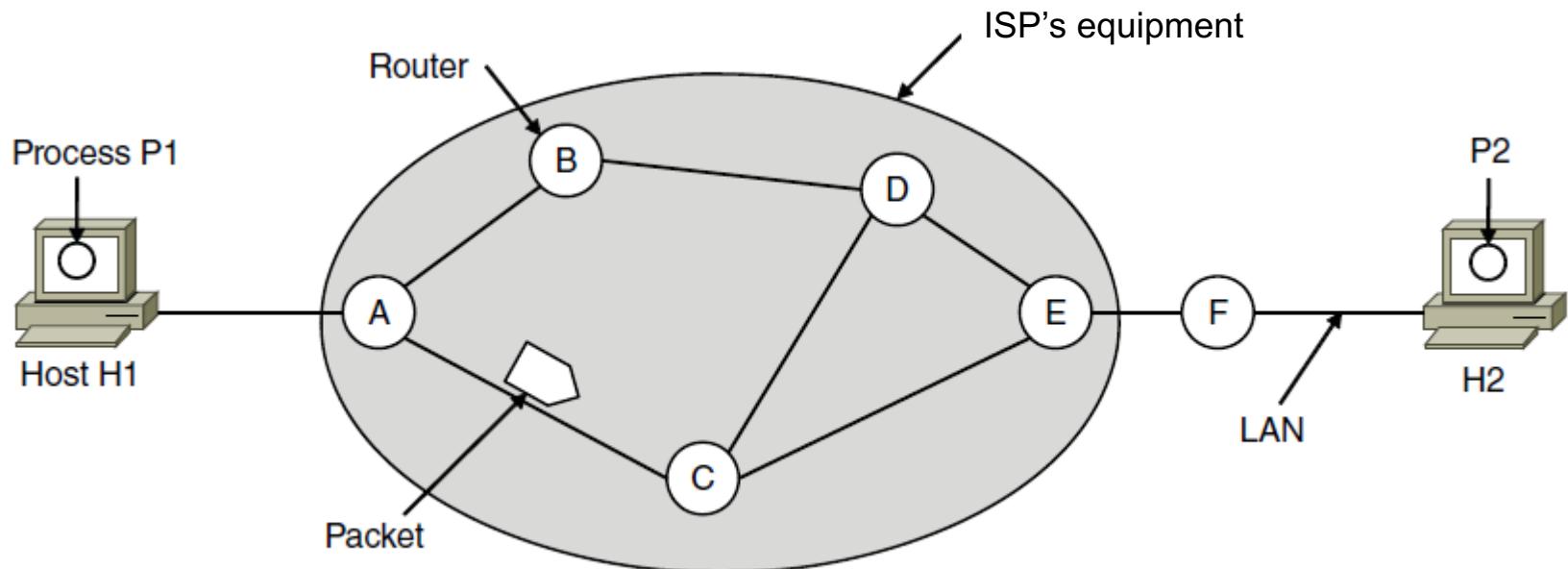


Network Layer Design Issues

- Which services are provided to the transport layer ?
- How do design issues affect performance ?
 - Store-and-forward packet switching
 - Connectionless service – datagrams
 - Connection-oriented service – virtual circuits
- Comparison of virtual-circuits and datagrams

Store and Forward Packet Switching

- Hosts generate packets and injects into the network
- Routers treat packets as messages, receiving (storing) them and then forwarding them based on how the message is addressed
- Router routes packets through the network



Services Provided to the Transport Layer

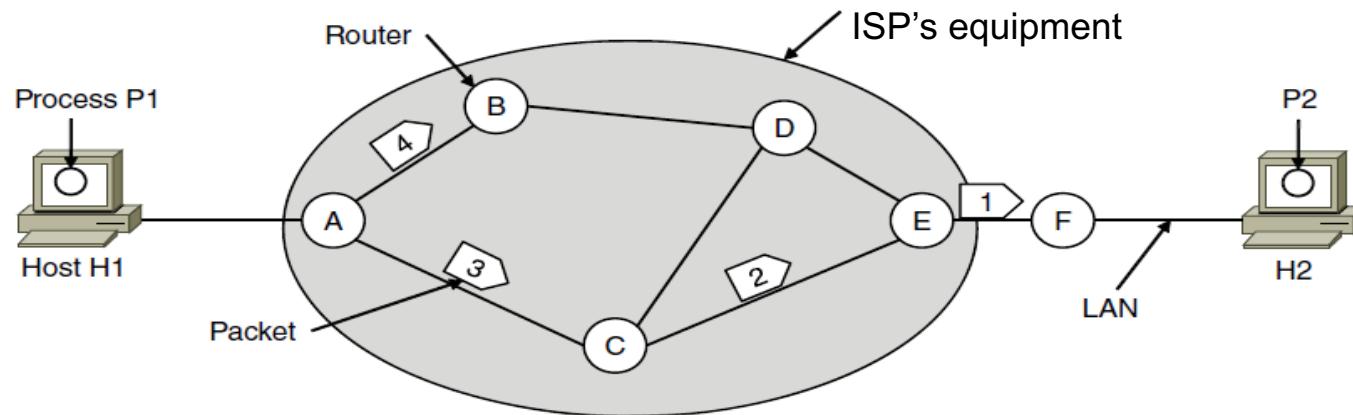
- Design goals:
 - Services should be independent of router technologies
 - Transport layer should be shielded from number, type and topology of routers
 - Network addressing should use a uniform numbering plan (network identifier)

Types of Services

1. **Connectionless:** Packets (datagrams) injected into subnet independently and packets individually routed to destination
 - ❑ Internet: move packets in a potentially unreliable subnet - QoS is not easily implemented
 - ❑ Flow and error control done by the hosts
2. **Connection-oriented:** Packets travelling between destinations all use the same route
 - ❑ Telco: guarantee reliability of subnet - QoS is important

Routing within a datagram subnet

- Post office model: packets are routed individually based on destination addresses in them
- Packets can take different paths
- E.g., P1 sends a long message to P2



A's table (initially)

| | |
|---|---|
| A | ☒ |
| B | B |
| C | C |
| D | B |
| E | C |
| F | C |

A's table (later)

| | |
|---|---|
| A | ☒ |
| B | B |
| C | C |
| D | B |
| E | B |
| F | B |

C's Table

| | |
|---|---|
| A | A |
| B | A |
| C | ☒ |
| D | E |
| E | E |
| F | E |

E's Table

| | |
|---|---|
| A | C |
| B | D |
| C | C |
| D | D |
| E | ☒ |
| F | F |

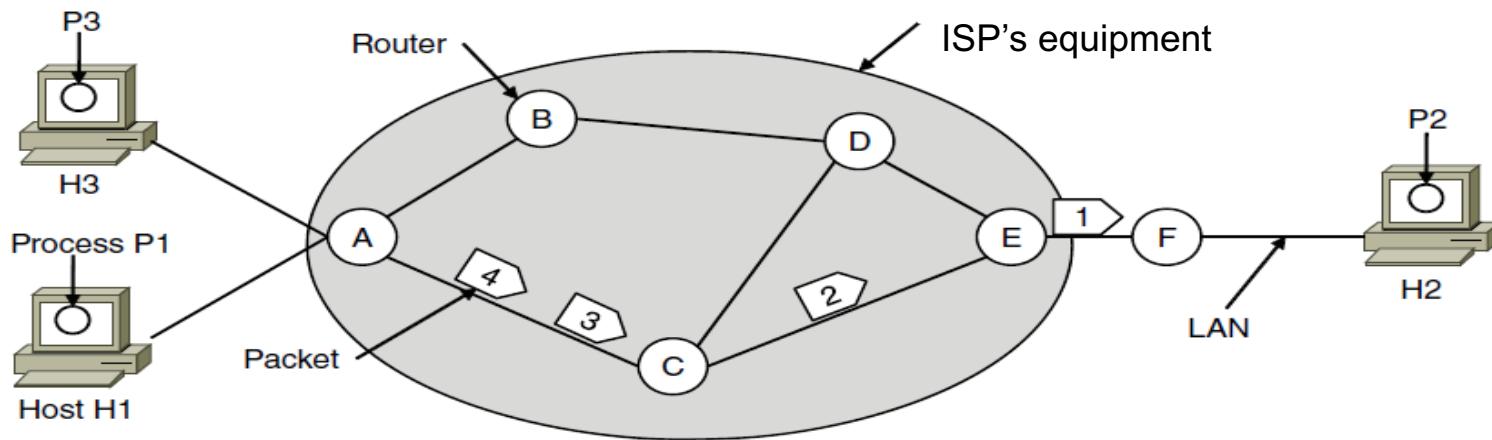
Routing table (can be fixed, can change over time)

Routing algorithm – manages the routing table

Routing within a virtual-circuit subnet

- Model is like telephone network

- Packets are routed through virtual circuits (created earlier) based on tag number (not full address but unique at a given link) in them
- Packets take the same path (to avoid having to choose a new route for every packet sent)
- E.g., Multi-protocol Label Switching Network (to provide QoS) – 20 bit label or conn. Identifier



connection identifier

A's table

| | | | |
|----|---|-----|---|
| H1 | 1 | C | 1 |
| In | | Out | |

C's Table

| | | | |
|---|---|---|---|
| A | 1 | E | 1 |
|---|---|---|---|

E's Table

| | | | |
|---|---|---|---|
| C | 1 | F | 1 |
|---|---|---|---|

Differences in Virtual Circuit and Datagram Subnets

| Issue | Datagram network | Virtual-circuit network |
|---------------------------|--|--|
| Circuit setup | Not needed | Required |
| Addressing | Each packet contains the full source and destination address | Each packet contains a short VC number |
| State information | Routers do not hold state information about connections | Each VC requires router table space per connection |
| Routing | Each packet is routed independently | Route chosen when VC is set up; all packets follow it |
| Effect of router failures | None, except for packets lost during the crash | All VCs that passed through the failed router are terminated |
| Quality of service | Difficult | Easy if enough resources can be allocated in advance for each VC |
| Congestion control | Difficult | Easy if enough resources can be allocated in advance for each VC |

Compromises in VC and Datagram Subnets

- Compromises:
 - Memory vs bandwidth
 - VC's require more space in router memory, and save potential overhead in full addressing of each packet
 - Setup time vs address parsing time
 - VC's require setup time and resources, but packet transmission is very fast
 - Amount of memory
 - datagram subnets require large tables of every possible destination routes, whereas VC does not. Really?
 - QoS and congestion avoidance
 - VC's can use a tighter QoS - able to reserve CPU, bandwidth and buffer in advance
 - Longevity
 - VC's can exist for a long time eg Permanent VC's
 - Vulnerability
 - VC's particularly vulnerable to hardware/software crashes - all VC's aborted and no traffic until they are rebuilt; datagram uses an alternative route

Outline

- Network Layer Design Issues
- Packet Forwarding
- Internetworking
- IPv4
- Routing Algorithms

Internetworking

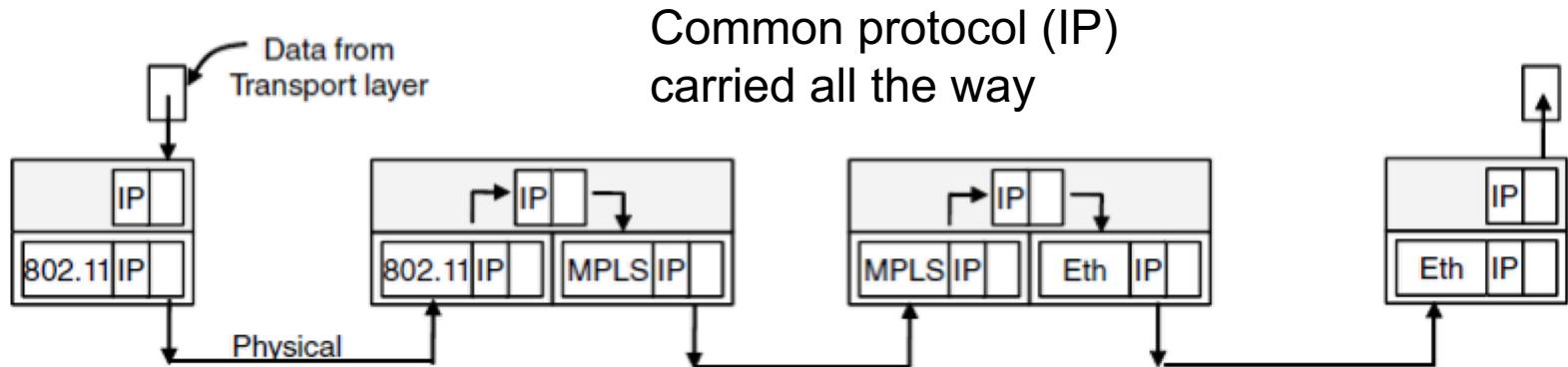
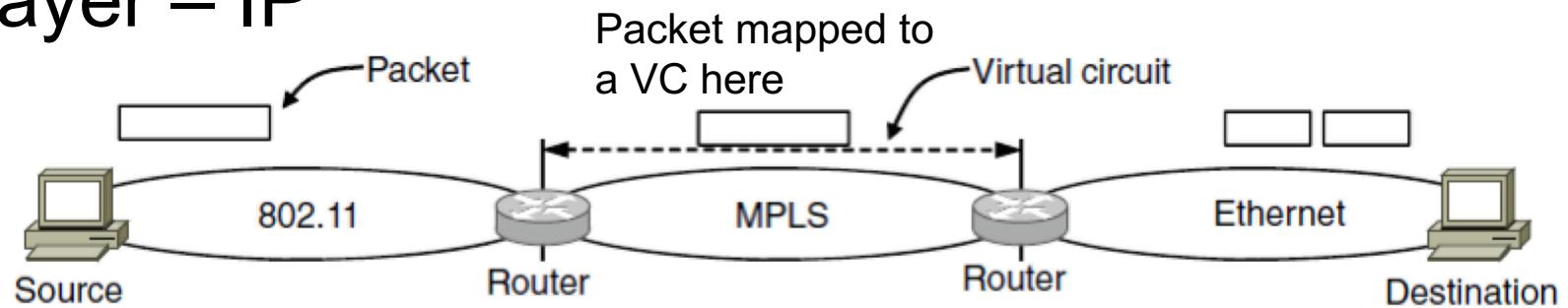
- Until now we assumed a single homogeneous network
- Internetworking joins multiple, different networks into a single larger network
- Issues when connecting networks:
 - Different network types and protocols
 - Different motivations for network choices
 - Different technologies at both hardware and software levels

Differences at the Network Layer

| Item | Some Possibilities |
|--------------------|---|
| Service offered | Connectionless versus connection oriented |
| Addressing | Different sizes, flat or hierarchical |
| Broadcasting | Present or absent (also multicast) |
| Packet size | Every network has its own maximum |
| Ordering | Ordered and unordered delivery |
| Quality of service | Present or absent; many different kinds |
| Reliability | Different levels of loss |
| Security | Privacy rules, encryption, etc. |
| Parameters | Different timeouts, flow specifications, etc. |
| Accounting | By connect time, packet, byte, or not at all |

How Different Networks are Connected

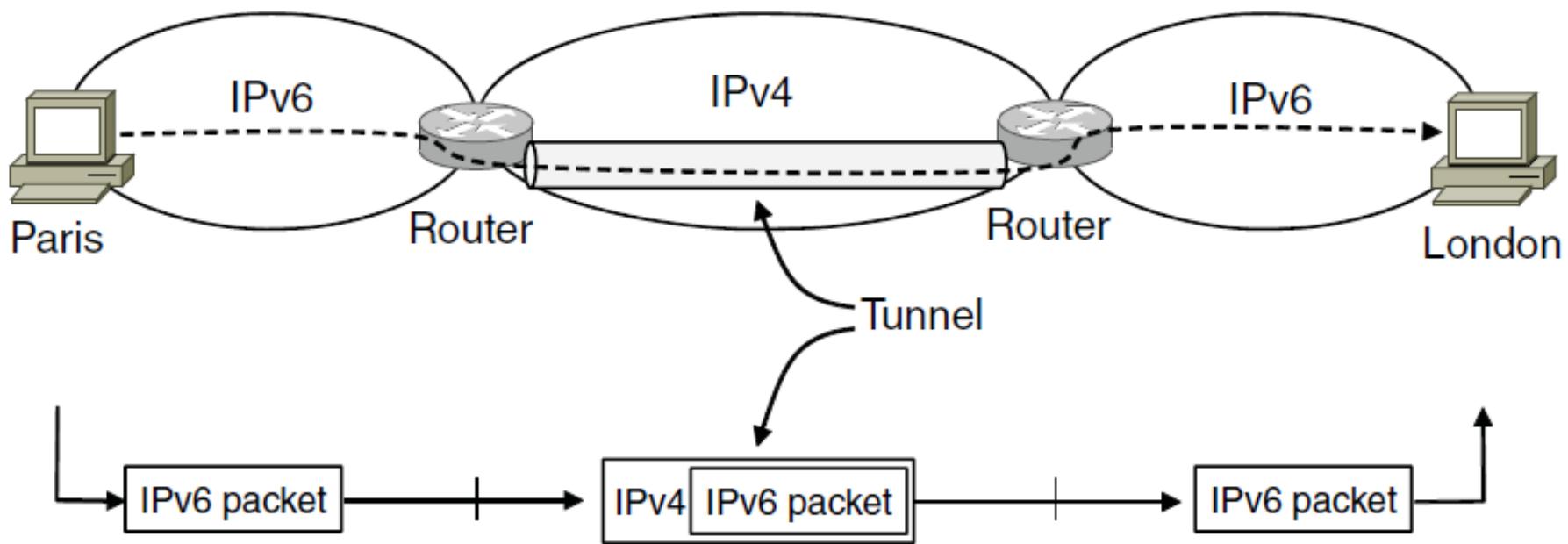
- Internetworking based on a common network layer – IP



Tunneling

- Tunneling is used when the source and destination are on the same network, but there is a different network in between.
 - Source Packets are encapsulated over the packets in the connecting network

Tunneling IPv6 packets through IPv4

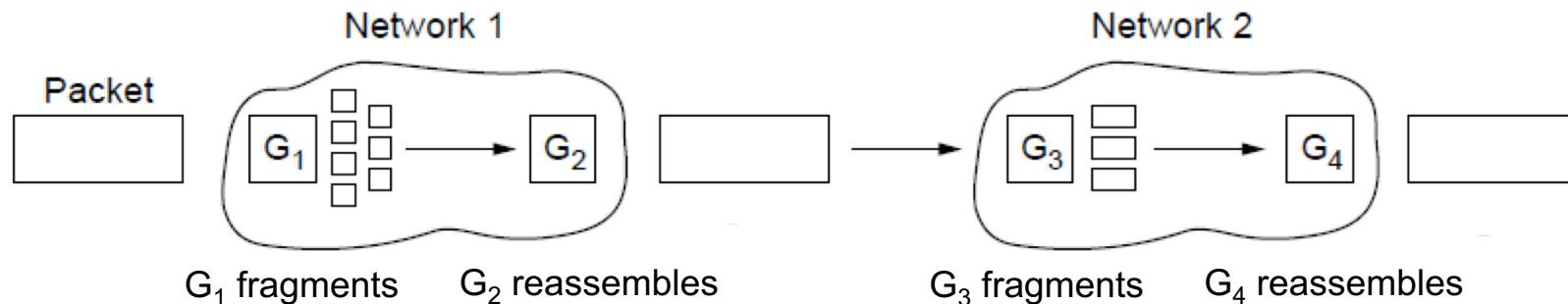


Fragmentation

- All networks have a maximum size for packets, could be motivated by:
 - Hardware
 - Operating system
 - Protocols
 - Standards compliance
 - Desire to reduce transmissions due to errors
 - Desire for efficiency in communication channel
- **Fragmentation** (division of packets into fragments) allows network gateways to meet size constraints

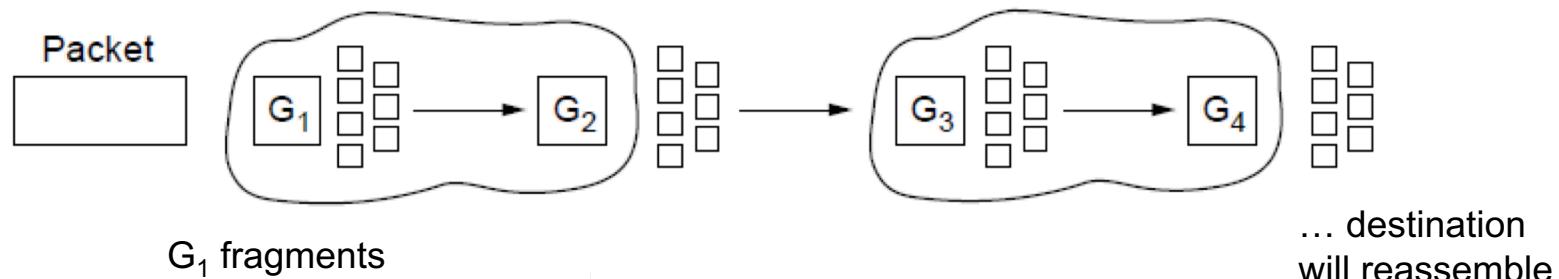
Types of Fragmentation

- Large packets need to be routed through a network whose maximum packet size is too small.
- Fragmentation and Reassembly is a solution.



Transparent – packets fragmented / reassembled in each network

- Route constrained, more work



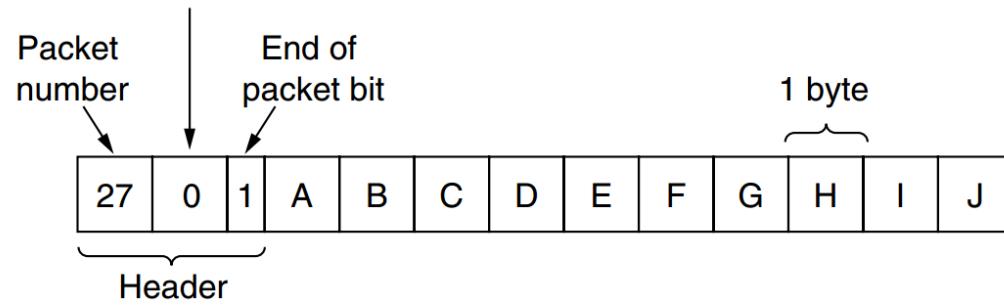
Non-transparent – fragments are reassembled at destination

- Less work (IP works this way) – packet number, byte offset, end of packet flag

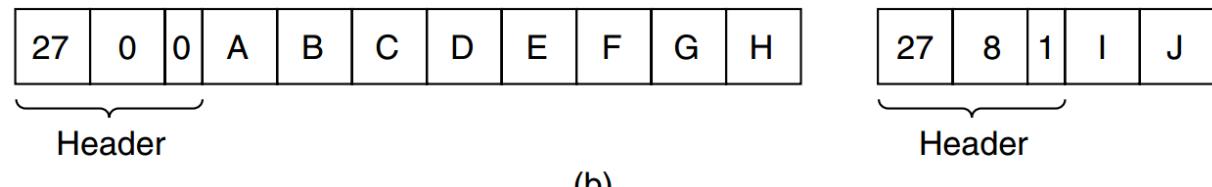
Example: IP Style Fragmentation

Original packet:
(10 data bytes)

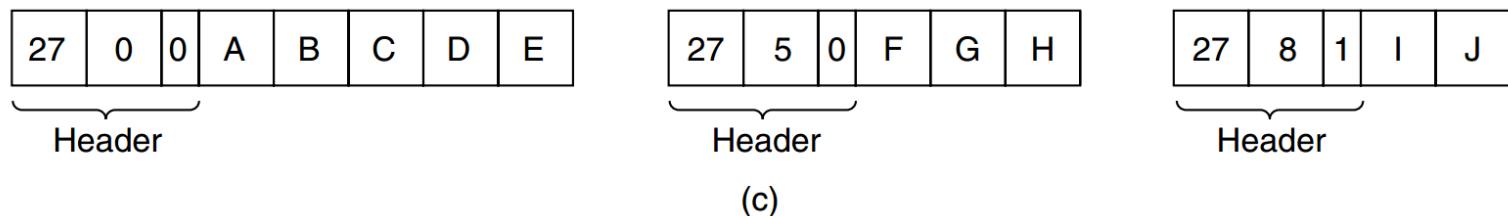
Number of the first elementary fragment in this packet



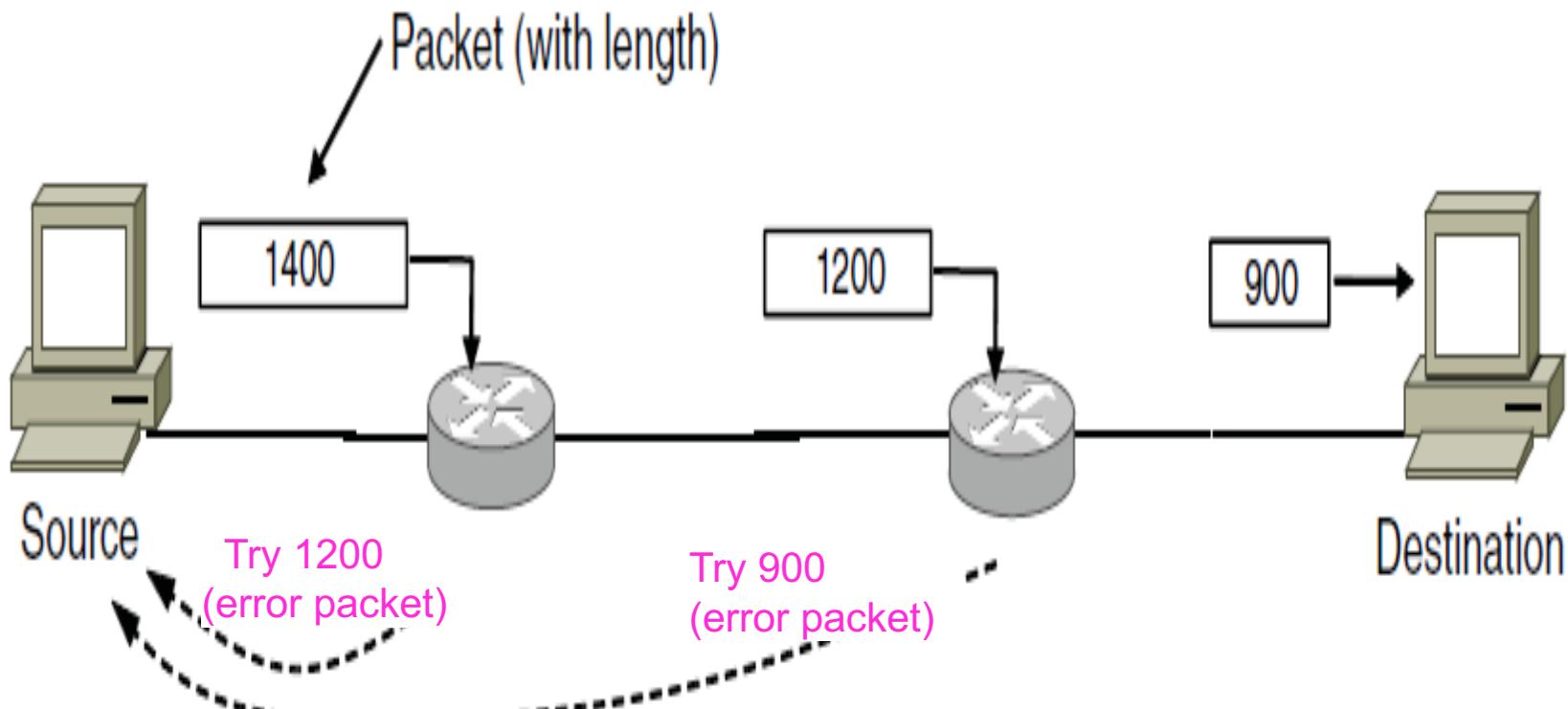
Fragmented:
(to 8 data bytes)



Re-fragmented:
(to 5 bytes)



Path MTU Discovery: Alternative to Fragmentation



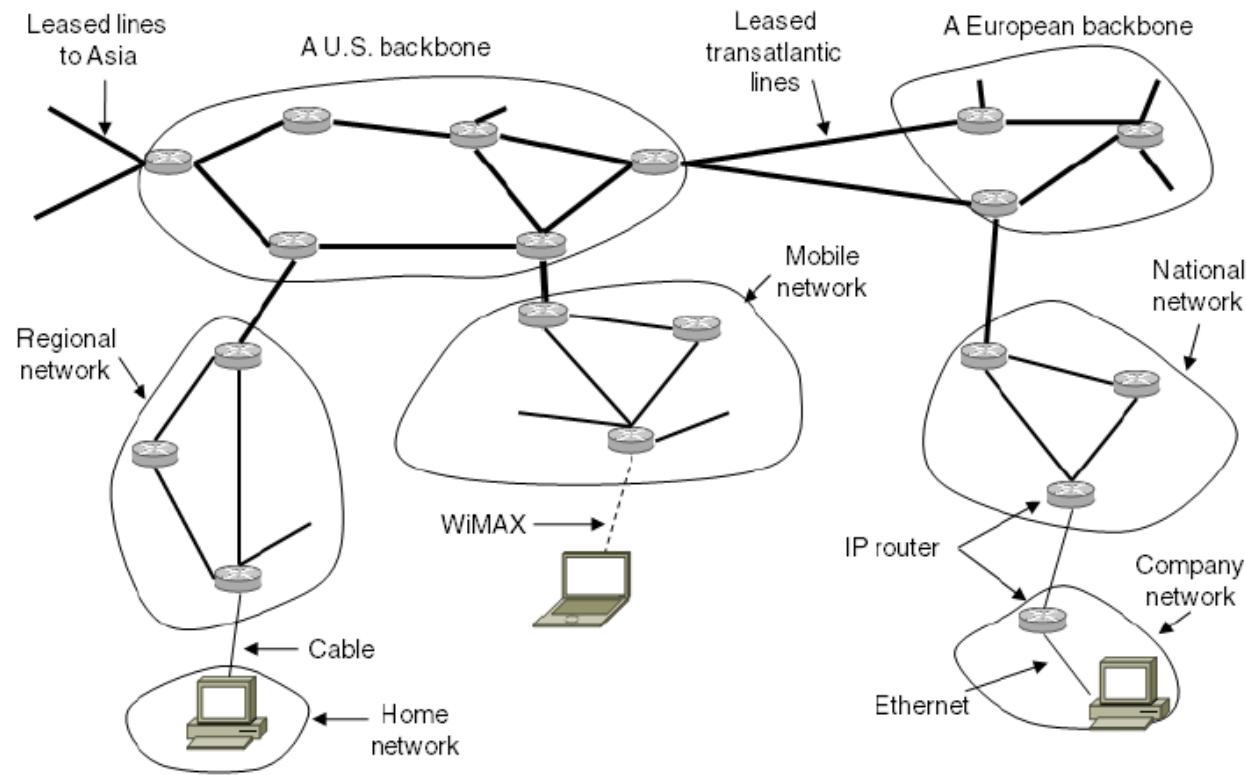
Advantage: The source now knows what length packet to send but if the routes and path MTU change, new error packets will be triggered and the source will adapt to the new path

Principles of Internet Design

- RFC 1958 “Architectural Principles of the Internet” (supplementary reading)
- Core Principles
 - Make sure it works
 - Keep it simple
 - Make clear choices
 - Exploit modularity
 - Expect heterogeneity
 - Avoid static options and parameters
 - Choose a good, but not necessarily perfect design
 - Be strict in sending and tolerant in receiving
 - Consider scalability
 - Consider performance vs costs

Network Layer in the Internet

- Internet is an interconnected collection of many networks of Autonomous systems that is held together by the IP protocol

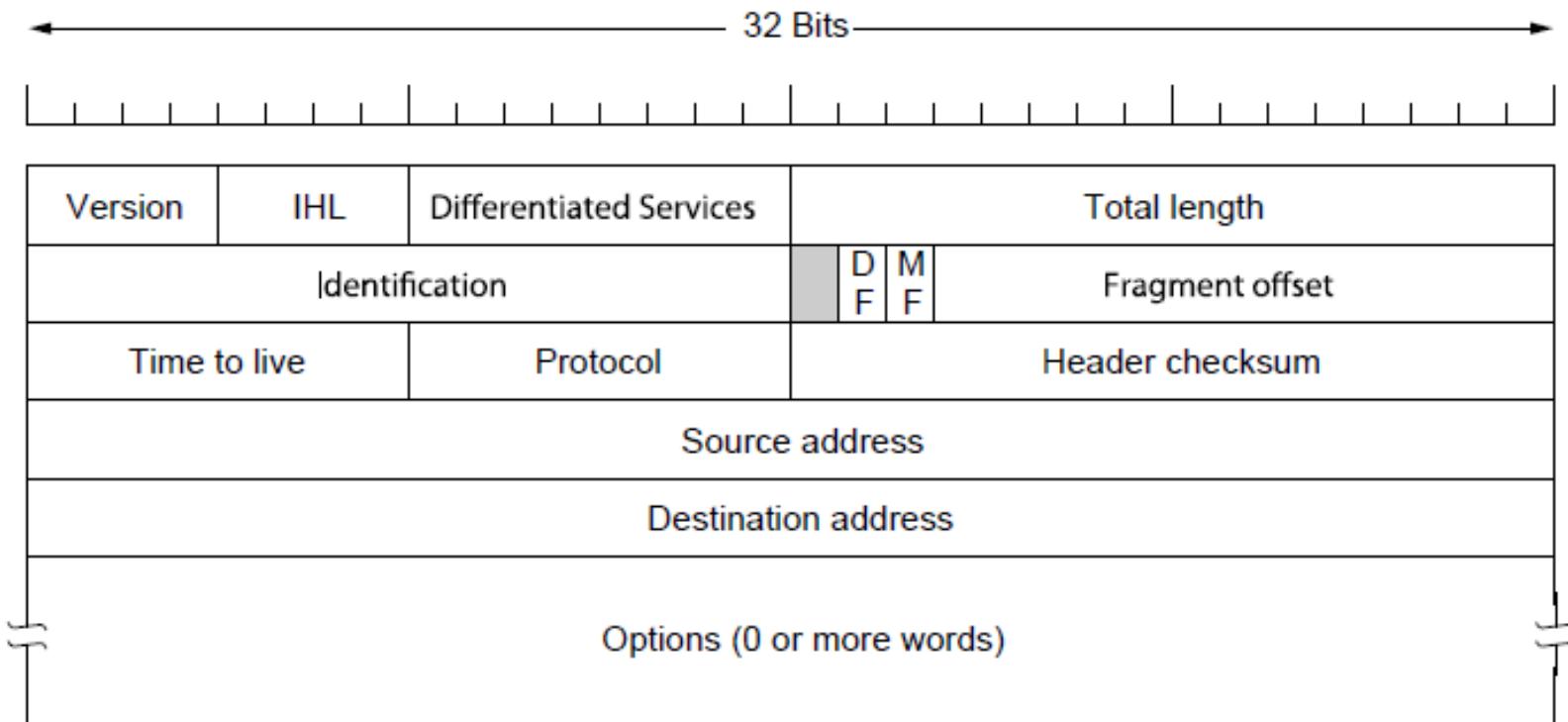


Internet Protocol (IP)

- The glue that holds the whole Internet together is the network layer protocol, IP (Internet Protocol)
- Provides a “best-effort” service to route datagrams from source host to destination host
- These hosts may be
 - On **same** network
 - On **different** networks
- Each network is called an **Autonomous System (AS)**

IPv4 Frame Structure Illustrated

- IPv4 (Internet Protocol) header is carried on all packets and has fields for the key parts of the protocol



IPv4 Frame Structure in Detail

- IPv4 Frame consists of a header and some text
- header is 20 byte fixed part + variable length optional part
- Version: IPv4 or IPv6
- IHL: Header Length – in 32bits units, min 5 and max is 15
- Type: differentiates different classes of service
- Total Length: header and payload, maximum length 65535 bytes
- Identification: allows host to determine which datagram the new fragment belongs to - all fragments of same datagram have same ID
- DF: Don't Fragment byte
 - Originally, it was intended to support hosts incapable of putting the pieces back together again.
 - Now it is used as part of the process to discover the path MTU, which is the largest packet that can travel along a path without being fragmented

IPv4 Frame Structure in Detail (continued)

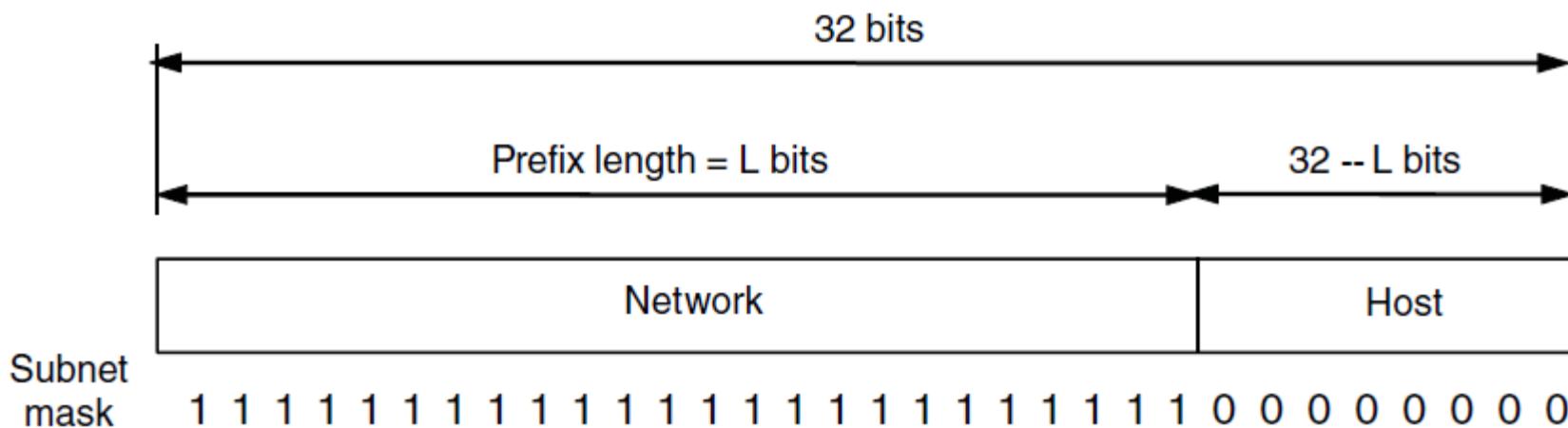
- MF: More Fragment byte - are there more or is this the last one ?
- Fragment offset: where in the datagram the current fragment belongs
- TTL: limits packet lifetimes - hops or seconds
- Protocol: TCP, UDP, others ...
- Header Checksum: verifies the header only
- Source Address: IP - host/network
- Destination Address: IP - host/network
- Options: eg security, strict vs loose source routing, record route, timestamp

IP Addresses

Example:

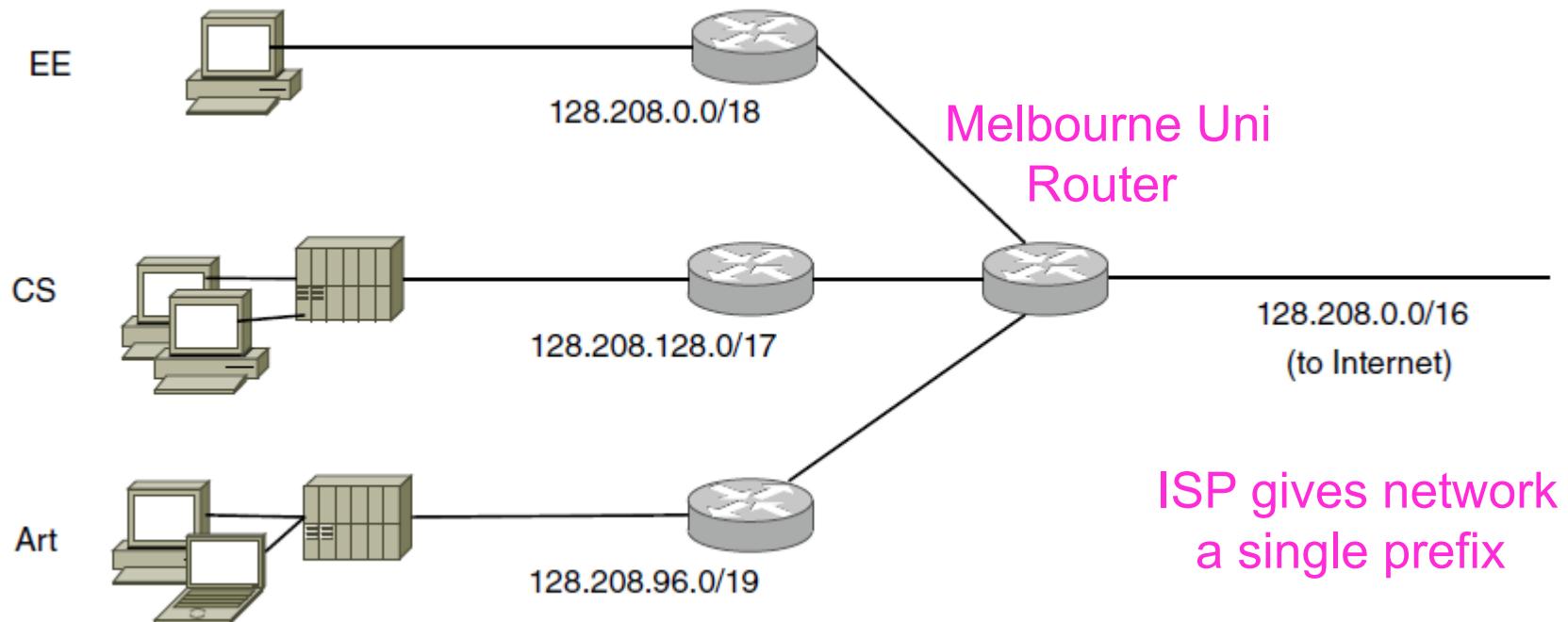
| | | | | | | | |
|-------|-------|-------|-------|-------|-------|-------|-------|
| 2^7 | 2^6 | 2^5 | 2^4 | 2^3 | 2^2 | 2^1 | 2^0 |
| 128 | 64 | 32 | 16 | 8 | 4 | 2 | 1 |
| 0 | 0 | 0 | 1 | 0 | 0 | 1 | 0 |

- Addresses are allocated in blocks called prefixes
 - Prefix is determined by the network portion
 - IP addresses are written in dotted decimal notation
 - Written lowest address/length, e.g., 18.0.31.0/24
- Overall IP allocation responsibility of Internet Corporation for Assigned Names and Numbers (ICANN) by delegation to IANA and Regional Internet Registries (RIR's)



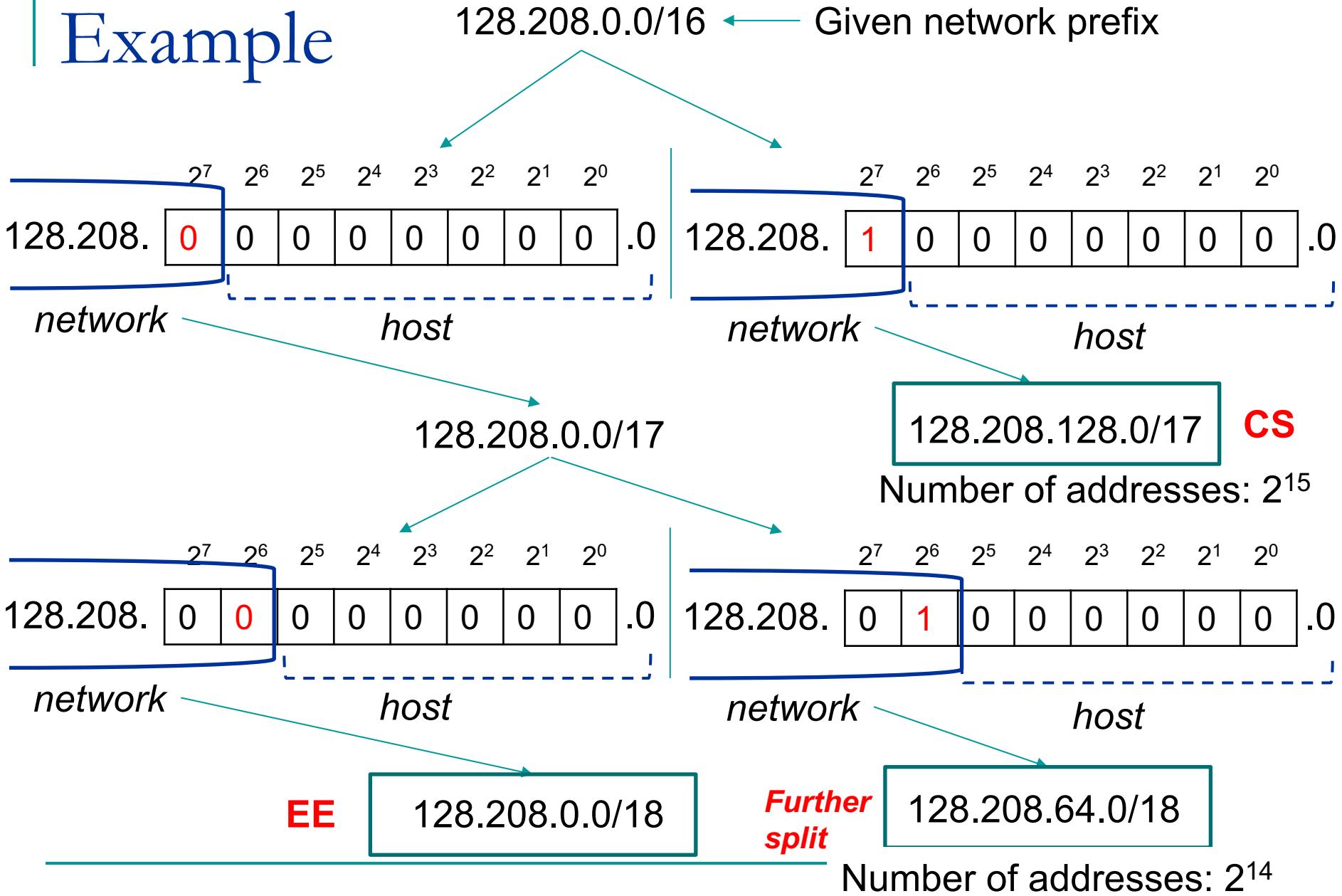
Subnets

- Subnetting allows networks to be split into several parts for internal uses whilst acting like a single network for external use
 - Looks like a single prefix outside the network

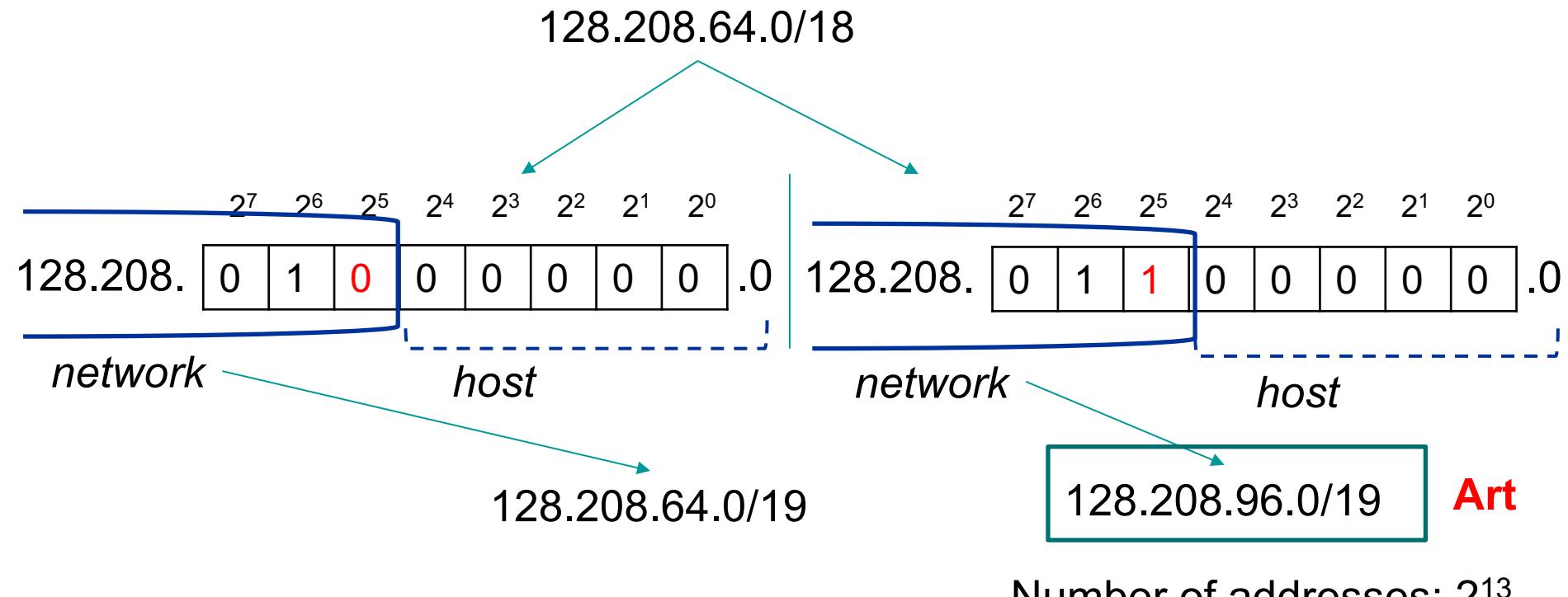


Network divides it into subnets internally

Example



Example (cont.)



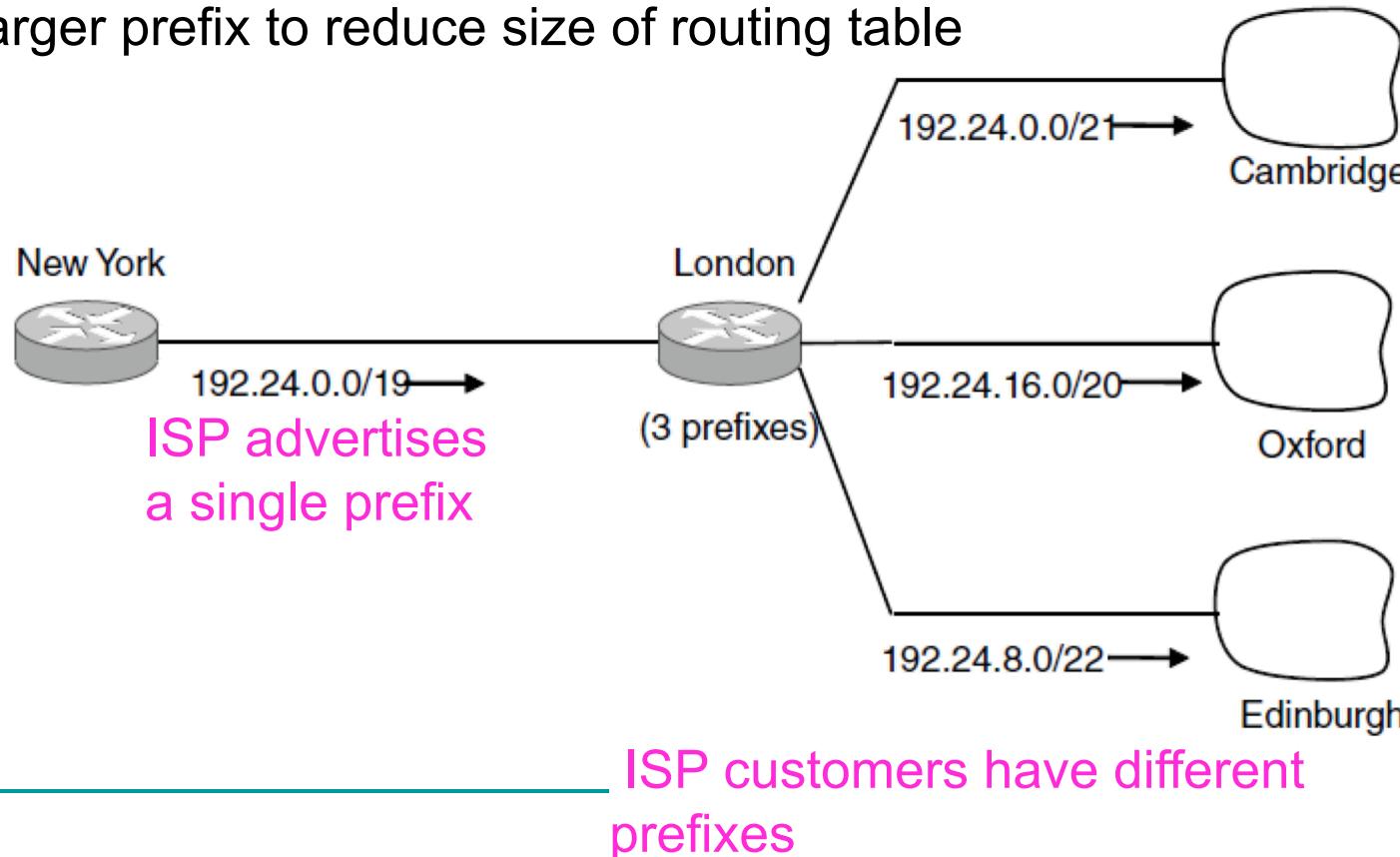
IP Addressing and Routing Tables

- Routing tables are typically based around a triplet:
 - IP Address
 - Subnet Mask
 - Outgoing Line (physical or virtual)
- Eg: A row of a routing table:

| Prefix addr | Subnet Mask | Interface |
|-------------|---------------|-----------|
| 203.32.8.0 | 255.255.255.0 | Eth 0 |

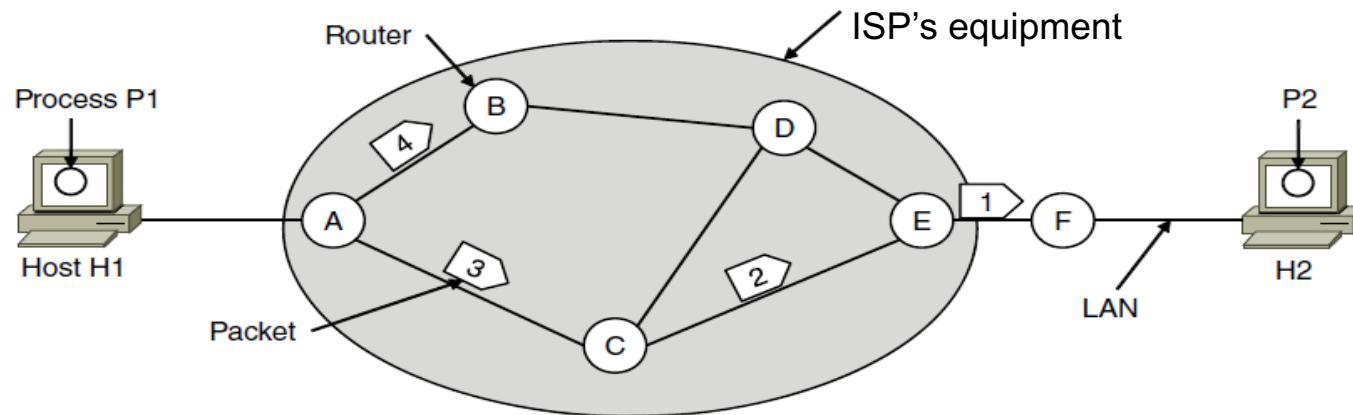
Aggregation of IP addresses

- Backbone router connecting networks around the world → 300k networks
- So we search each line for each incoming packet?
- Aggregation – Process of joining multiple IP prefixes into a single larger prefix to reduce size of routing table



Reminder: Routing within a datagram subnet

- Post office model: packets are routed individually based on destination addresses in them
- Packets can take different paths



A's table (initially)

| | |
|---|---|
| A | X |
| B | B |
| C | C |
| D | B |
| E | C |
| F | C |

A's table (later)

| | |
|---|---|
| A | X |
| B | B |
| C | C |
| D | B |
| E | B |
| F | B |

C's Table

| | |
|---|---|
| A | A |
| B | A |
| C | X |
| D | E |
| E | E |
| F | E |

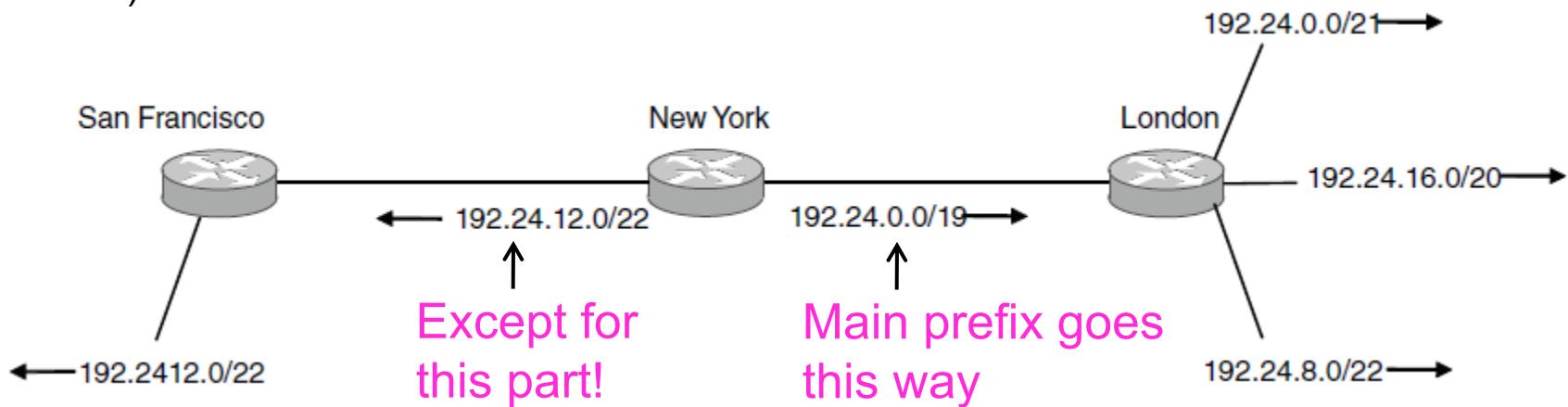
E's Table

| | |
|---|---|
| A | C |
| B | D |
| C | C |
| D | D |
| E | X |
| F | F |

Dest. Line

Longest Matching Prefix

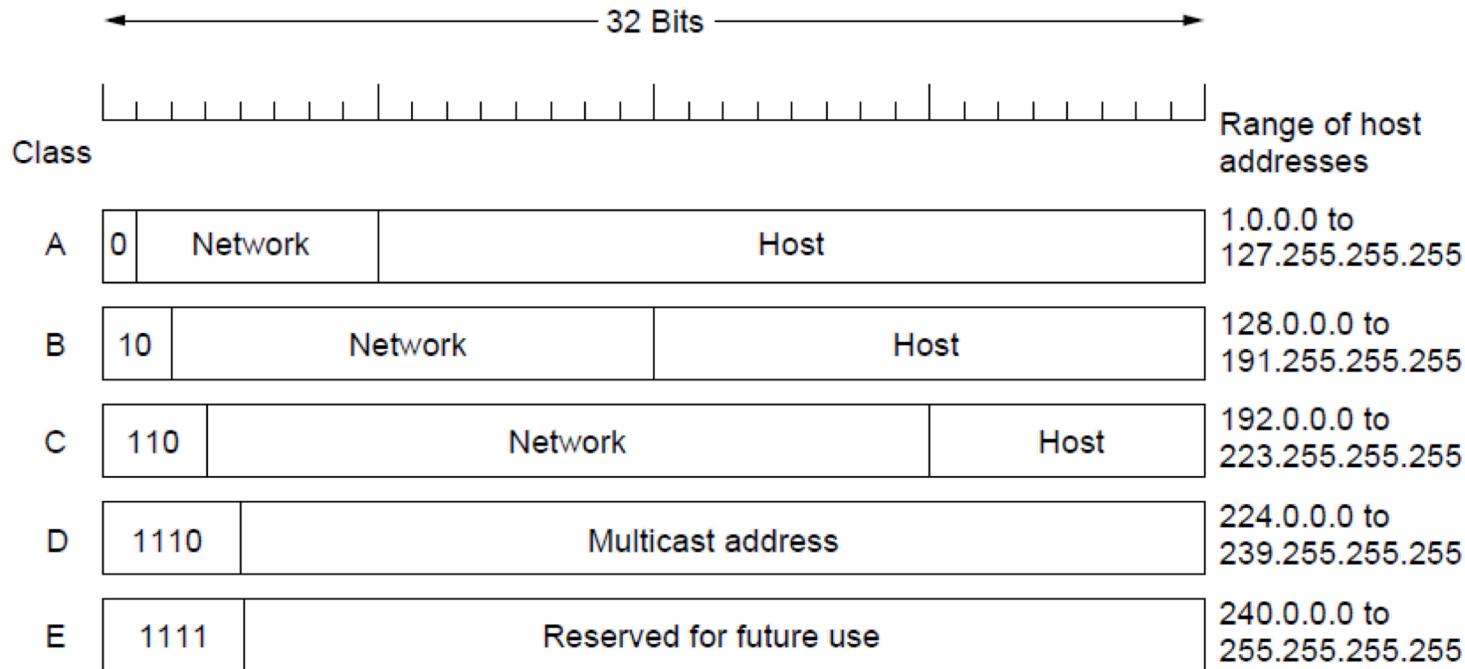
- Packets are forwarded to the entry with the longest matching prefix or smallest address block
 - Complicates forwarding but adds flexibility
- 1) Check address whether matches the longest prefix → /22
 - 2) If not the see if it matches /19



| Prefix addr | Subnet Mask | Interface |
|-------------|---------------|-----------|
| 192.24.12.0 | 255.255.252.0 | Eth 0 |
| 192.24.0.0 | 255.255.224.0 | Eth 1 |

Classful Addressing

- Part of history now-old addresses came in blocks of fixed size (A, B, C)
 - Carries size as part of address, but lacks flexibility
 - Called classful (vs. classless) addressing

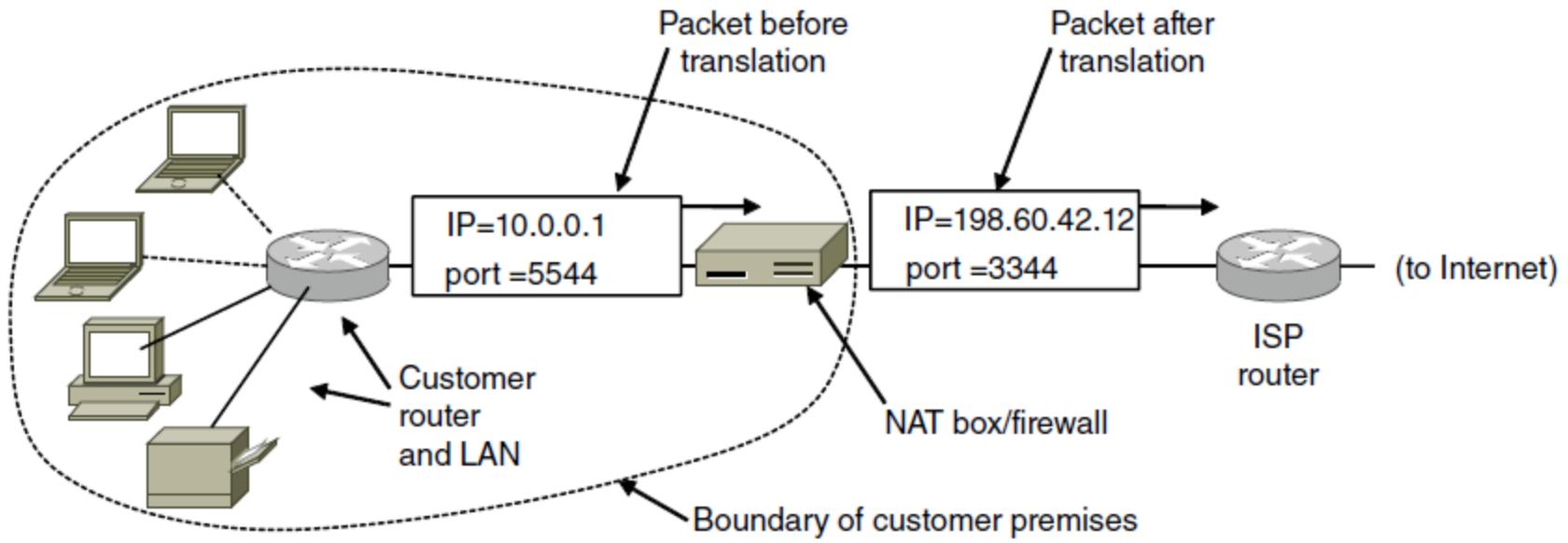


Private IP Ranges

- Range of IP addresses that CANNOT appear in the Internet
- Only for private networks
- 10.0.0.0/8 (16,777,216 hosts)
- 172.16.0.0/12 (1,048,576 hosts)
- 192.168.0.0 /16 (65,536 hosts)

Network Address Translation (NAT)

- NAT box maps one external IP address to many internal IP addresses
 - Uses TCP/UDP port to tell connections apart
 - Violates layering; very common in homes, etc.
- <https://youtu.be/QBqPzHEDzvo>



Internet Control Protocols

- IP works with the help of several control protocols:
 - ICMP is a companion to IP that returns error info
 - Required, and used in many ways, e.g., for traceroute
 - ARP finds MAC address of a local IP address
 - Glue that is needed to send any IP packets
 - Host queries an address and the owner replies
 - DHCP assigns a local IP address to a host
 - Gets host started by automatically configuring it
 - Host sends request to server, which grants a lease

ICMP

- Internet Control Message Protocol
- Used for testing and monitoring ambient conditions between hosts and routers

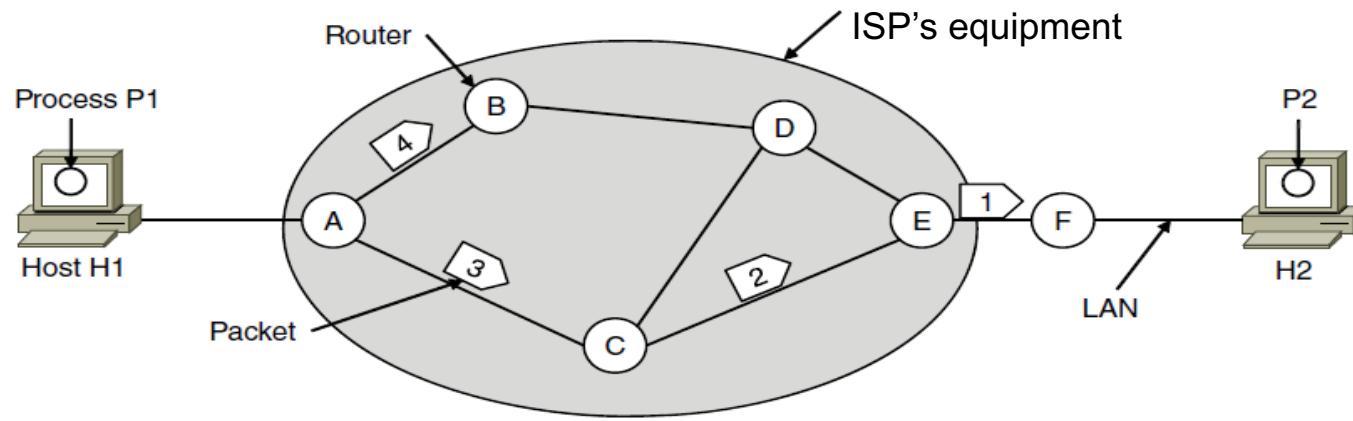
| Message type | Description |
|-----------------------------------|----------------------------------|
| Destination unreachable | Packet could not be delivered |
| Time exceeded | Time to live field hit 0 |
| Parameter problem | Invalid header field |
| Source quench | Choke packet |
| Redirect | Teach a router about geography |
| Echo and Echo reply | Check if a machine is alive |
| Timestamp request/reply | Same as Echo, but with timestamp |
| Router advertisement/solicitation | Find a nearby router |

Outline

- Routing Algorithms
- Routing and Routing Control
- Quality of Service

Reminder: Routing within a datagram subnet

- Post office model: packets are routed individually based on destination addresses in them
- Packets can take different paths



| A's table (initially) | A's table (later) | C's Table | E's Table | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
|--|-------------------|-----------|-----------|---|---|---|---|---|---|---|---|---|--|---|---|---|---|---|---|---|---|---|---|---|---|--|---|---|---|---|---|---|---|---|---|---|---|---|--|---|---|---|---|---|---|---|---|---|---|---|---|
| <table border="1"><tr><td>A</td><td>X</td></tr><tr><td>B</td><td>B</td></tr><tr><td>C</td><td>C</td></tr><tr><td>D</td><td>B</td></tr><tr><td>E</td><td>C</td></tr><tr><td>F</td><td>C</td></tr></table> | A | X | B | B | C | C | D | B | E | C | F | C | <table border="1"><tr><td>A</td><td>X</td></tr><tr><td>B</td><td>B</td></tr><tr><td>C</td><td>C</td></tr><tr><td>D</td><td>B</td></tr><tr><td>E</td><td>B</td></tr><tr><td>F</td><td>B</td></tr></table> | A | X | B | B | C | C | D | B | E | B | F | B | <table border="1"><tr><td>A</td><td>A</td></tr><tr><td>B</td><td>A</td></tr><tr><td>C</td><td>X</td></tr><tr><td>D</td><td>E</td></tr><tr><td>E</td><td>E</td></tr><tr><td>F</td><td>E</td></tr></table> | A | A | B | A | C | X | D | E | E | E | F | E | <table border="1"><tr><td>A</td><td>C</td></tr><tr><td>B</td><td>D</td></tr><tr><td>C</td><td>C</td></tr><tr><td>D</td><td>D</td></tr><tr><td>E</td><td>X</td></tr><tr><td>F</td><td>F</td></tr></table> | A | C | B | D | C | C | D | D | E | X | F | F |
| A | X | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| B | B | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| C | C | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| D | B | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| E | C | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| F | C | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| A | X | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| B | B | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| C | C | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| D | B | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| E | B | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| F | B | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| A | A | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| B | A | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| C | X | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| D | E | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| E | E | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| F | E | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| A | C | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| B | D | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| C | C | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| D | D | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| E | X | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| F | F | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |

Routing Algorithms

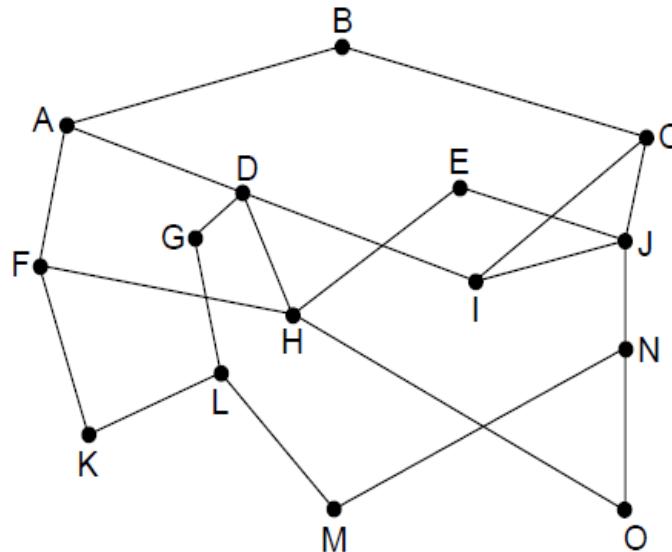
- Consider the network as a graph of nodes and links:
 - Decide what to optimize (e.g., fairness vs efficiency)
 - Update routes for changes in topology (e.g., failures)
 - Routing is the process of discovering network paths
- The routing algorithm is responsible for deciding on which output line an incoming packet should be transmitted
- Non-Adaptive Algorithms
 - Static decision making process (e.g., static routing)
- Adaptive Algorithms
 - Dynamic decision making process (e.g., dynamic routing)
 - Changes in network topology, traffic, etc.

Optimality Principle

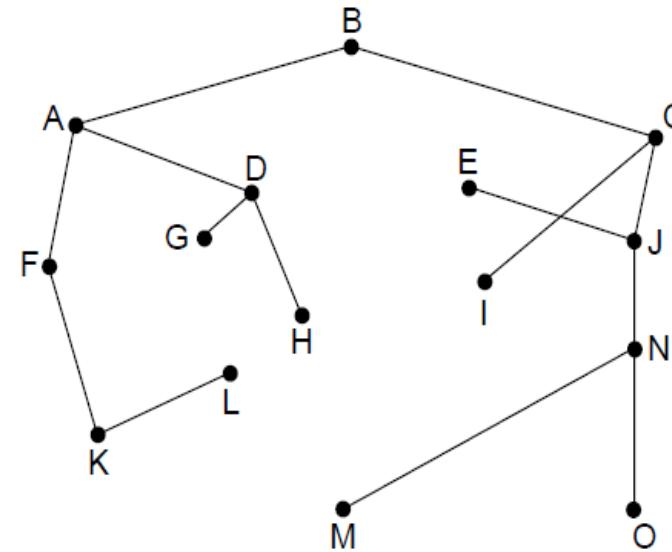
- “If router B is on the optimal path from router A to router C, then the optimal path for B to C also falls along the same route”.

Sink Tree

- The set of optimal routes from all sources to a given destination form a tree rooted at the destination - “sink tree”
- The goal of a routing algorithm is to discover and utilise the sink trees for all routers



Network



Sink tree of best paths to router B

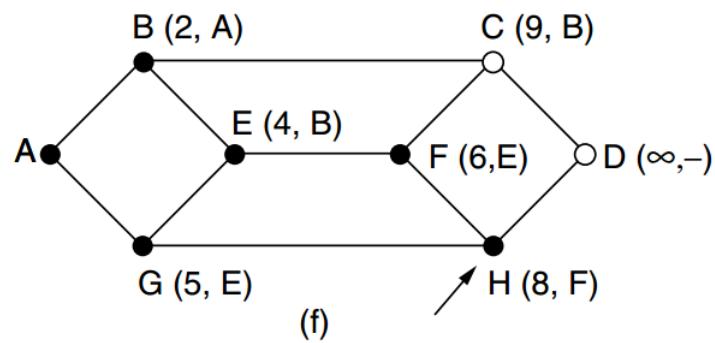
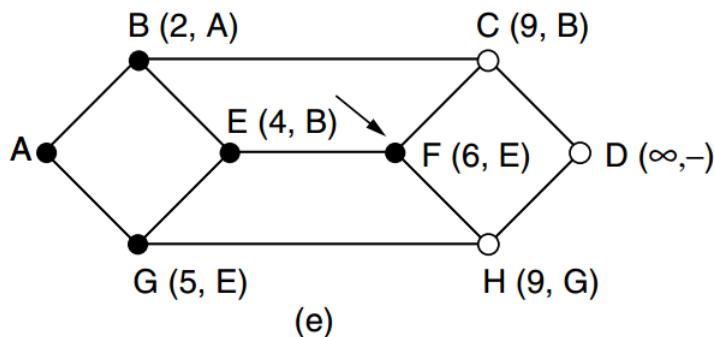
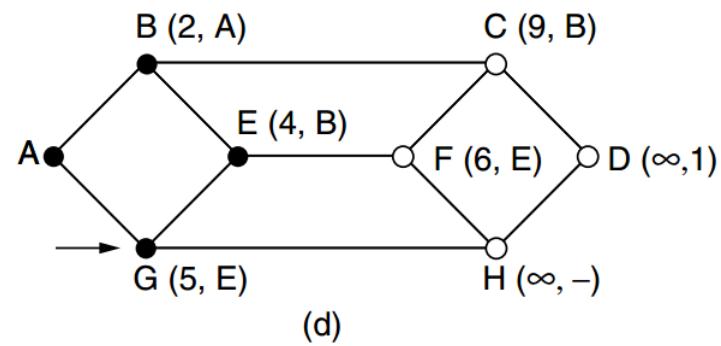
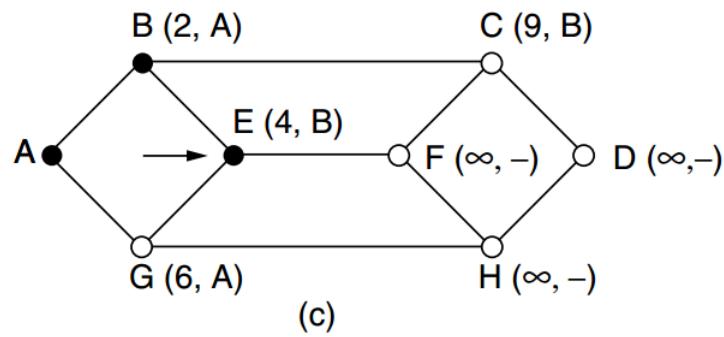
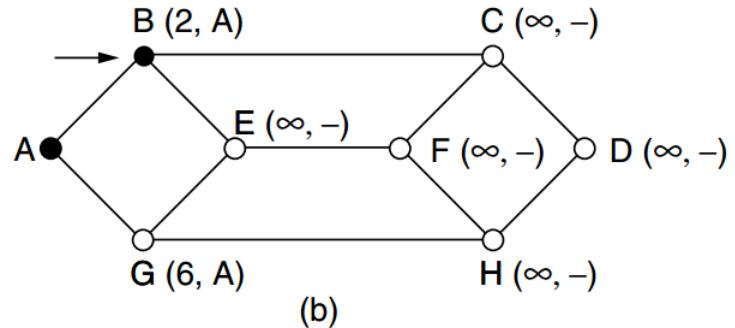
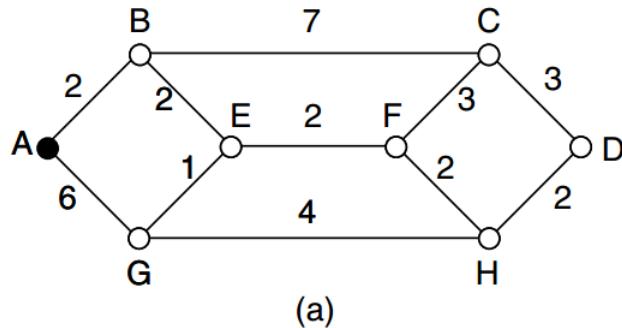
Shortest Path Routing

- A non-adaptive algorithm
- Shortest path can be determined by building a graph with each node representing a router, and each arc representing a communication link
- To choose a path between 2 routers, the algorithm finds the shortest path between them on the graph

Dijkstra's algorithm for computing a sink tree

- Dijkstra's algorithm computes a sink tree on the graph:
 - Each link is assigned a non-negative weight/distance
 - Shortest path is the one with lowest total weight
 - Using weights of 1 gives paths with fewest hops
- Algorithm:
 - Start with sink, set distance at other nodes to infinity
 - Relax distance to other nodes
 - Pick the lowest distance node, add it to sink tree
 - Repeat until all nodes are in the sink tree

Shortest Path Algorithm (2)



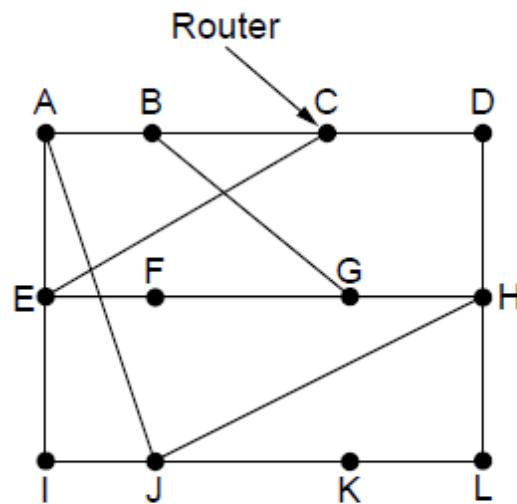
Flooding

- A non-adaptive algorithm
- Every incoming packet is sent out on every outgoing line except the one on which it arrived
- Generates a large number of duplicate packets - inefficient
- Selective flooding (where routers send packets only on links which are in approximately the right direction) is an improved variation

Distance Vector Routing

- A **dynamic** algorithm
- Each router maintains a **table** which includes the **best known distance** to each destination (a metric) and which line to use to get there.
- Tables are exchanged with **neighbouring routers**
- “**Global information shared locally**”
- Algorithm:
 - Each node knows distance of links to its neighbors
 - Each node advertises vector of lowest known distances to all neighbors
 - Each node uses received vectors to update its own
 - Repeat periodically

Distance Vector Routing (2)



Network

| To | A | I | H | K | New estimated delay from J | Line |
|----|----|----|----|----|----------------------------|------|
| A | 0 | 24 | 20 | 21 | 8 | A |
| B | 12 | 36 | 31 | 28 | 20 | A |
| C | 25 | 18 | 19 | 36 | 28 | I |
| D | 40 | 27 | 8 | 24 | 20 | H |
| E | 14 | 7 | 30 | 22 | 17 | I |
| F | 23 | 20 | 19 | 40 | 30 | I |
| G | 18 | 31 | 6 | 31 | 18 | H |
| H | 17 | 20 | 0 | 19 | 12 | H |
| I | 21 | 0 | 14 | 22 | 10 | I |
| J | 9 | 11 | 7 | 10 | 0 | - |
| K | 24 | 22 | 22 | 0 | 6 | K |
| L | 29 | 33 | 9 | 9 | 15 | K |

JA delay is 8 JI delay is 10 JH delay is 12 JK delay is 6

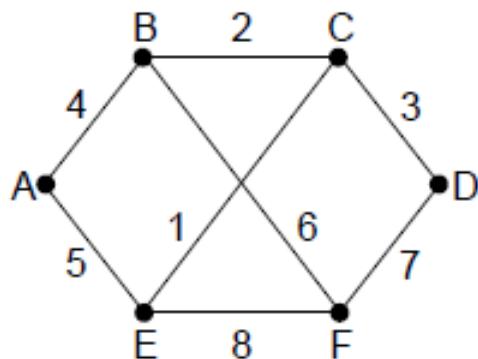
New vector for J
Vectors received at J from
Neighbors A, I, H and K

Link State Routing

- A dynamic algorithm
 - An alternative to distance vector
 - DV - primary problem that caused its demise was that the algorithm often took too long to converge after the network topology changed
 - Widely used in the Internet (OSPF, ISIS)
 - More computation but simpler dynamics
- Each router has to do 5 steps:
 1. Discover neighbours and learn network addresses
 2. Measure delay or cost to each neighbour
 3. Construct packet resulting from previous steps
 4. Send this packet to all other routers
 5. Compute the shortest path to every other router
- “**Local information shared globally**” using flooding

Building link state packets

- LSP (Link State Packet) for a node lists neighbors and weights of links to reach them



Network

| Link | State | Packets |
|------|-------|---------|
| A | C | E |
| Seq. | Seq. | Seq. |
| Age | Age | Age |
| B 4 | B 2 | A 5 |
| E 5 | C 2 | B 6 |
| | D 3 | C 1 |
| | F 6 | D 7 |
| | E 1 | F 8 |
| | | |
| | | |

LSP for each node

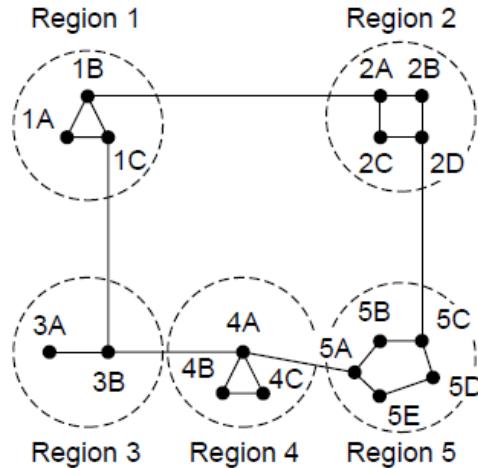
- The hard part is determining when to build LSP.
- Periodically, that is, at regular intervals
- Build them when some significant event occurs, such as a line or neighbour going down or coming back up again or changing its properties appreciably

Hierarchical Routing

- As networks grow in size, routing tables expand but this impacts CPU and memory requirements
- Dividing all routers into regions allows efficiencies
 - Each router knows everything about other routers in its region but nothing about routers in other regions
 - Routers which connect to two regions act as exchange points for routing decisions

Hierarchical routing, cont

- Hierarchical routing reduces the work of route computation but may result in slightly longer paths than flat routing



Full table for 1A

| Dest. | Line | Hops |
|-------|------|------|
| 1A | - | - |
| 1B | 1B | 1 |
| 1C | 1C | 1 |
| 2A | 1B | 2 |
| 2B | 1B | 3 |
| 2C | 1B | 3 |
| 2D | 1B | 4 |
| 3A | 1C | 3 |
| 3B | 1C | 2 |
| 4A | 1C | 3 |
| 4B | 1C | 4 |
| 4C | 1C | 4 |
| 5A | 1C | 4 |
| 5B | 1B | 5 |
| 5C | 1B | 5 |
| 5D | 1C | 6 |
| 5E | 1C | 5 |

Hierarchical table for 1A

| Dest. | Line | Hops |
|-------|------|------|
| 1A | - | - |
| 1B | 1B | 1 |
| 1C | 1C | 1 |
| 2 | 1B | 2 |
| 3 | 1C | 2 |
| 4 | 1C | 3 |
| 5 | 1C | 4 |

Broadcast Routing

- Broadcast routing allows hosts to send messages to many or all other hosts
 - Single distinct packet (inefficient, source needs all destination addresses)
 - Flooding
 - Multi-destination routing (efficient but source needs to know all the destinations)
 - a router receives a single packet which encapsulates the list of destinations, and then constructs a specific packet for each one (acts as a relay)
- Reverse path forwarding
- When a broadcast packet arrives at a router, the router checks to see if the packet arrived on the line normally used for sending packets to the source of the broadcast. If so there is a high probability that the route used to transmit the received packet is the best route. The router then forwards the packet onto all other lines.
- If the broadcast packet arrived on a link other than the preferred one for reaching the source, the packet is discarded as a likely duplicate

Multicast Routing

- A routing algorithm used to send a message to a well-defined group within the whole network
- Each router computes a spanning tree covering all other routers – the first router to receive the packet prunes the spanning tree to eliminate all lines which do not lead to members of the group

Internetwork Routing

- Need to route:
- Within a network using an interior gateway protocol (eg OSPF)
- Between networks using an exterior gateway protocol (eg BGP)

Routing and Routing Control

- Congestion Control Algorithms
 - Handling congestion is the responsibility of the Network and Transport layers working together
 - We look at the Network portion here
- Quality of Service (QoS)

Summary

- Type of services provided by Network Layer
- Internetworking
 - Tunneling
 - Fragmentation
 - Path MTU discovery
- Internet Protocol (IP)
 - Explain principles of Internet design
 - Analyse structure of IP addresses, subnetting, IP add. aggregation, NAT
 - Explain roles of different Internet Control Protocols
- Routing
 - Dijkstra's algorithm
 - Distance Vector routing
 - Link State routing