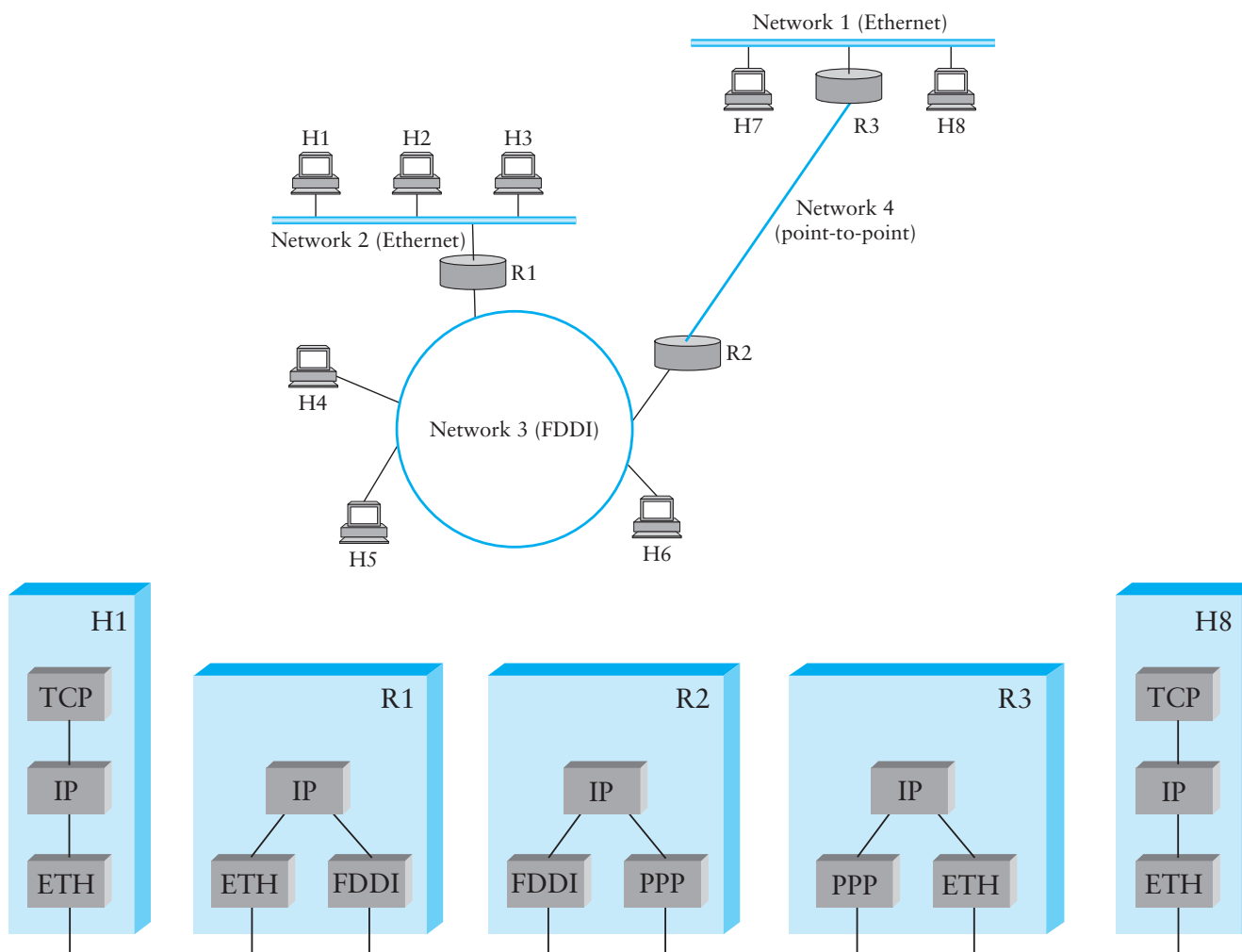


Internet Protocol

- Goal: Glue lower-level networks together
- Wasn't that the goal of switching?



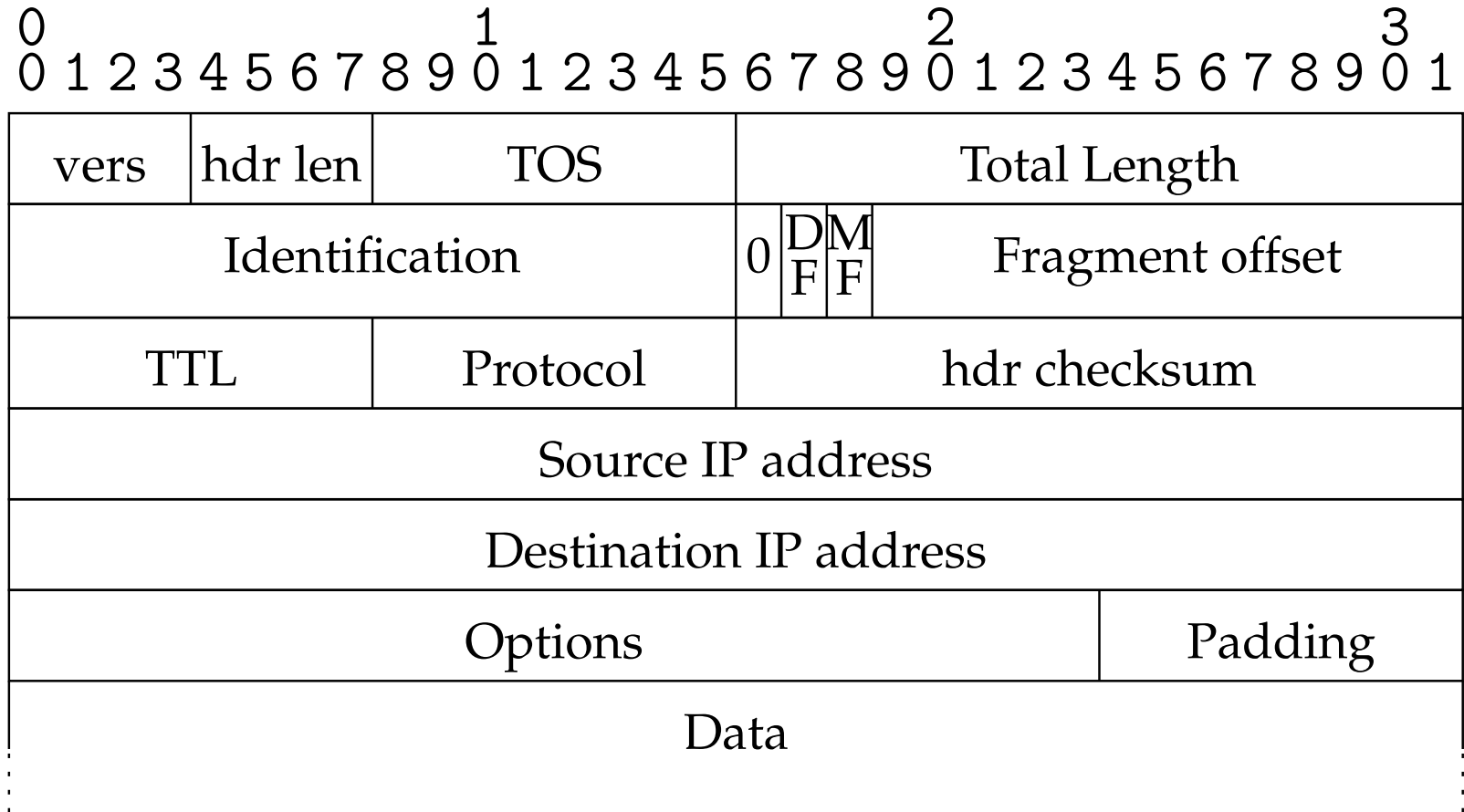
Internetworking Challenges

- Addresses are different.
- Service models differ.
- Allowable packet sizes differ.
- Congestion control: MAC equivalent
- **Scaling matters.**

Service model

- **Connectionless (datagram-based)**
- **Best-effort delivery (unreliable service)**
 - packets are lost
 - packets are delivered out of order
 - duplicate copies of a packet are delivered
 - packets can be delayed for a long time
- **It's the lowest common denominator.**

IP packet format



IP header details

- **Routing is based on destination address**
- **TTL (time to live) decremented at each hop.**
 - Originally was in seconds (no longer)
 - TTL mostly saves from routing loops
 - But other cool uses. . .
- **Fragmentation possible for large packets**
 - Fragmented in network if crosses link w. small frame size
 - MF bit means more fragments for this IP packet
 - DF bit says “don’t fragment” (returns error to sender)
- **Following IP header is “payload” data**
 - Typically beginning with TCP or UDP header

Internet Control Message Protocol (ICMP)

- Echo (ping)
- Redirect (from router to source host)
- Destination unreachable (protocol, port, or host)
- TTL exceeded (so datagrams don't cycle forever)
- Checksum failed
- Reassembly failed
- Cannot fragment
- **Many ICMP messages include part of packet that triggered them**
 - Example: Traceroute

Translating IP to lower-level addresses

- **Map IP addresses into physical addresses**
 - *E.g.*, Ethernet address of destination host
 - Or ethernet address of next hop router
- **Techniques**
 - Encode physical address in host part of IP address (IPv6)
 - Each network node maintains a lookup table (IP→phys)
- **ARP – *address resolution protocol***
 - Dynamically builds table of IP to physical address bindings.
 - Broadcast request if IP address not in table.
 - Everybody learns physical address of requesting node. (broadcast)
 - Target machine responds with its physical address.
 - Table entries are discarded if not refreshed.

ARP Ethernet packet format

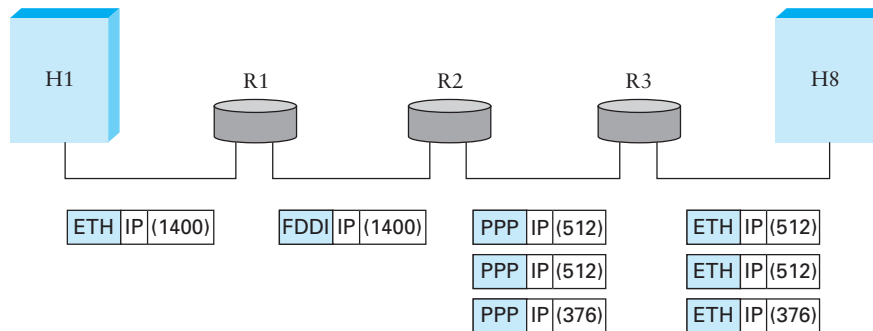
0	8	16	31
Hardware type = 1		ProtocolType = 0x0800	
HLen = 48	PLen = 32		Operation
SourceHardwareAddr (bytes 0–3)			
SourceHardwareAddr (bytes 4–5)		SourceProtocolAddr (bytes 0–1)	
SourceProtocolAddr (bytes 2–3)		TargetHardwareAddr (bytes 0–1)	
TargetHardwareAddr (bytes 2–5)			
TargetProtocolAddr (bytes 0–3)			

Why include the source hardware address? Why not?

Fragmentation & Reassembly

- Each network has some MTU
- Strategy
 - Fragment when necessary ($MTU < \text{Datagram}$)
 - Re-fragmentation is possible
 - Fragments are self-contained datagrams
 - Use CS-PDU (not cells) for ATM
 - Delay reassembly until destination host
 - Do not recover from lost fragments
When fragment is lost, whole packet must be retransmitted!

Fragmentation example



(a)

Start of header			
Ident = x		0	Offset = 0
Rest of header			
1400 data bytes			

(b)

Start of header			
Ident = x		1	Offset = 0
Rest of header			
512 data bytes			

Start of header			
Ident = x		1	Offset = 64
Rest of header			
512 data bytes			

Start of header			
Ident = x		0	Offset = 128
Rest of header			
376 data bytes			

Sending data

- **Transports should send path-MTU sized packets.**
 - Chosen to avoid fragmentation (e.g., 1460 on ethernet LAN)
 - Write of 8K might use 6 segments.
 - Problem: How to know what MTU to use?
- **Solution: Exploit ICMP messages**
 - Set **DF** (don't fragment) bit in IP header
 - If too big, will get back ICMP Cannot Fragment
- **Can do binary search on packet sizes**
 - But better: Base algorithm on most common MTUs

Congestion Control

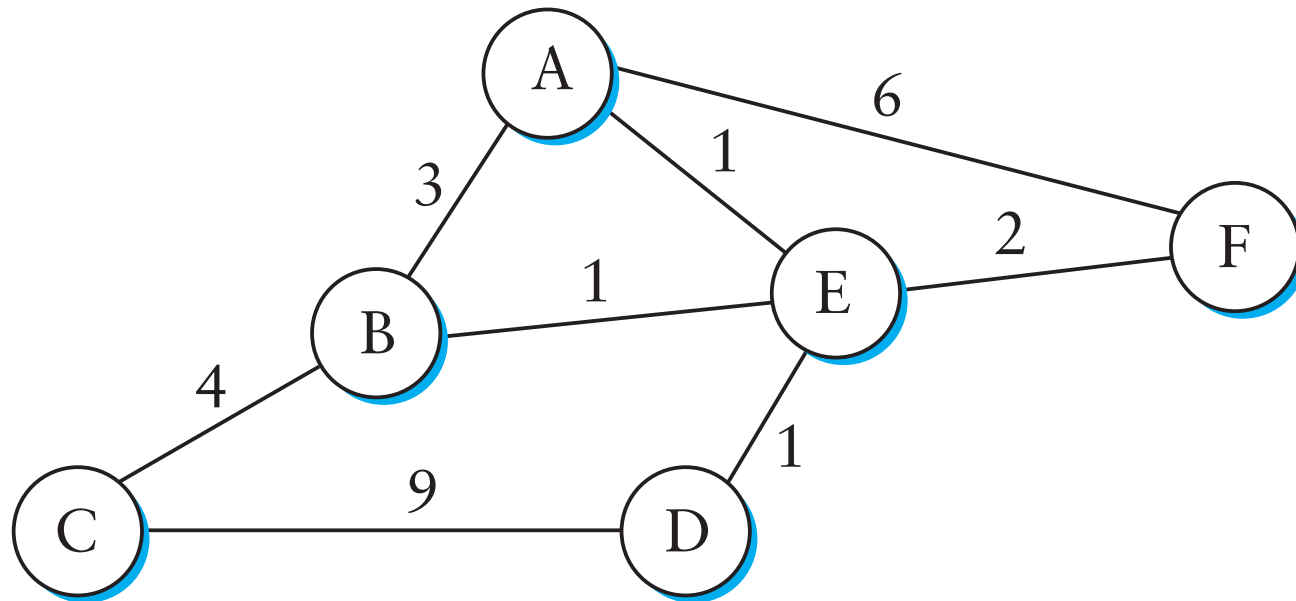
At the network layer, there is not a lot to be done.

- **Drop packets.**
 - Upper layers (TCP) interpret loss.
 - Complicates non-congestive loss recovery.
- **Explicitly signal higher layers.**
 - ICMP Source Quench
 - ECN bits.
 - Eliminates confusion
- **Active Queue Management (AQM)**
 - Random Early Detection (RED), Weighted Fair Queuing
 - Can be used *before* queues fill (best with ECN)

What is routing?

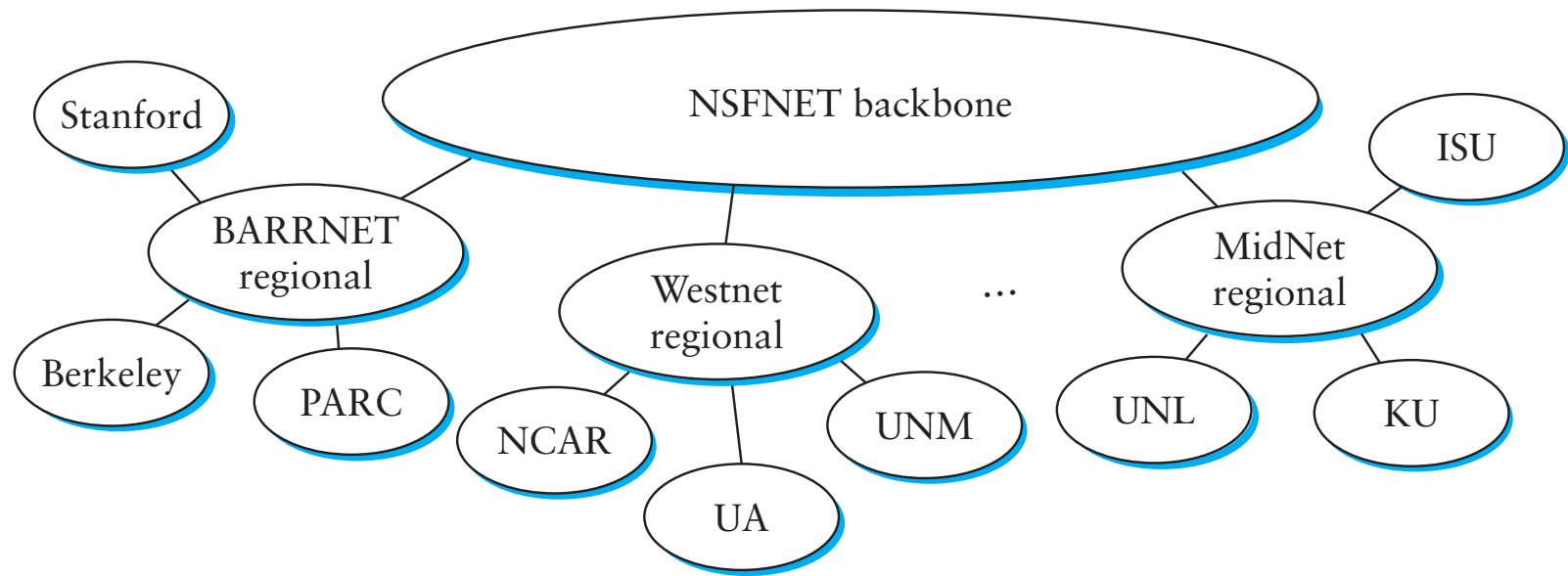
- **Packet *forwarding* – moving packets between ports**
 - Look up destination address in forwarding table
 - Find *out-port* or $\langle out-port, MAC\ addr \rangle$ pair
- ***Routing* is process of populating forwarding table**
 - Routers exchange messages about nets they can reach
 - Goal: Find optimal route for every destination, or maybe good route, or maybe *any* route (depending on scale)
- ***Intra-domain* vs. *Inter-domain* routing**
 - Intra-: All routers under same administrative control
 - Intra-: Scale to ~ 100 networks (e.g., campus like Brown)
 - Inter-: Decentralized, scale to Internet

Optimality



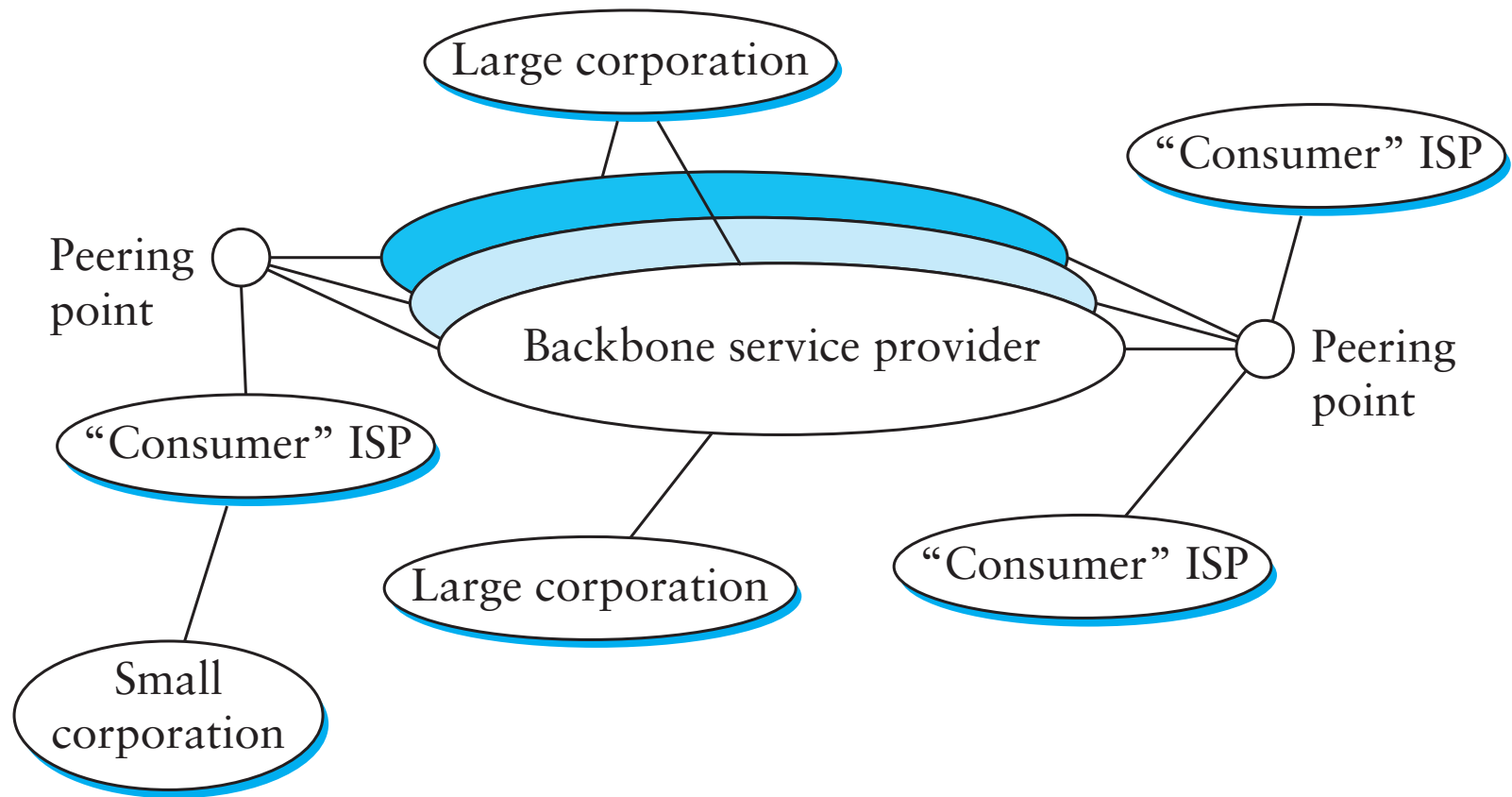
- **View network as a graph**
- **Assign *cost* to each edge**
 - Can be based on latency, b/w, utilization, queue length, ...
- **Problem: Find lowest cost path between two nodes**
 - Must be computed in *distributed* way

The Internet, 1990



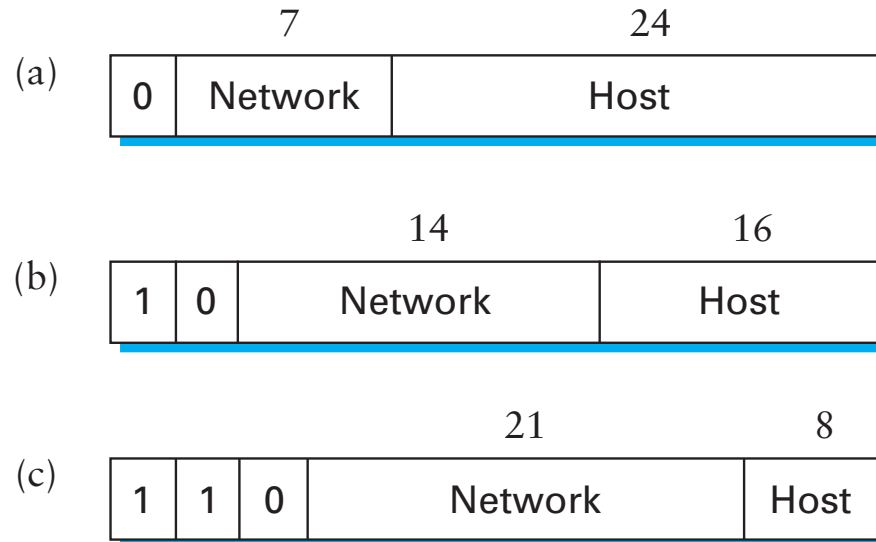
- Hierarchical structure

The Internet, today



- Multiple "backbones"

Exploit structure of network



- **Treat IP address as structured tuples.**
 - Class A (8-bit prefix), B (16-bit), C (24-bit)
 - Routers need only know route for each network

Forwarding Tables

Network	Next Address
212.31.32.*	0.0.0.0
18.*.*.*	212.31.32.5
128.148.*.*	212.31.32.4
default	212.31.32.1

- Like an Ethernet switch's table, but more compact.
- Keyed by network portion, not the entire address.
- Next address should be local.

Classed routing

- **Hierarchical: network + host**
 - Saves memory in backbone routers (no default route)
 - Originally, routing prefix embedded in address
- **Inefficient use of Hierarchical Address Space**
 - Class C with 2 hosts ($2/255 = 0.78\%$ efficient)
 - Class B with 256 hosts ($256/65535 = 0.39\%$ efficient)
 - Causes shortage of IP addresses (esp. class B)
 - Makes address authorities reluctant to give out class Bs
- **Still Too Many Networks**
 - routing tables do not scale
 - route propagation protocols do not scale

Subnetting

Network number	Host number
----------------	-------------

Class B address

111111111111111111111111	00000000
--------------------------	----------

Subnet mask (255.255.255.0)

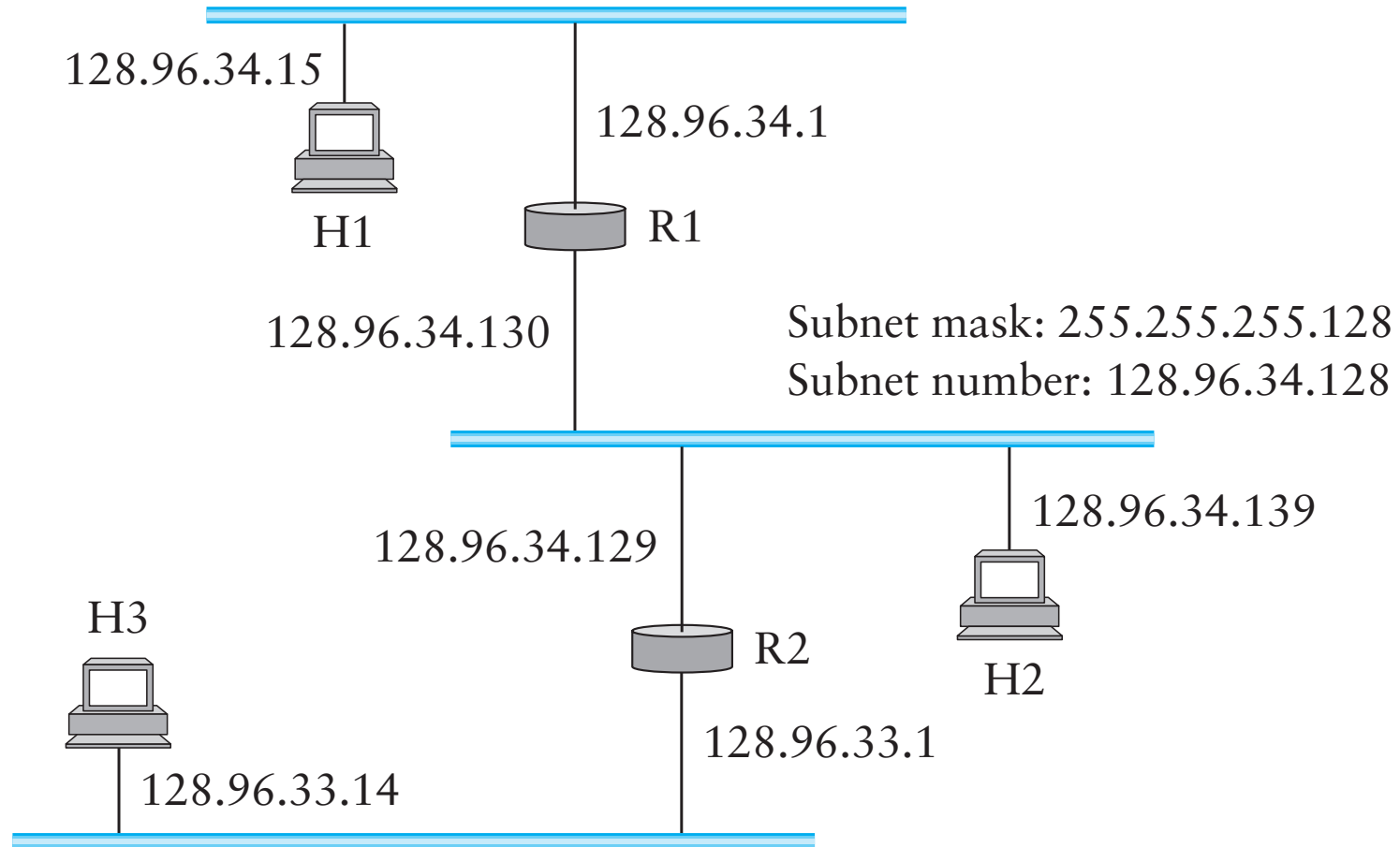
Network number	Subnet ID	Host ID
----------------	-----------	---------

Subnetted address

- Add another level to address/routing hierarchy
- Subnet masks define variable partition of host part
- Subnets visible only within site

Example

Subnet mask: 255.255.255.128
Subnet number: 128.96.34.0



Subnet mask: 255.255.255.0
Subnet number: 128.96.33.0

Supernetting

- **Assign block of contiguous network numbers to nearby networks**
- **Called CIDR: Classless Inter-Domain Routing**
- **Represent blocks with a single pair**
(first network address, count)
- **Restrict block sizes to powers of 2**
- **Use a bit mask (CIDR mask) to identify block size**
- **All routers must understand CIDR addressing**

CIDR Forwarding Tables

Network	Next Address
212.31.32/24	0.0.0.0
18/8	212.31.32.5
128.148/16	212.31.32.4
128.148.128/17	212.31.32.8
0/0	212.31.32.1

IPv6

- **Features**

- 128-bit addresses (classless), includes multicast addresses
- real-time service
- authentication and security
- autoconfiguration
- end-to-end fragmentation
- protocol extensions

- **40-byte “base” header, points to next extension hdr**

- fragmentation
- source routing
- authentication and security
- other options

Network Address Translation

- Address space is IPv6's main selling point.
- Despite CIDR, it's still difficult to allocate addresses. (2^{32} is only 4 billion.)
- NAT "hides" entire networks behind one address.
- Hosts are given *private* addresses.
- Routers map outgoing packets to a free address/port.
- Routers reverse map incoming packets
- Problems?

Coming up

- **Now: Get a partner! Start assignment #1**
- **Today, 4:30pm: NetApp [Lubrano]**
- **Tomorrow, 7pm: Microsoft [CIT 165]**
- **Thu, 4pm: Securing Vehicular Communication**
- **Thu, 5:30pm: Oracle [Lubrano]**
- **Thu, 11:59pm: Homework 1**
- **Oct 13th, 10am-3pm: ACM Programming Contest**