Bridges, IP Basics

18 December 2024
Lecture 7

Some Slides Credits: Steve Zdancewic (UPenn)

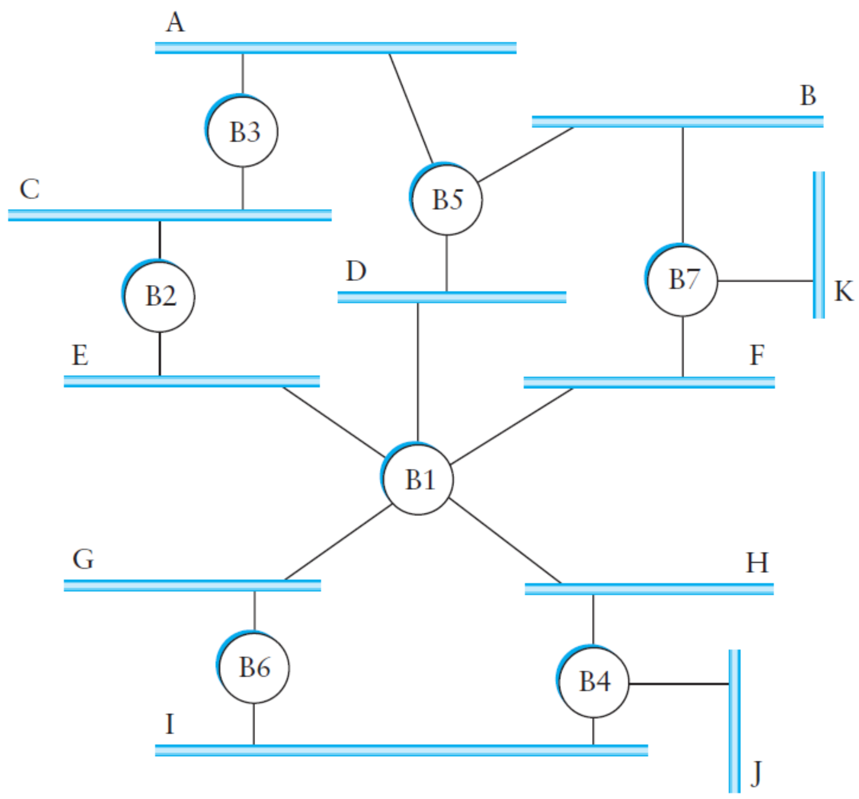
Topics for Today

- Bridges and Spanning Tree Algorithm
- IP
  - Basics
  - Addresses
  - Fragmentation and Reassembly

Spanning Tree Concepts

Key concepts:

- A single root bridge is elected
  - Each subnet must have a single path to reach the root bridge
- Each bridge may be connected to (and receive packets from) multiple subnets
  - Only the designated bridge will forward packets toward the root
- Every bridge knows which of its ports is closest to the root bridge
  - Called the root port



Spanning Tree Algorithm

Advertisement
$(ROOT, dist, SENDER)$

- $ROOT$ root node ID
- $dist$ how many hops to the root $ROOT$
- $SENDER$ ID who sent it

Each node begins thinking it's the root and starts advertising that

If a node receives a better advertisement, it stops broadcasting its own messages

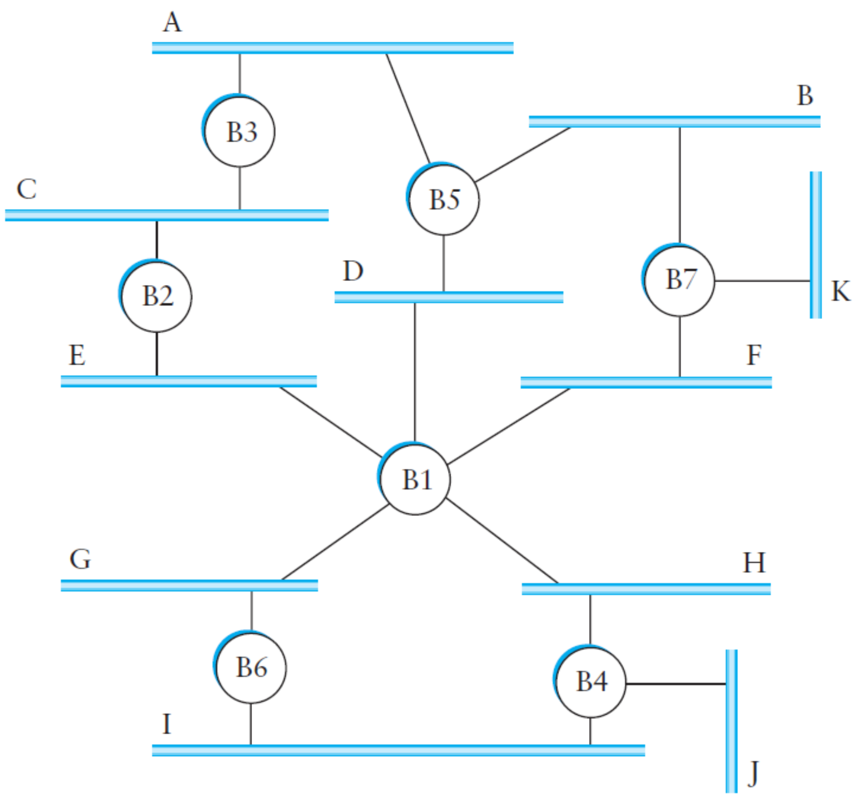
- Better is smaller $ROOT$ ID or same $ROOT$ ID and smaller $dist$
- Last one generating ads wins as root
- Bridge remembers where the shortest, best path – that's the Root Port

Election also for designated bridge (at the same time)

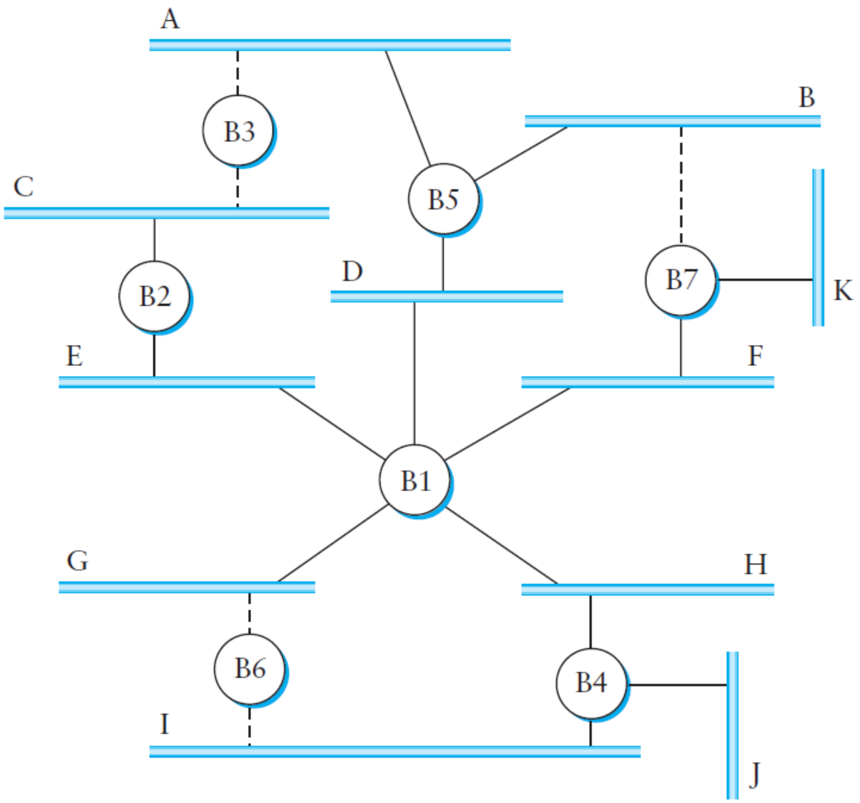
- Smallest $dist$ to $ROOT$ or same $dist$ but smaller ID
- If a bridge hears a shorter, better path on a port, it knows it's not the Designated Bridge for that subnet

Spanning Tree Example

Before



After



Spanning Tree Maintenance

The root bridge is the last one generating advertisements

- It sends out advertisements every so often
- If a bridge notices that it hasn't heard an advertisement in a while (timer), it starts the algorithm again

- Automatic detection of failures and network topology changes:
  - Hello time – how often does the bridge send out messages
  - Max age – maximum age for a message before it's dropped
  - Forward delay – how long it takes to move bridge from listening to forwarding

Limitations of Bridges

Scaling

- Connections on order of dozens
- Spanning tree algorithm scales linearly
- Transparency incomplete

Congestion can be visible to higher protocol layers

Latency can be larger and more variable

Heterogeneity

- Limited to compatible (similarly addressed) link layers

Rapid Spanning Tree Protocol

- Updates protocol from 2004

Less bridge port states

- Discarding
- Learning
- Forwarding

Alternate Port role

- A second way to get to the root
- Alternate to the root port

Backup Port role

- Another way to reach the same LAN
- Same switch, just different ports

Every bridge sends packets every Hello time

- Not just in response to root’s messages

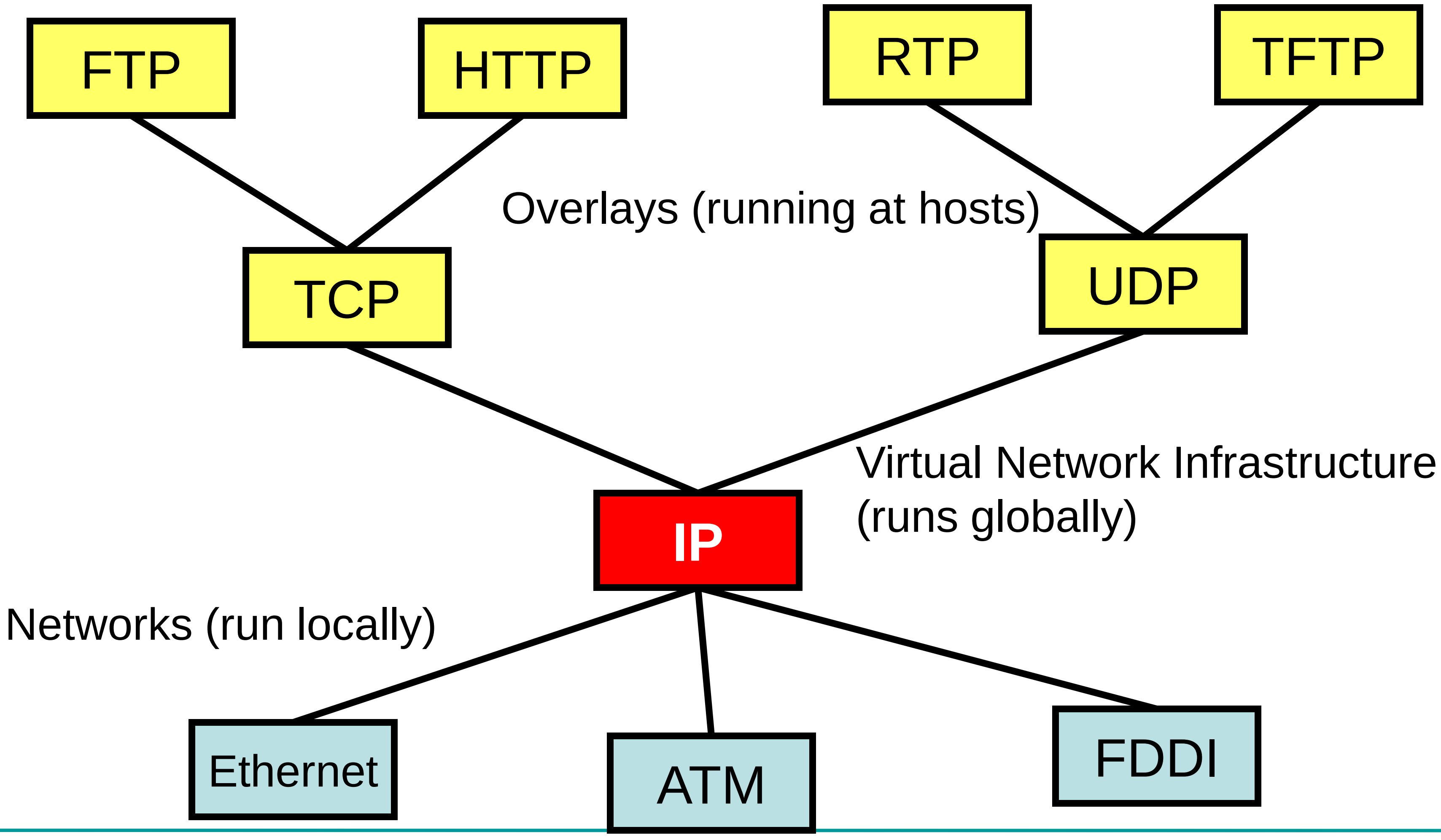
Faster sync between bridges

- Negotiate, don’t just listen for advertisements
- Finish in 3x Hello time

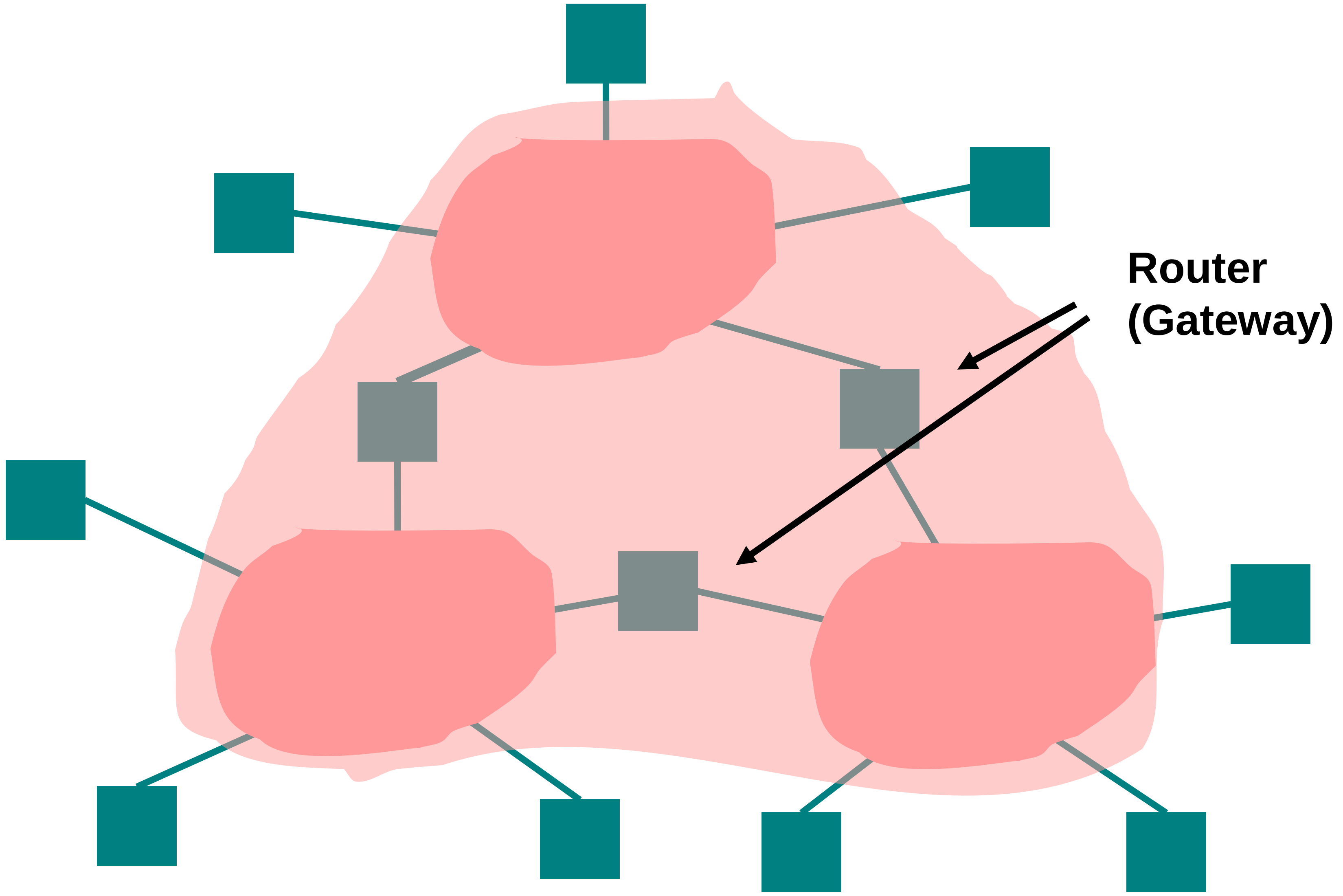
So Far

- Bridges and Spanning Tree Algorithm
- IP
    - Basics
    - Addresses
    - Fragmentation and Reassembly

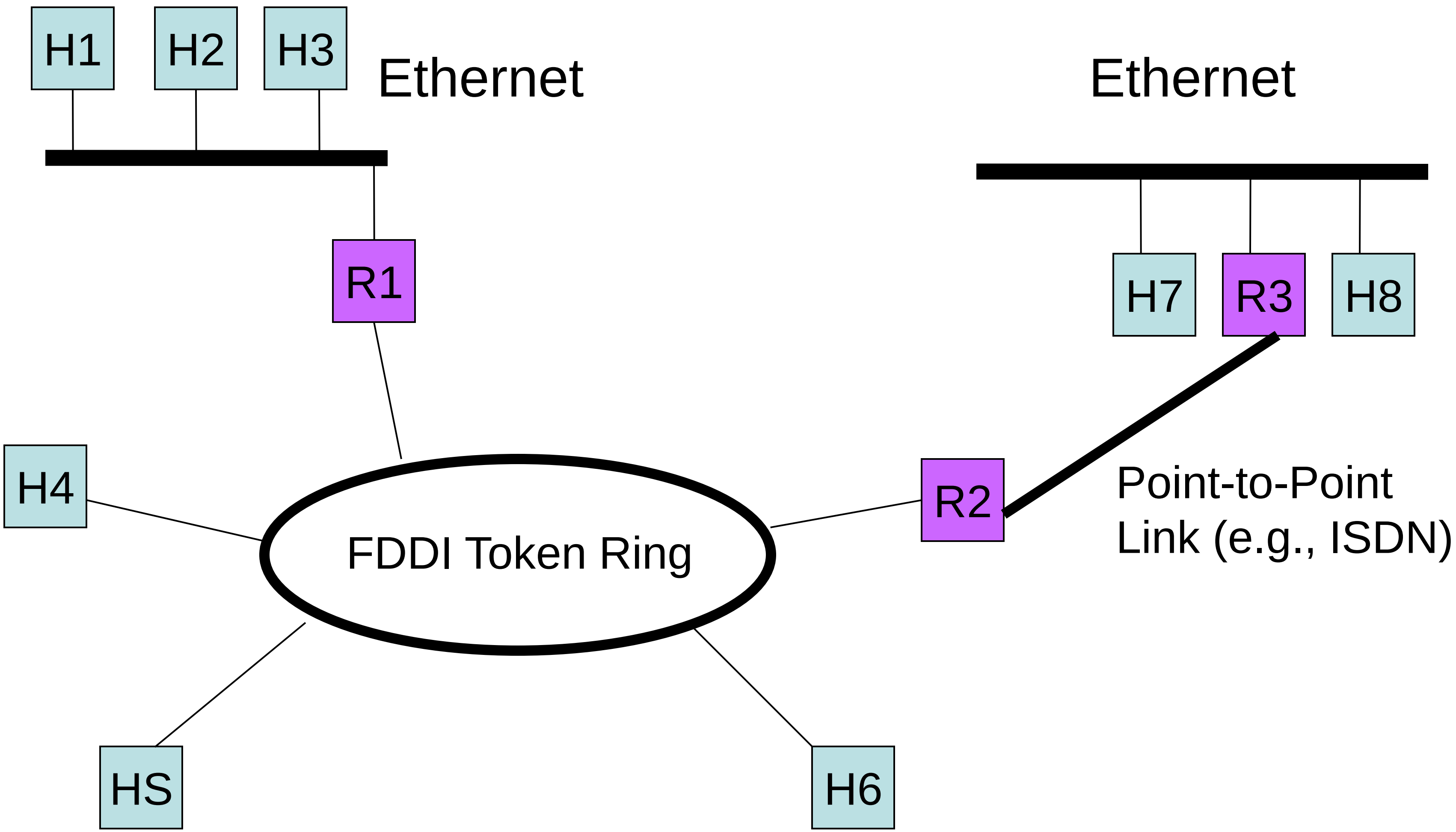
Internet Protocol Interoperability



Internetworks



Internetworks



IP Concepts

Address

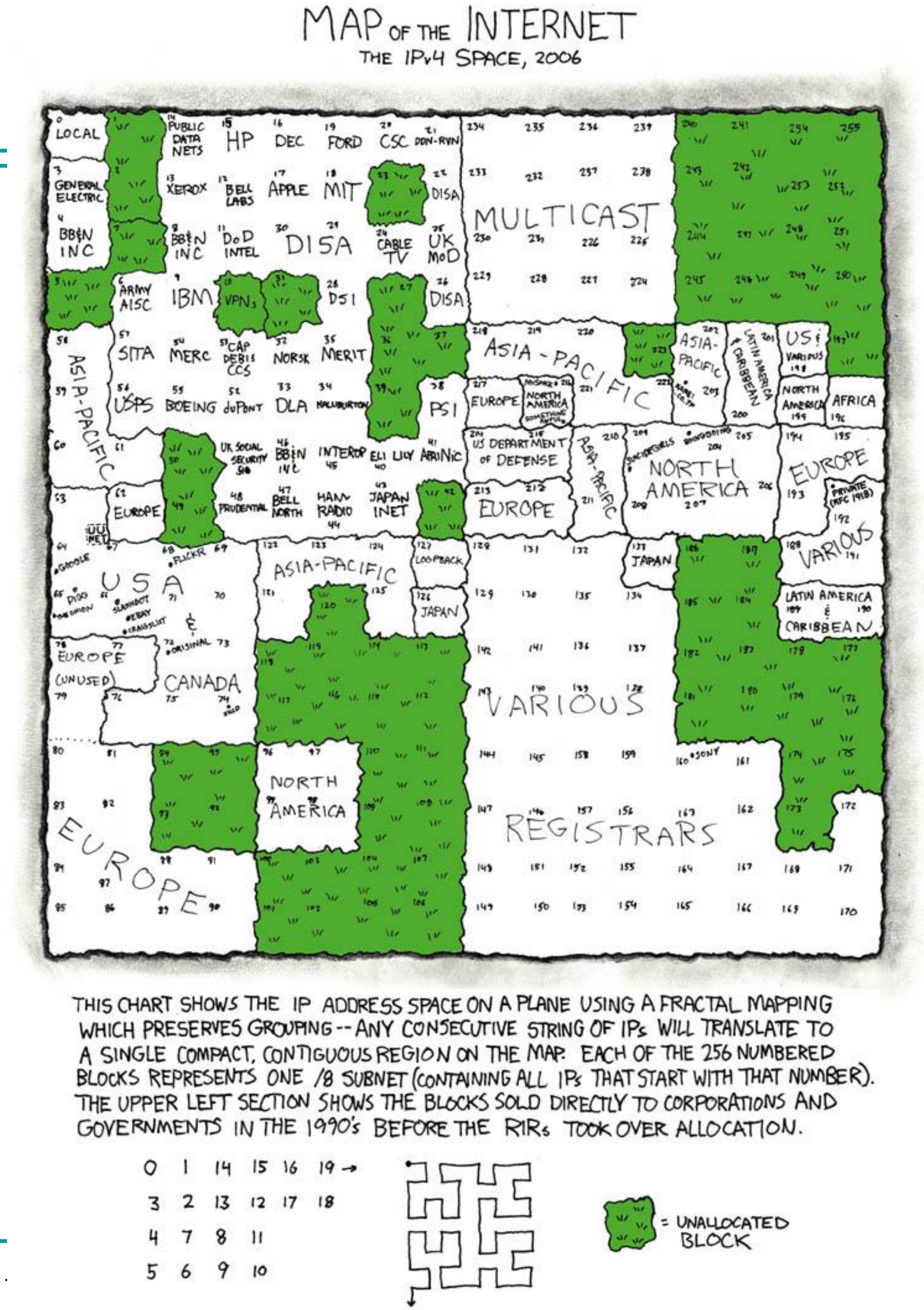
Router

Forwarding

Fragmentation

IP Addresses

- 4 byte addresses
  - Written 255.255.255.255
- Hierarchical



Service Model

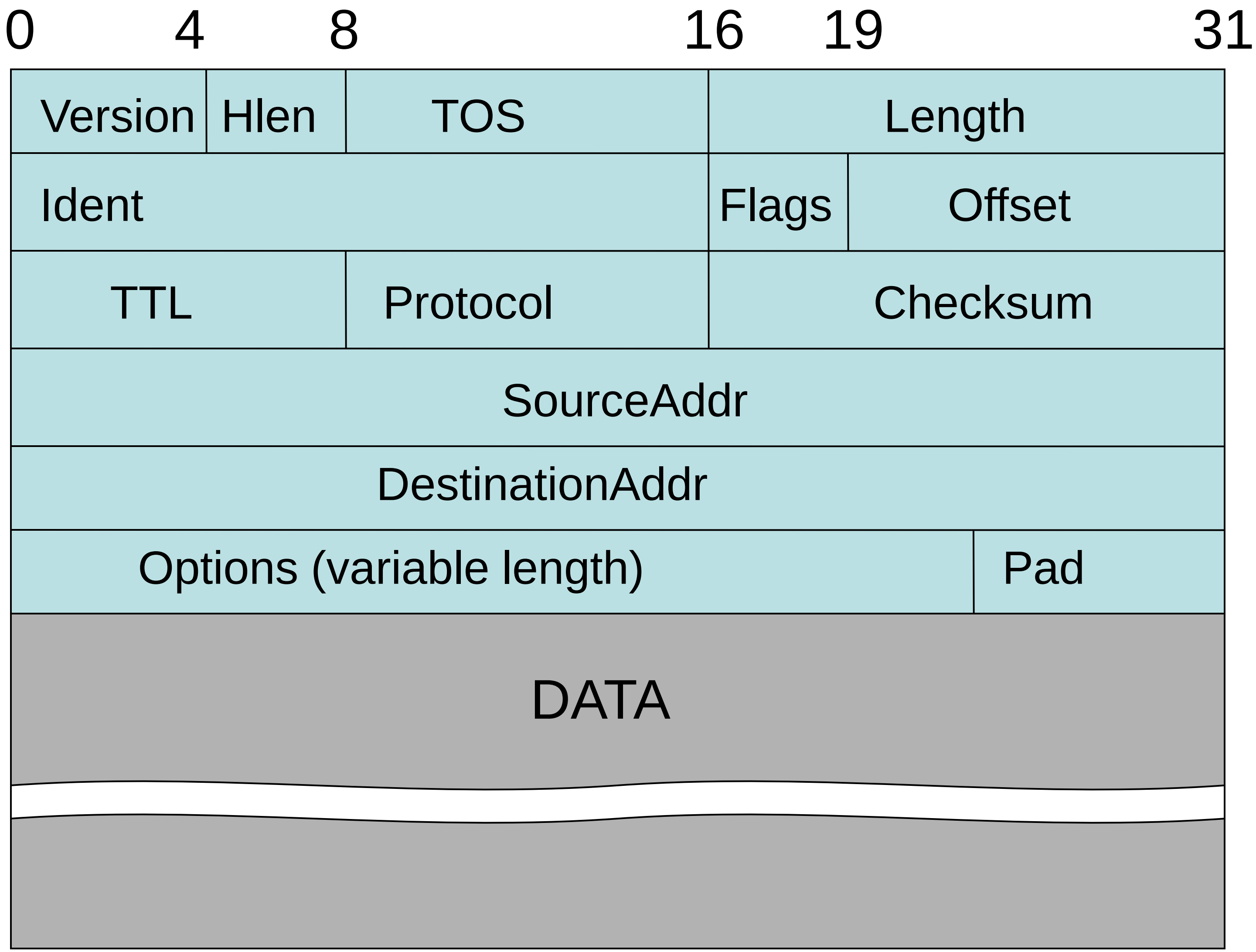
Minimal service mode

- All nets can implement
- “Tin cans and a string” extremum

Features:

- Standard packet format
- Best-effort datagram delivery (unreliable)
- “*Run over anything*”

IPv4 Packet Format



Fields of IPv4 Header

Version

- Version of IP, example header is IPv4
- First field so easy to implement case statement

Hlen

- Header length, in 32-bit *words*

TOS

- Type of Service (rarely used)
- Priorities, delay, throughput, reliability

Length

- Length of datagram, in *Bytes*
- 16 bits, hence max. of 65,536 Bytes

Fields for *fragmentation* and *reassembly*

- Identifier
- Flags
- Offset

Header fields, continued

TTL

- Time to live (in reality, hop count)
- 64 is the current default (128 also used)

Protocol

- Examples: TCP (6), UDP(17)

Checksum

- Checksum of header (not CRC)
- If header fails checksum, discard the whole packet

SourceAddr, DestinationAddr

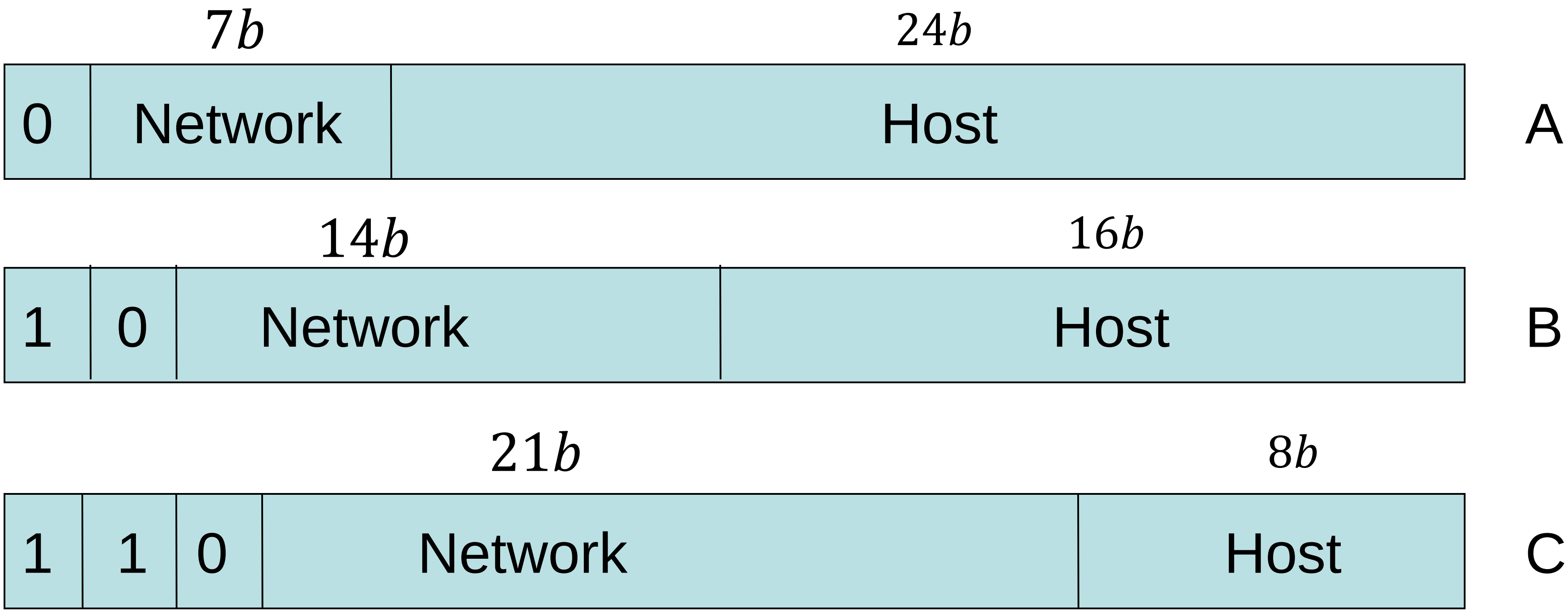
- 32-bit IP addresses
- Global, IP-defined

Options

- Length can be computed using Hlen

IP addresses

- Hierarchical, not flat as in Ethernet



- Written as four decimal numbers separated by dots:
  158.130.14.2

IP Address Ranges

0 0 0 0 0 0 0 0	0 0 0 0 0 0 0 0	0 0 0 0 0 0 0 0	0 0 0 0 0 0 0 0	0.0.0.0	Any IP
0 0 0 0 0 0 0 1	0 0 0 0 0 0 0 0	0 0 0 0 0 0 0 0	0 0 0 0 0 0 0 0	1.0.0.0	
0 0 0 0 0 0 1 0	0 0 0 0 0 0 0 0	0 0 0 0 0 0 0 0	0 0 0 0 0 0 0 0	2.0.0.0	Class A
0 1 1 1 1 1 1 0	0 0 0 0 0 0 0 0	0 0 0 0 0 0 0 0	0 0 0 0 0 0 0 0	126.0.0.0	
0 1 1 1 1 1 1 1	0 0 0 0 0 0 0 0	0 0 0 0 0 0 0 0	0 0 0 0 0 0 0 0	127.0.0.0	Localhost
1 0 0 0 0 0 0 0	0 0 0 0 0 0 0 0	0 0 0 0 0 0 0 0	0 0 0 0 0 0 0 0	128.0.0.0	
1 0 0 0 0 0 0 0	0 0 0 0 0 0 0 1	0 0 0 0 0 0 0 0	0 0 0 0 0 0 0 0	128.1.0.0	
1 0 0 0 0 0 0 0	0 0 0 0 0 0 1 0	0 0 0 0 0 0 0 0	0 0 0 0 0 0 0 0	128.2.0.0	Class B
1 0 0 0 0 0 0 0	0 0 0 0 0 0 1 1	0 0 0 0 0 0 0 0	0 0 0 0 0 0 0 0	128.3.0.0	
1 0 1 1 1 1 1 1	1 1 1 1 1 1 1 0	0 0 0 0 0 0 0 0	0 0 0 0 0 0 0 0	191.254.0.0	
1 0 1 1 1 1 1 1	1 1 1 1 1 1 1 1	0 0 0 0 0 0 0 0	0 0 0 0 0 0 0 0	191.255.0.0	
1 1 0 0 0 0 0 0	0 0 0 0 0 0 0 0	0 0 0 0 0 0 0 0	0 0 0 0 0 0 0 0	192.0.0.0	
1 1 0 0 0 0 0 0	0 0 0 0 0 0 0 0	0 0 0 0 0 0 0 1	0 0 0 0 0 0 0 0	192.0.1.0	
1 1 0 0 0 0 0 0	0 0 0 0 0 0 0 0	0 0 0 0 0 0 1 0	0 0 0 0 0 0 0 0	192.0.2.0	Class C
1 1 0 0 0 0 0 0	0 0 0 0 0 0 0 0	0 0 0 0 0 0 1 1	0 0 0 0 0 0 0 0	192.0.3.0	
1 1 0 1 1 1 1 1	1 1 1 1 1 1 1 1	1 1 1 1 1 1 1 0	0 0 0 0 0 0 0 0	223.255.254.0	
1 1 0 1 1 1 1 1	1 1 1 1 1 1 1 1	1 1 1 1 1 1 1 1	0 0 0 0 0 0 0 0	223.255.255.0	
1 1 1 0 0 0 0 0	0 0 0 0 0 0 0 0	0 0 0 0 0 0 0 0	0 0 0 0 0 0 0 0	224.0.0.0	Class D
1 1 1 0 1 1 1 1	1 1 1 1 1 1 1 1	1 1 1 1 1 1 1 1	1 1 1 1 1 1 1 1	239.255.255.255	
1 1 1 1 0 0 0 0	0 0 0 0 0 0 0 0	0 0 0 0 0 0 0 0	0 0 0 0 0 0 0 0	240.0.0.0	Class E
1 1 1 1 1 1 1 1	1 1 1 1 1 1 1 1	1 1 1 1 1 1 1 1	1 1 1 1 1 1 1 1	255.255.255.255	

Network Classes

Class	# of nets	# of hosts per net
A	128	$\sim 16\ million$
B	$16{,}384$	$65{,}534$
C	$\sim 2\ million$	254

IP addresses and networks

Every network device has an IP address

Every IP packet (datagram) contains the destination IP address

Network part of the IP address uniquely identifies a single physical network

- Part of the larger Internet.

Routers are connected to multiple network interfaces

- A router has multiple network adapters
- Routers can exchange packets on any network they're attached.

IP Forwarding algorithm

Executed by the router:

If I'm on the same network as the destination:

→ deliver packet to destination (ARP)

else: look up the *forwarding table*:

if the destination network is in forwarding table:

→ deliver packet to <u>NextHop</u> router

else: deliver packet to *default router*

- Forwarding tables
  - Contain (Network #, NextHop) pairs
  - Additional information
  - Built by routing protocol

So Far

- Bridges and Spanning Tree Algorithm
- IP
  - Basics
  - Addresses
  - Fragmentation and Reassembly

Fragmentation and Reassembly

Why?

- Networks differ on maximum packet size

How?

- Fragment packets into pieces
- Each fragment is itself a complete packet
- Receiving host reassembles them

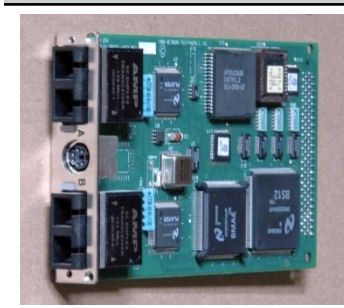
Maximum Transmission Unit (MTU)

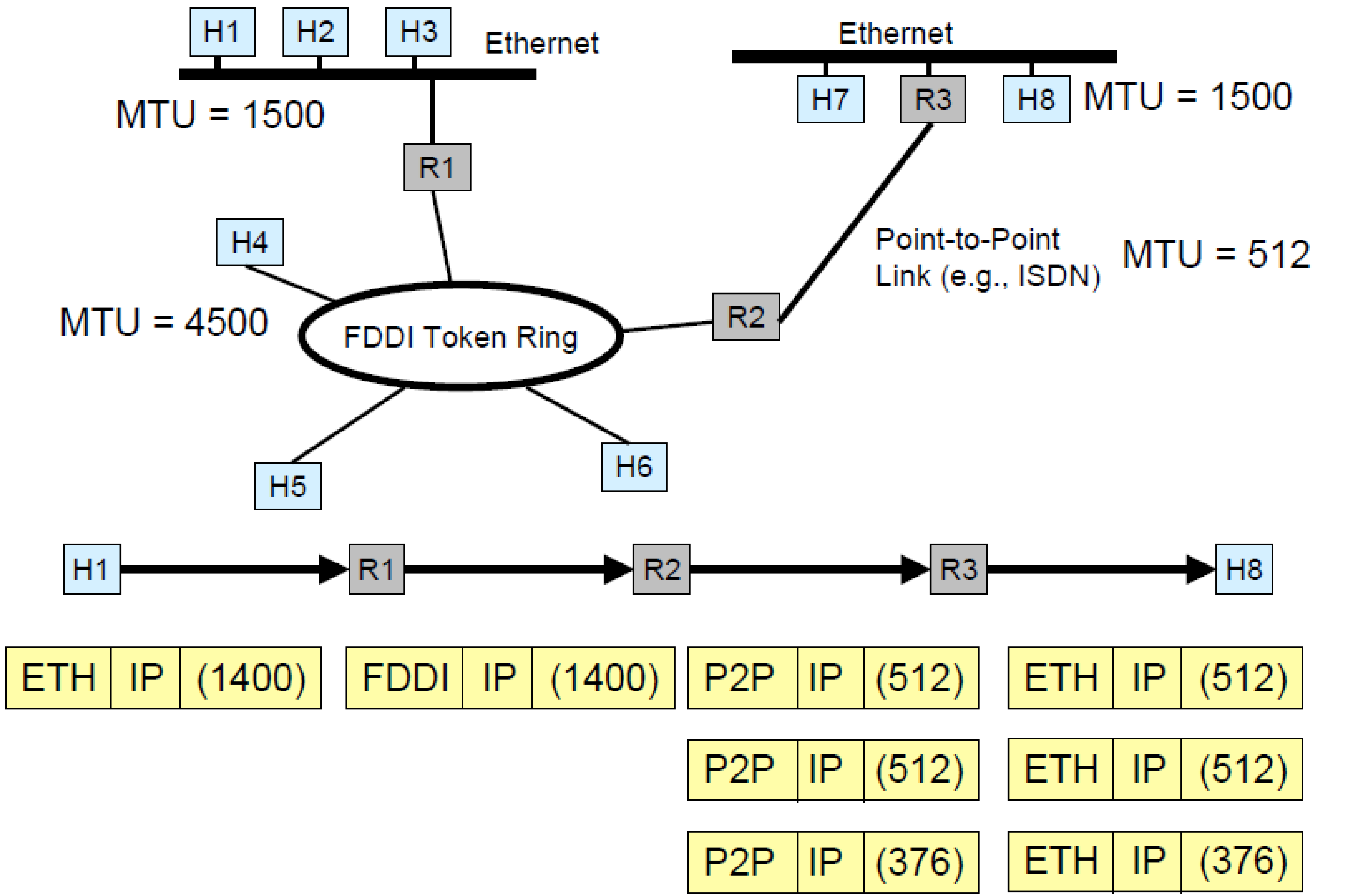
- Path MTU is min MTU for path
- Sender typically sends at MTU of first hop

https://www.lehighvalleylive.com/news/2019/05/truck-stuck-in-tunnel-closes-route-519-in-warren-county.html

Some MTU Numbers

Link Layer Technology	Notes	MTU
Ethernet	Standard home/office networking	1500B
PPPOE over Ethernet	Used for connecting DSL modems to ISP networks	1492B
802.11 (Wi-Fi)	Standard Wireless	2304B
Token Ring (802.5)	(Bottom right)	4464B
FDDI	(Bottom left)	4352B




H1
H2
H3
Ethernet
MTU = 1500
R1
Ethernet
H7
R3
H8
MTU = 1500
H4
Point-to-Point
Link (e.g., ISDN)
MTU = 512
MTU = 4500
FDDI Token Ring
R2
H5
H6
H1
R1
R2
R3
H8
ETH IP (1400)
FDDI IP (1400)
P2P IP (512)
ETH IP (512)
P2P IP (512)
ETH IP (512)
P2P IP (376)
ETH IP (376)

Packet Fragmentation

Unfragmented Packet

Start of Header				
Ident = x			0	Offset = 0
Rest of header				
1400 Bytes of data				

More to come flag

Fragmented Packet

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

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

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

$\text{Offset} \times 8 = \#bytes$

IPv4 vs IPv6

IPv4

Router or source may fragment packets

- Fragments arrive at destination for reassembly

Sender can set Don’t Fragment flag in header to prevent

- Router sends to ICMP TooBig error as needed

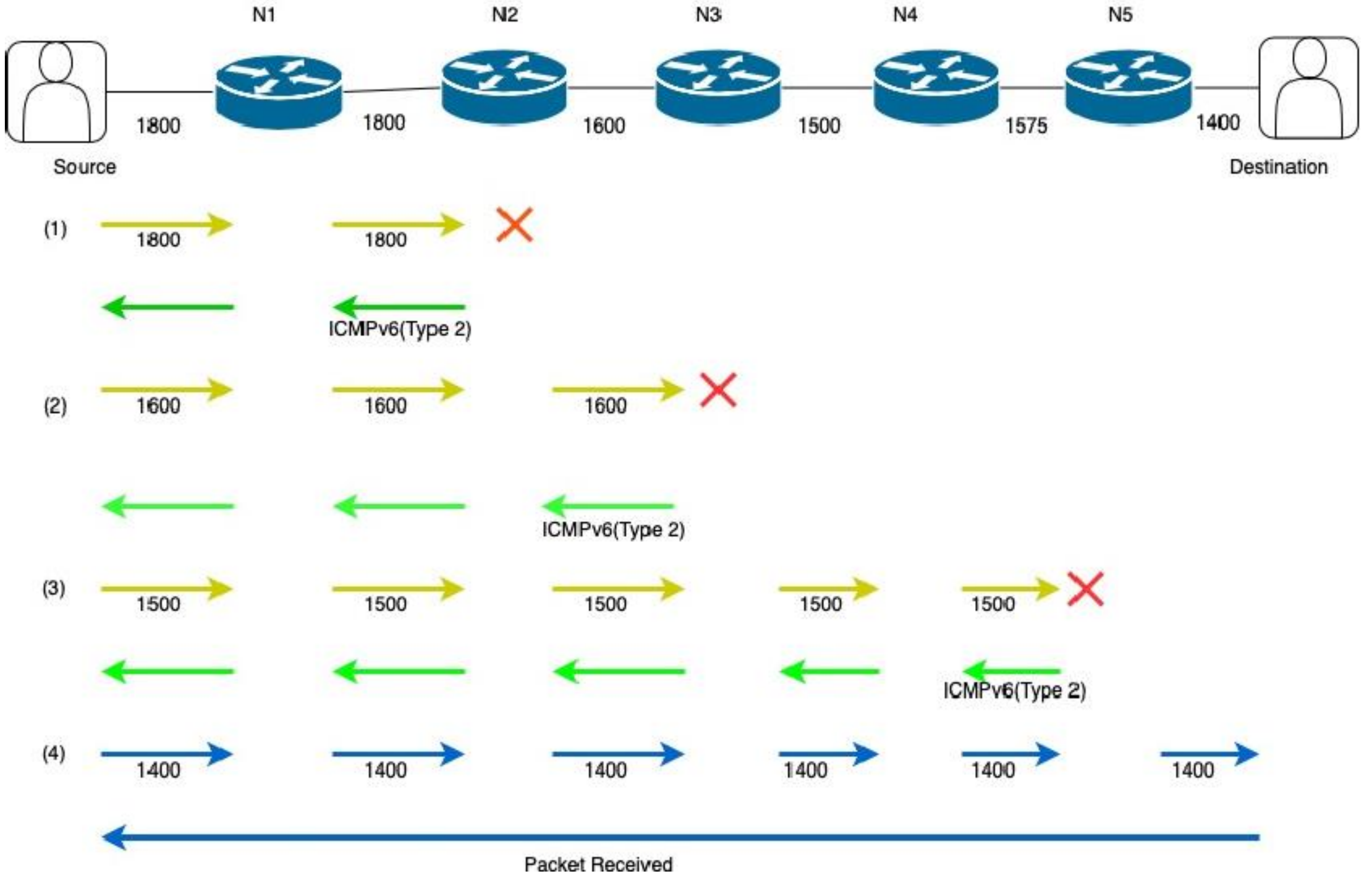
IPv6

Only source fragments

- Fragments arrive at destination for reassembly

Router sends ICMPv6 TooBig message as needed

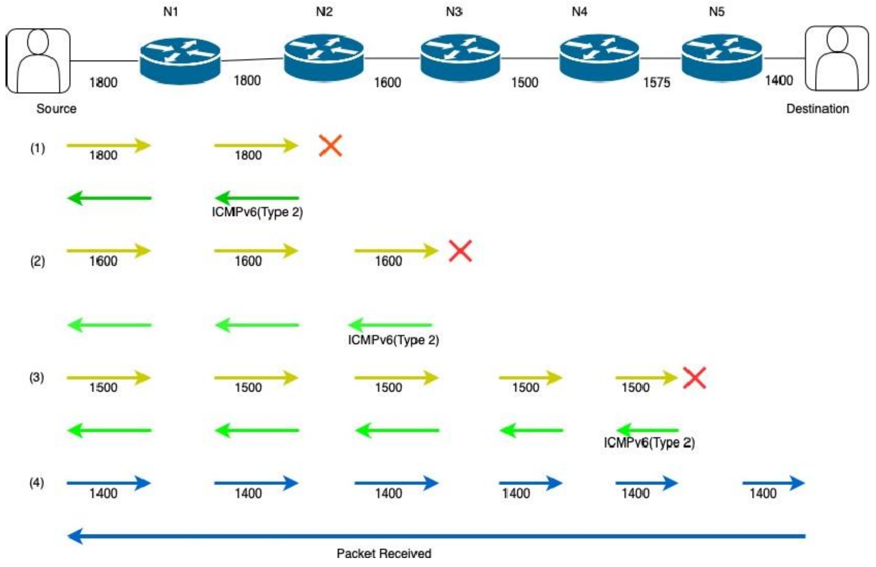
Path MTU Discovery



Hussain, Ishfaq and Janibul Bashir. "Measuring time delay in path MTU discovery in transmitting a packet in IPv4 and IPv6 network." ArXiv abs/2011.06935 (2020): n. pag.

Path MTU Discovery

1. Sender sends with DF flag in IPv4 or and IPv6
2. Router responds with TooBig message with its MTU
3. Sender records response and updates packet size
4. Sender discovers path MTU



Conclusion

- Bridges and Spanning Tree Algorithm
- IP
  - Basics
  - Addresses
  - Fragmentation and Reassembly