

15-441

Computer Networking

Internetworking - Forwarding

Sept. 27, 2004

Topics

- What's an Internet?
- IP forwarding

Slides

- Hui Zhang, Randy Bryant, Dave Eckhardt

Synchronization

Textbook

- Start reading Chapter 4 (mammoth!)
- Today
 - “Internetworking”
 - Section 4.1, plus some, minus some
- Upcoming
 - Routing (3 lectures): Section 4.2 (more or less)

The other kind of learning

- Which coding standard did you choose?
 - BSD/Linux/PDL?
- Which source control mechanism?
- Consider scheduled P2 meetings with your partner
 - You will be responsible reading for your partner's code...

Prelude

What does “TCP” stand for?

- What does it do?

What does “IP” stand for?

- What does it do that TCP doesn't???

“Internetworking”??

How can we study Internetworking: only one Internet?

- Lots of world-spanning digital networks
 - Telegraph, telephone
 - IBM VNET
 - DEC DECNET
 - X.25 public data network (Europe)
 - IBM SNA
 - Xerox Vines

What makes “The” Internet special?

“The Internet Way”

What makes “The Internet” special?

- (Eckhardt's opinion)

1. Heterogeneity

- The “Internet Problem” (Cerf & Kahn): three networks
 - Land-based computer network (ARPANET, ≠ Internet)
 - Mobile packet radio network
 - Satellite radio network
- Each network “had its own ideas”
 - Node address, packet size range, medium access control, ...
 - Each network carefully designed to do “its thing” well
 - » Environments very different, solutions very different
 - » No way to declare “one way to do things”

“The Internet Way”

2. TCP ≠ IP

- This decision was not immediately obvious
 - ARPANET “NCP” had one protocol
 - X.25 had one protocol
 - Everybody liked & understood reliable stream protocols
- In some ways this decision was bad...
 - ...(forward ref to congestion control lecture)...
 - ...“See, it really wasn't obvious!”

3. Semi-accidental “open standards” approach

- “Supercomputers” were very different
- From the beginning, lots of implementations of protocols
- Cooperative/competitive interoperability “bake-off” events
 - ...paid for by the (sole-source) funding agency.
- Result: “of course” protocol docs were widely available

“The Internet Way”

4. The “RFC approach”

- They're not really “standards documents”
- More like “Hey, look what I did!”
- As opposed to: “Observe my cool design for the future...”
- “IETF credo” (Dave Clark, 1992)
 - We reject kings, presidents, and voting.
 - We believe in rough consensus and running code.

Results

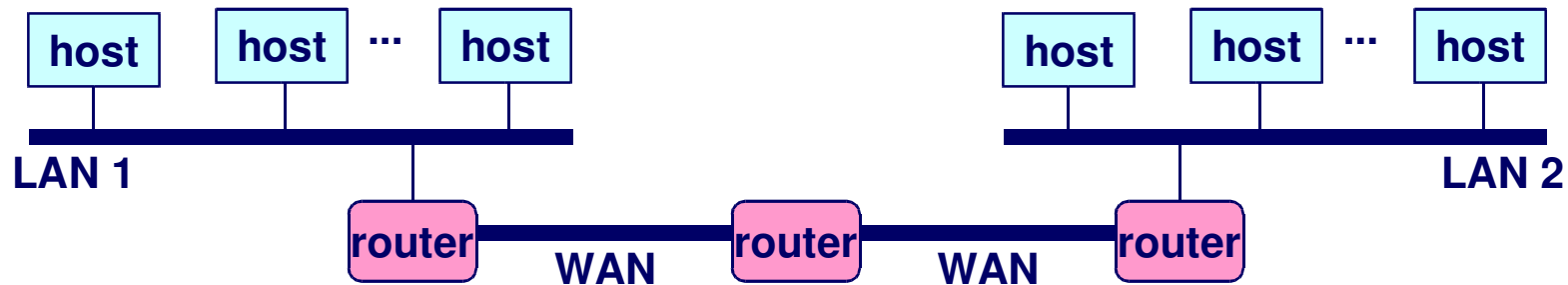
- Not just one network technology
- Not just one vendor's boxes, OS, database, ... (cf. IBM SNA)
- Room in the protocols for innovation
- Room in the culture for innovation

What is an Internetwork?

Multiple incompatible LANs can be physically connected by specialized computers called *routers*.

The connected networks are called an *internetwork*.

- The “*Internet*” is one (very big & successful) example of an internetwork



LAN 1 and LAN 2 might be completely different, totally incompatible LANs (e.g., Ethernet and ATM)

Issues in Designing an Internetwork

How do I designate a distant host?

- Addressing / naming

How do I send information to a distant host?

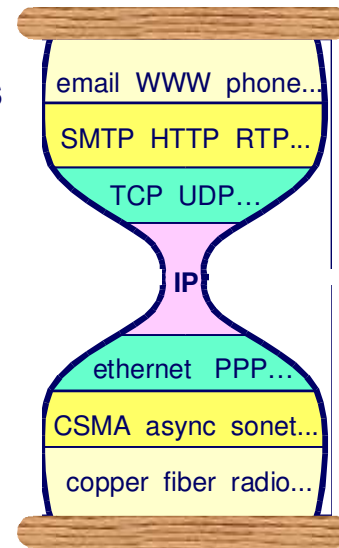
- “Service model”
 - What gets sent?
 - How fast will it go?
 - How often will it get there? Or else what?
- Routing – which path(s) will my information take?

Challenges

- Heterogeneity
 - Assembly from variety of different networks
- Scalability
 - Ensure ability to grow to worldwide scale

Internet Protocol (IP)

Network applications



Network technology

Steve Deering, CISCO

“Hour-glass” Model

- Abstraction layer hides underlying technology from network application software
- Make “as minimal as possible”
- Allow range of current & future technologies
- Can support many different types of applications

Agreeing, Disagreeing

How to address a distant host

- Or else we'll never get information there
- Ignore/work-around/use addressing method of local net

How to “containerize” data

- What goes in a link-layer “frame”?
 - Potentially many kinds of data (mux/de-mux)
 - Format of key control items (sender, receiver)
- Managing the size issue

We do not agree on

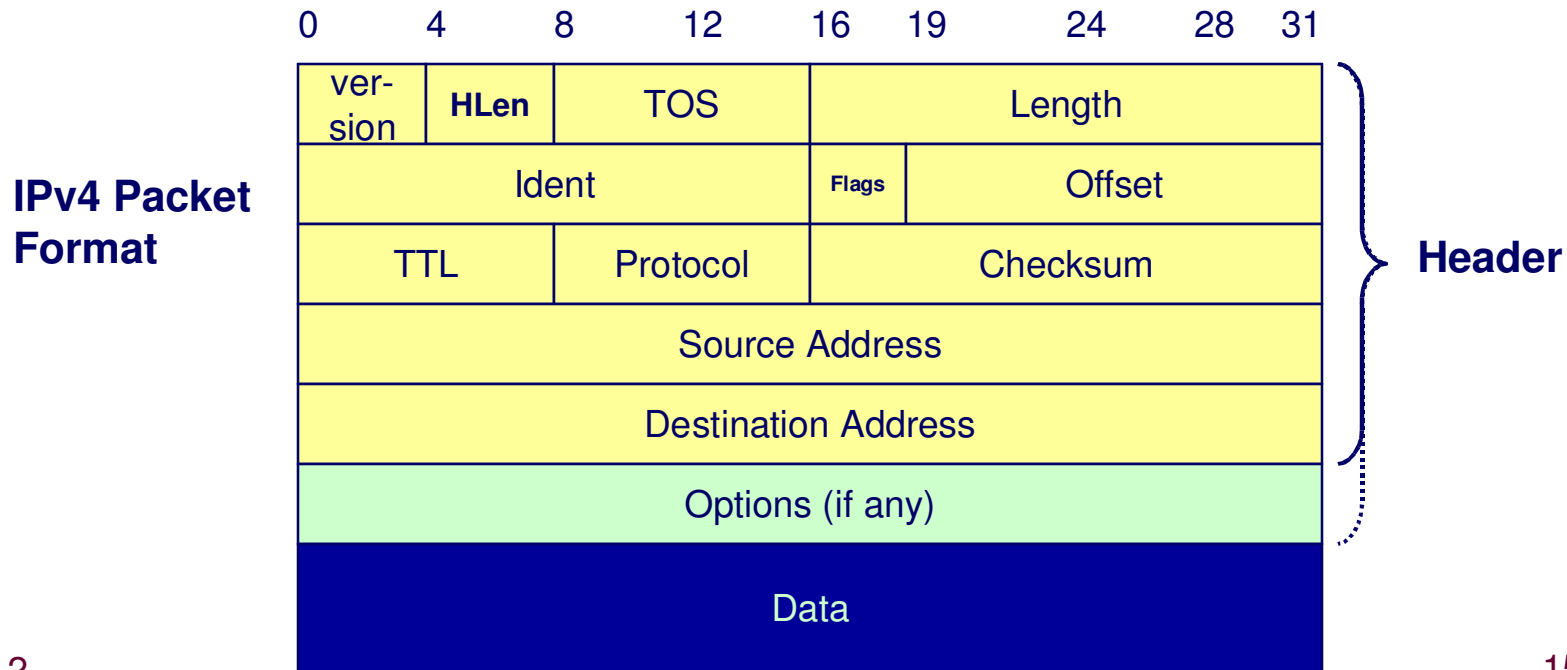
- Routing – Ethernet \neq van-mounted packet radios
- Precise service model – Token ring has link-level ACK

IP Service Model

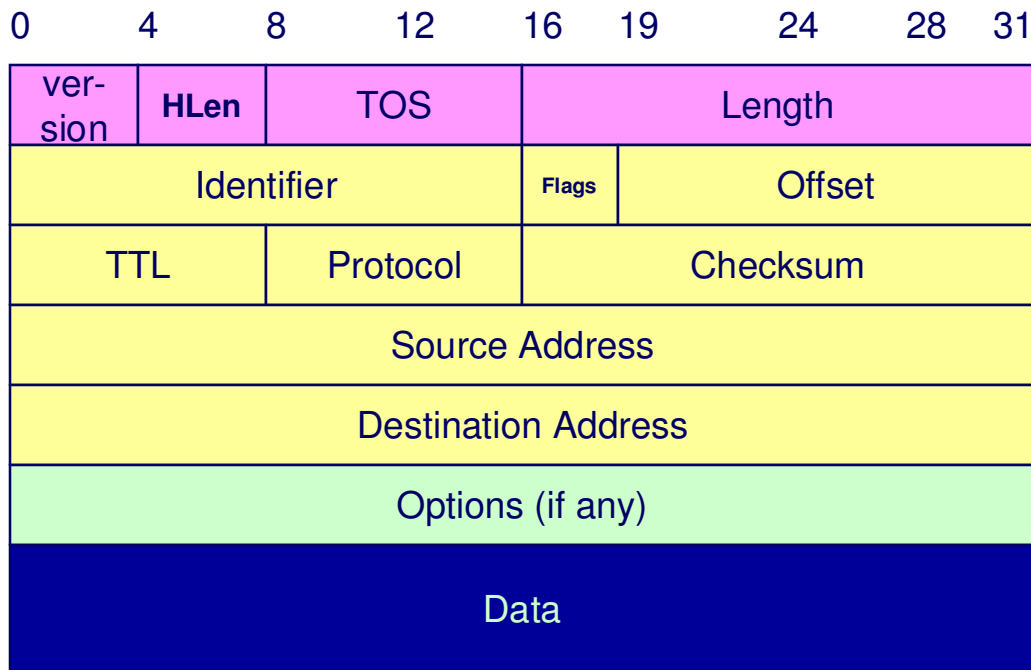
- Low-level communication model provided by Internet

Datagram

- Each packet self-contained
 - All information needed to get to destination
 - No advance setup or connection maintenance
- Analogous to letter or telegram



IPv4 Header Fields: Word 1



Version: IP Version

- 4 for IPv4

HLen: Header Length

- 32-bit words (typically 5)

TOS: Type of Service

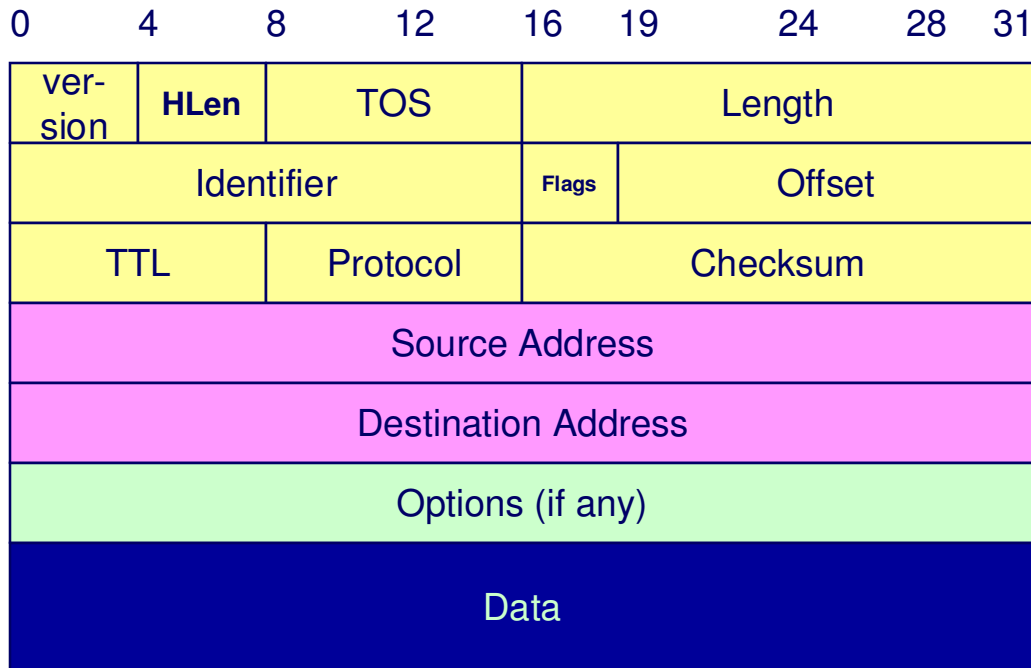
- Priority information

Length: Packet Length

- Bytes (including header)

- Header format can change with versions
 - First byte identifies version
- Length field limits packets to 65,535 bytes
 - In practice, most “sub” networks work better with smaller packets than that

IPv4 Header Fields: Words 4&5



Source Address

- 32-bit IP address of sender

Destination Address

- 32-bit IP address of destination

- Like the addresses on an envelope
- In principle, globally unique identification of sender & receiver
 - In practice, there are contexts where either source or destination are not the ultimate addressees

IP Addressing

IPv4: 32-bit addresses

- Typically written in “dotted quad” format
 - E.g., 128.2.198.135
 - Each number is decimal representation of byte
- Big-Endian Order

0	8	16	24	31	
128	2	198	135		Decimal
80	02	c6	87		Hexadecimal
0100 0000	0000 0010	1100 0110	1000 0111		Binary

Translation from Network Names

- Performed by Domain Name System (DNS)

```
unix> host bryant.vlsi.cs.cmu.edu
```

```
bryant.vlsi.cs.cmu.edu has address 128.2.198.135
```

IP Delivery Model

Best Effort Service

- Network will do its best to get packet to destination

Does NOT Guarantee...

- Any maximum latency or even ultimate success
- Sender will be informed if packet doesn't make it
- Packets will arrive in same order sent
- Only one copy of packet will arrive

Implications

- Scales very well
- Higher level protocols must make up for shortcomings
 - Reliably delivering ordered sequence of bytes
- Some services not feasible
 - Latency or bandwidth guarantees

Basic Internet Components

An ***Internet backbone*** is a collection of routers (nationwide or worldwide) connected by high-speed point-to-point networks.

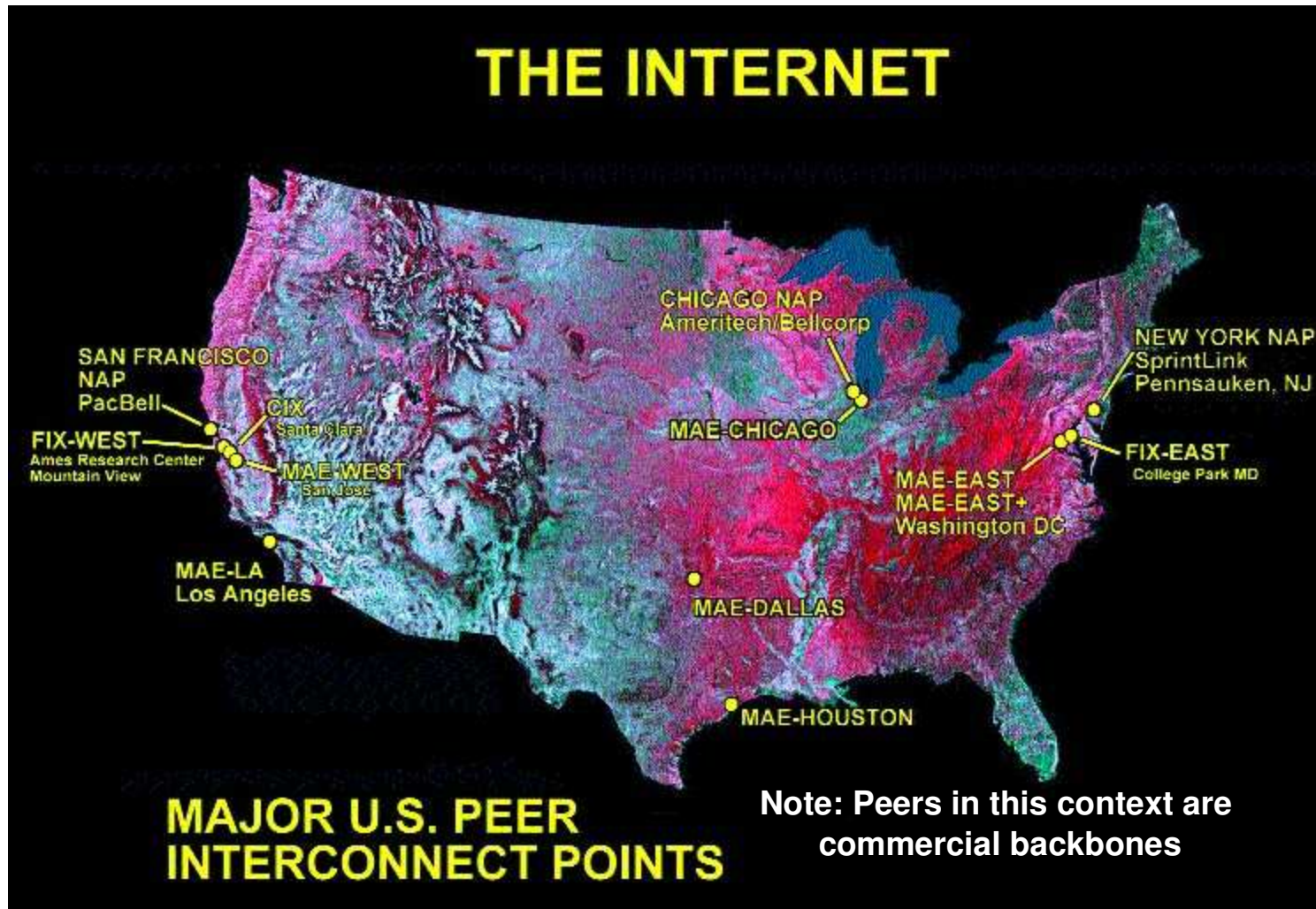
A ***Network Access Point (NAP)*** is a router that connects multiple backbones (sometimes referred to as *peers*).

Regional networks are smaller backbones that cover smaller geographical areas (e.g., cities or states)

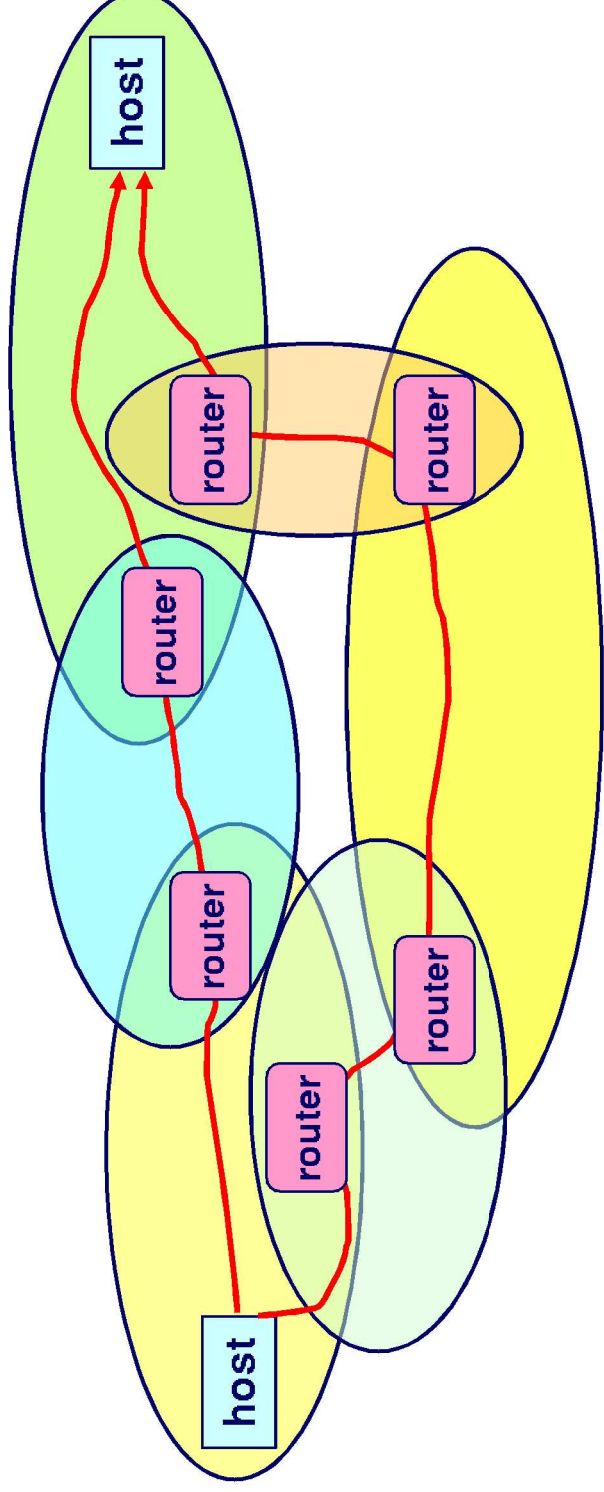
A ***point of presence (POP)*** is a machine that is connected to the Internet.

Internet Service Providers (ISPs) provide dial-up or direct access to POPs.

Network Access Points (NAPs)

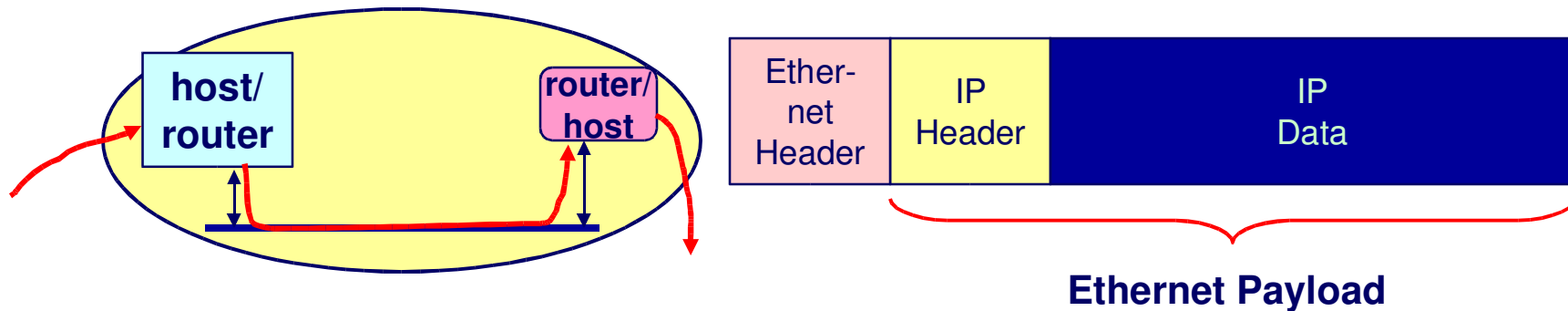


Logical Structure



- Ad hoc interconnection of networks
 - No particular topology
 - Vastly different router & link capacities
- Send packets from source to destination by hopping through networks
 - Router forms ?bridge? from one network to another
 - Different packets may take different routes
 - » OK, since IP doesn't guarantee ordering

Routing Through Single Network



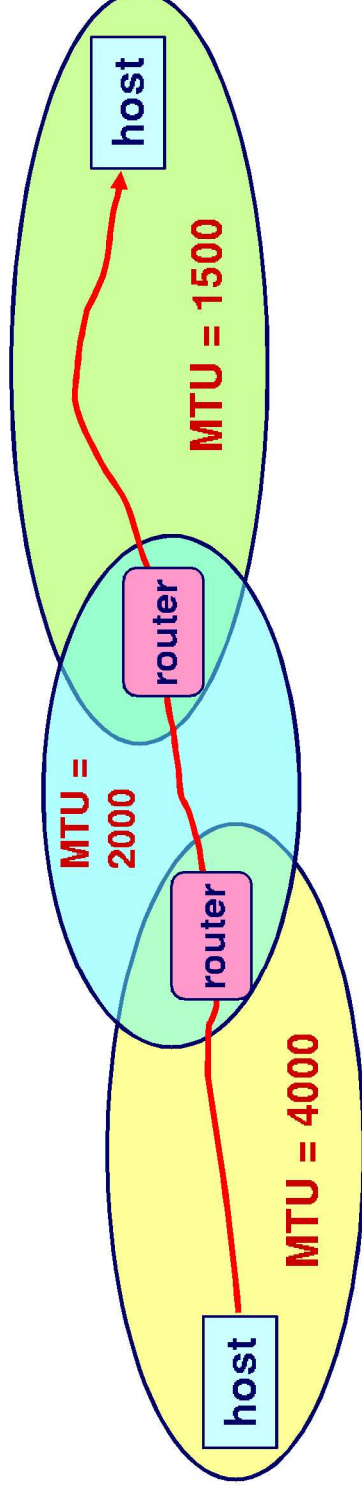
Path = series of hops

- Source \Rightarrow Router
- Router \Rightarrow Router (typically high-speed, point-to-point link)
- Router \Rightarrow Destination

Each hop crosses one link, uses that link-layer protocol

- Hop destination = function(IP destination)
- Encapsulate IP packet as payload for local subnetwork
 - Key step: link-layer address of next-hop router on this subnetwork
 - » Multi-point network (Ethernet): "MAC address" (see ARP, later)
 - » Point-to-point network (PPP, 15-441): no need for address

IP Fragmentation



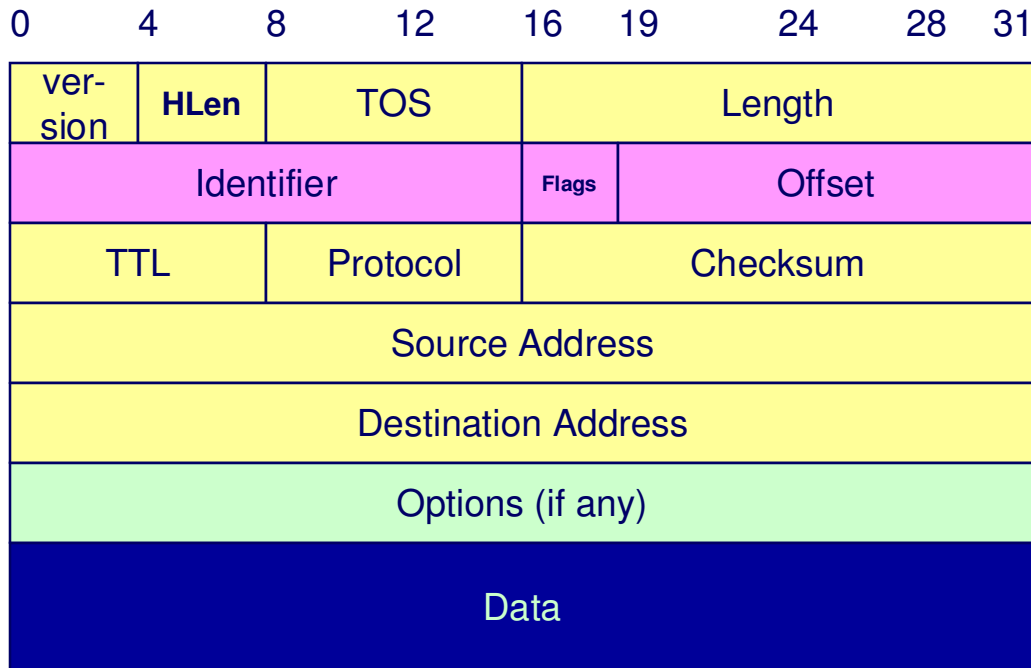
Every Network has Own Maximum Transmission Unit (MTU)

- Largest IP datagram it can carry within its own packet frame
 - E.g., Ethernet is 1500 bytes
- Don't know MTUs of all intermediate networks in advance

IP Solution

- When hit network with small MTU, fragment packet
 - Packet may fragment further on a later link
- Reassemble at the destination
 - If any fragment disappears, delete entire packet

IPv4 Header Fields: Word 2



Identifier

- Unique identifier for original datagram
 - Typically, source increments counter every time sends packet

Flags (3 bits)

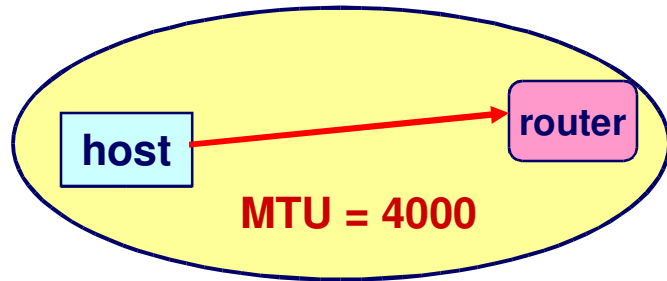
- M flag: This is not the last fragment

Offset

- Byte position of first byte in fragment $\div 8$
- Byte position must be multiple of 8

- Each fragment carries copy of IP header
 - All information required for delivery to destination
- All fragments comprising original datagram have same identifier
- Offsets indicate positions within datagram

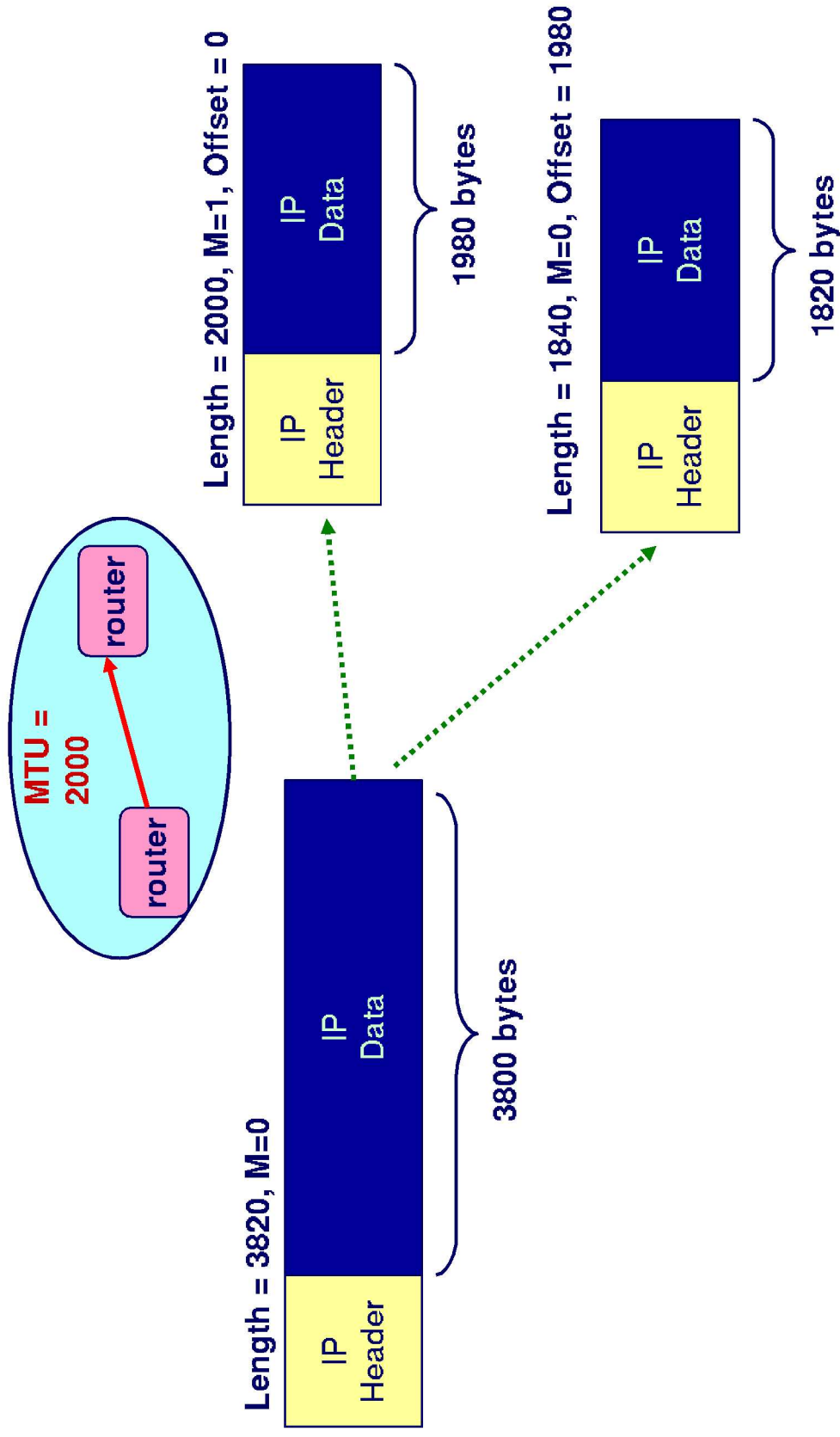
IP Fragmentation Example #1



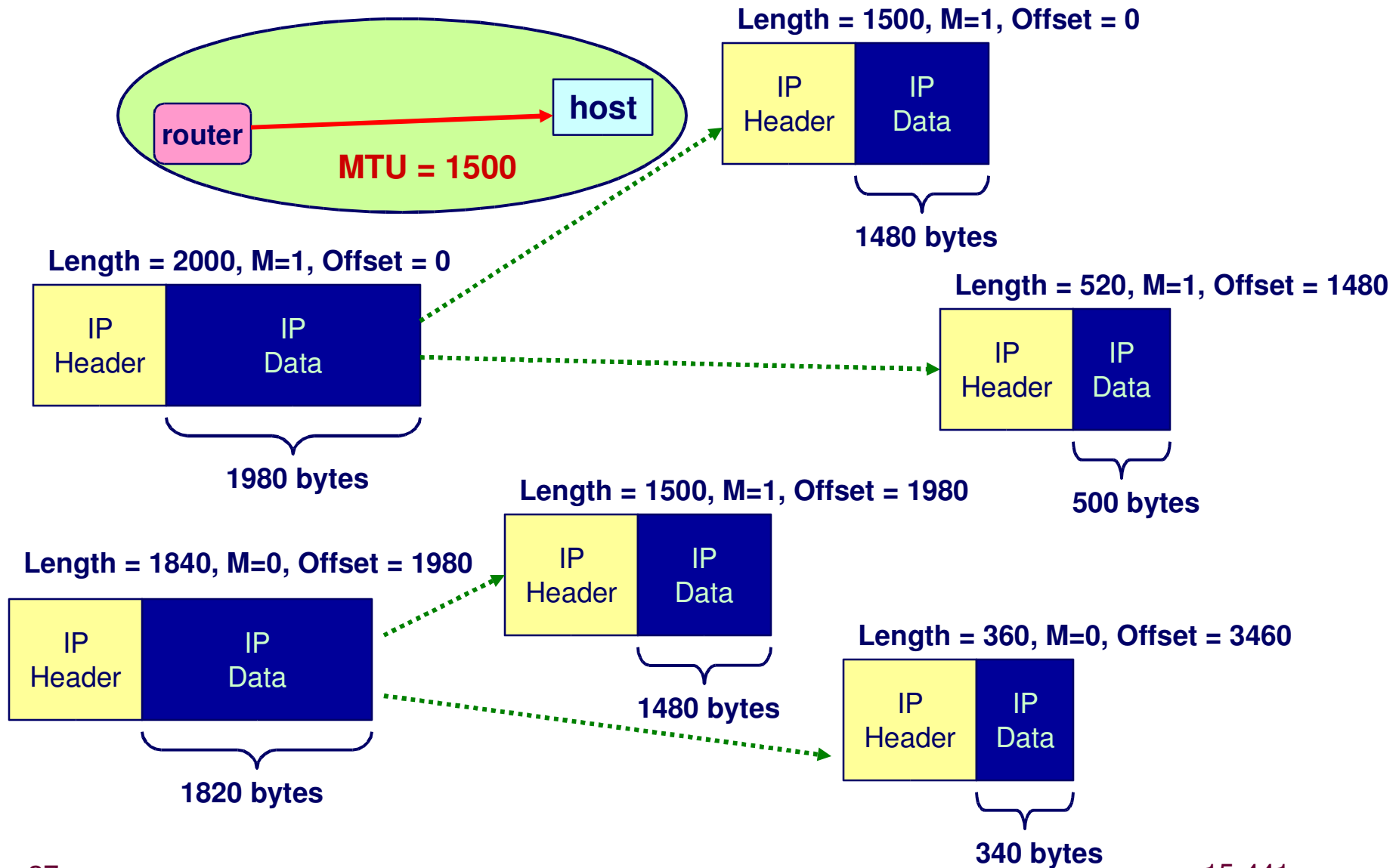
Length = 3820, M=0



IP Fragmentation Example #2



IP Fragmentation Example #3



IP Reassembly

Length = 1500, M=1, Offset = 0



Length = 520, M=1, Offset = 1480



Length = 1500, M=1, Offset = 1980



Length = 360, M=0, Offset = 3460



- Performed at final destination
- Fragment with M=0 determines overall length
 - $(360-20)+3460$

Challenges

- Fragments might arrive out-of-order
 - Don't know how much memory required until receive final fragment
- Some fragments may be duplicated
 - Keep only one copy
- Some fragments may never arrive
 - After a while, give up entire process
- Significant memory management issues
 - See code in book

Frag. & Reassembly Concepts

Demonstrates Many Internet Concepts

Decentralized

- Every network can choose MTU

Connectionless Datagram Protocol

- Each (fragment of) packet contains full routing information
- Fragments can proceed independently and along different routes

Fail by Dropping Packet

- Destination can give up on reassembly
- No need to signal sender that failure occurred

Keep Most Work at Endpoints

- Reassembly

Frag. & Reassembly Reality

Reassembly Fairly Expensive

- Copying, memory allocation
- Want to avoid

MTU Discovery Protocol

- Protocol to determine MTU along route
 - Send packets with “don’t fragment” flag set
 - Keep decreasing message lengths until packets get through
- Assumes every packet will follow same route
 - Routes tend to change slowly over time

Common Theme in System Design

- Assure correctness by implementing complete protocol
- Optimize common cases to avoid full complexity

Summary: Forwarding Layer Tasks

Hacker's inventory approach

- Examine each bit field, read what RFC says

Higher-level approach

- Mini-integrity check
 - Sanity-check link layer and IP layer of your neighbor (“peer”)
- Housekeeping/manipulation step
 - Loop detection, ...
- Consult your “next-hop oracle” (routing system)
- Adapt datagram to next-hop network
 - Size, priority/queue policy, ...
- Transmit