

ELEC / COMP 177 – Fall 2012

Computer Networking

→ Internet Protocol (IP)

Some slides from Kurose and Ross, *Computer Networking*, 5th Edition

Schedule

- Topics
 - Next 2 weeks - Network layer
 - IP, ARP, ICMP, Routers, Routing protocols
- Assignments
 - **Project 1:** Tonight by Midnight
 - **How's it coming?**
 - **Homework 4:** Due Thursday, Nov 1st
 - **Project 2:** Due Tuesday, Nov 6th

Project #1

- What was the most difficult part of this project?
 - *What solution did you find?*

Network Layer – IP

Why not just use Ethernet?

- Most computer systems use Ethernet networking
- Ethernet provides facilities to
 - Locate computers
 - Forward packets directly
 - Prevent loops
 - ...
- What are the drawbacks of Ethernet for global communication?

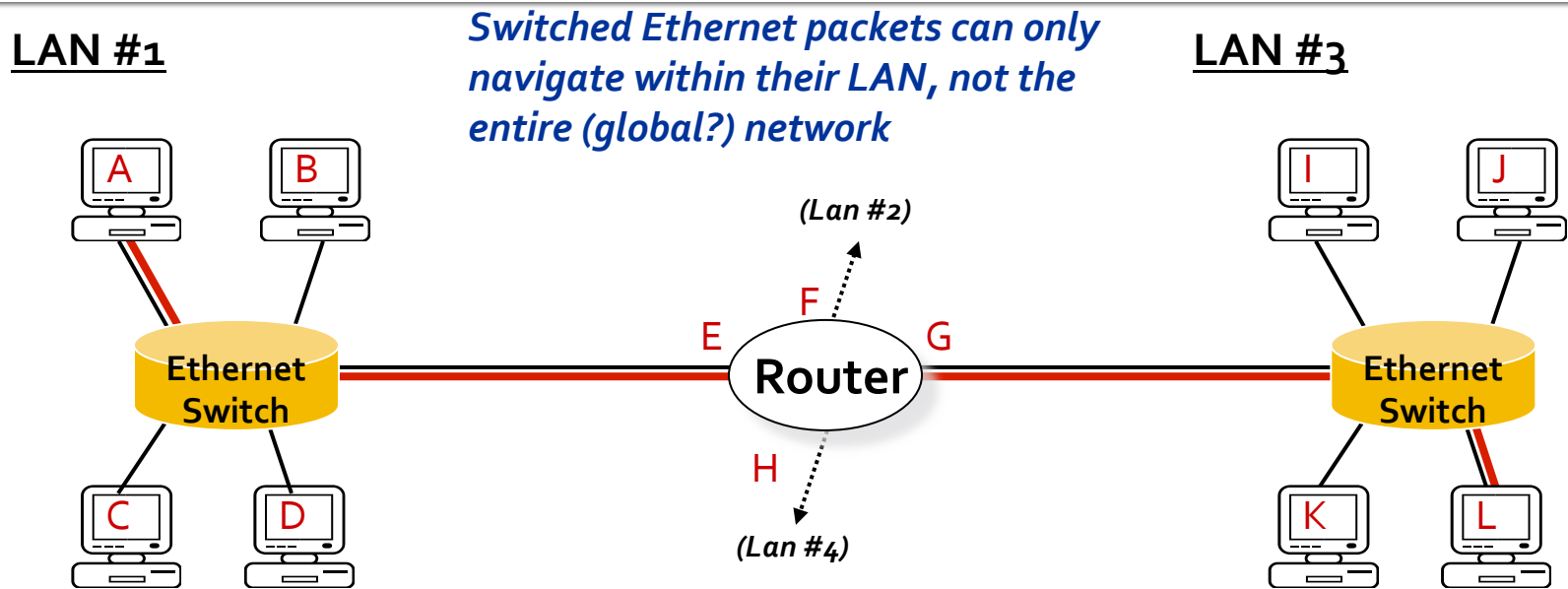
Ethernet Drawbacks

- Locating computers
 - Do we really want to broadcast across the Internet?
- Preventing loops
 - Do we really want to rebuild an Internet-wide spanning tree whenever the topology changes?
 - Do we really want packets to live forever if loops remain?
- Unreachable computers
 - What happens if the destination is unreachable?
 - I.e., it doesn't exist, is turned off, is broken, ...

Network vs. Link Layer

- “Most” ≠ “All”
 - Not all networks are Ethernet
 - Why limit choice, innovation, etc. at the link level?
- Link layer
 - What is the best way to move local traffic (single hop)?
 - Old/new network, wired/wireless network, ...
 - Different links can use different networks!
- Network layer
 - What is the best to handle multi-hop communication?
 - Addressing
 - Unreliable delivery mechanisms (routing)

Routing Between LANs



(1) A transmits to L using higher-level protocol (e.g. IP)
 Ethernet frame destination is router

(3) Router uses higher-level protocol to determine destination, and updates Ethernet frame destination, source and CRC

Frame:

DA (E)	SA (A)	Type / Data	CRC
--------	--------	-------------	-----

Frame:

DA (L)	SA (G)	Type / Data	CRC
--------	--------	-------------	-----

(2) Switch forwards frame to router

(4) Switch forwards frame to destination

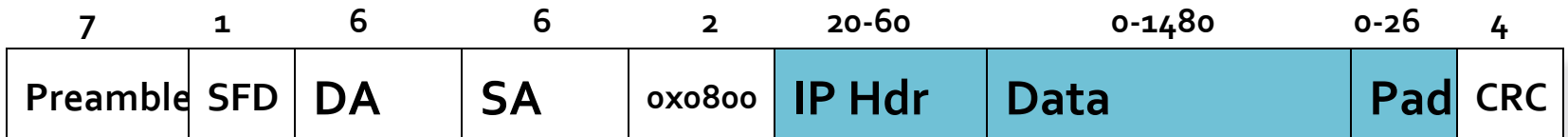
The Internet Protocol

- Datagram
 - Each packet is individually routed
 - Packets may be fragmented or duplicated
 - Due to underlying networks
- Connectionless
 - No guarantee of delivery in sequence
- Unreliable
 - No guarantee of delivery
 - No guarantee of integrity of data
- Best effort
 - Only drop packets when necessary
 - No time guarantee for delivery

IP and Ethernet

- IP datagrams can be *encapsulated* in Ethernet frames

Bytes:



IP Datagram

Understanding IP

- Datagram lifetimes
 - Time-to-live
- Handling disparate link layers
 - Fragmentation
- IP integrity
 - Header checksum
- What do datagrams look like?
 - Header format
- Addressing

Time-to-Live

- Sender sets a TTL value for each datagram
- Each router decrements the TTL
- When the TTL reaches 0
 - The router drops the datagram
 - The router sends an ICMP error (more later) to the sender
- Effectively a “maximum hop count”
- **Why is this useful / necessary?**

Traceroute

- Tool to find the route IP packets take through the Internet
- Exploits the TTL field
 - Send packets with successively increasing TTL values
 - See what routers respond with ICMP “Time Exceeded” errors
- Only shows us routers, not Ethernet switches
 - Only works at network layer, not link layer

Using Traceroute

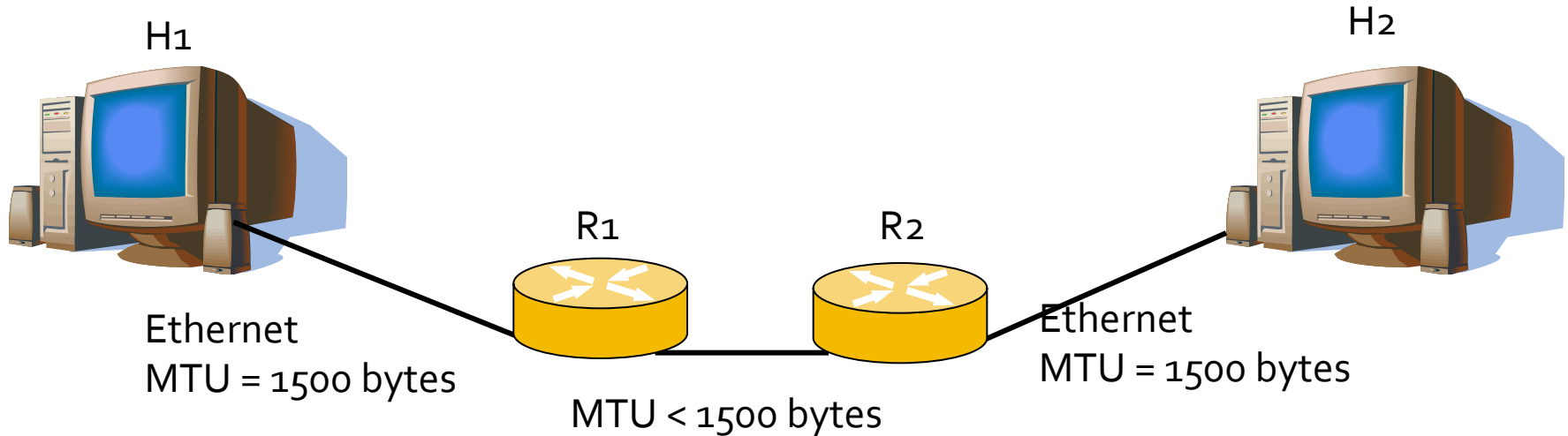
```
traceroute to www.l.google.com (74.125.224.52), 64 hops max, 52 byte packets
 1  10.10.207.254 (10.10.207.254)  0.737 ms  0.296 ms  0.304 ms
 2  10.0.0.133 (10.0.0.133)  0.387 ms  0.328 ms  0.344 ms
 3  10.0.0.105 (10.0.0.105)  0.401 ms  0.353 ms  0.357 ms
 4  138.9.253.252 (138.9.253.252)  0.748 ms  0.664 ms  1.319 ms
 5  74.202.6.5 (74.202.6.5)  6.836 ms  7.932 ms  5.986 ms
 6  paol-pr1-ge-3-0-0-0.us.twtelecom.net (66.192.242.70)  7.275 ms  7.643 ms  7.529 ms
 7  216.239.49.250 (216.239.49.250)  8.627 ms  58.979 ms  17.394 ms
 8  64.233.174.15 (64.233.174.15)  17.445 ms  9.161 ms  8.453 ms
 9  74.125.224.52 (74.125.224.52)  8.016 ms  8.325 ms  8.505 ms
```

TTL Value

Router name/IP address

Round trip time of 3 probes

The Maximum Transmission Unit



- Not all networks have the same maximum transmission unit (MTU)
- IP datagrams can be larger (64KB) than the Ethernet MTU (1500 bytes)
- Would like IP datagrams to still get delivered!

Fragmentation

- **Routers fragment IP datagrams**
 - Break datagram into MTU-sized datagrams
 - Set "MF" (more fragments) flag, as necessary
 - Set fragment offset field appropriately
- May need to fragment already fragmented datagrams!
- Datagrams are reassembled by the receiver
 - Routers do not reassemble fragments
 - Once fragmented, a datagram remains fragmented

Routing Fragments

- Fragments are routed independently through the network
- Fragment could be lost or duplicated
- Receiver must reassemble fragments
 - May arrive out of order
 - May not all arrive – must drop entire datagram

Avoiding Fragmentation

- Source can prevent fragmentation with “DF” (don’t fragment) flag
 - Routers send “ICMP” (more later) error to the sender if packet is too large for link MTU
- Hosts commonly try to avoid fragmentation
 - Use “path MTU Discovery” to find the smallest MTU on the path
 - Repeatedly send successively smaller datagrams until no fragmentation occurs
 - Try it yourself with “tracert -F *host size*”

IP Checksum

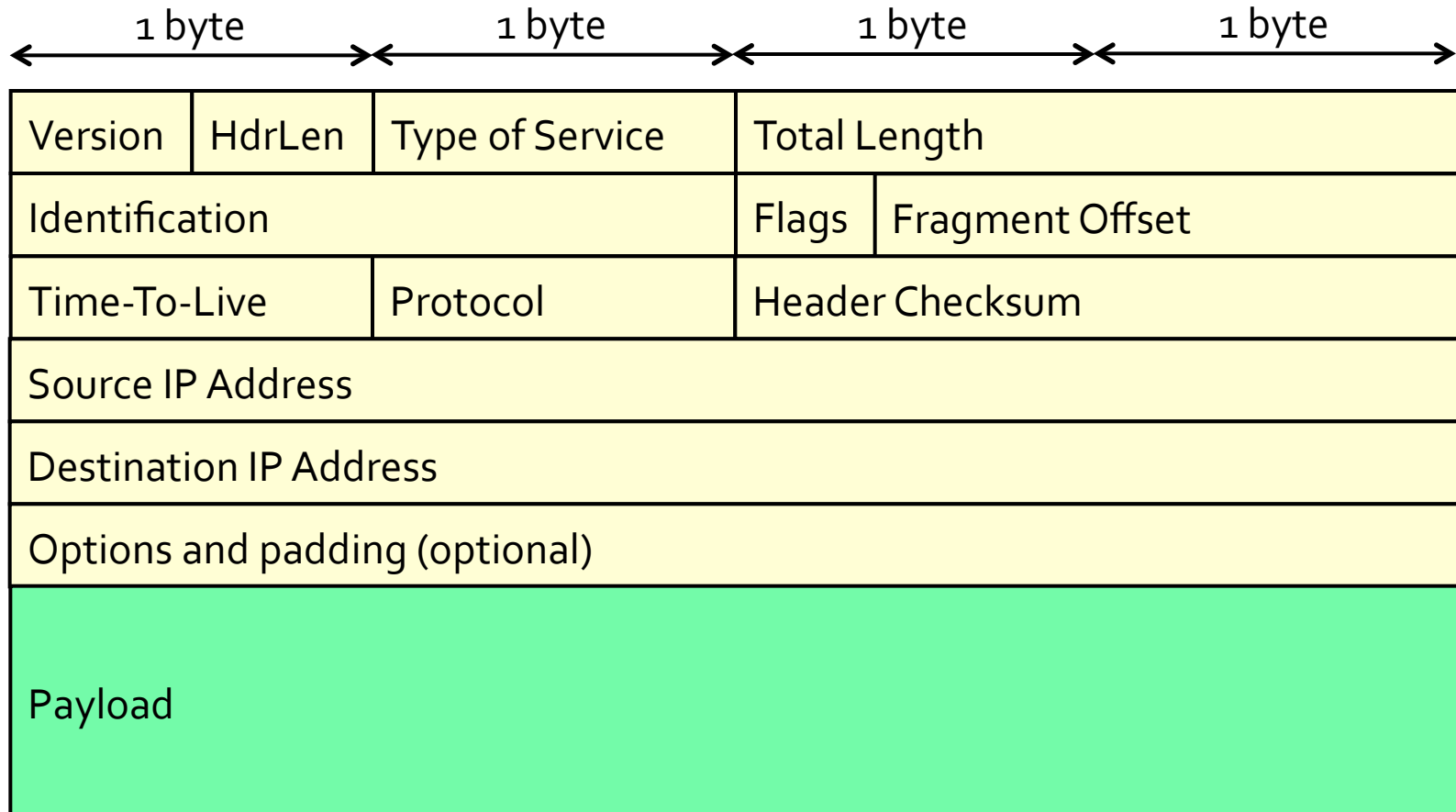
- 16-bit one's complement checksum
 - Allows hosts/routers to detect corruption in the IP header
 - Payload is not covered
 - Not enough information to correct errors
- Routers must modify the TTL of every IP datagram
 - Thus, checksum must be recomputed **by each router!**
- **Do checksums provide reliability?**
 - **Why are they included in the IP header?**

IP Header Format

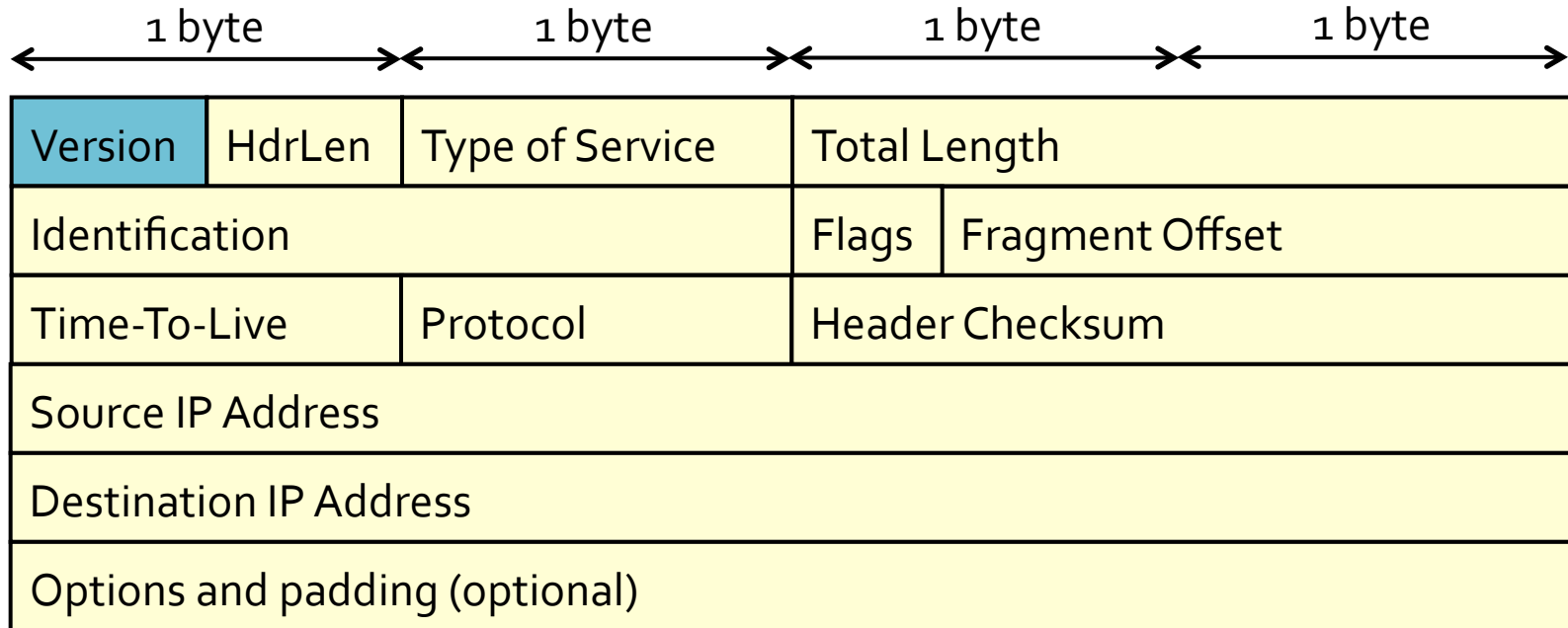
Packet Header Formats

- You will be dealing with these headers throughout the class
- It is easy to be off by a byte or forget something
- www.networksorcery.com is a great resource

An IP Datagram

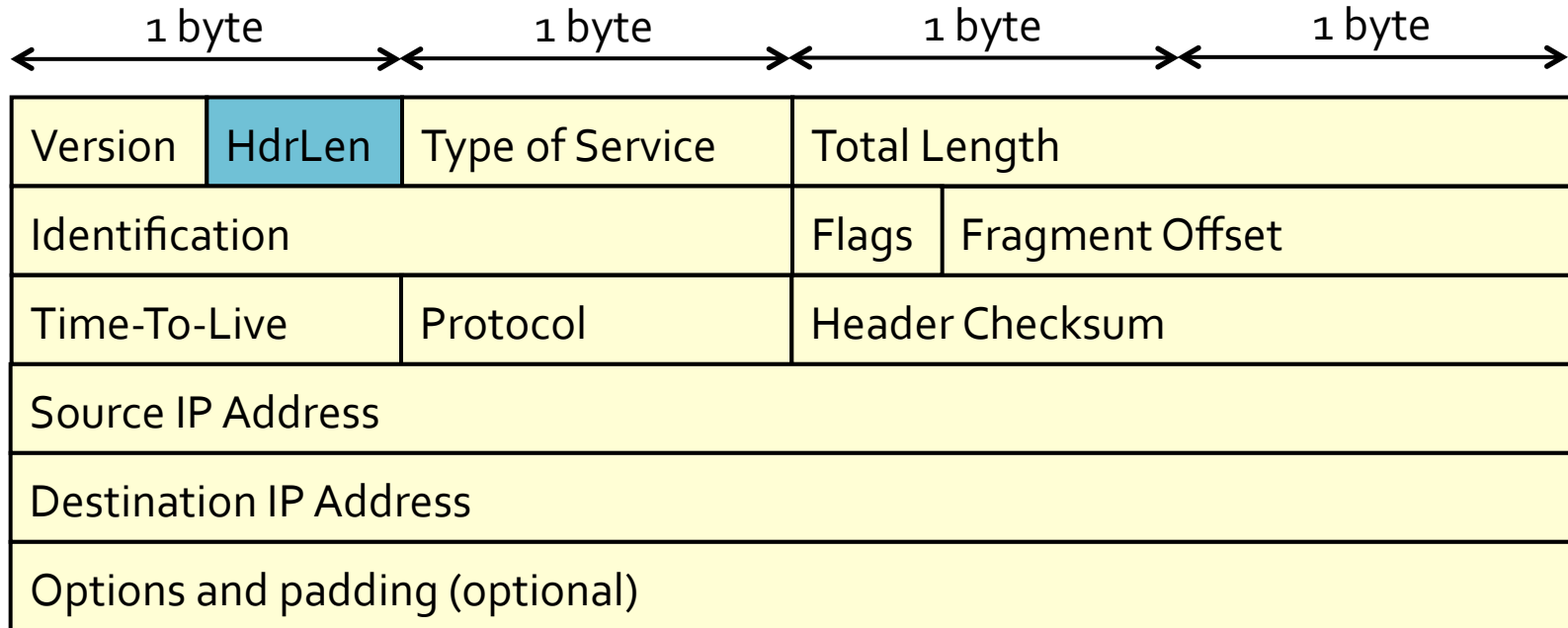


IP Version



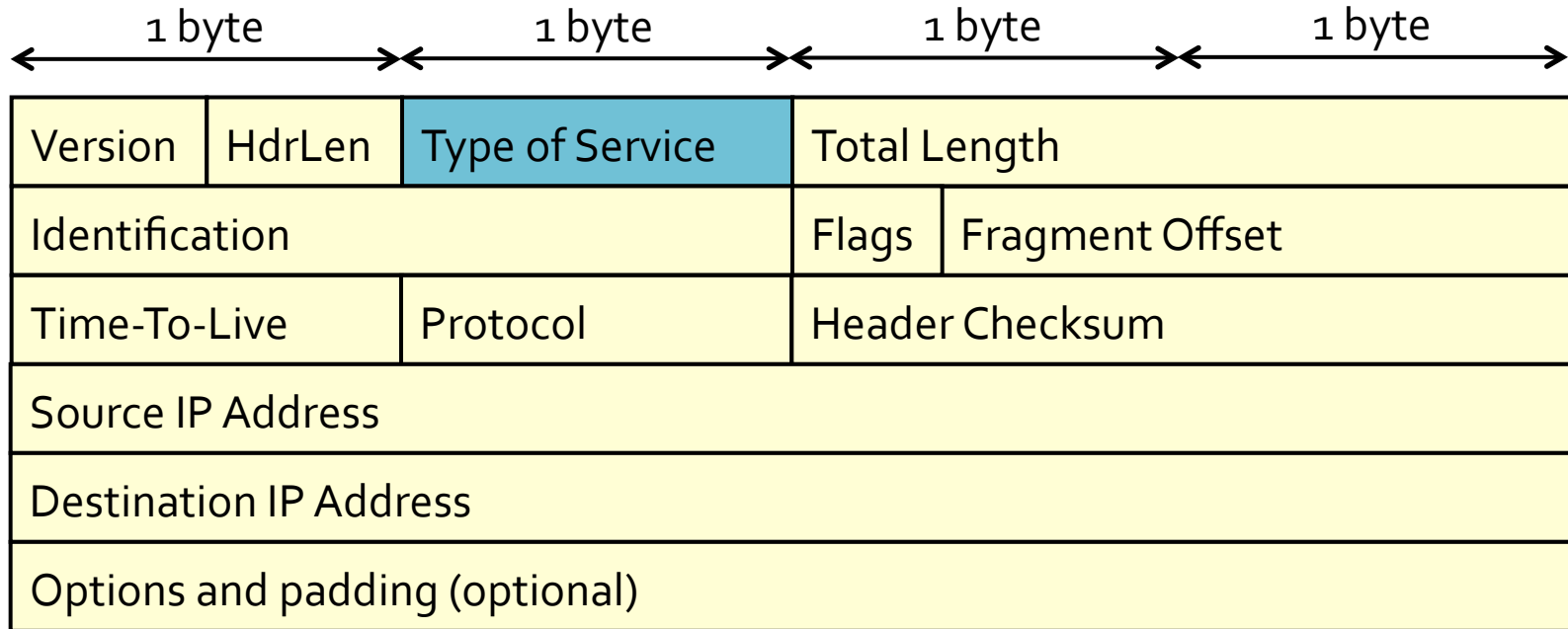
- IPv4 or IPv6
 - Also other, uncommon, options

Header Length



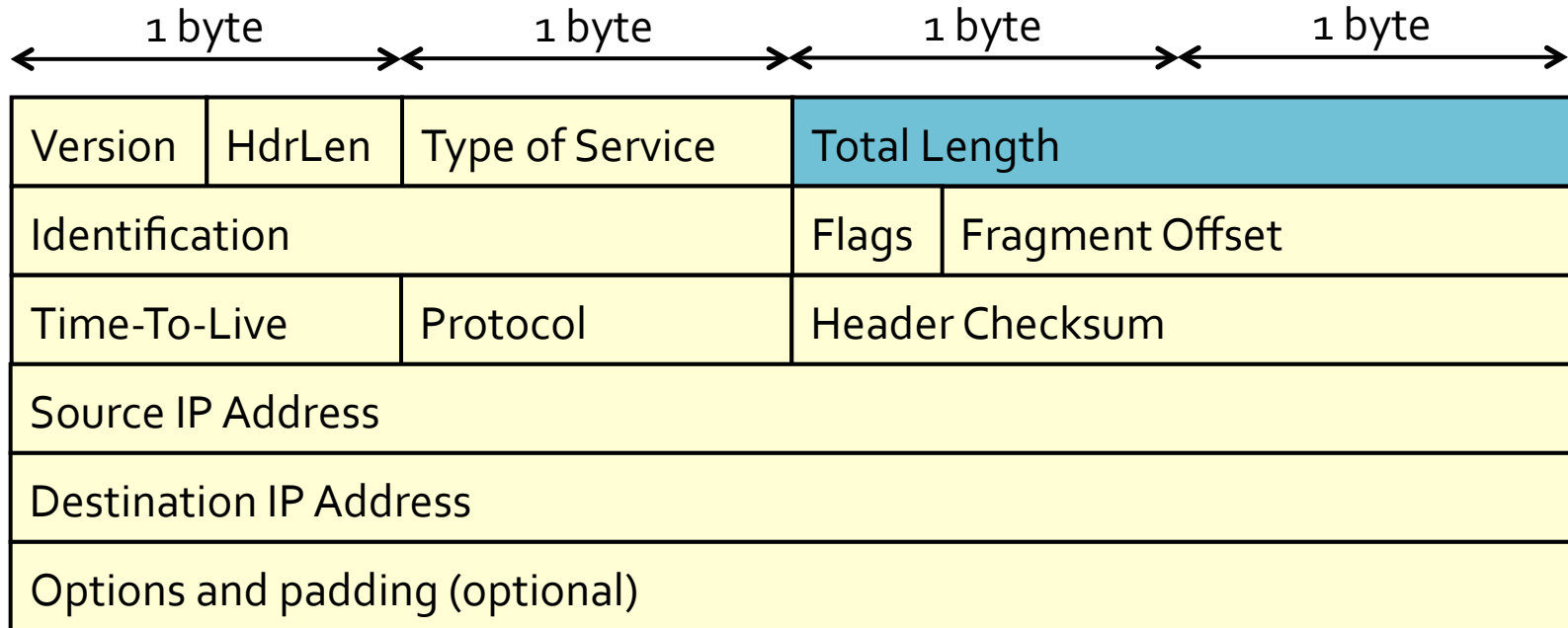
- Number of 32-bit words in the header
- Need to know how many options are present

Type of Service



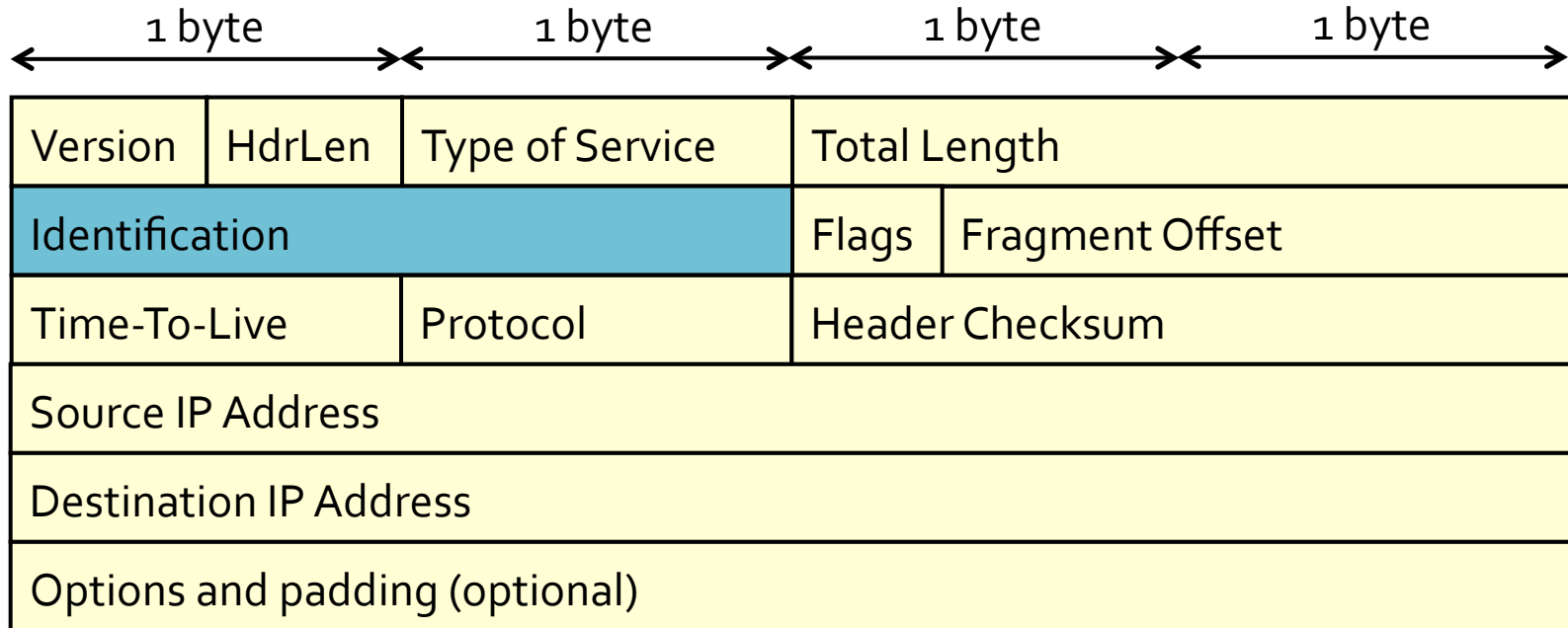
- Routine, priority, immediate, control, etc.
- High/low delay/throughput/reliability

Type of Service



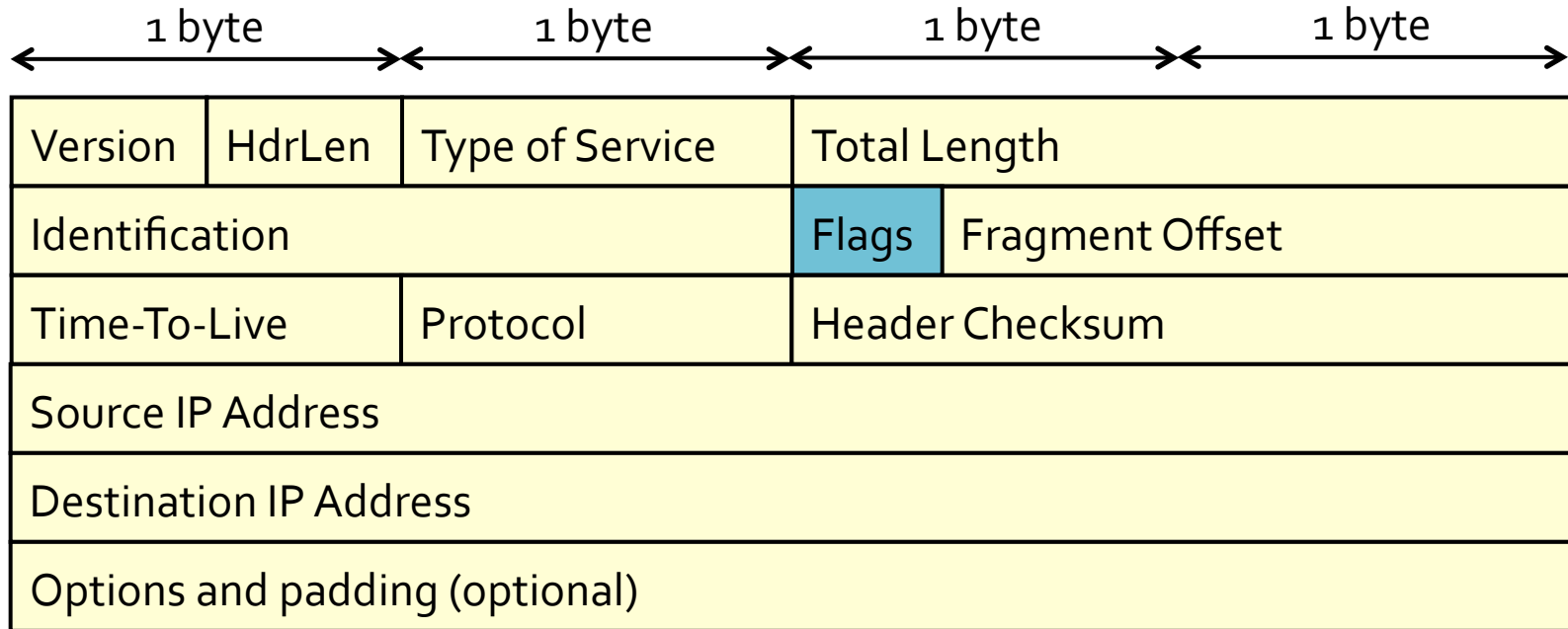
- Need to know the payload length
- Number of total bytes of IP packet

Identification



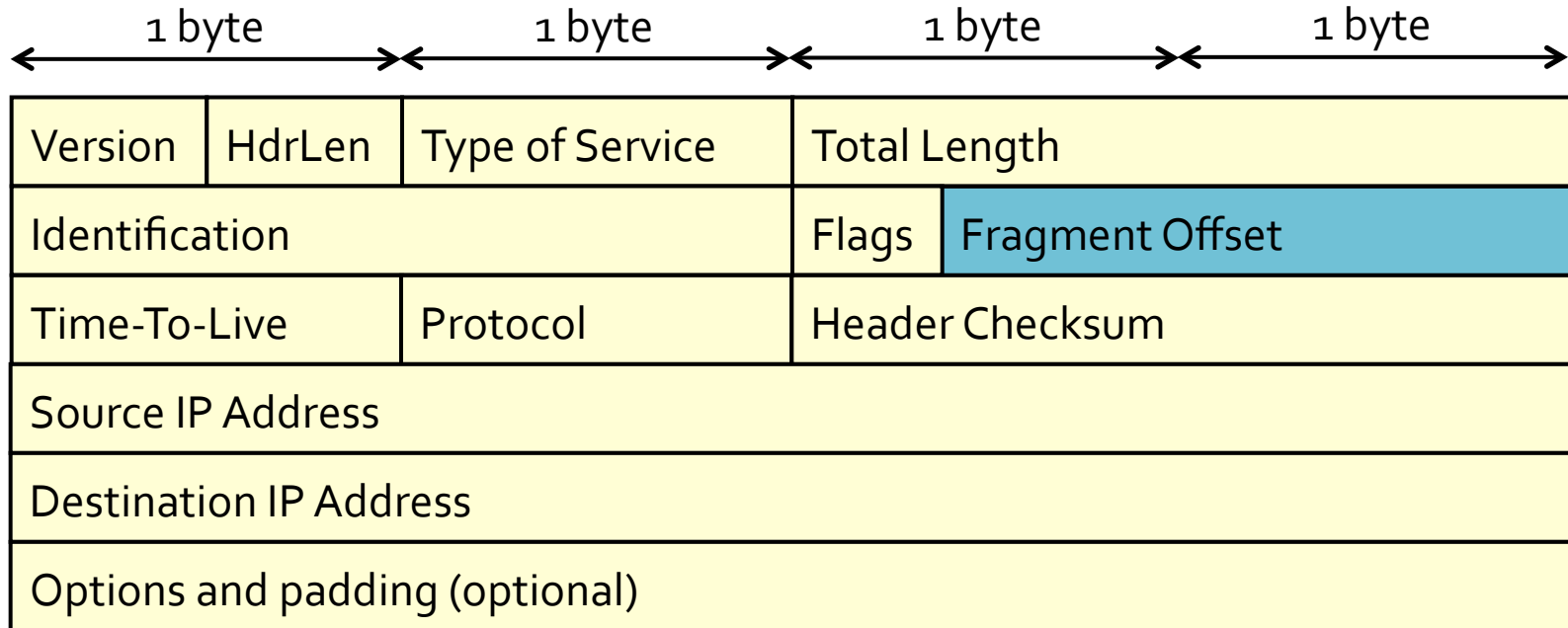
- Unique for each source/destination/protocol combination while the datagram is in-flight
- Can be used to determine which datagram a fragment came from

Flags



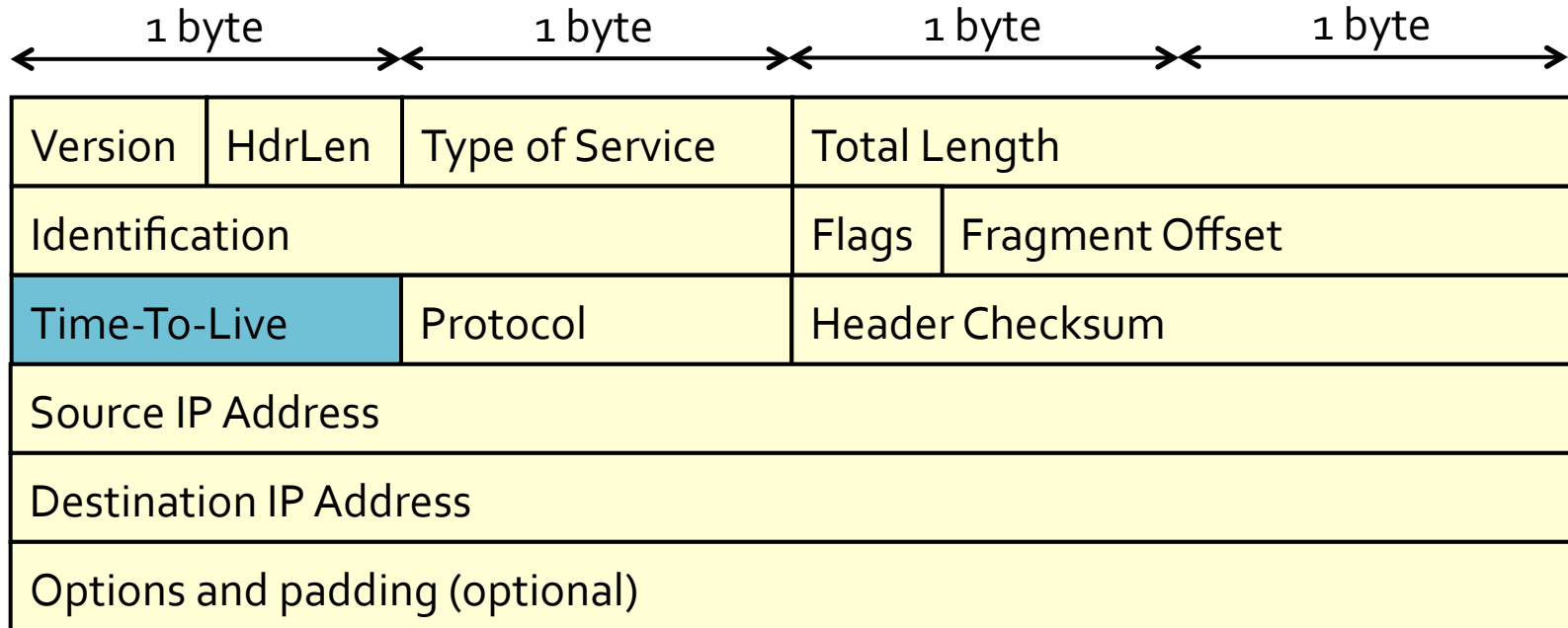
- 3 bits: Reserved/DF/MF
- DF: Don't fragment this packet
- MF: More fragments follow this packet

Fragment Offset



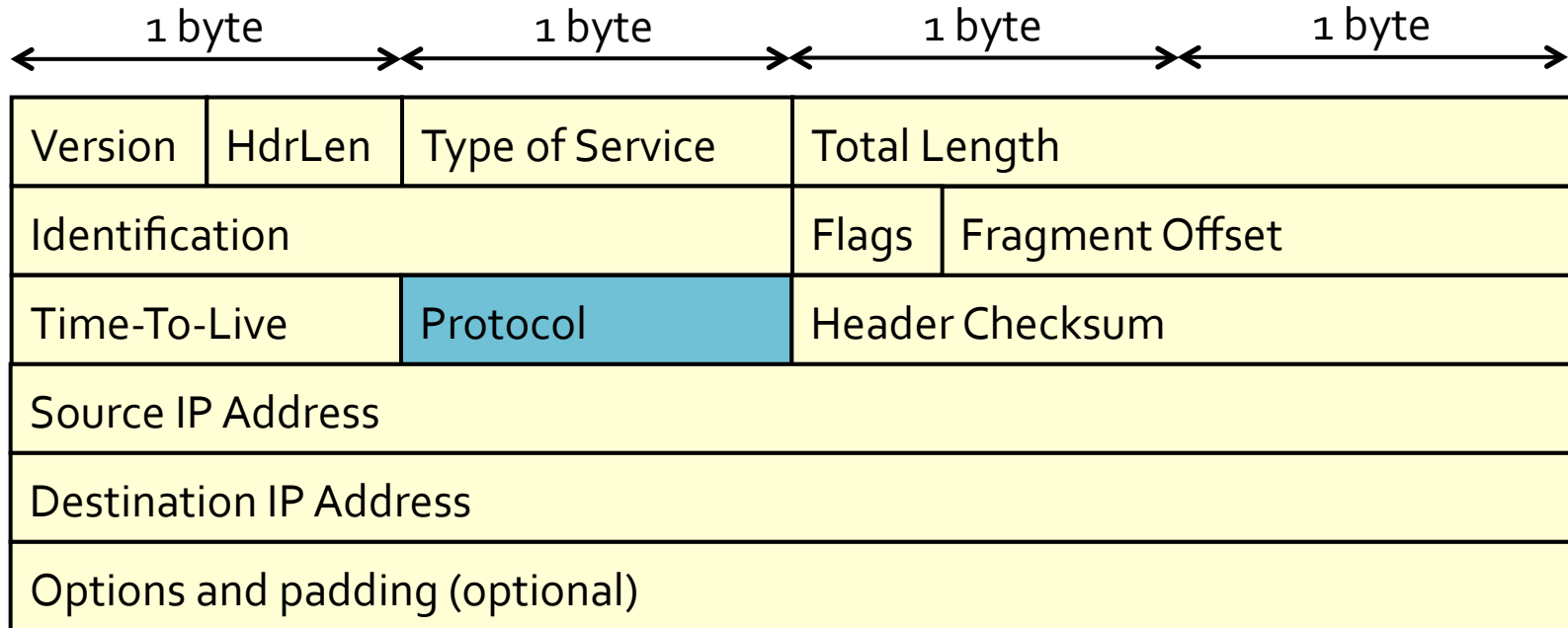
- Offset of fragment in original datagram
- 13 bits wide, but measures units of 8 bytes each
 - $2^{13} * 8 = 64k$ (so we can fragment the entire packet)

Time-To-Live



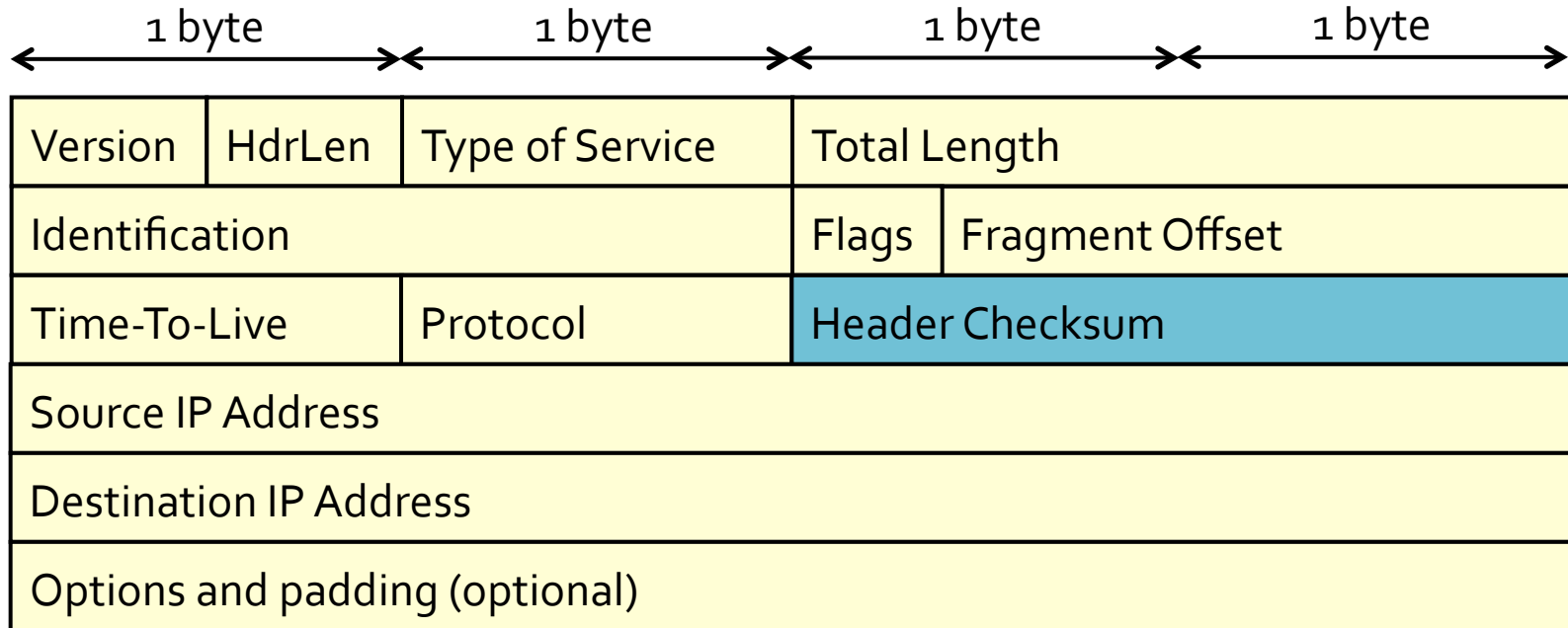
- “Hop count” – decrement each hop
- Discard datagrams with 0 TTL

Protocol



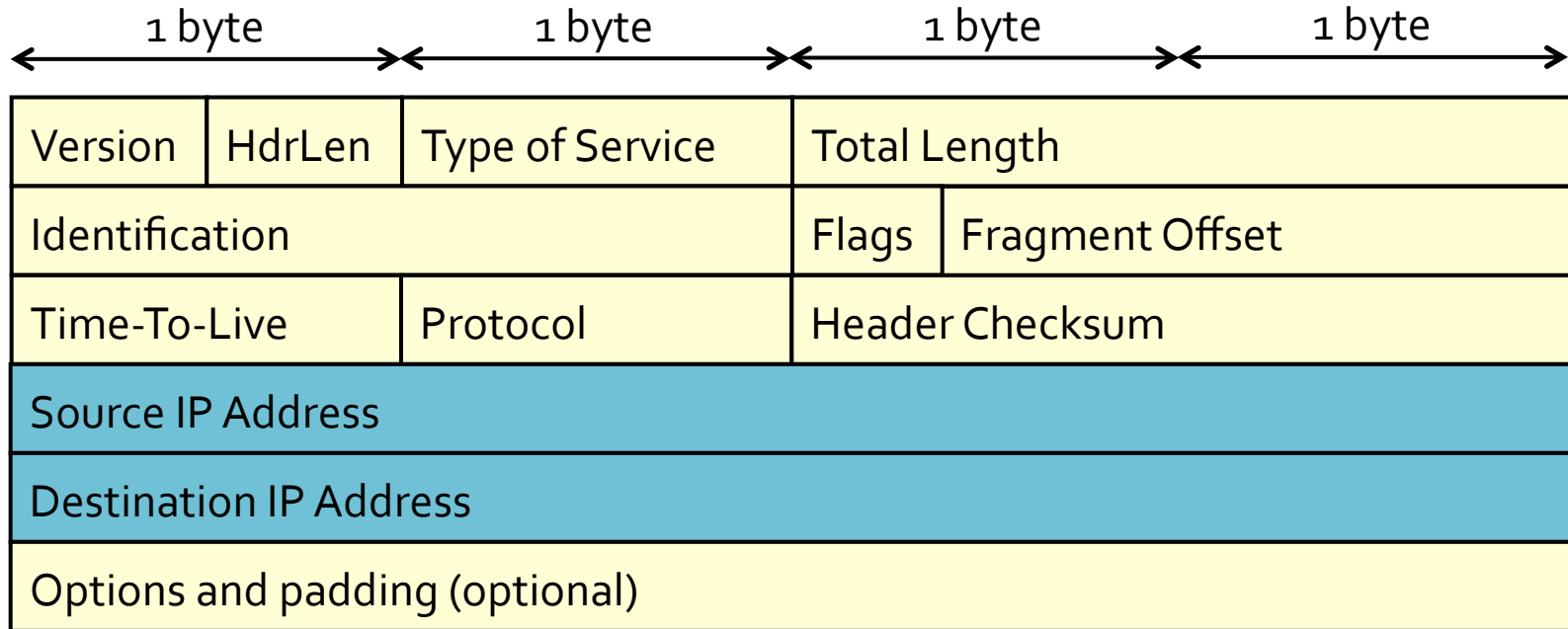
- What is encapsulated in this IP datagram?
 - 1 = ICMP, 6 = TCP, 17 = UDP, etc...

Header Checksum



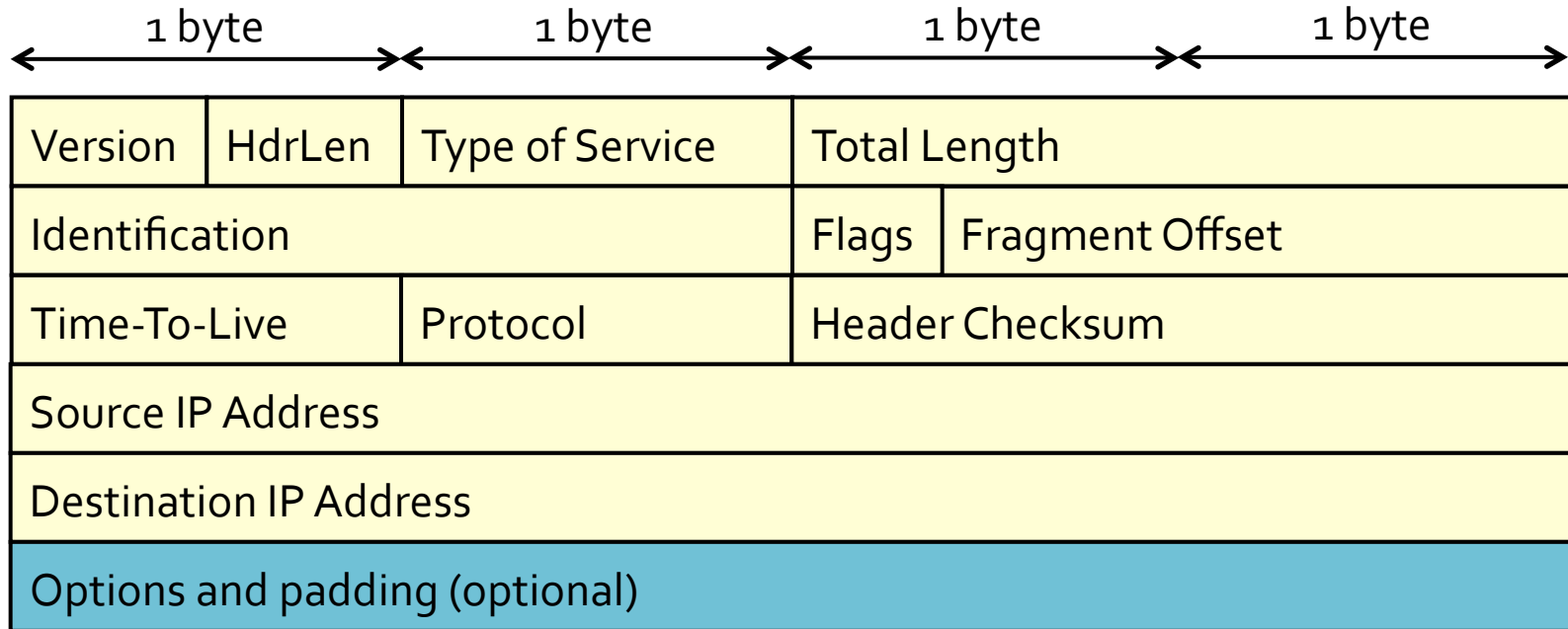
- Inverted 16-bit one's complement sum of the IP header and options

IP Addresses



- IP address of source and destination

Options

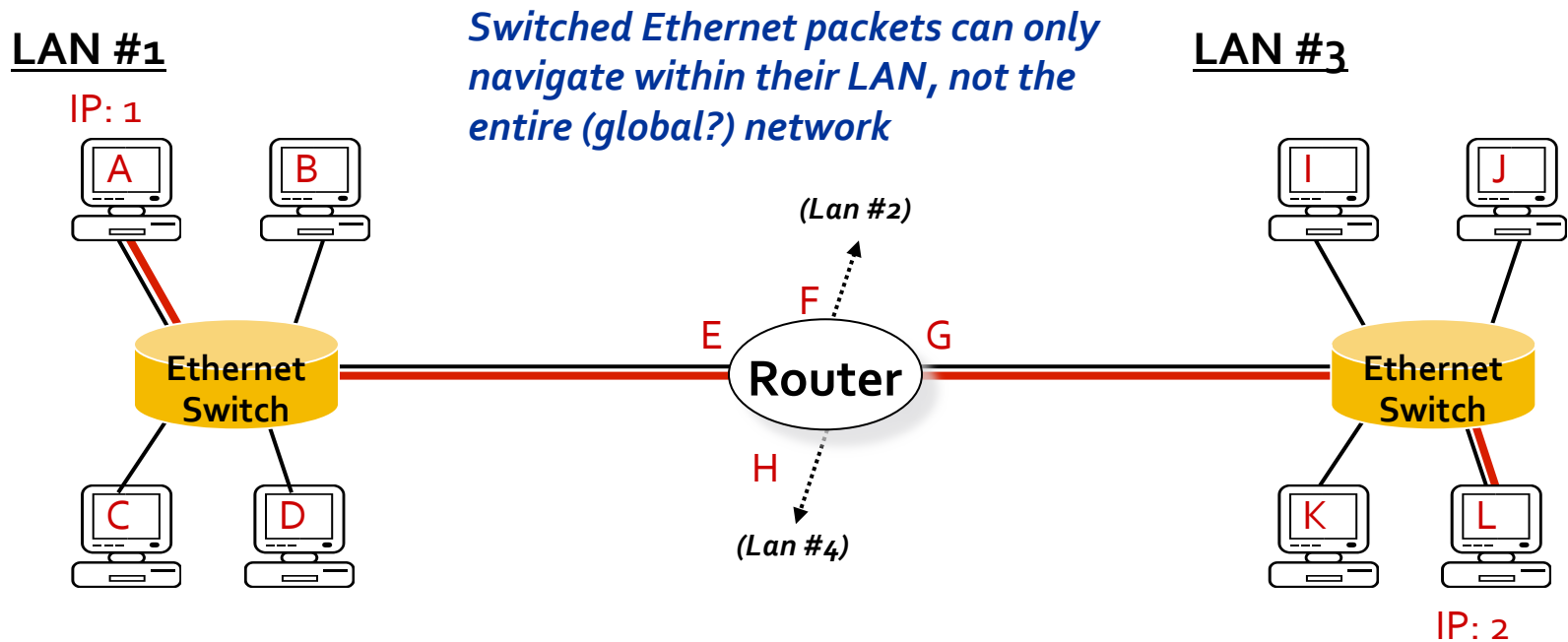


- Additional route information, timestamps, experimental information, etc.

IP encapsulated in Ethernet

Destination MAC Address				
Destination MAC Address		Source MAC Address		
Source MAC Address				
Type (0x0800)		Version	HdrLen	Type of Service
Total Length		Identification		
Flags	Fragment Offset		Time-To-Live	Protocol
Header Checksum		Source IP Address		
Source IP Address		Destination IP Address		
Destination IP Address		Options and Padding		
Options and Padding		Payload		
Payload				
Ethernet CRC				

Routing Between LANs



(1) A (1) transmits to L (2) using IP.
Ethernet frame destination is router

(3) Router uses IP protocol to forward data.
Eth: update src/dst/crc IP: update TTL/checksum

Frame:

EDA (E)	ESA (A)	0x0800	IPDA (2)	IPSA (1)
---------	---------	--------	----------	----------

Frame:

EDA (L)	ESA (G)	0x0800	IPDA (2)	IPSA (1)
---------	---------	--------	----------	----------

(2) Switch forwards frame to router

(4) Switch forwards frame to destination

IP Address Format

IP Addresses

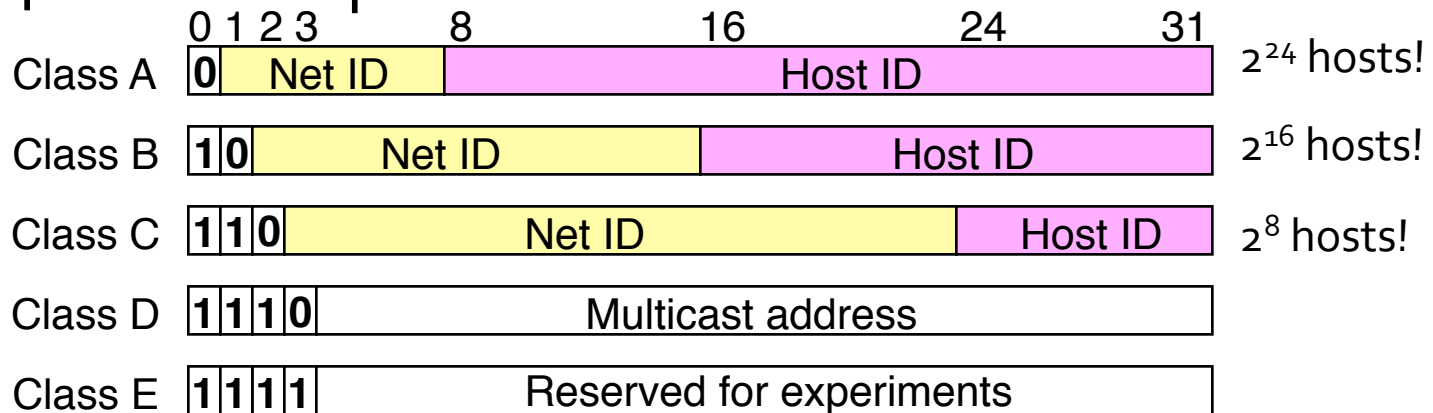
- IP version 4 addresses are 32-bits
 - Version 6 address are 128 bits
- Every network interface has at least one IP address
 - A computer might have 2 or more IP addresses
 - A router has many IP addresses
 - These addresses can be assigned statically or dynamically
- IP addresses are always in big-endian byte order (network byte order)
 - True in general for any integer transferred in a packet header from one machine to another
 - E.g., the port number used to identify a TCP connection

IP Address Format

- IPv4 addresses are usually displayed in dotted decimal notation
 - Each byte represented by decimal value
 - Bytes are separated by a period
 - IP address $0x8002C2F2 = 128.2.194.242$
- IP addresses are hierarchical
 - Address is composed of a network ID and a host ID
 - `www.pacific.edu: 138.9.110.12`

IP Address Structure

- IPv4 Address space divided into classes:



- Special IP addresses
 - Loop-back address: 127.0.0.1
 - Unrouted (private) IP addresses:
 - 10.0.0.0 – 10.255.255.255
 - 172.16.0.0 – 172.31.255.255
 - 192.168.0.0 – 192.168.255.255

A Joke



<http://xkcd.com/742/>

Routing

- Routers in the Internet should only need to know about network IDs
- Routers inside an organization can route based on host IDs
- Problems?

Problems

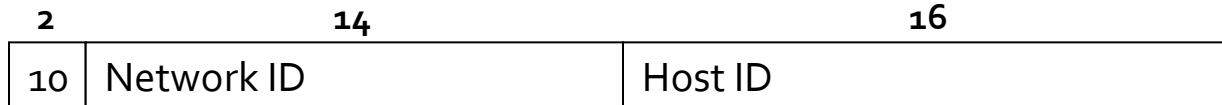
- Address classes were too “rigid”
 - Class C is too small and Class B is too big in many situations
 - Inefficient use of address space
 - Leads to a shortage of addresses
- Small organizations wanted Class B networks
 - In case they grew to more than 255 hosts
 - But there are only about 16,000 Class B network IDs
- Larger organizations wanted many Class C networks
 - Separate network ID for each router link
- Every router in the Internet had to know about every network ID in every organization
 - Leads to large address tables in every router

Subnetting

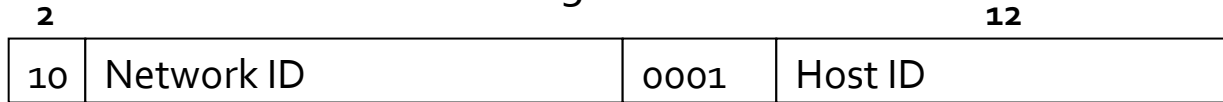
- Divide the network within an organization
 - Basically consider one Class B network to be a collection of many smaller networks
 - Size of smaller networks can be selected by the organization (don't have to be Class C sized networks)
- Internet routers don't need to know about subnetting within an organization
 - Just route their traffic to the organization

Subnetting

Company's
Class B Network

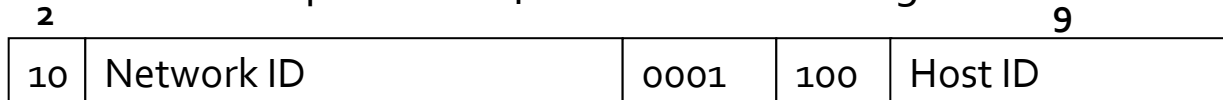


Building 1 Network



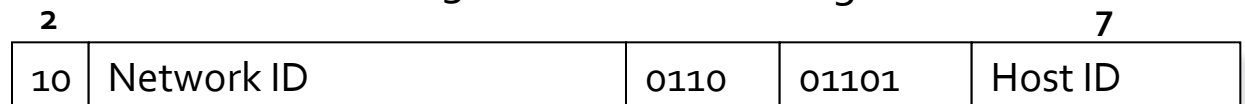
Subnet ID (18)

Department 4 network in Building 1



Subnet ID (21)

Floor 13 network in Building 6



Subnet ID (23)

Subnetting

- Can recursively subnet addresses down to as fine a granularity as you want
 - Almost...
 - Minimum-sized subnet has 4 addresses
 - Address 00 names the subnet
 - Address 01 and 10 names hosts
 - Address 11 is the broadcast IP address
- Subnet sizes don't have to be the same
 - One building divided by department, one by floor
 - Department/floor subnets not the same size

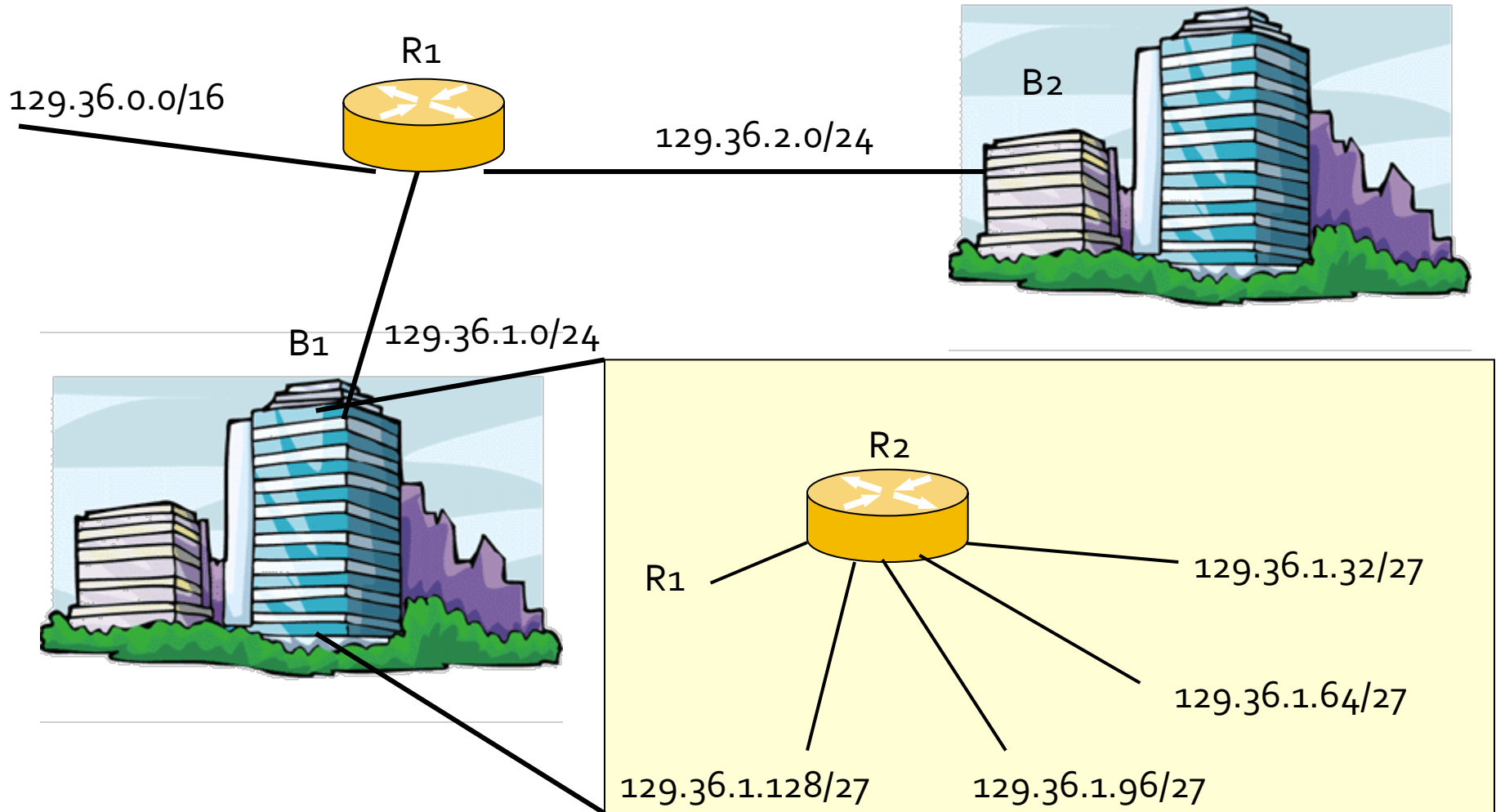
Subnet Notation

- A.B.C.D/X
 - IP address of the subnet (with 0's in all host ID bits)
 - X = number of bits in the subnet network address
- Examples:
 - 17.0.0.0/8 – Apple's entire class A address space
 - 17.2.3.0/24 – A class C sized subnet in Apple's network
- Alternatively represented by subnet IP and a bit mask (netmask)
 - 17.0.0.0/255.0.0.0
 - 17.2.3.0/255.255.255.0

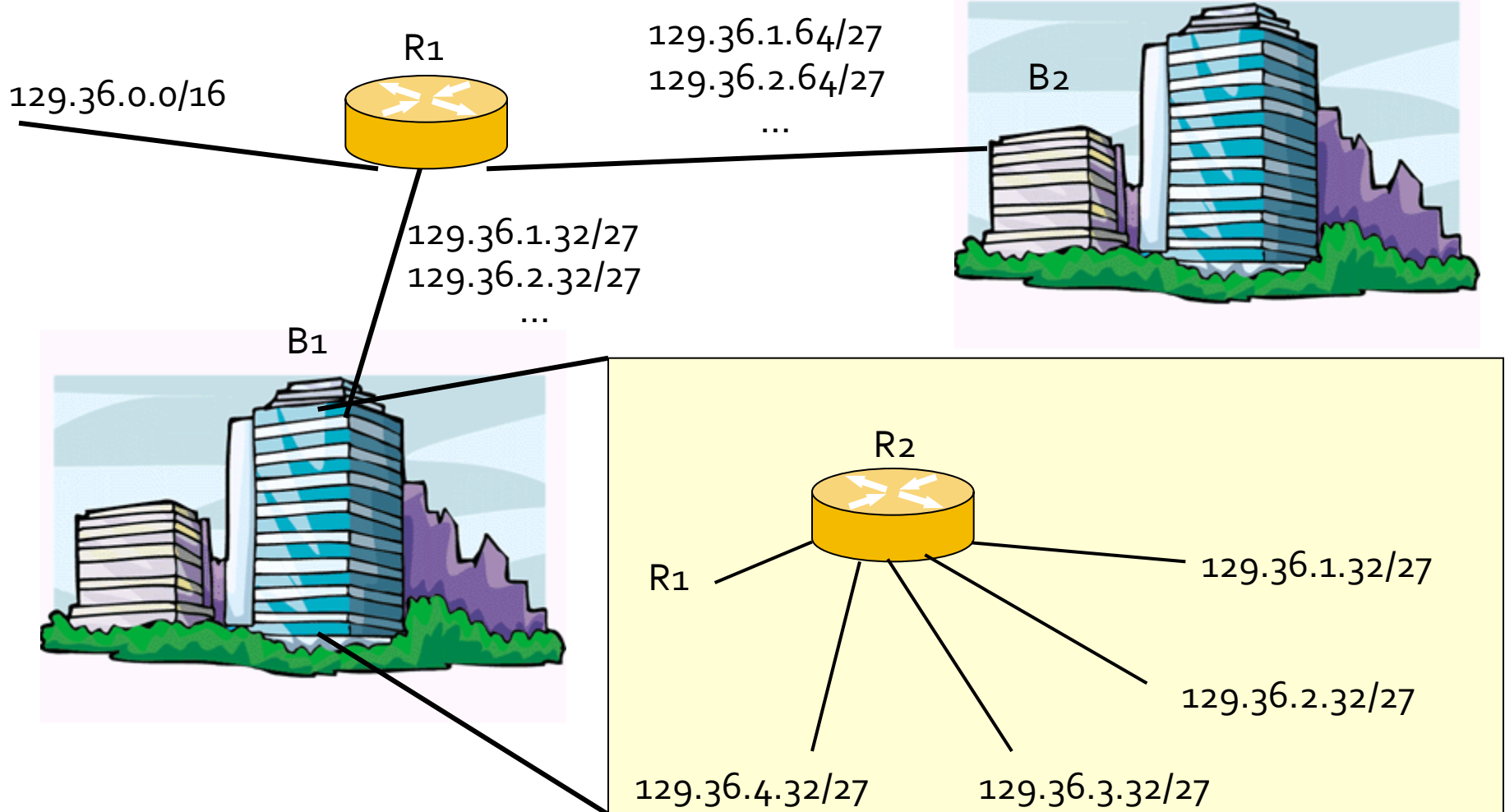
Subnet Meaning

- Subnets don't have to have physical meaning
 - Although easier to keep track of if they do...
- Good subnet assignment simplifies routing for internal routers
 - All traffic for "building 1" goes through this port
 - All traffic for "department 3" goes through that port
 - ...

Route Aggregation



Little/No Route Aggregation



Classless InterDomain Routing

- CIDR introduced in 1993
 - Meant to provide more flexible routing
 - Eliminate dependences on “class” networks in routing
- “Supernetting”
 - Combine multiple contiguous networks into one larger network
 - Effectively reduces the number of entries needed in each routing table
 - Inverse of subnetting which takes one larger network and breaks it into multiple contiguous smaller networks

CIDR Idea

- Break up IP address space into *prefixes*
 - Same idea as subnets (138.9/16)
- Each prefix has its own routing entry
 - All traffic to Pacific (138.9/16) within the Internet should be routed the same way, regardless of how Pacific subnets its address space

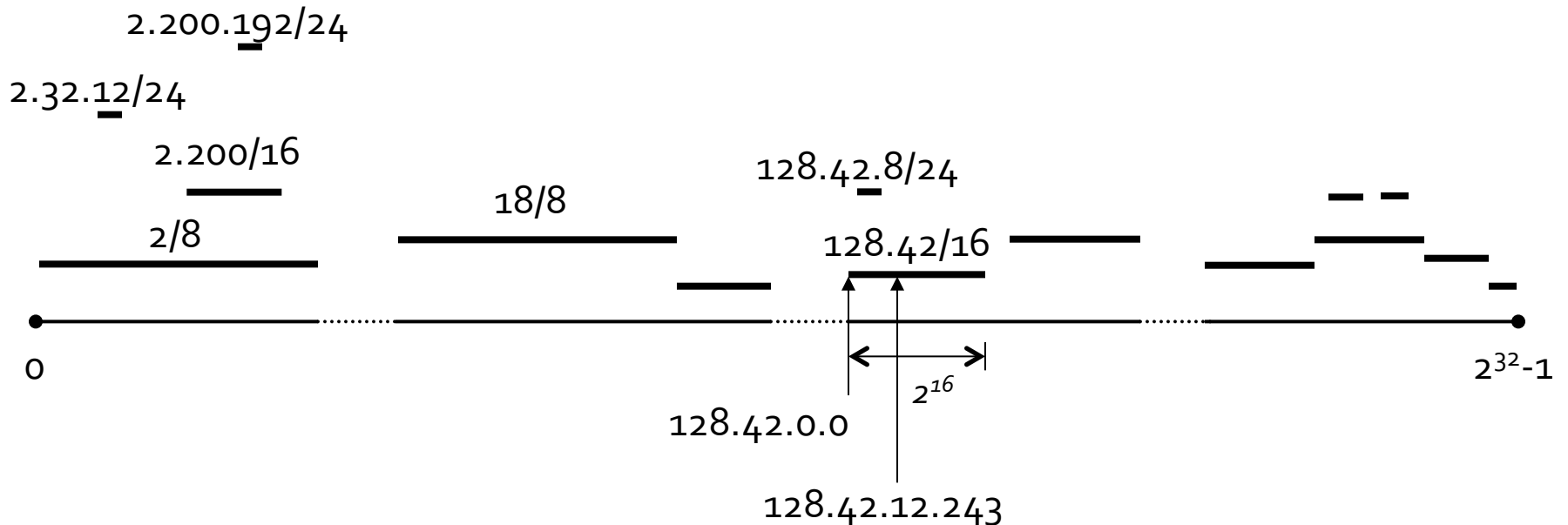
Route Aggregation

- Fundamental idea behind supernetting
 - Same type of aggregation as with subnets, but on a larger scale
- E.g., one ISP handles traffic for two corporate networks (129.32/16 and 129.33/16)
 - Aggregate route to 129.32/15 for both networks
 - Only break them apart when necessary for the last (few) hop(s)
- E.g., one ISP handles all 64/8 networks
 - All other routers know to route 64/8 traffic to that ISP
 - External routers don't care how the ISP breaks up the network addresses internally!
 - Internal ISP routers must route at a finer grain

What if there are holes?

- Pacific builds a 4th campus
 - 128.42/16 needs to be routed to Pacific
 - 128.42.8/24 needs to be routed to our satellite campus in Hawaii...
- Do we need to break routes up?
 - 128.42.0/20 (.0-.7)
 - 128.42.8/24 (.8)
 - 128.42.9/24 (.9), 128.42.10/24 (.10), 128.42.11/24 (.11)
 - 128.42.12/22 (.12-.15)
 - 128.42.16/20 (.16-.31)
 - 128.42.32/19 (.32-.63)
 - 128.42.64/18 (.64-.127)
 - 128.42.128/17 (.128-.255)

IP Prefixes



- IP address space can be viewed as a number line
 - Each segment represents an aggregated route
 - Segments can overlap
- Look for smallest segment that matches the destination address

Longest Prefix Match

- Allow more specific entries to supersede more general ones
 - 128.42.8/24
 - Route this traffic to Hawaii
 - 128.42/16
 - Route this traffic to Stockton
 - Except for addresses that match a route with a longer prefix (i.e., 128.42.8/24)
- Allows significantly more route aggregation
- Simplifies things if companies move (physically or to another ISP) their block of IP addresses

IP Address Classes

- CIDR makes address classes less important
- With CIDR, routing is based on arbitrary subdivisions of the address space
 - Aggregate routes into largest possible group
 - Use longer prefixes to deal with exceptions
- Routing
 - Routers use longest prefix matching to determine routes
 - No longer deal with exact matches on class network IDs

Implementing Software LPM

- Simple software solution
 - Keep list of prefixes and masks
 - Search for all prefix matches
 - Choose result with longest prefix
- Trick
 - Sort list based on decreasing prefix length (stop search after first match)
- This is extremely inefficient for large forwarding tables!

ARIN WHOIS Database Search

Relevant Links: [ARIN Home Page](#) [ARIN Site Map](#) Training: [Querying ARIN's WHOIS](#)

Search ARIN WHOIS for: 138.9.1.21

Submit

OrgName: University of the Pacific
OrgID: **UNIVER-95**
Address: 3601 Pacific Ave.
City: Stockton
StateProv: CA
PostalCode: 95211
Country: US

NetRange: **138.9.0.0 - 138.9.255.255**
CIDR: 138.9.0.0/16
NetName: **UOP**
NetHandle: **NET-138-9-0-0-1**
Parent: **NET-138-0-0-0-0**
NetType: Direct Assignment
NameServer: NS1.PACIFIC.EDU
NameServer: NS2.PACIFIC.EDU
Comment:
RegDate: 1990-01-17
Updated: 2007-09-07

RAbuseHandle: **DAVEA-ARIN**
RAbuseName: Lundy, Dave A.
RAbusePhone: +1-209-946-3951
RAbuseEmail: dlundy@pacific.edu

RTechHandle: **EES7-ARIN**
RTechName: Escalante, Edgar
RTechPhone: +1-209-946-3190
RTechEmail: eescalante@pacific.edu

OrgTechHandle: **DAVEA-ARIN**
OrgTechName: Lundy, Dave A.
OrgTechPhone: +1-209-946-3951
OrgTechEmail: dlundy@pacific.edu