CS 330: Network Applications & Protocols

Network Layer

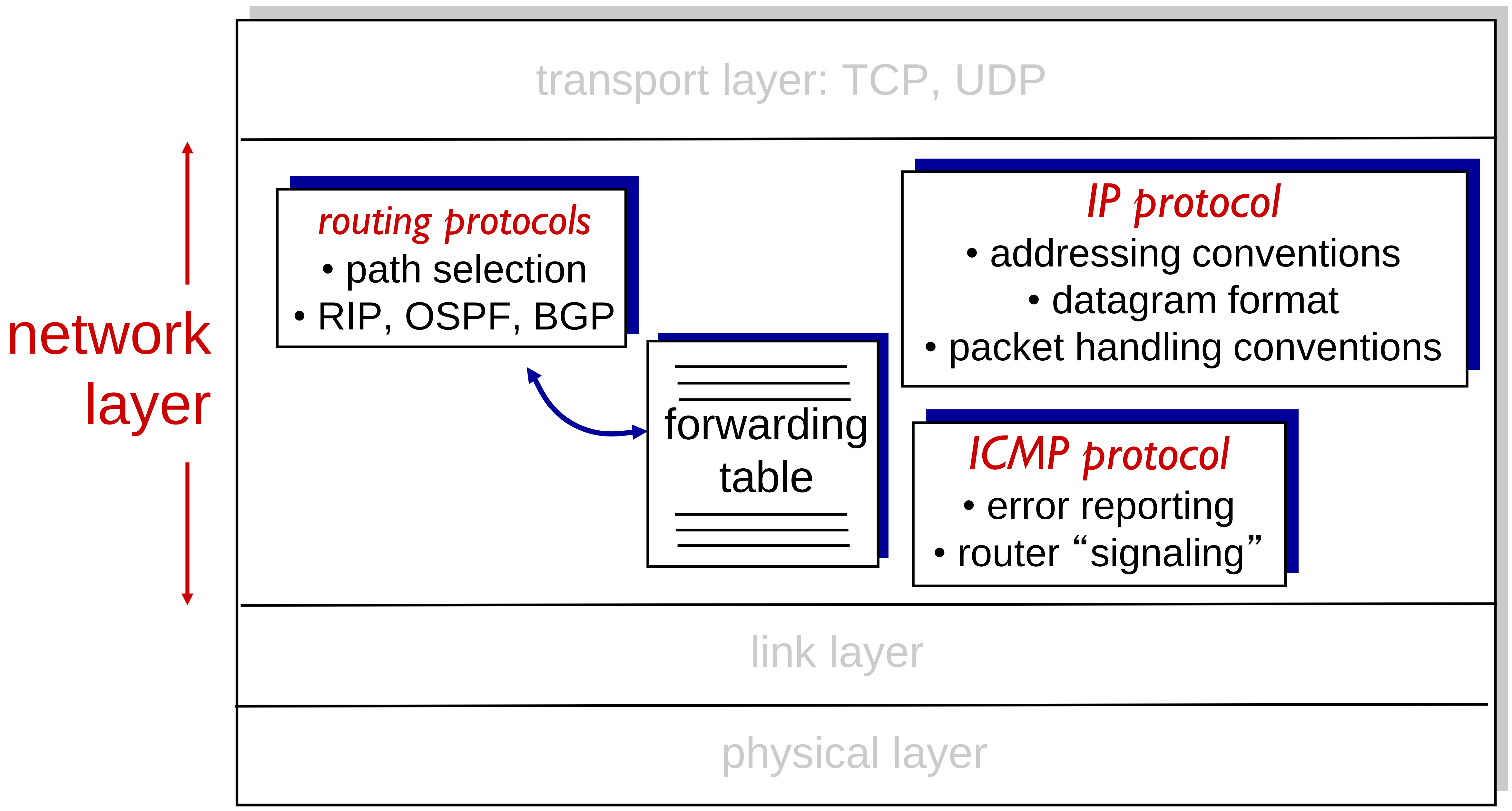
Galin Zhelezov
Department of Physical Sciences
York College of Pennsylvania

Overview of Network Layer

- **Virtual Circuit and Datagram Networks**
- **Router Architectures**
- **IP: Internet Protocol**
  - Datagram Format
  - IPv4 Addressing
  - ICMP
  - IPv6
- **Routing algorithms**
- **Routing in the Internet**
- **Broadcast and multicast routing**

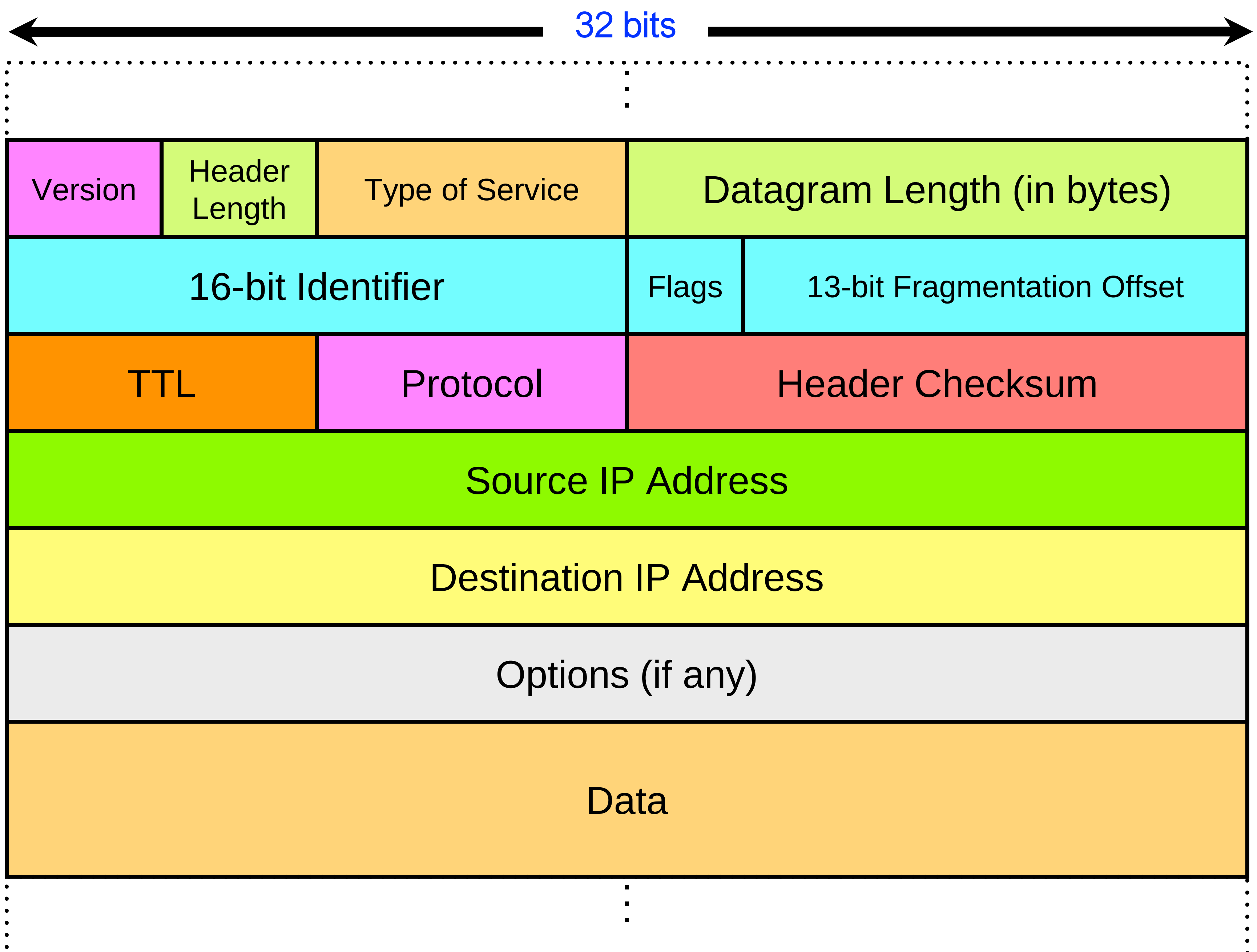
The Internet network layer

host, router network layer functions:



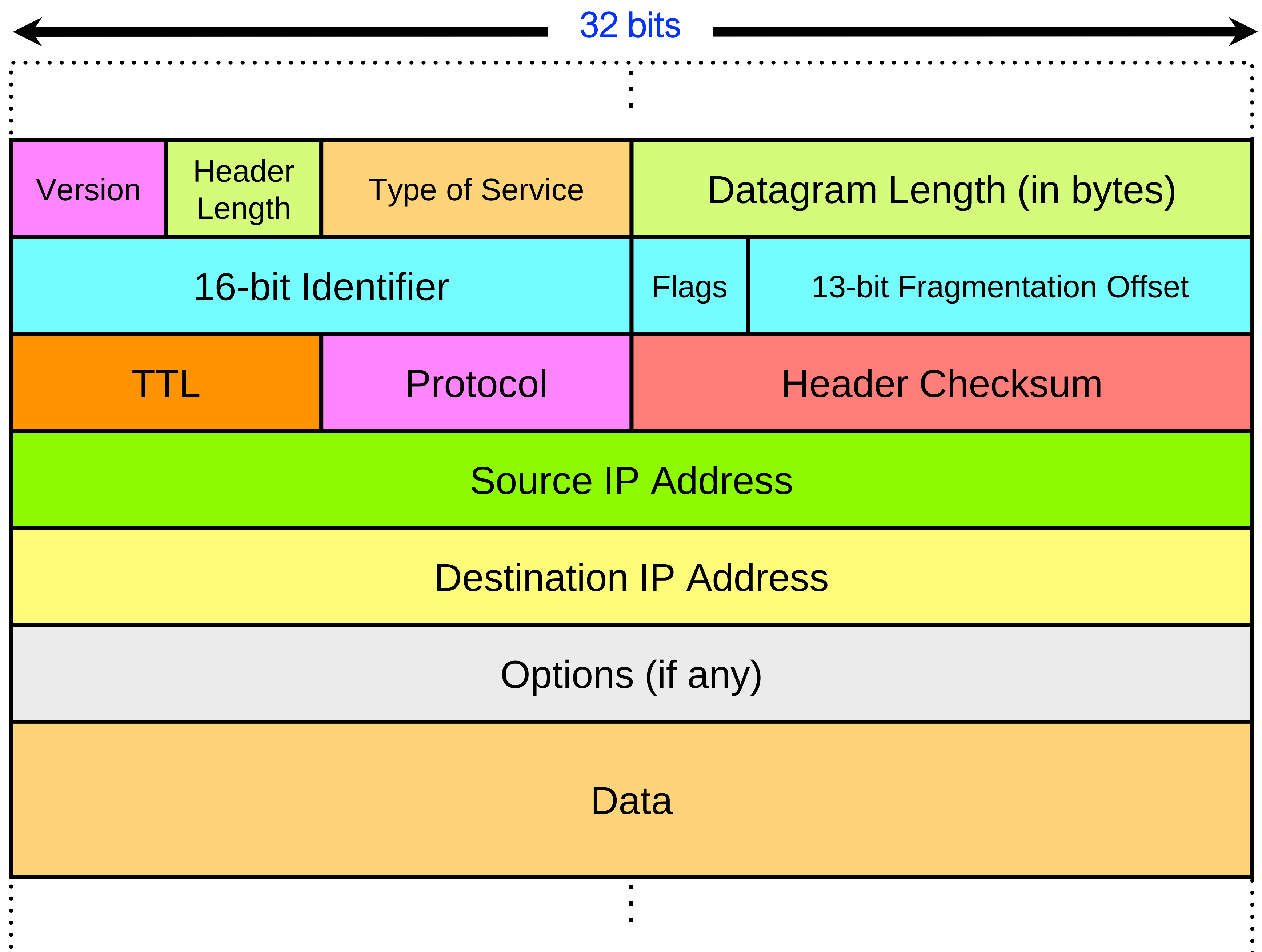
IP Datagram Format

- **Version - 4-bit value that specifies the IP protocol version of the datagram (e.g. 0x4 for IPv4)**
- **Header Length - 4-bit value that indicates how many 32-bit words are in the header (can vary with options)**
  - Min value = 5
- **Type of Service - 8-bit value that indicates the type of service requested (e.g. priority service, low delay, etc.)**
  - TOS has been superseded by Differentiated Services field



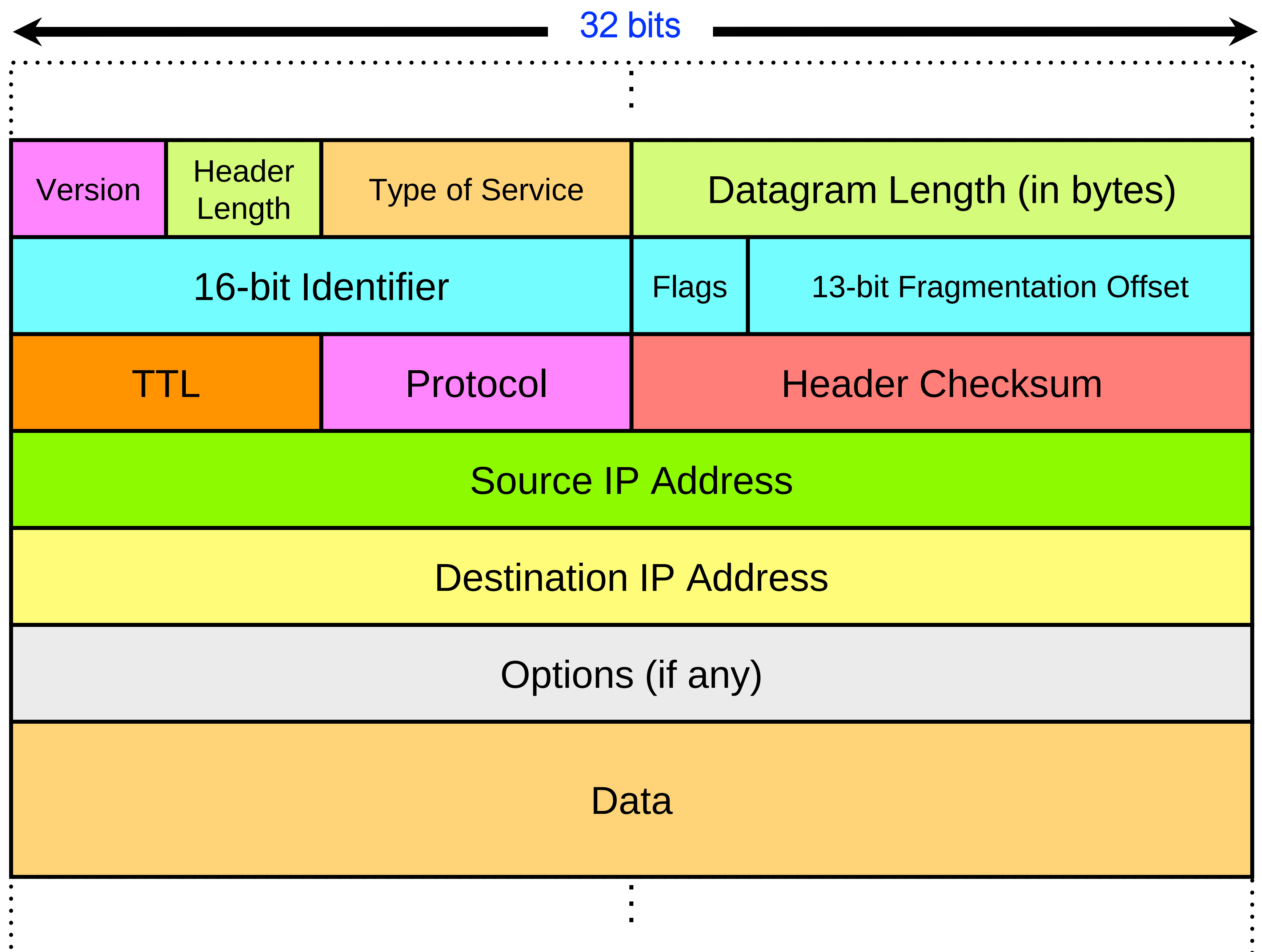
IP Datagram Format

- **Datagram Length - 16-bit value that specifies the total length of the IP datagram including header**
  - Rarely larger than 1500 bytes
- **TTL - 8-bit value that specifies the *time-to-live* for the datagram**
  - Decremented by 1 at each router, if value reaches 0 packet is dropped
  - Used to prevent packets from circulating network forever
- **Protocol - 8-bit value that specifies that transport-layer protocol contained within the datagram**
  - Protocol = 6 indicates TCP
  - Protocol = 17 indicates UDP



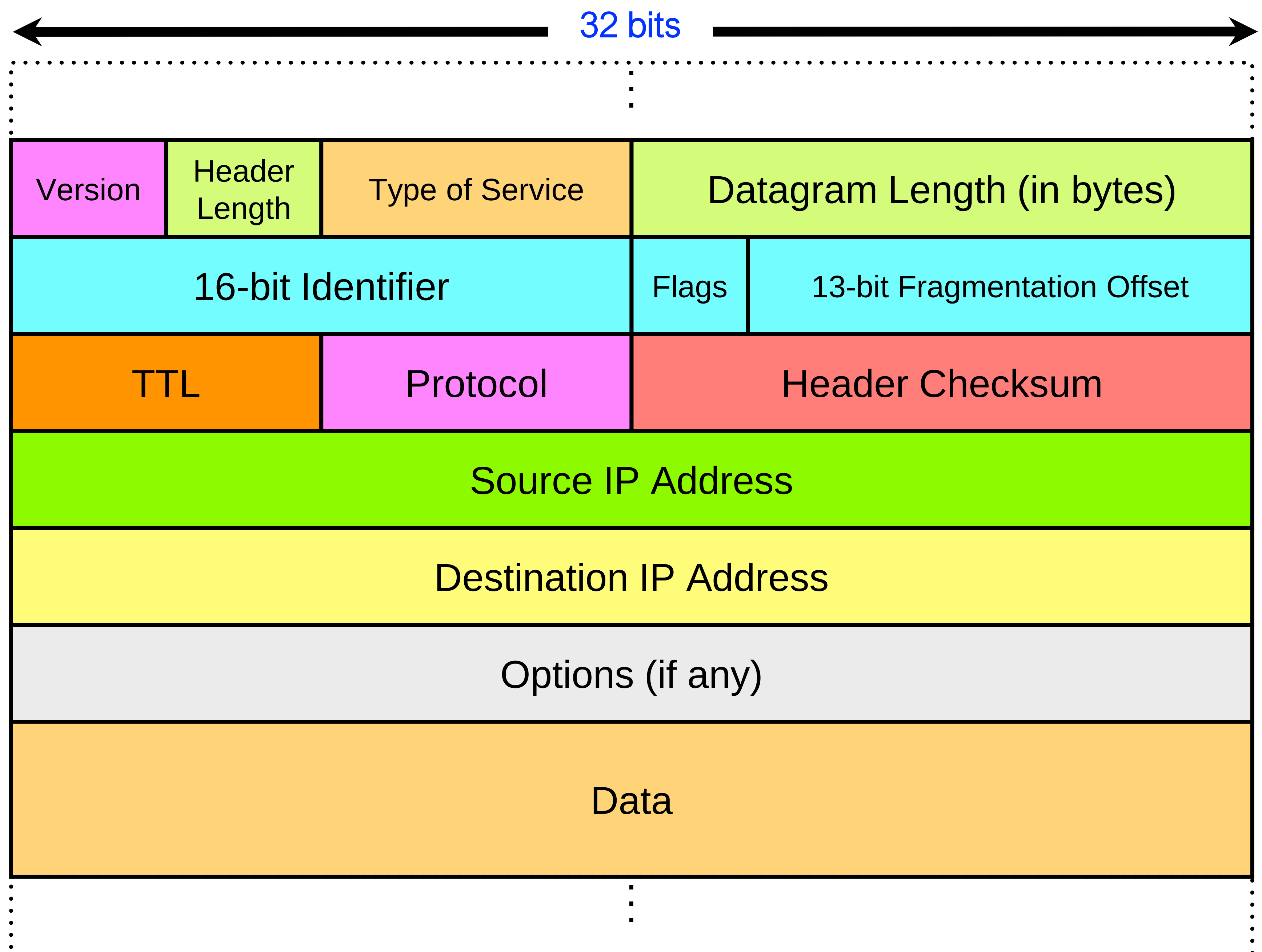
IP Datagram Format

- **Header Checksum - 16-bit 1's complement checksum computed over IP header**
  - Checked at each router along route to destination (checked again at destination)
    - Datagram is dropped if error is detected
  - Checksum is recomputed at each router before datagram is retransmitted since header changes
    - TTL is decremented
    - Options may change
- **Source IP Address - 32-bit address of the machine sending the datagram**
- **Destination IP Address - 32-bit address of the intended recipient**



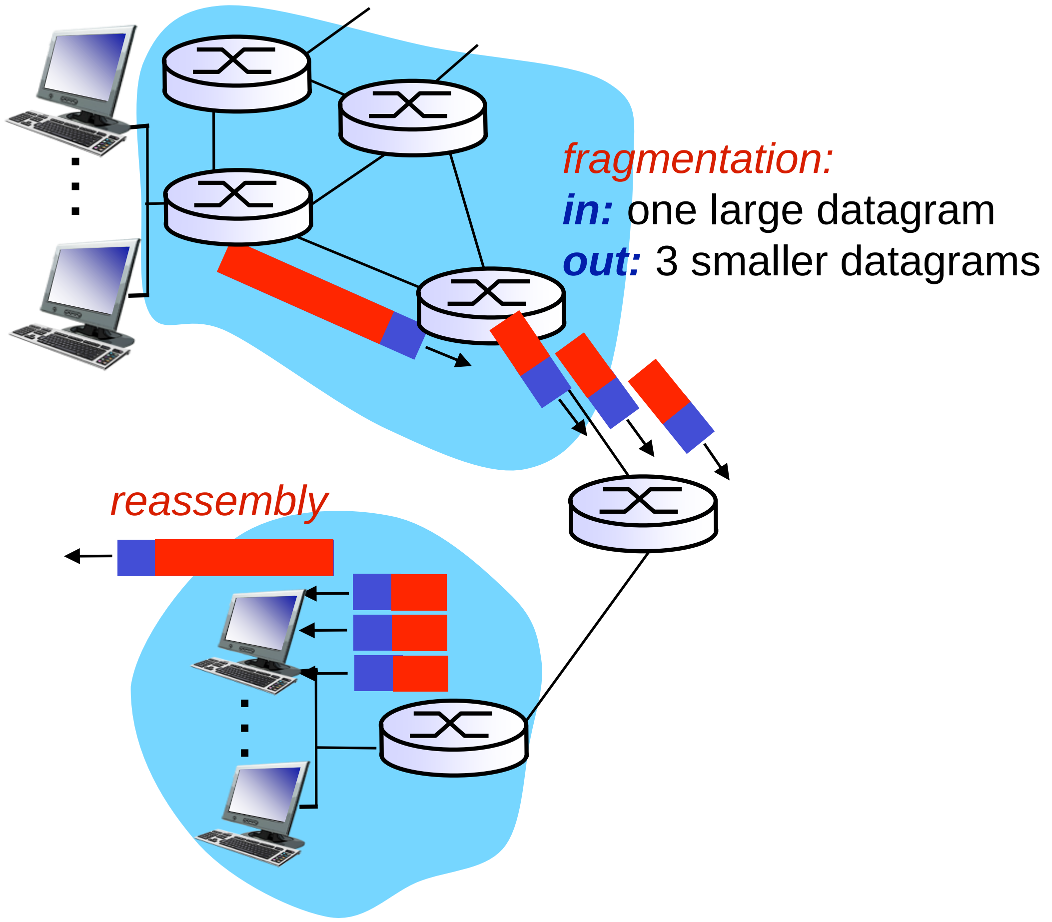
IP Datagram Format

- **Options - optional header fields**
  - IPv4 supports a variable number of options fields
- **Data - contains the data for the IP datagram**
  - Most often will contain Transport Layer segment (TCP/UDP) or some other protocol such as ICMP
- **Fragmentation fields - used to support IP fragmentation**
  - 16-bit Identifier is created by original sender
  - Flags
  - 13-bit Fragmentation Offset



IP Fragmentation & Reassembly

- **Network links have MTU (Maximum Transfer Unit)**
  - Largest possible Link-Level frame
  - Different link types provide different MTUs
- **Large IP datagrams may get divided ("fragmented") within network**
  - One large IP datagram becomes several smaller IP datagrams
  - Fragmented IP datagrams are reassembled at final destination
- **Fragmentation fields are used to identify and order related fragments**



IP Fragmentation & Reassembly

- **Example - sending an IP datagram of 4000 bytes that encounters a link that has an MTU of 1500 bytes**
  - Original datagram is fragmented in the network into multiple smaller datagram (fragments)
  - Fragment sizes (except the last) must be multiples of 8 bytes

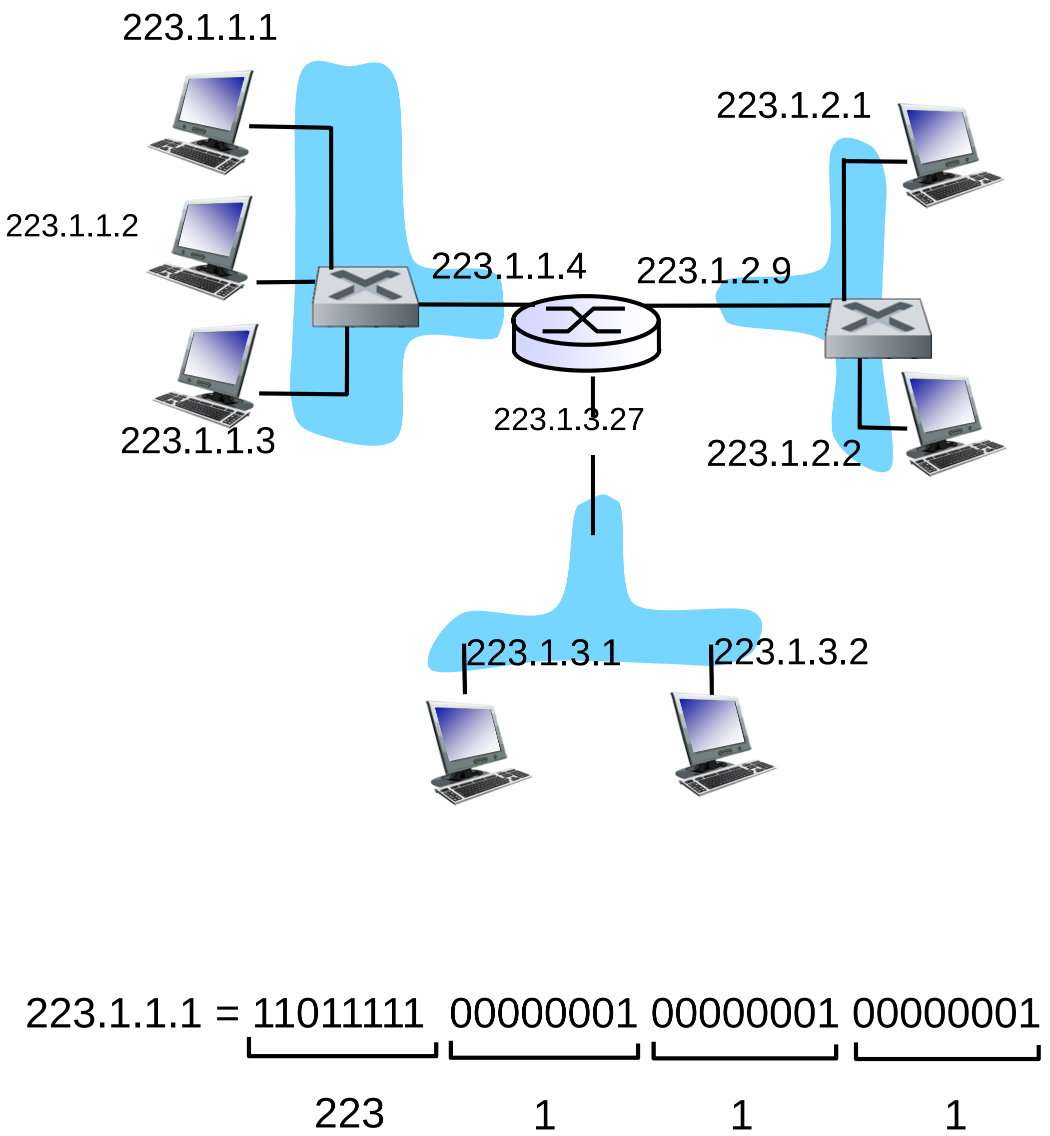
Fragment	Bytes	ID	Offset	Flag
Original / Unfragmented	4000 bytes (20 hdr + 3980 data)	562	0 Insert data at byte 0	0 Last fragment
1st Fragment	1480 bytes of data (plus 20 for header)	562	0 Insert data at byte 0	1 More fragments exist
2nd Fragment	1480 bytes of data (plus 20 for header)	562	185 Insert data at byte 185*8 = 1480	1 More fragments exist
3rd Fragment	1020 bytes of data (plus 20 for header)	562	370 Insert data at byte 370*8 = 2960	0 Last fragment

Overview of Network Layer

- **Virtual Circuit and Datagram Networks**
- **Router Architectures**
- **IP: Internet Protocol**
  - Datagram Format
  - IPv4 Addressing
  - ICMP
  - IPv6
- **Routing algorithms**
- **Routing in the Internet**
- **Broadcast and multicast routing**

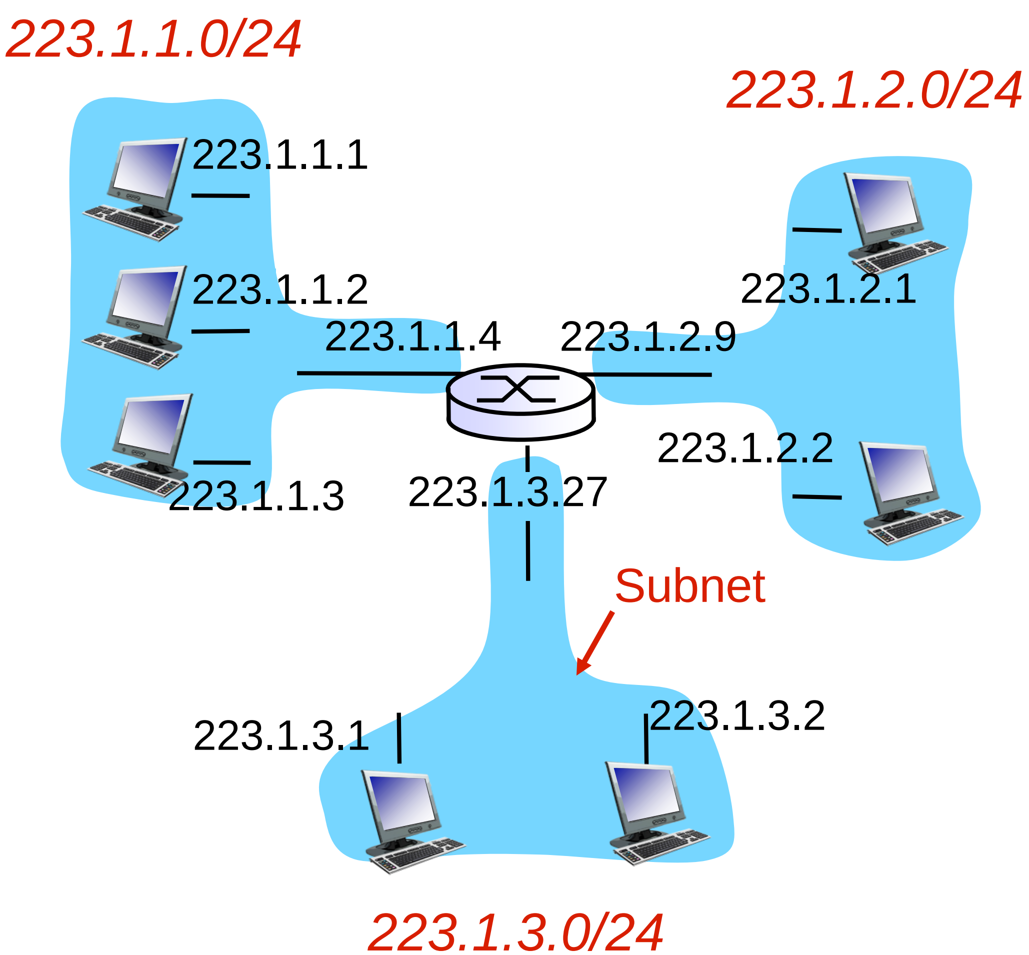
Introduction to IP Addressing

- **IP address is a 32-bit identifier for a host/router interface**
  - Format:
    ###.###.###.###
    four values where each value is between 0-255
- **An interface is the connection between host/router and the physical link**
  - Routers typically have multiple interfaces
  - Hosts typically have one or two interfaces (e.g. wired Ethernet, wireless 802.11)
- **An IP addresses is associated with *each* interface**



Subnets

- **Subnets are logical subdivisions of an IP network**
- **IP address:**
  - subnet part - high order bits
  - host part - low order bits
- **Hosts on the same subnet:**
  - Have common most-significant bits
    - Number of bits in common represented with subnet mask or CIDR notation
  - Can physically reach each other without intervening router

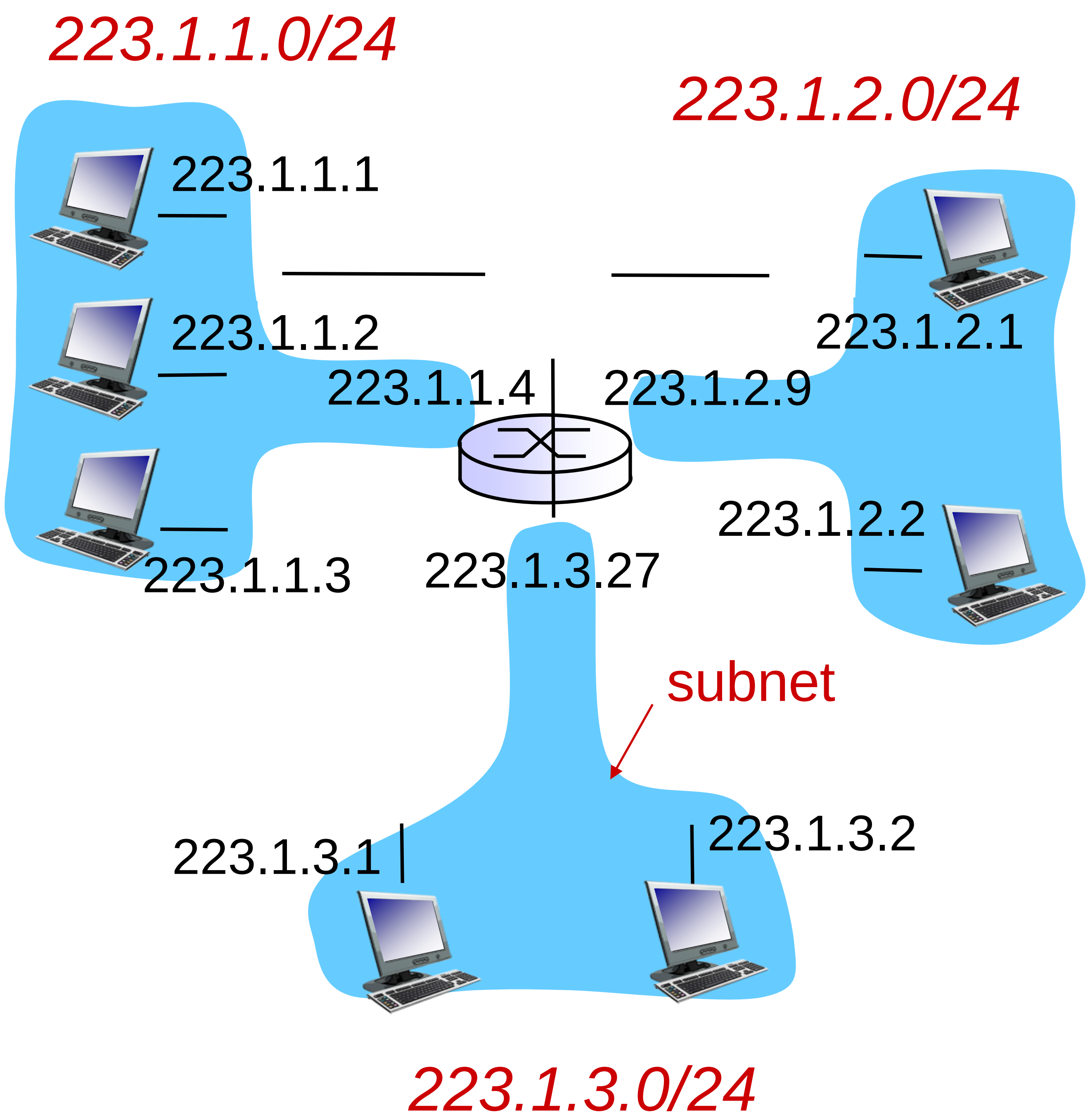


Network consisting of 3 subnets

Subnets

recipe

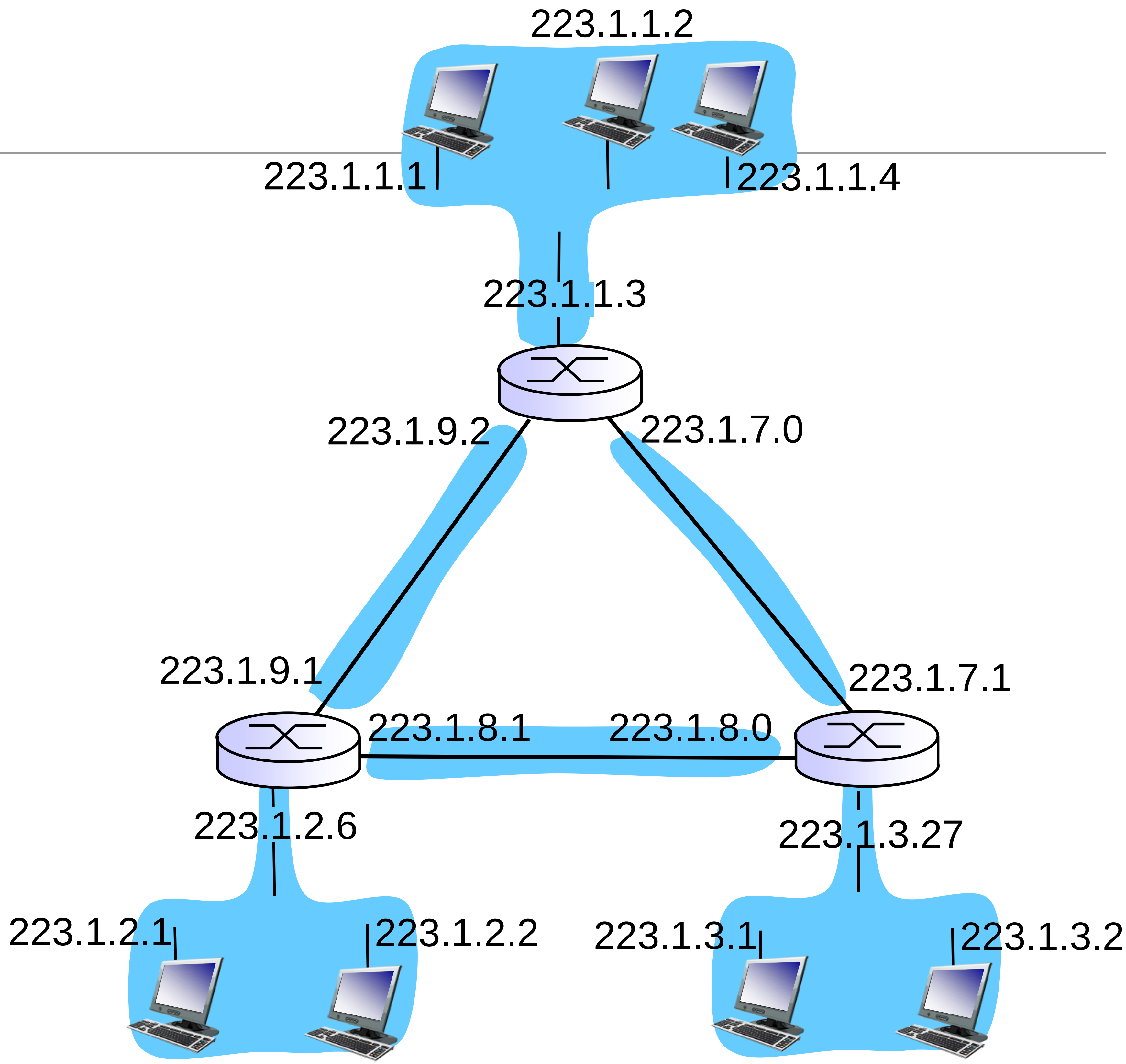
- to determine the subnets, detach each interface from its host or router, creating islands of isolated networks
- each isolated network is called a ***subnet***



subnet mask: /24

Subnets

how many?



IP Addressing: CIDR

- **CIDR: Classless InterDomain Routing**
  - Subnet portion of address of arbitrary length
  - Address format: a.b.c.d/x, where x is the number of bits in subnet portion of address
    - For machines to reside on the same subnet and be able to communicate with each other, all bits in subnet part must match

Subnet Part **Host Part**

11001000 00010111 00010000 00000000

200.23.16.0/23

Equivalent to
200.23.16.0 with subnet mask of 255.255.254.0

Matches block of addresses:
200.23.16.0 - 200.23.17.255

IP addresses: how to get one?

Q: How does a *host* get IP address?

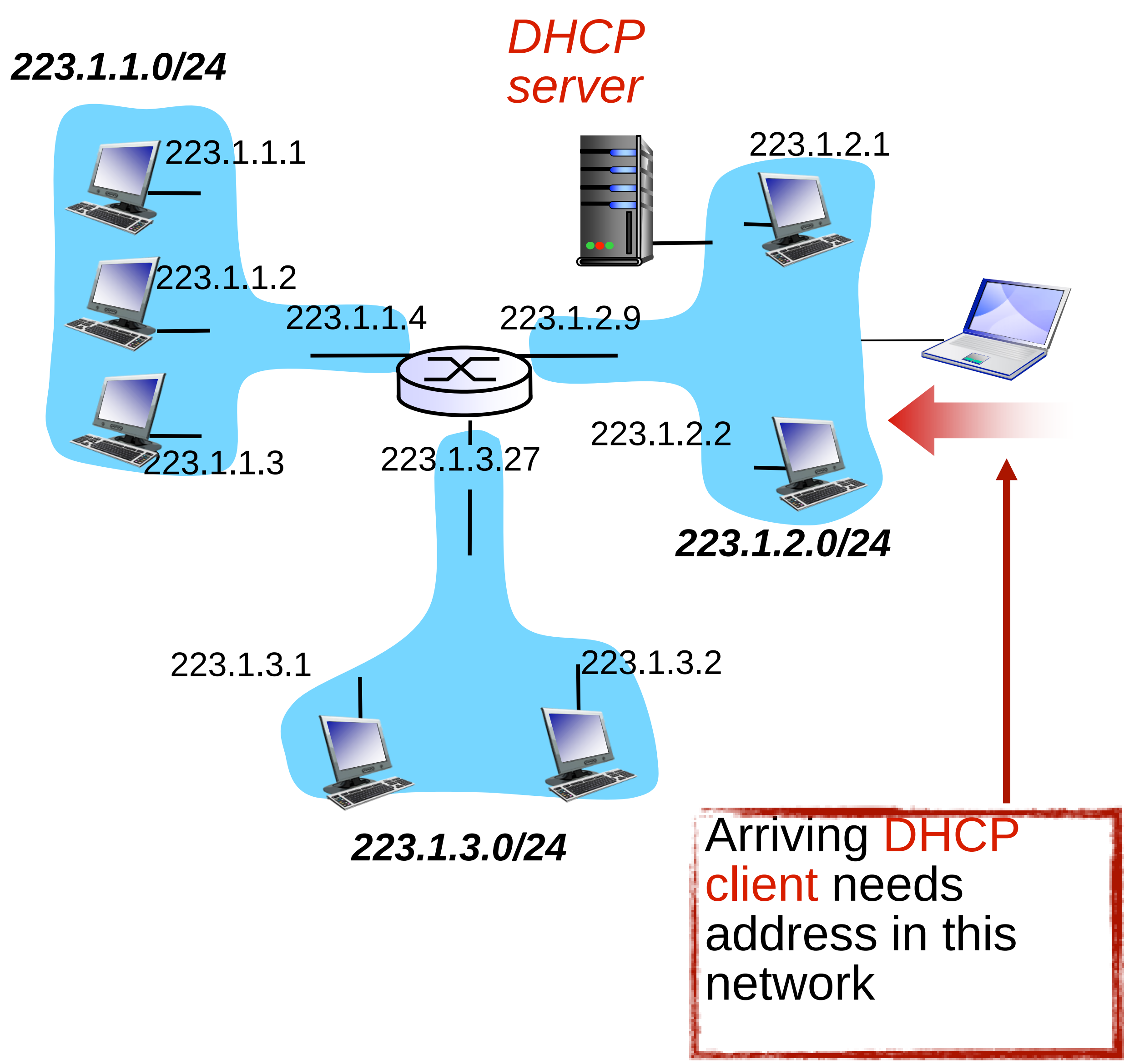
- **hard-coded by system admin in a file**
  - Windows: control-panel->network->configuration->tcp/ip->properties
  - UNIX: /etc/rc.config
- **DHCP: Dynamic Host Configuration Protocol: dynamically get address from as server**
  - "plug-and-play"

DHCP: Dynamic Host Configuration Protocol

- **Allows a host to dynamically obtain an IP address from network server when it joins network**
  - IP addresses are *leased* for some duration of time
  - Host can renew its lease on IP address before it expires
  - Allows reuse of addresses
    - Hosts only hold use address while connected to network
  - Great for mobile devices that hop from network to network

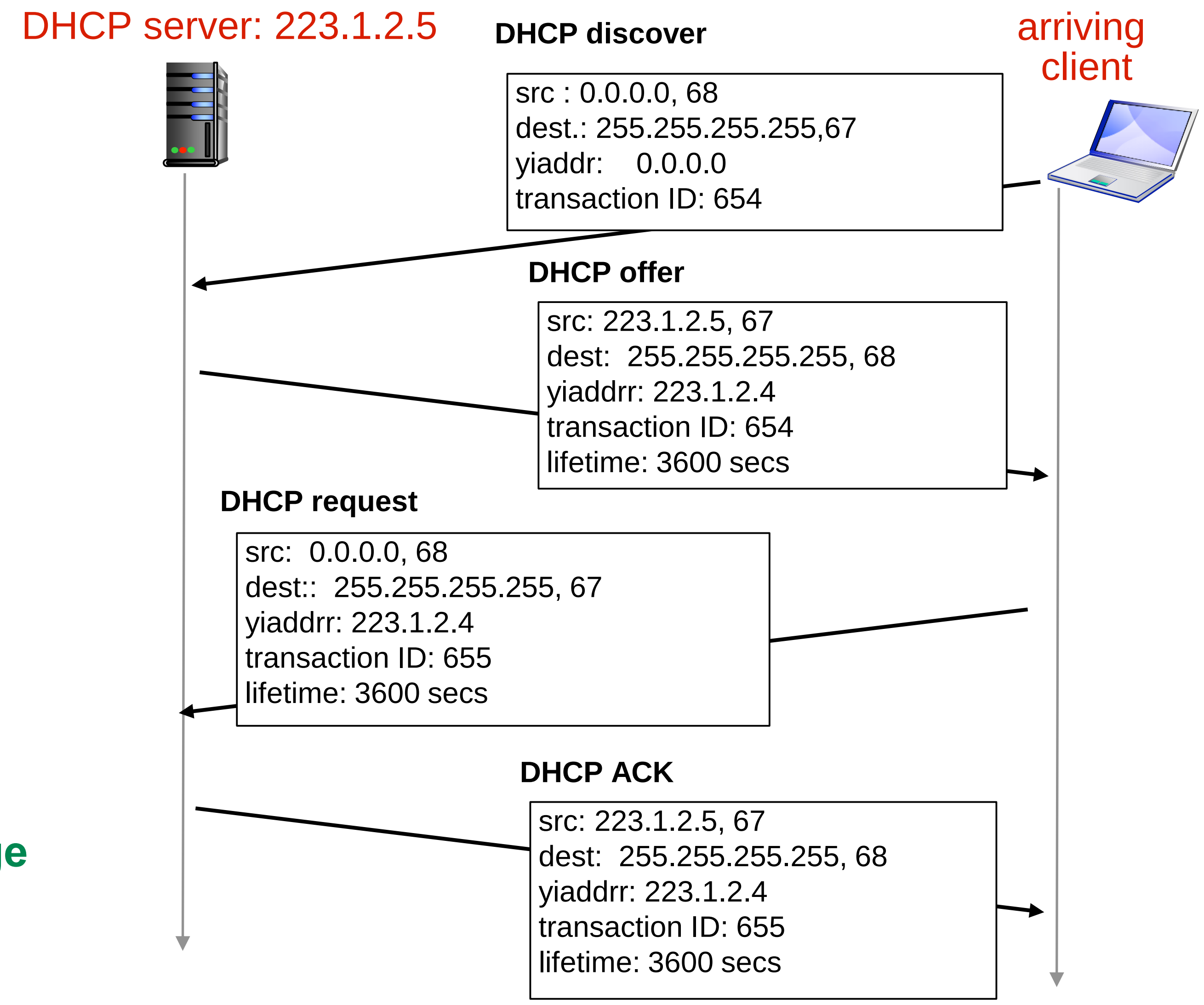
DHCP Client-Server Scenario

- **Host broadcasts "DHCP Discover" message to find a DHCP server [optional]**
  - Host doesn't know address of DHCP server, so send broadcast
- **DHCP server(s) respond with "DHCP Offer" message [optional]**
  - Offer is also broadcast message since newly connected client has no address yet!
- **Host requests IP address with "DHCP Request" message**
  - Can choose from multiple offers if there are multiple DHCP servers
  - Another broadcast message
- **DHCP server sends address with "DHCP ACK" message**
  - Again, this message is broadcast
- **Client can now begin to use new IP address**



DHCP Client-Server Scenario

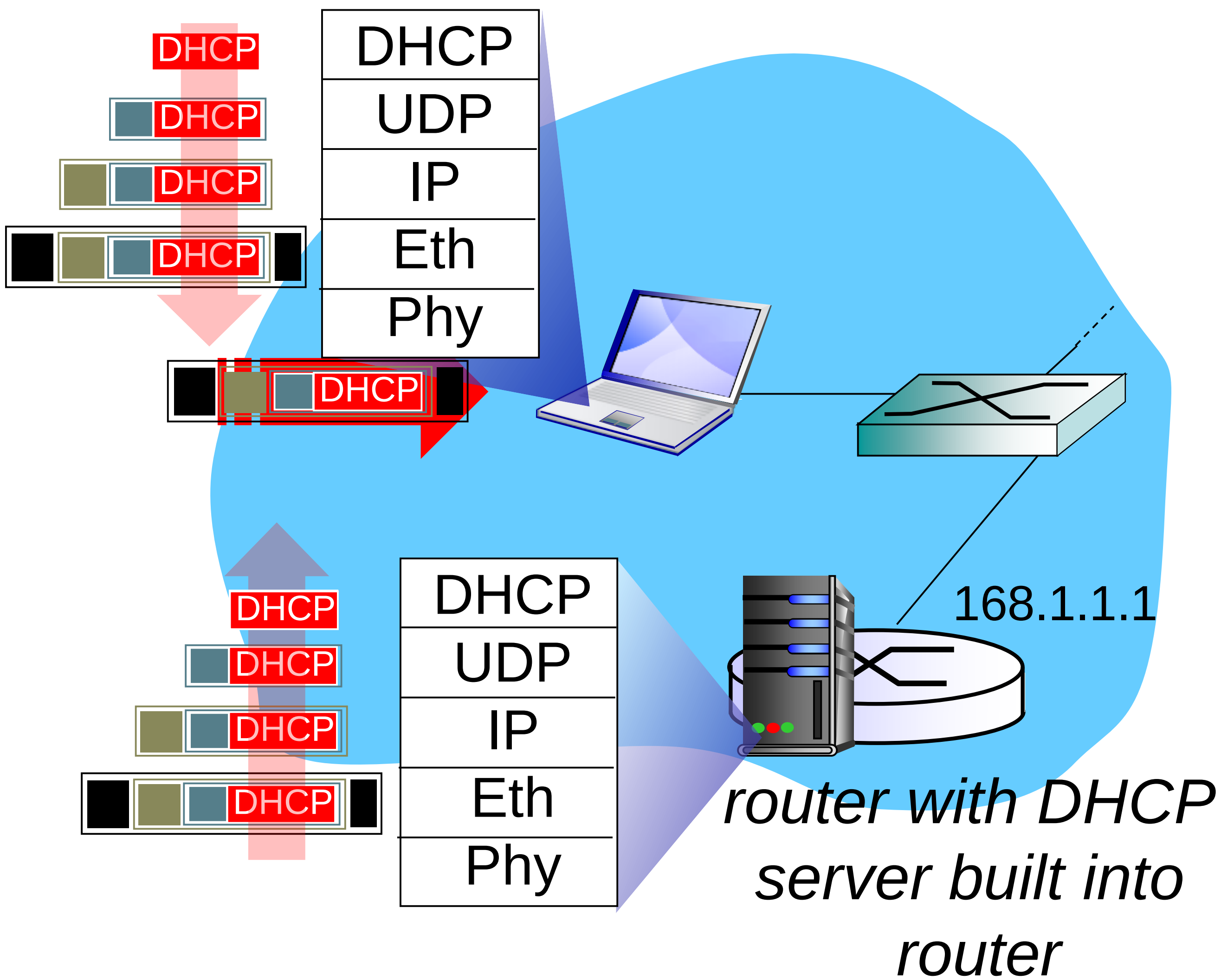
- **Host broadcasts "DHCP Discover" message to find a DHCP server [optional]**
  - Host doesn't know address of DHCP server, so send broadcast
- **DHCP server(s) respond with "DHCP Offer" message [optional]**
  - Offer is also broadcast message since newly connected client has no address yet!
- **Host requests IP address with "DHCP Request" message**
  - Can choose from multiple offers if there are multiple DHCP servers
  - Another broadcast message
- **DHCP server sends address with "DHCP ACK" message**
  - Again, this message is broadcast
- **Client can now begin to use new IP address**



Additional DHCP Services

- **DHCP can return more than just an allocated IP address on subnet**
  - Address of first-hop router for client
  - Name and IP address of DNS sever
  - Network mask (indicating network versus host portion of address)

DHCP: example



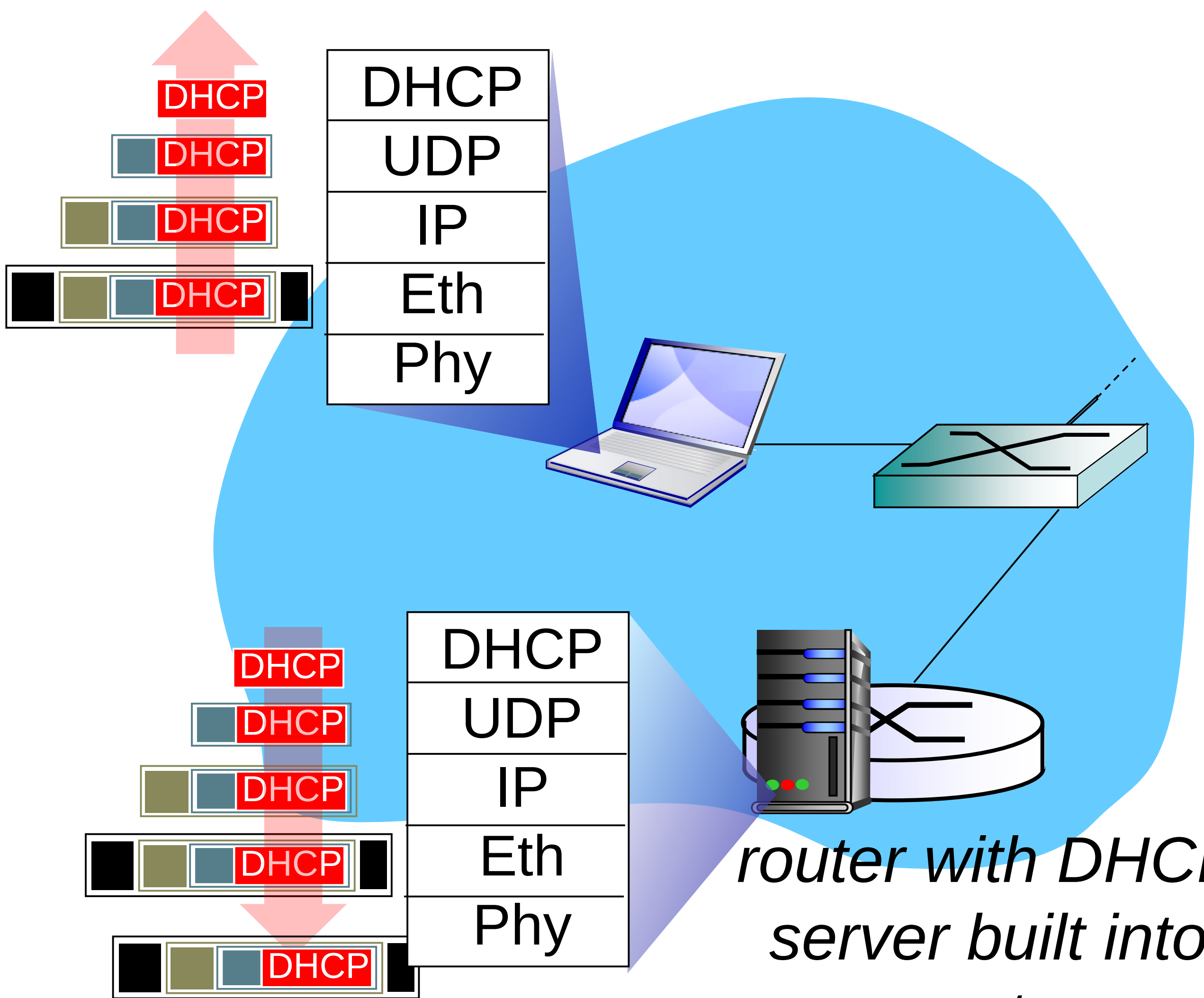
- **connecting laptop needs its IP address, addr of first-hop router, addr of DNS server: use DHCP**

- DHCP request encapsulated in UDP, encapsulated in IP, encapsulated in 802.1 Ethernet

- Ethernet frame broadcast (dest: FFFFFFFFFFFF) on LAN, received at router running DHCP server

- Ethernet demuxed to IP demuxed, UDP demuxed to DHCP

DHCP: example



router with DHCP server built into router

- **DCP server formulates DHCP ACK containing client's IP address, IP address of first-hop router for client, name & IP address of DNS server**
  - encapsulation of DHCP server, frame forwarded to client, demuxing up to DHCP at client
  - client now knows its IP address, name and IP address of DSN server, IP address of its first-hop router