# EITF25 – Internet: Technology and Applications

# Internet Protocols
# -1-
# Network Layer

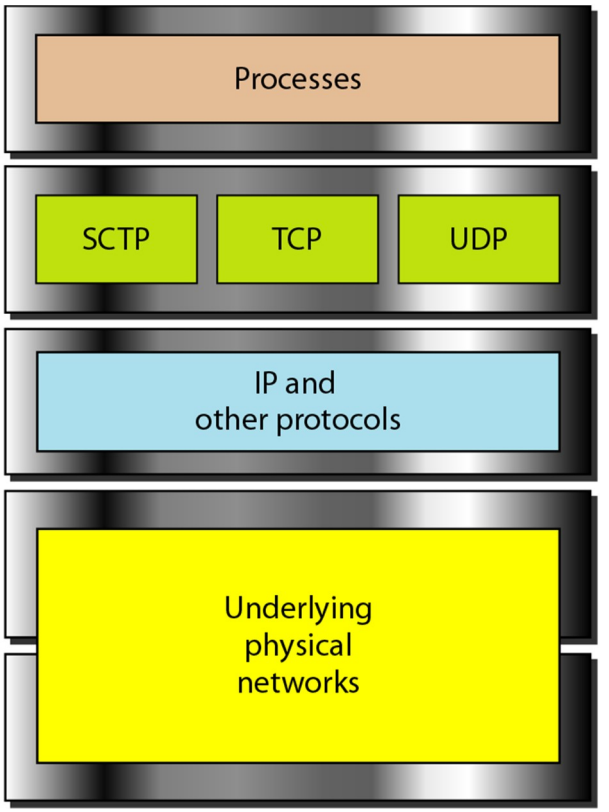

2013, Lecture 04

Kaan Bür, Stefan Höst


# Previously on EITF25

## Medium Access Control Sublayer

- Access methods
  - Slotted ALOHA, CSMA/CD
- Ethernet
  - Evolution of local area networks
- Wireless LAN
  - Hidden terminal problem
  - CSMA/CA



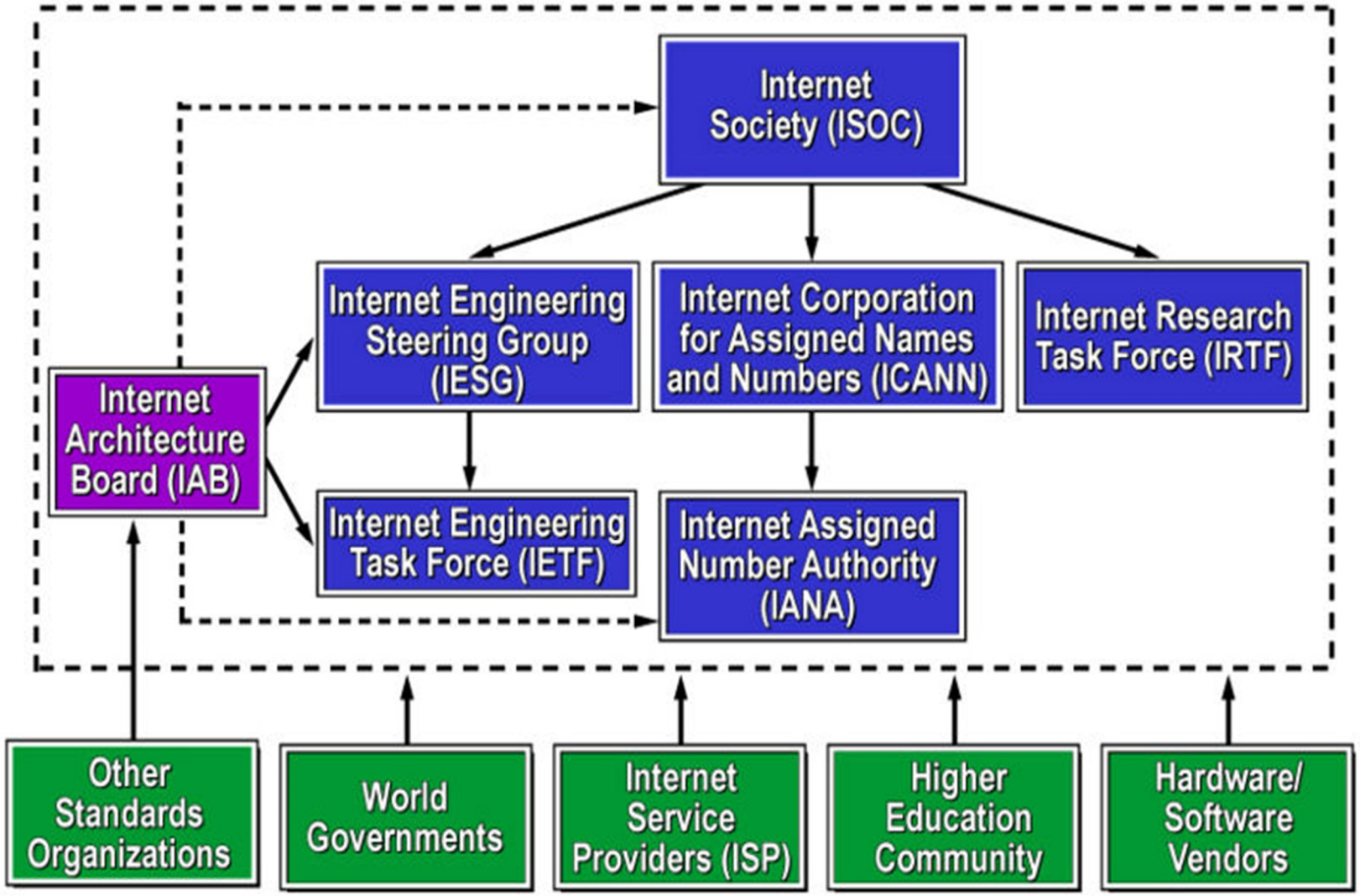
# Today: Internet Protocols (1)

## Network Layer

- Internetworking, routing
  *[Forouzan ed.5 ch.18.1-2]*
- Internet Protocol, v4 & v6
  *[Forouzan ed.5 ch.18.4-5, 19.1, 22.1-2]*

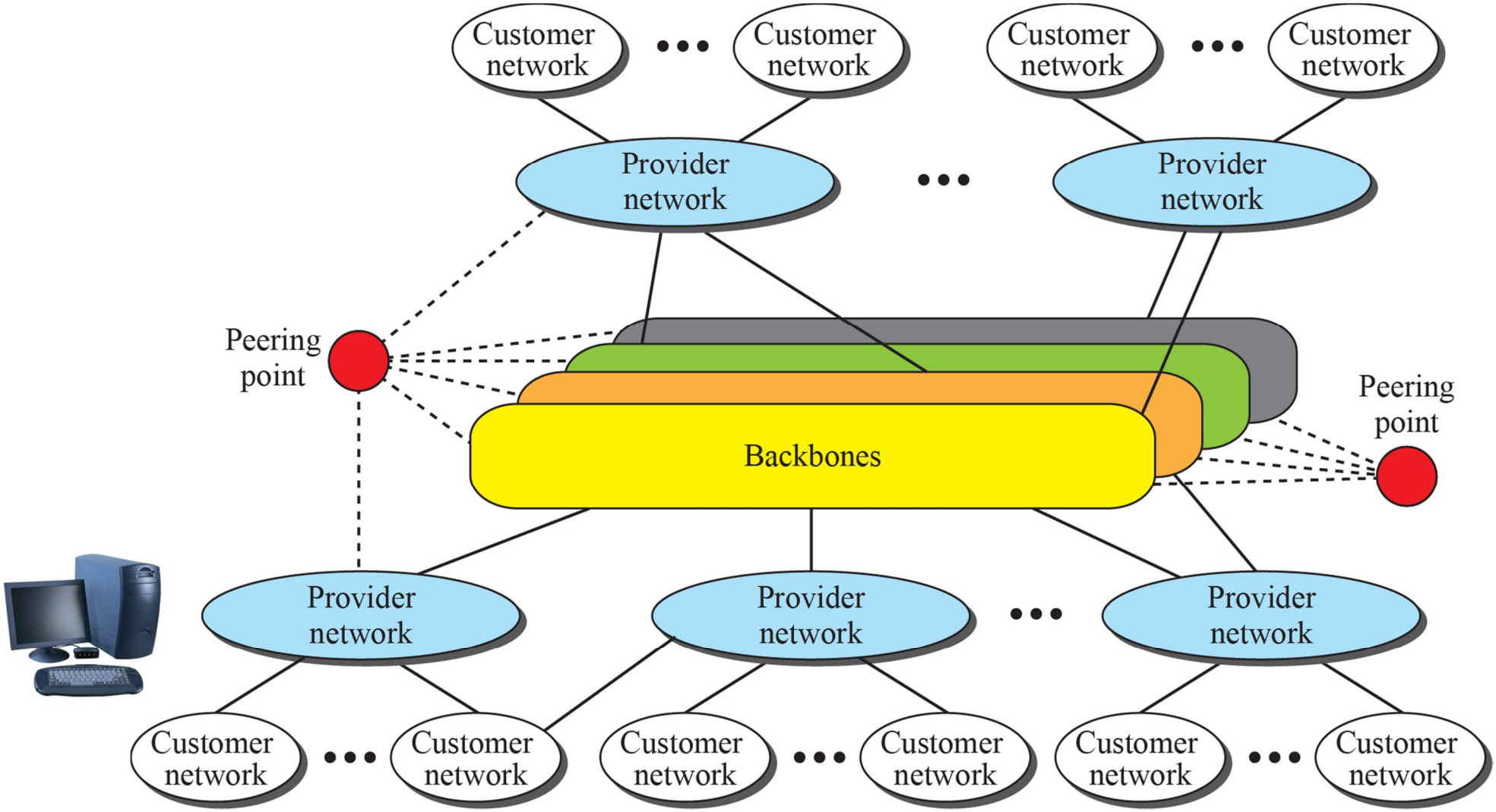
**[Kihl & Andersson: 6.1-3, 7.1-5, 7.8, 8.2]*

# Internet administration




Source: http://www.hill2dot0.com

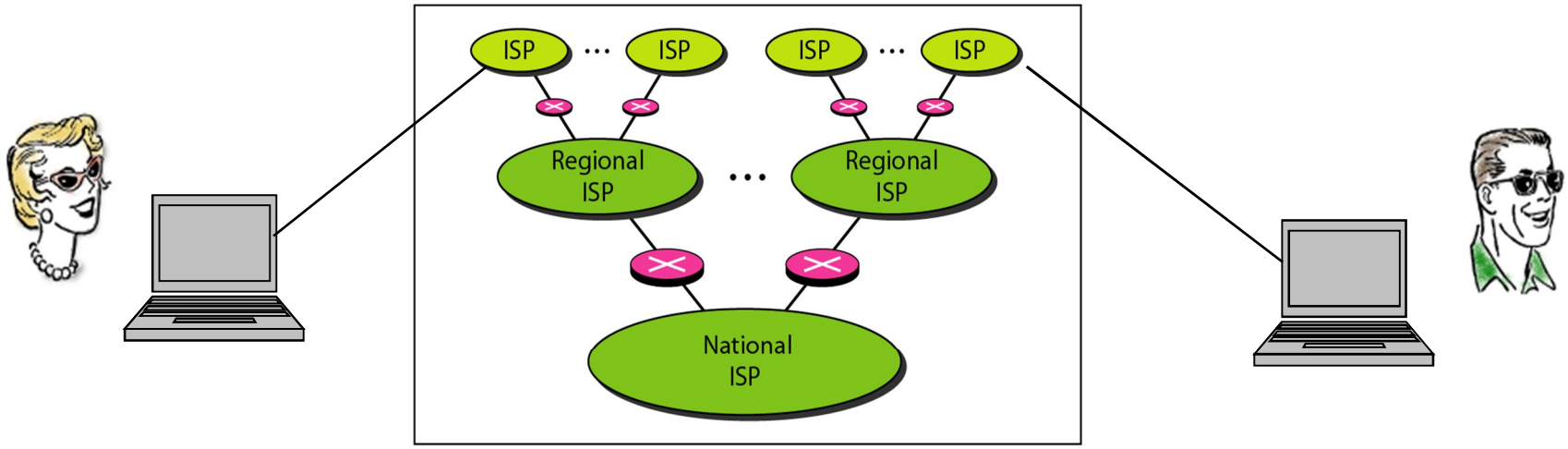
# Internet hierarchy



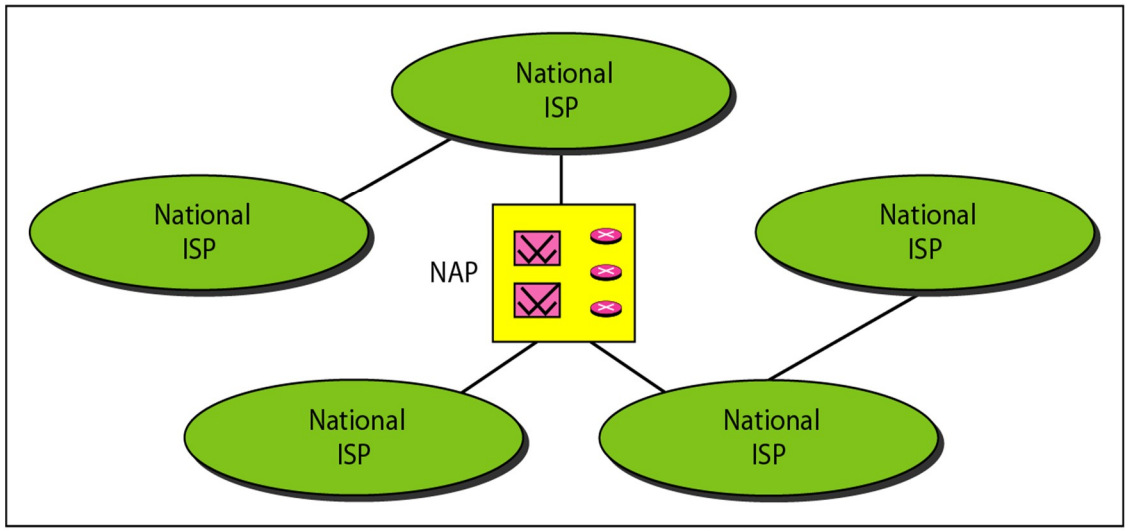
# Network architecture

- Two types of networks in Internet
  - Backbone networks
  - Access networks
- Internet Service Providers (ISP)
  - Today, the access networks and some of the backbone networks are owned by private ISP.
- Network Access Points (NAP)
  - Switching stations providing connection between backbone networks

# Organisation of Internet



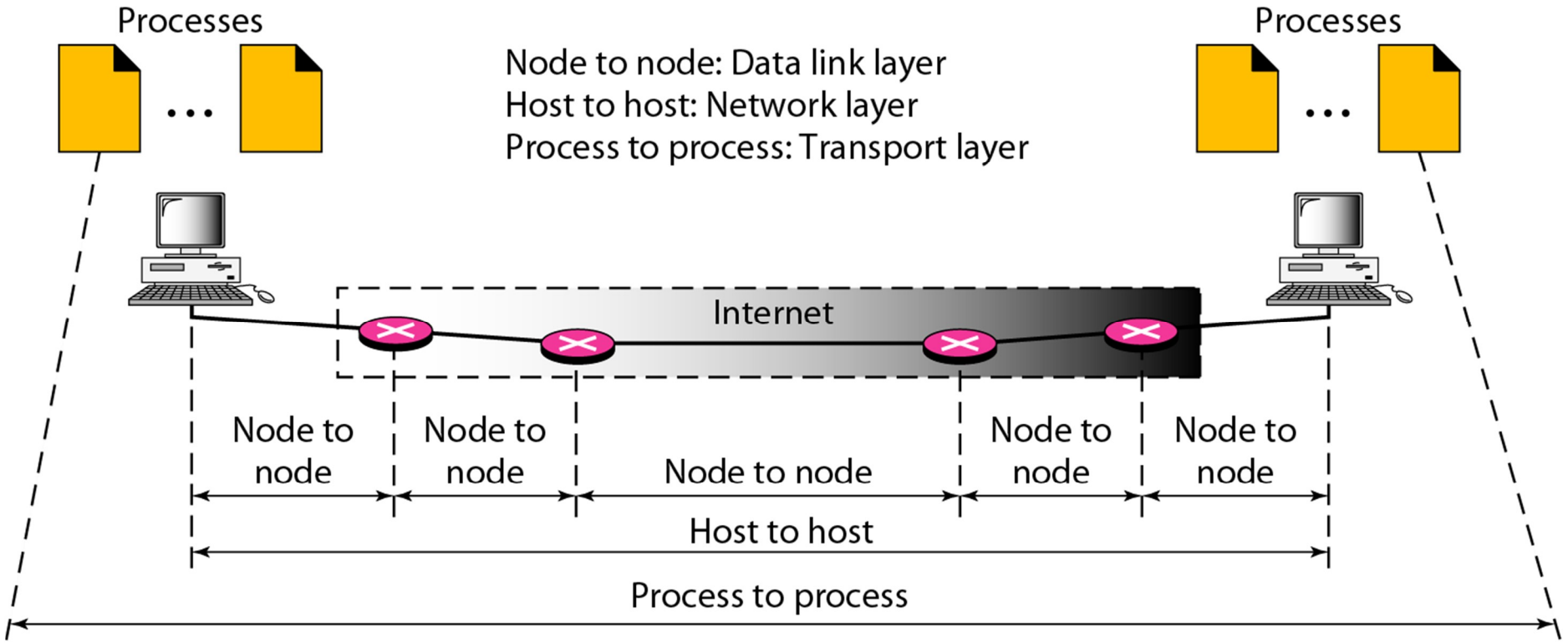

a. Structure of a national ISP




b. Interconnection of national ISPs

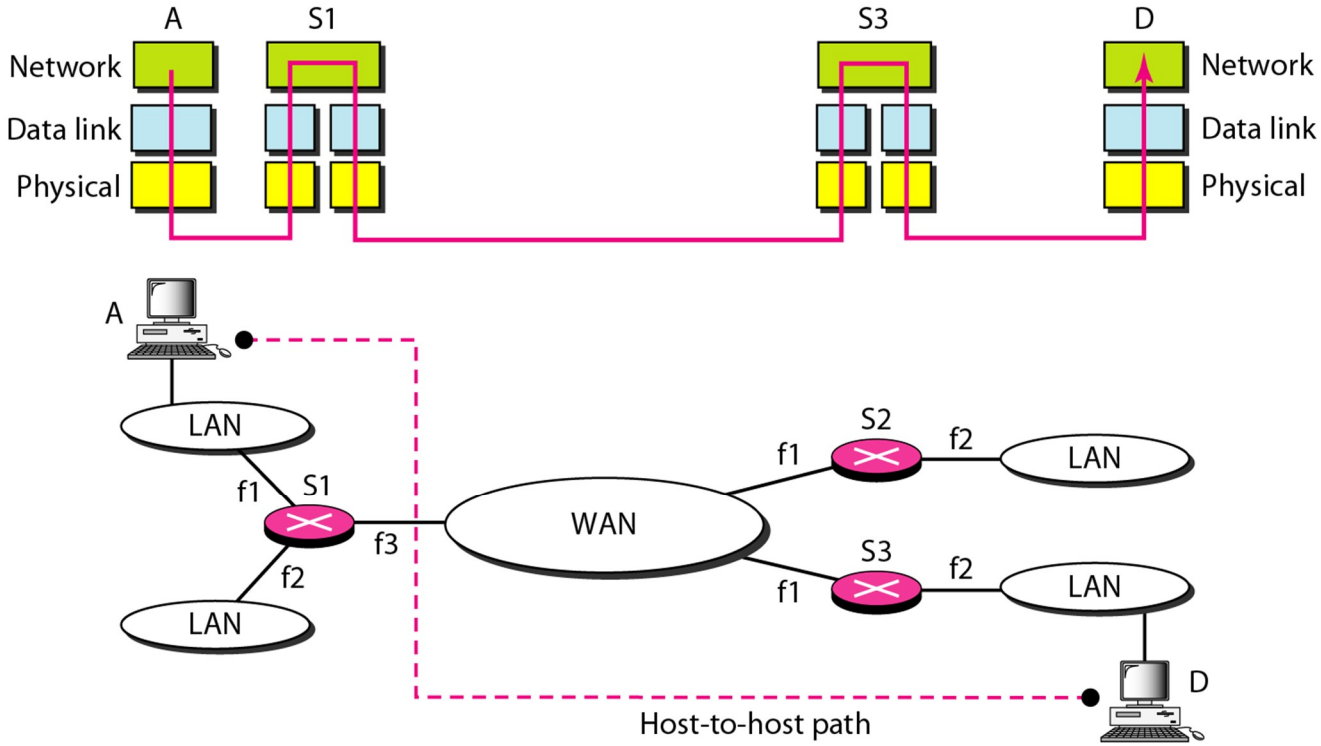
# Host-to-host delivery

- Multiple applications even on the same host



# Network layer

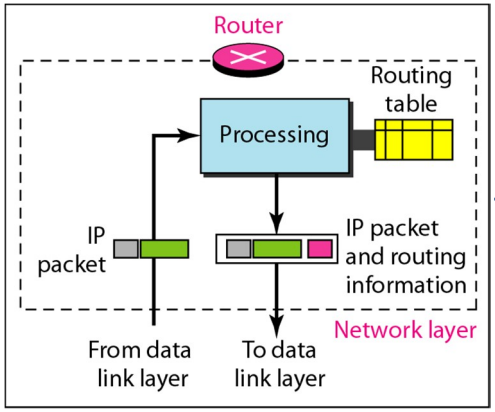
- L3
  - end-to-end



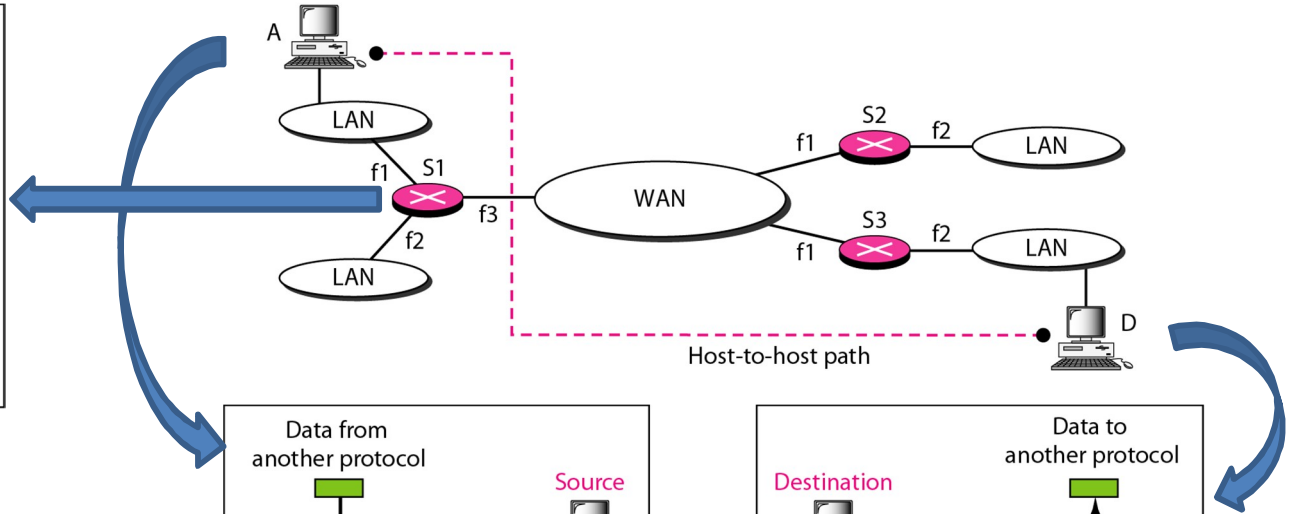

- L2
  - hop-by-hop

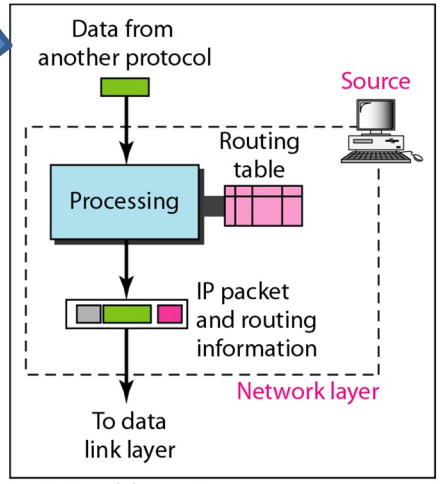
# Network layer: Routing

- L3 is end-to-end

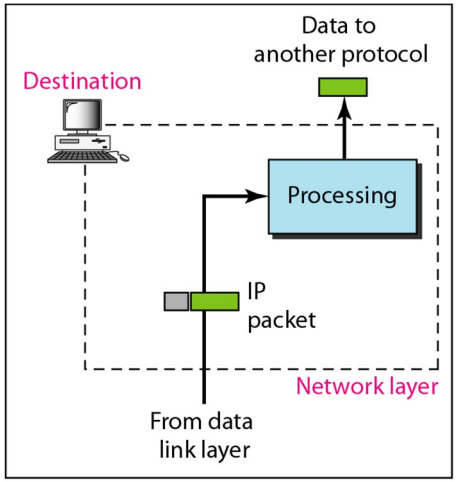




c. Network layer at a router




a. Network layer at source



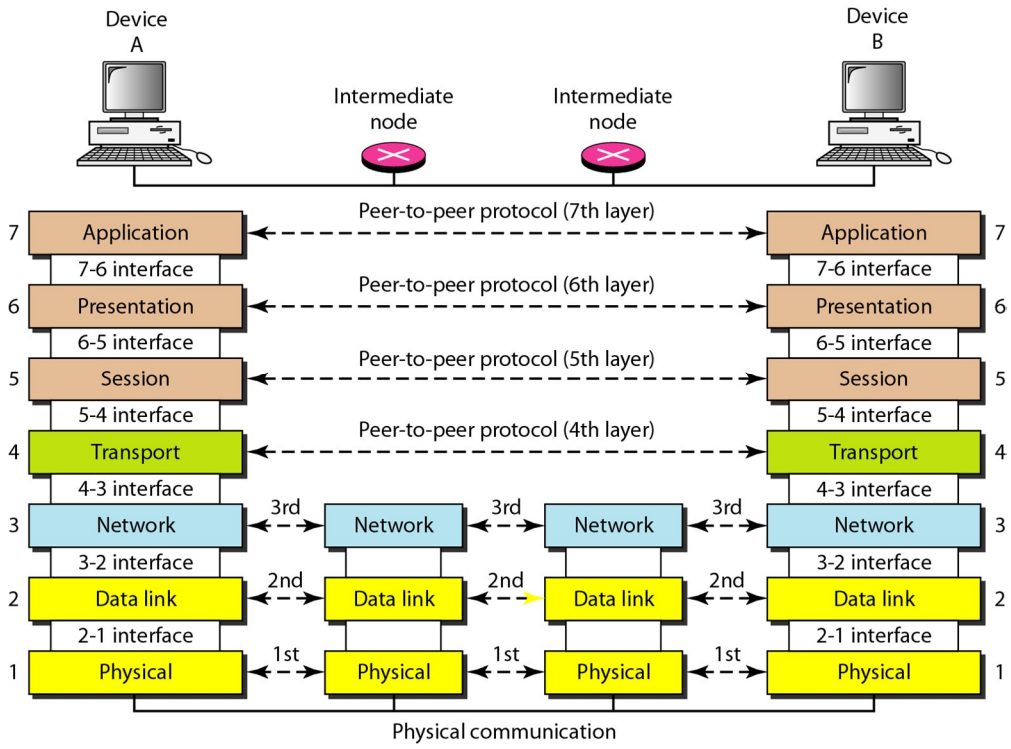

b. Network layer at destination

Two functions:

1. Addressing
2. Feedback

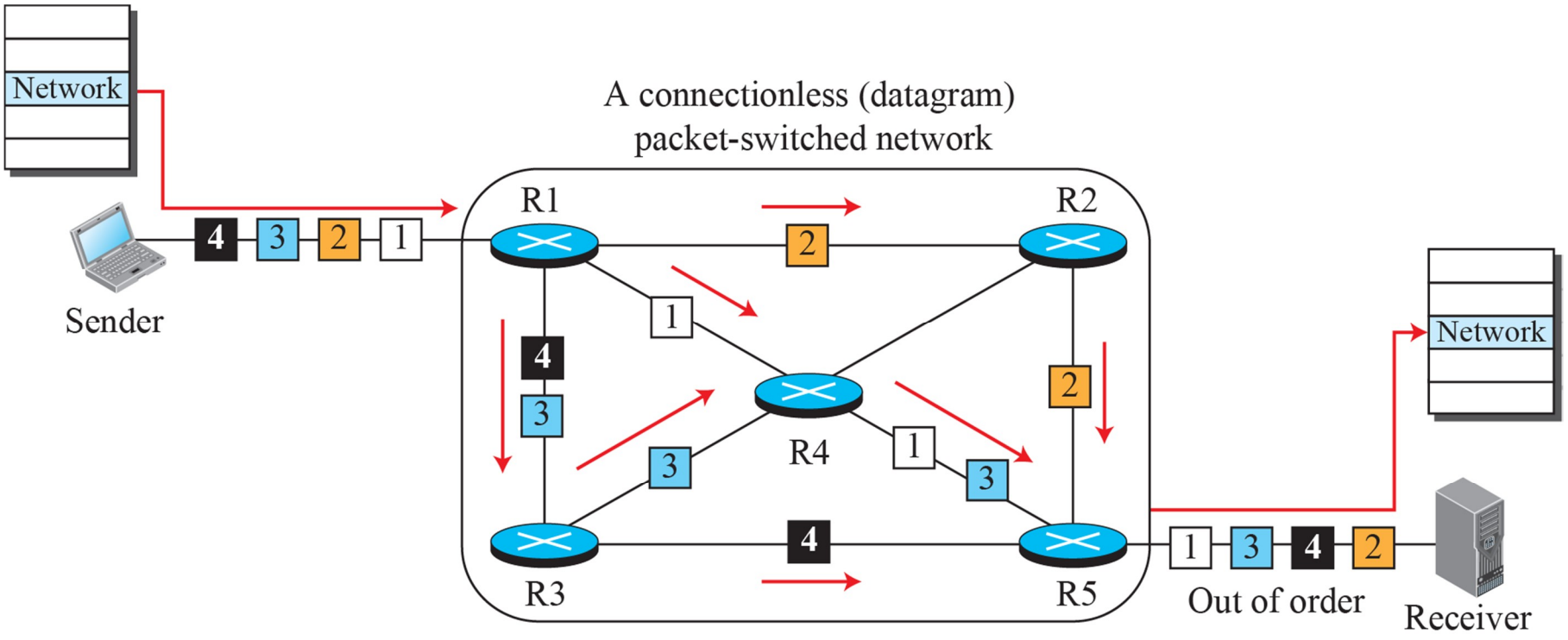
# Routing

- Choosing the optimal path
  - Using a cost metric
- Sharing information
  - Central
  - Distributed
- Algorithms
  - Rules and procedures
  - Updates



# Packet-switched routing

- Choosing the optimal path
  - Using a cost metric

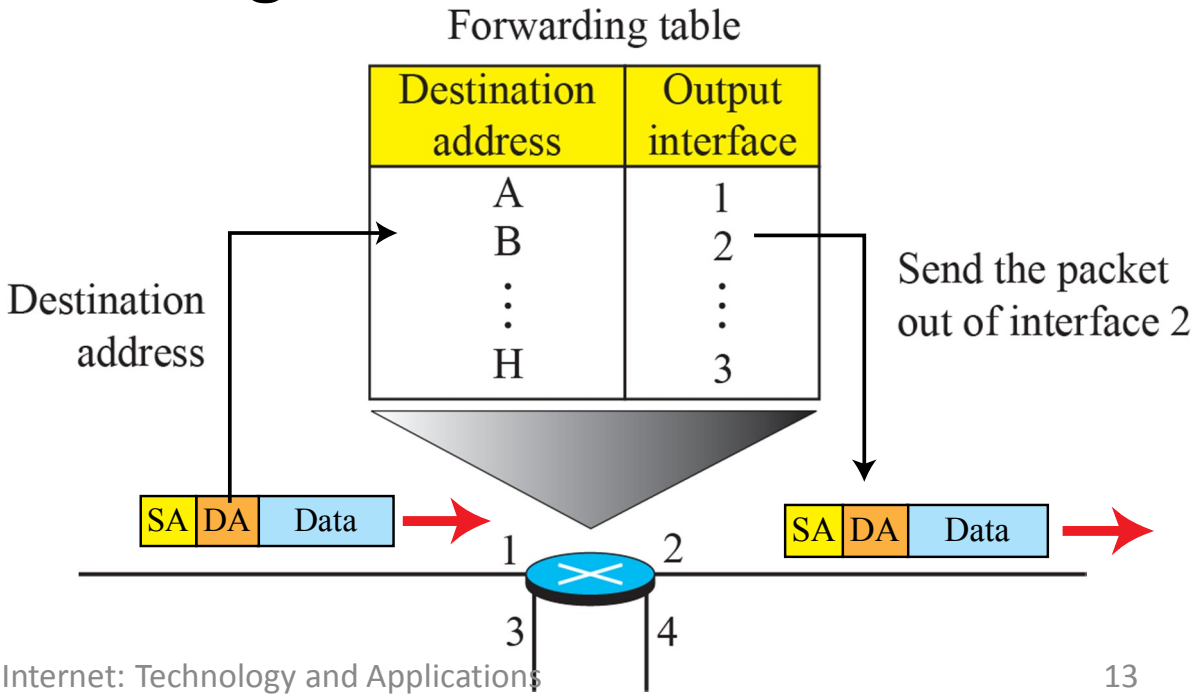


# Router

- Internetworking device
  - Passes data packets between networks
  - Checks ***Network Layer*** addresses
  - Uses Routing/forwarding tables

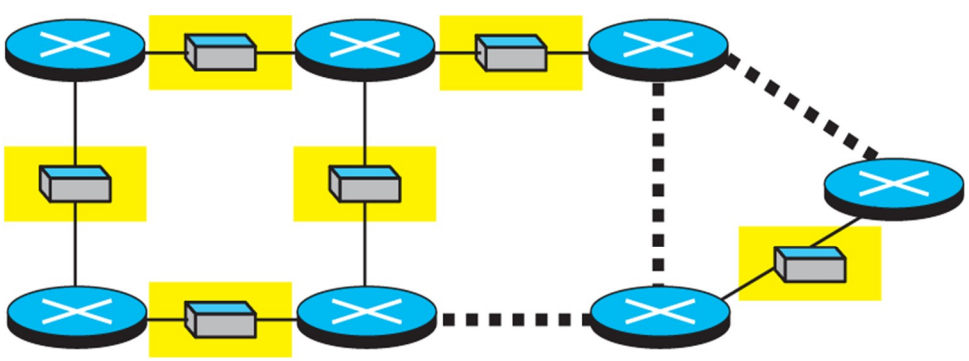
Two functions:

1. Routing
2. Forwarding

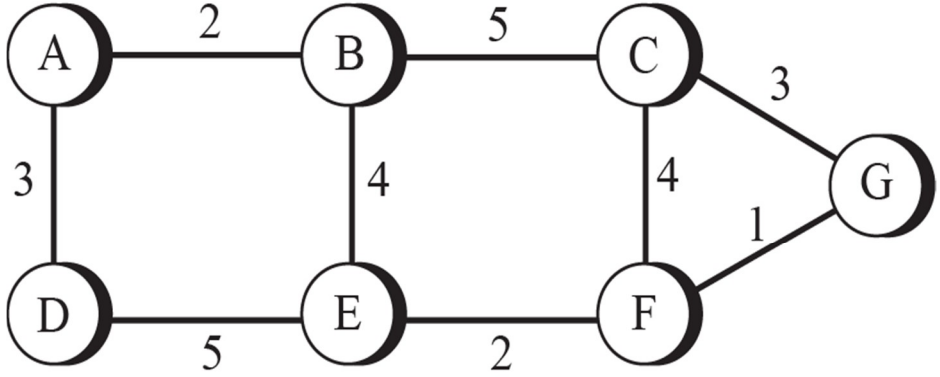


# Routing algorithm

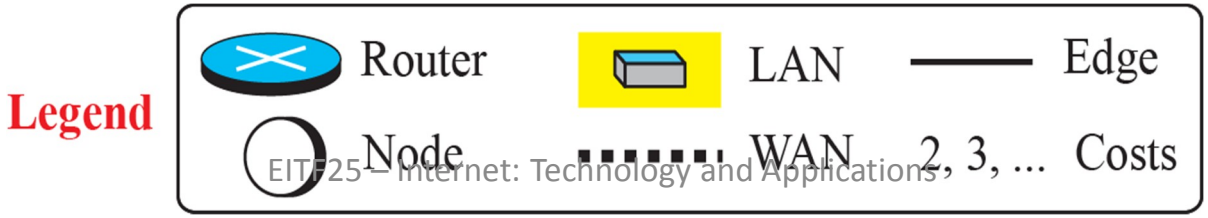
- Find route with least cost between source and destination.
- Update routing tables



a. An internet




b. The weighted graph



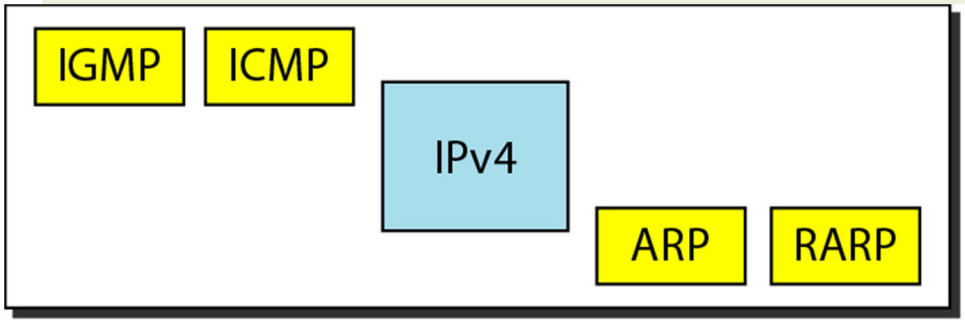
# Network layer protocols

- We need a universal address system. This is called the *network address*.
- We need rules for data forwarding. This is called *routing*.
- We need entities connecting several networks together and forwarding data between them. These are called *routers*.

# Internet Protocol

## *IPv4*

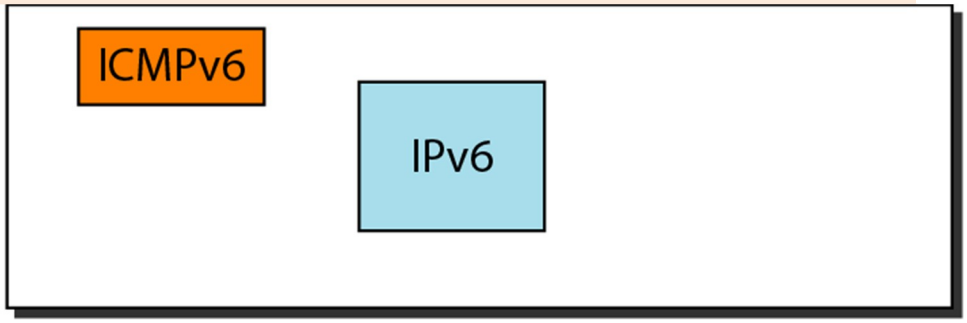
- Addressing scheme
  - Hierarchy
  - Configuration
  - Lookup
- Datagram format




Network layer in version 4

## *IPv6*

- Larger address space
- Better header format
  - Extendible
  - More secure
- Support for QoS




Network layer in version 6

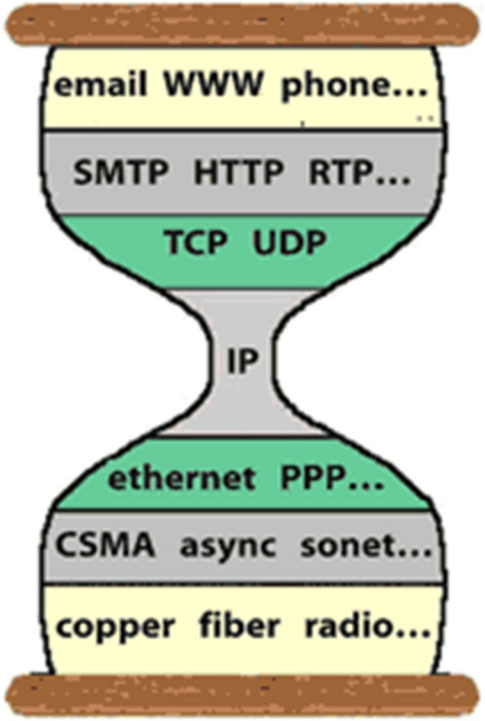
# Internet

- All networks that are part of Internet have one thing in common:

  **They all use the same network protocol, *Internet Protocol (IP)*!**
- They do not need to use the same link protocols, which means that all kinds of networks can be part of Internet.
- The protocol framework for Internet is called ***the TCP/IP model***.

# TCP/IP model

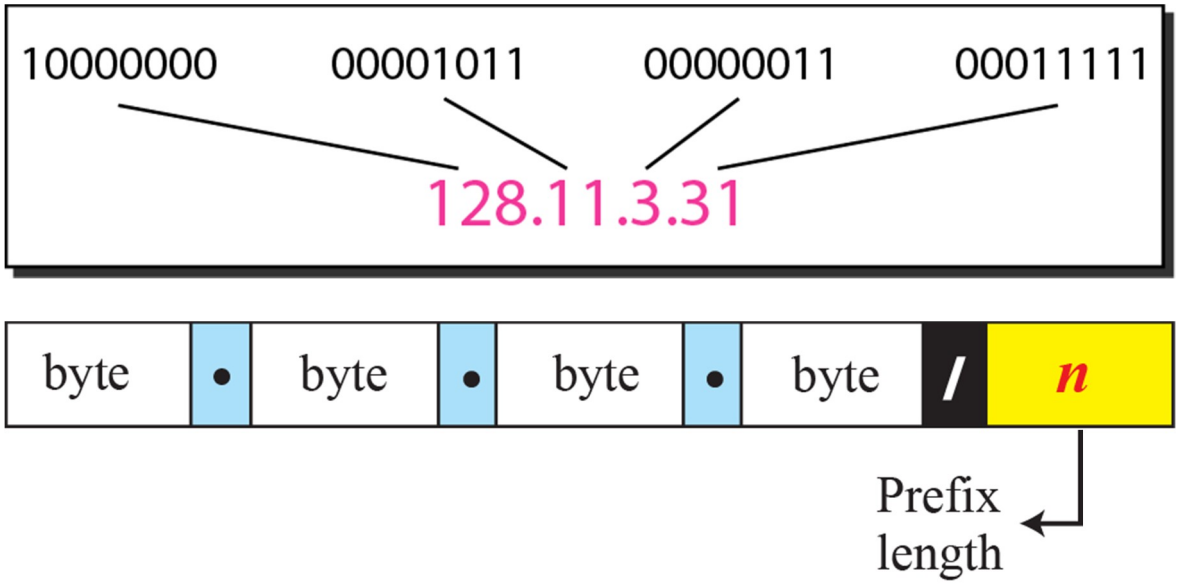
- Sometimes illustrated with a hourglass.
- IP can be used on all types of networks
- This has made Internet what it is today.



# IPv4 addresses

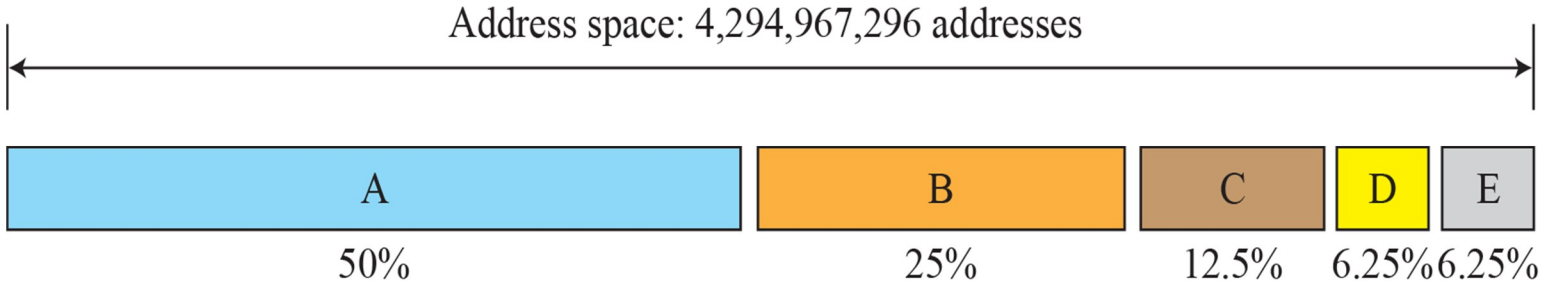
- 32 bits = 4 bytes
- $2^{32} = (2^8)^4 = 256^4 = 4\ 294\ 967\ 296$
- Classful vs. classless hierarchy

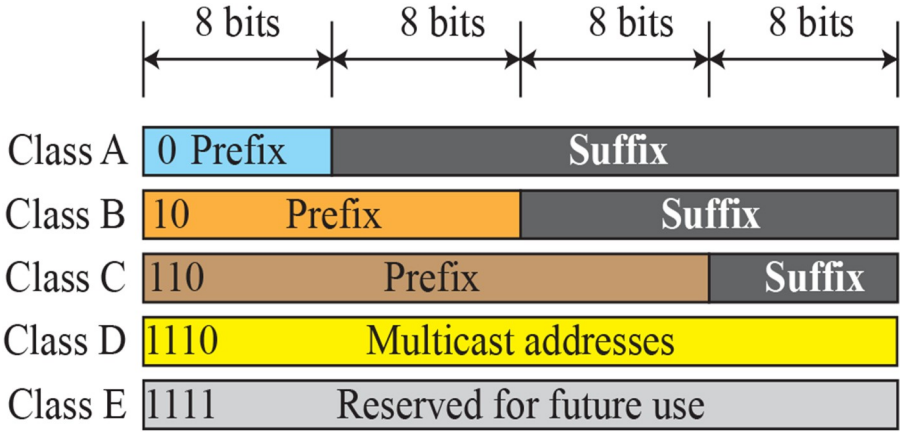
- Notations
  - Dotted decimal
  - Slash (CIDR)



# Classful addressing

- Five address classes defined: A, B, C, (D and E)




| Class | Prefixes | First byte |
|---|---|---|
| A | $n = 8$ bits | 0 to 127 |
| B | $n = 16$ bits | 128 to 191 |
| C | $n = 24$ bits | 192 to 223 |
| D | Not applicable | 224 to 239 |
| E | Not applicable | 240 to 255 |

# Classful addressing

- Organizations can only get addresses in one of the predefined blocks.

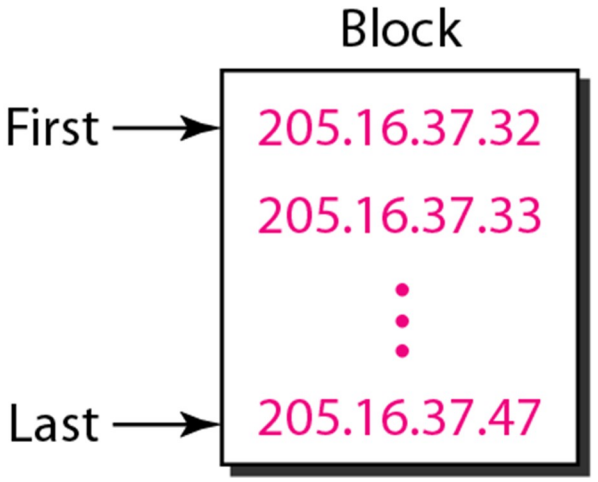
| *Class* | *Number of Blocks* | *Block Size* | *Application* |
|---|---|---|---|
| A | 128 | 16,777,216 | Unicast |
| B | 16,384 | 65,536 | Unicast |
| C | 2,097,152 | 256 | Unicast |

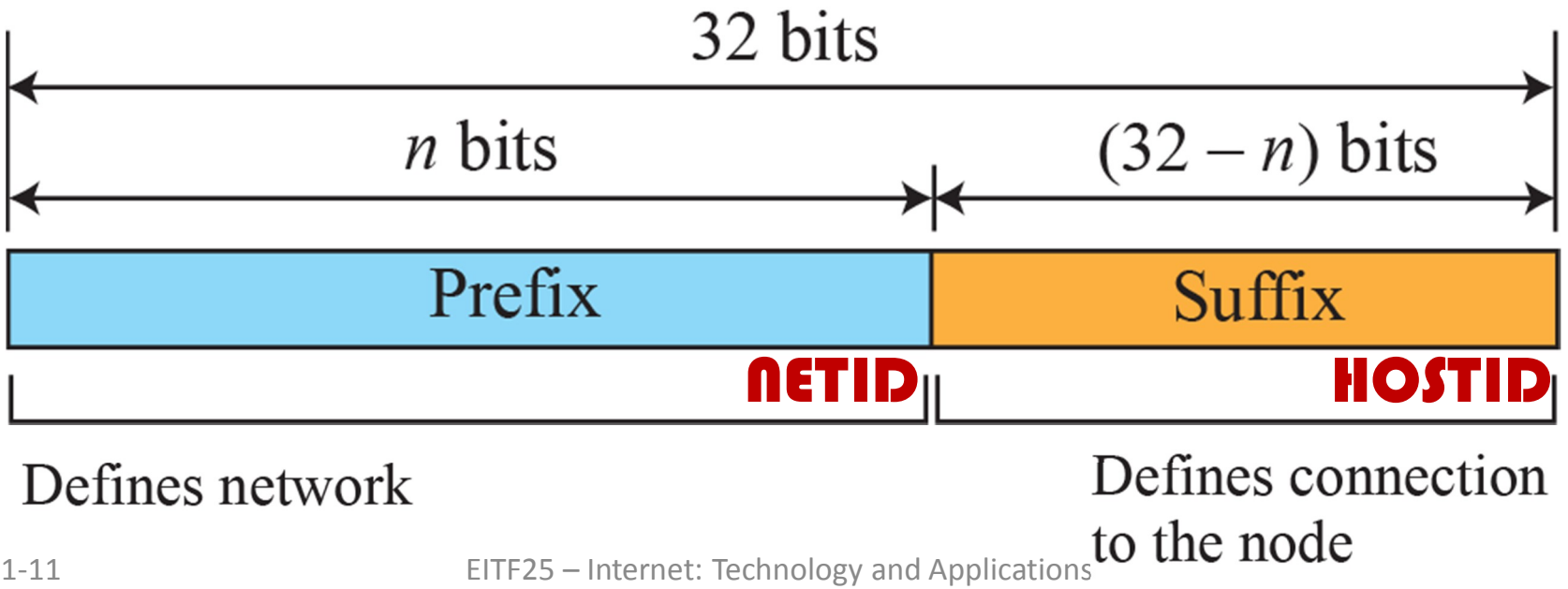
# Address depletion

- Classful addressing defined as there were very few networks connected to the Internet.
- With the growth of Internet, the address classes didn’t match the reality.
- Subnetting and supernetting was introduced.
  - Class A and B address blocks divided into subnets.
  - Several Class C address blocks combined into larger blocks called supernets.

# Classless addressing

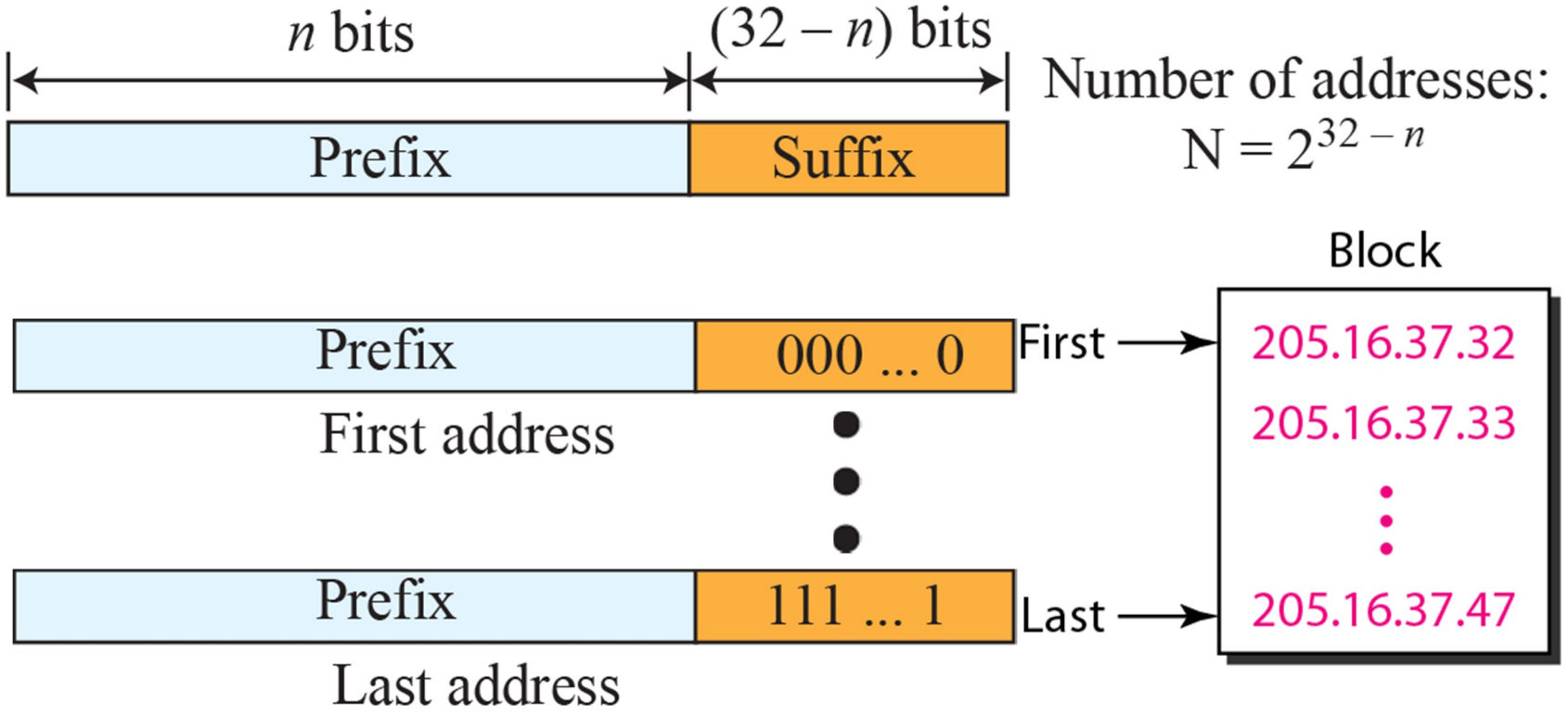
- Addresses in blocks
  - Block size power of 2
  - $N = 2^{32-n}$
  - First address divisible by $N$



# *Exercise:* Classless addressing

- CIDR = slash notation with mask ***/n***
- 205.16.37.39/28

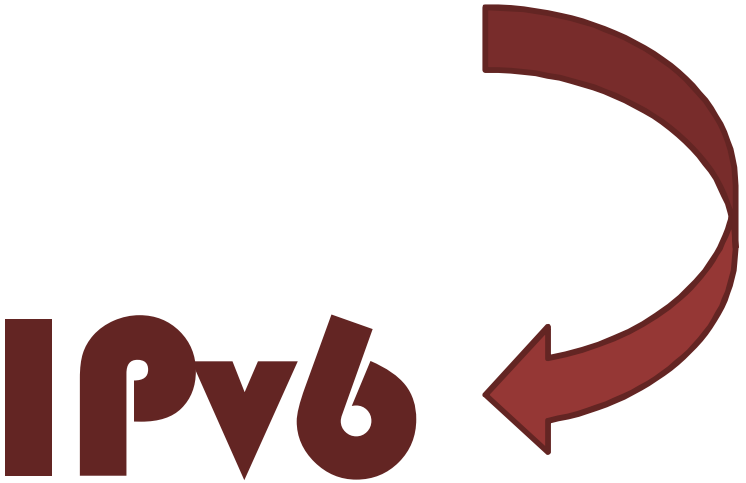


# See you in 15’ :)

- After the break
  - IPv6
  - Fragmentation
  - Forwarding, ARP

# Problems with IPv4

- Address space too small
- Not designed for real-time applications
- No support for encryption and authentication

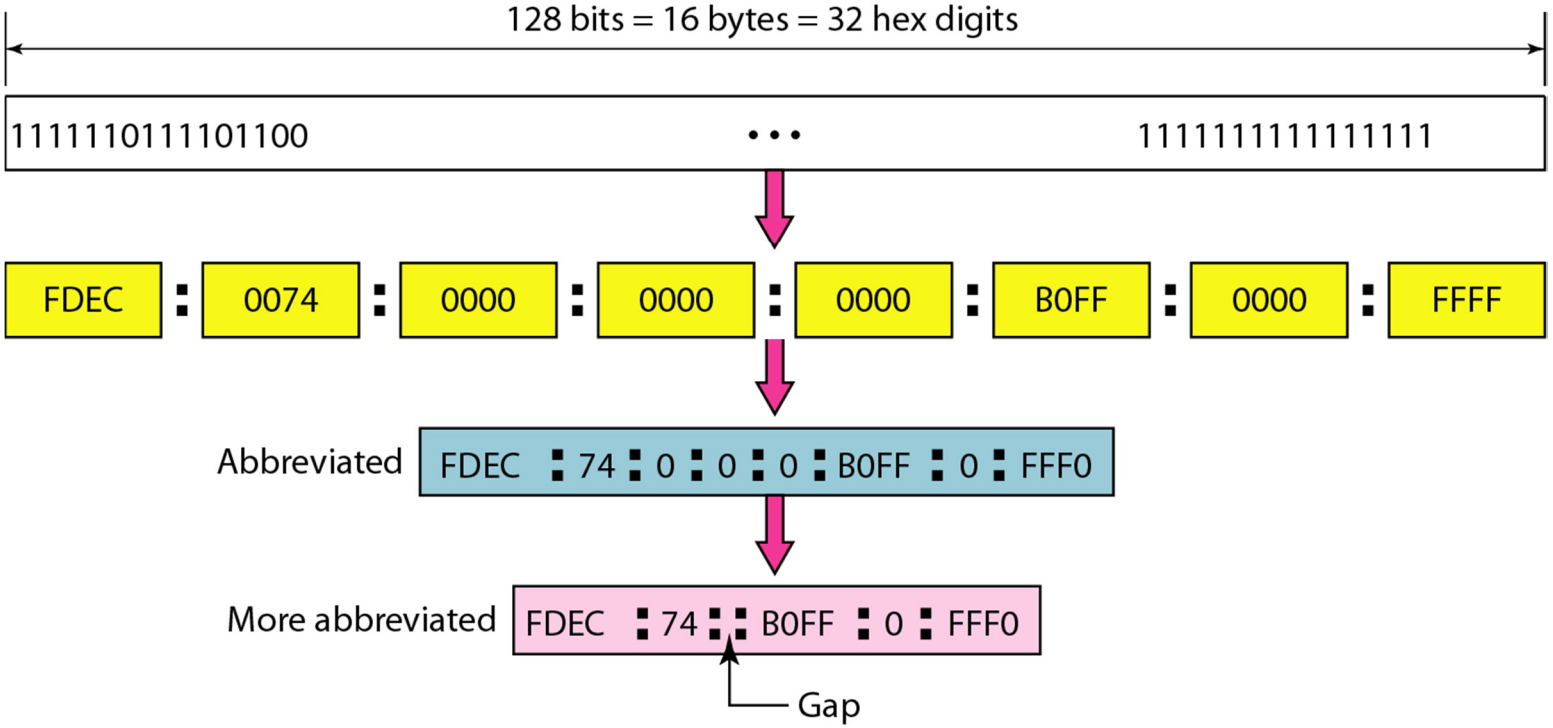


# Some advantages with IPv6

- **Larger address space:** 128 bit-long addresses.
- **Better header format:** base header has constant length (40 bytes). Options can be inserted when needed.
- **Support for more security:** Encryption and authentication options.
- **Support for real-time applications:** Special handling of datagram can be requested.

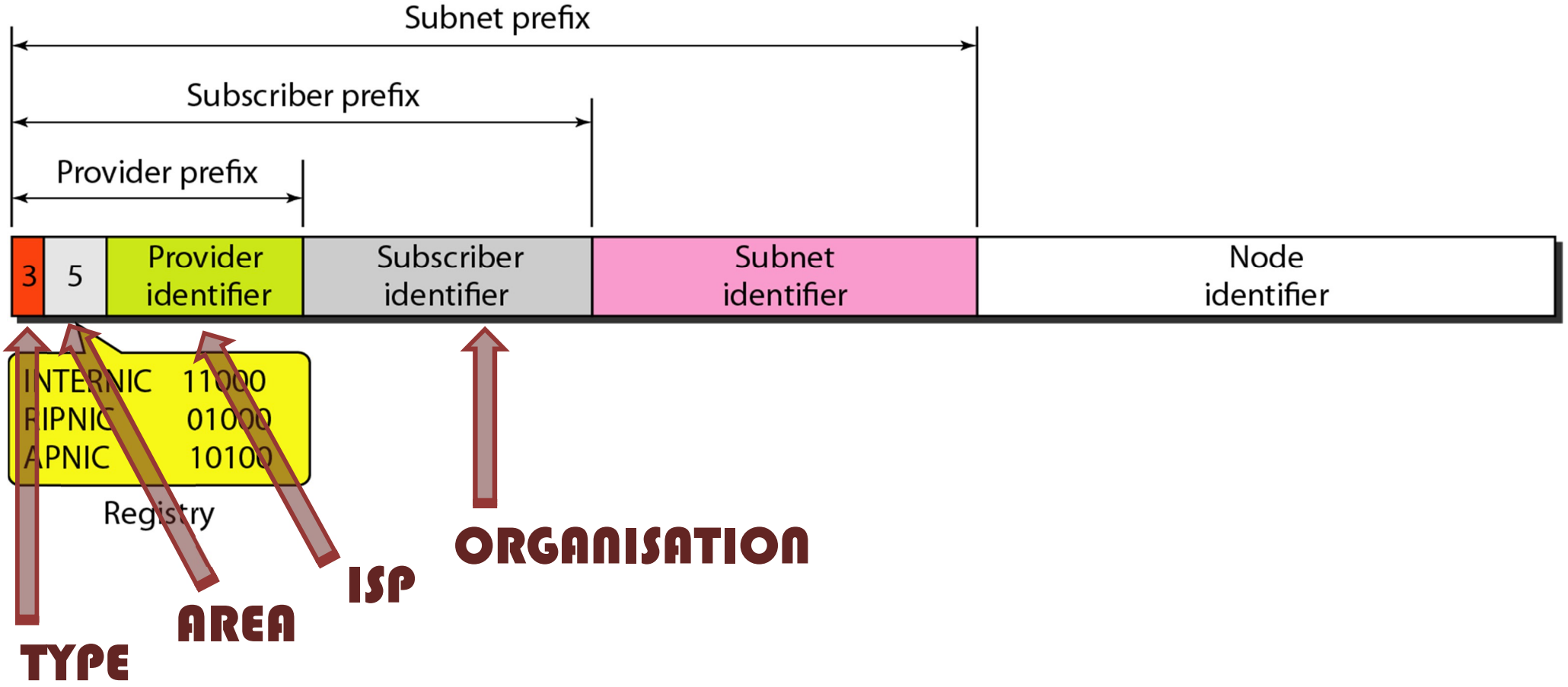
# IPv6 addresses

- 128 bits = 16 bytes
- $2^{128} = 2^{32} \cdot 2^{96} > 3 \cdot 10^{35}$
- Notations

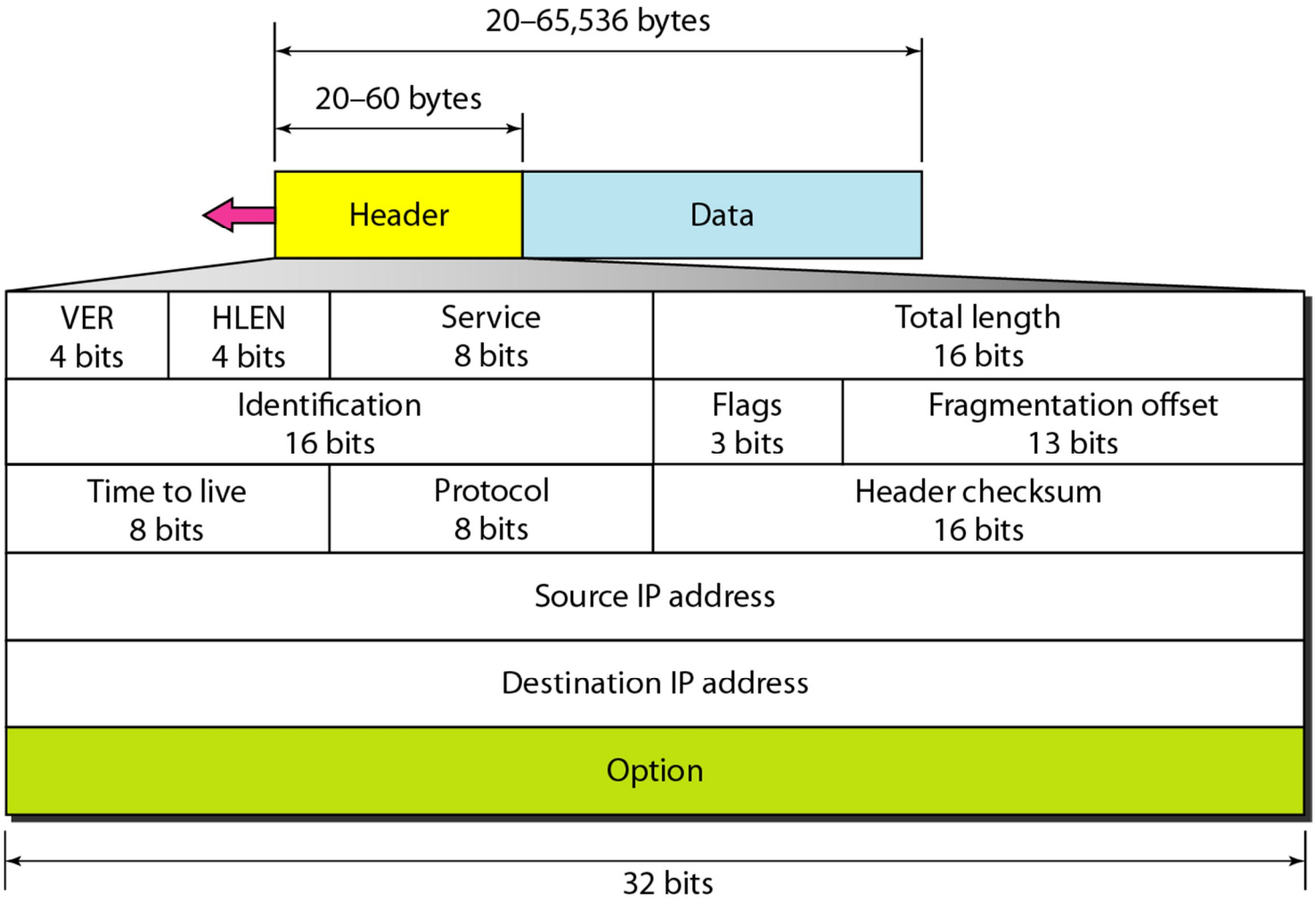


# Global unicast addresses

- Identify individual computers

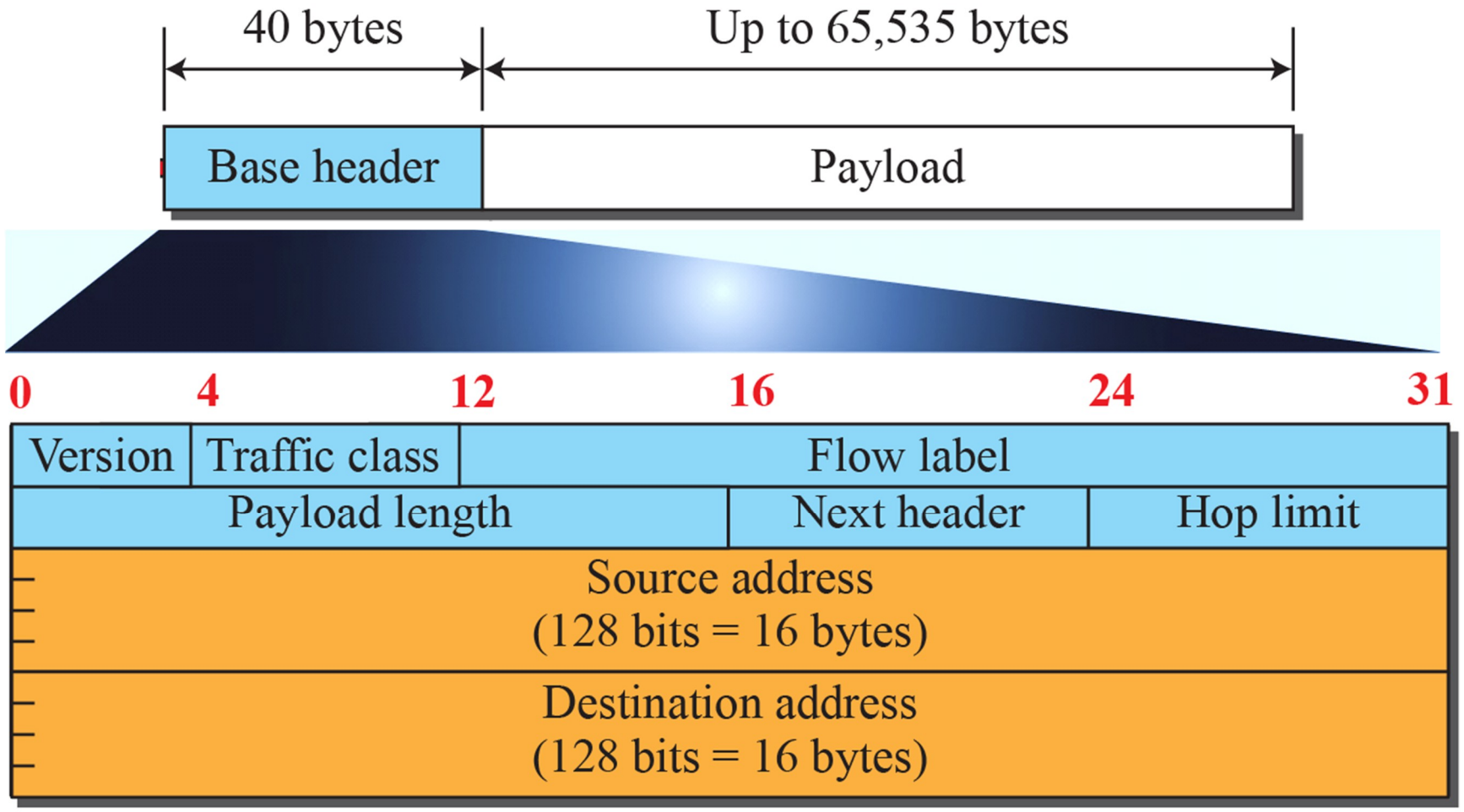


# IPv4 datagram

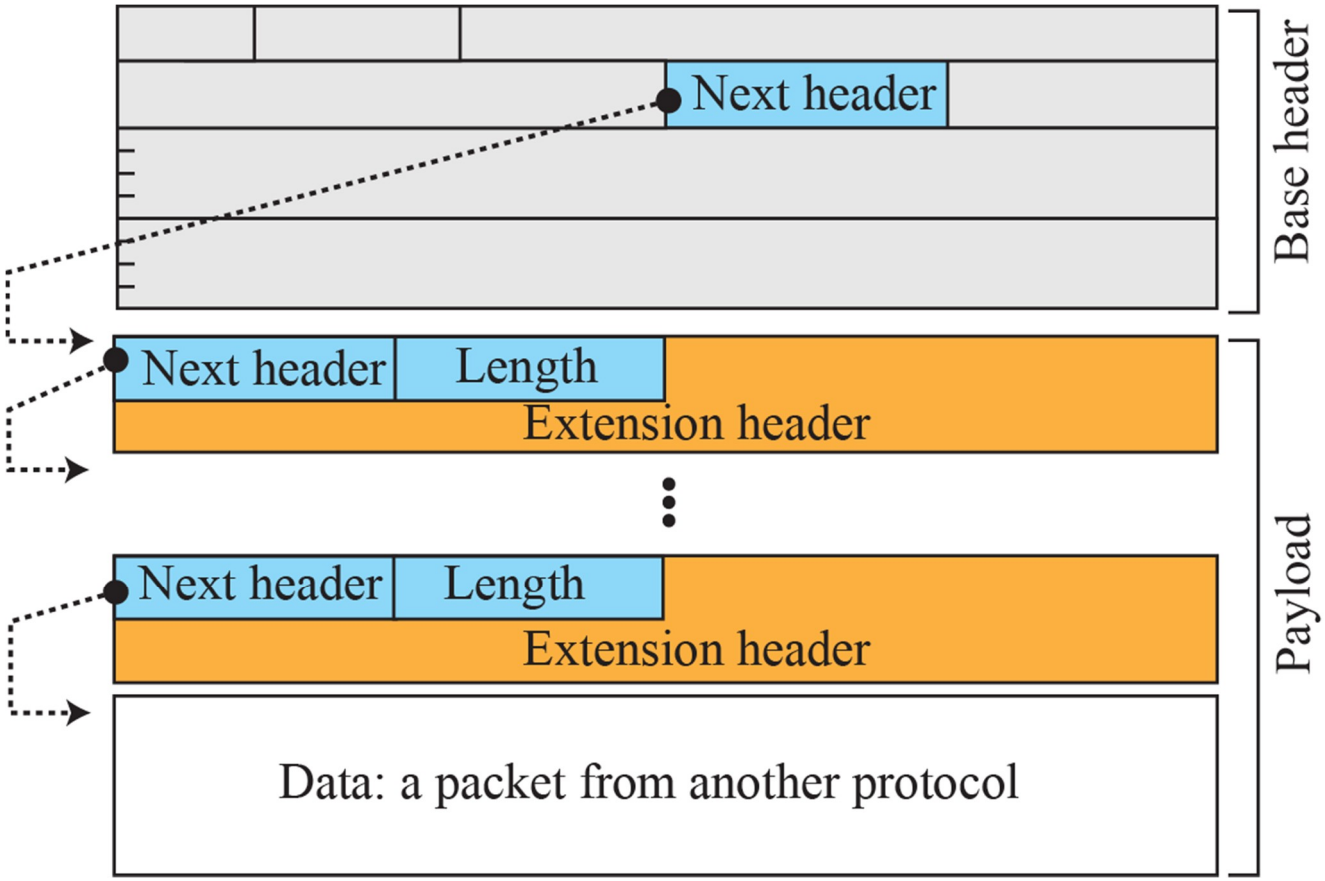


# IPv6 datagram

- Simpler base header, flexible for extensions



# IPv6 extension headers




**Some next-header codes**

00: Hop-by-hop option
02: ICMPv6
06: TCP
17: UDP
43: Source-routing option
44: Fragmentation option
50: Encrypted security payload
51:Authentication header
59: Null (no next header)
60: Destination option

# Traffic Classes → Packet priorities

- 0 .. 7
  - Congestion controlled

| *Priority* | *Meaning* |
|---|---|
| 0 | No specific traffic |
| 1 | Background data |
| 2 | Unattended data traffic |
| 3 | Reserved |
| 4 | Attended bulk data traffic |
| 5 | Reserved |
| 6 | Interactive traffic |
| 7 | Control traffic |

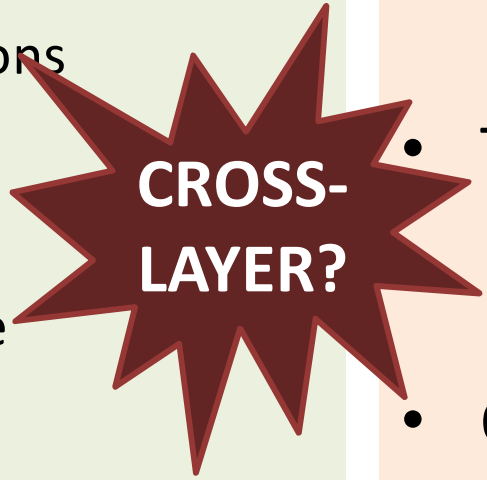
- 8 .. 15
  - Non-congestion controlled

| *Priority* | *Meaning* |
|---|---|
| 8 | Data with greatest redundancy |
| . . . | . . . |
| 15 | Data with least redundancy |

# IPv6 and QoS

## *Flow label*

- Identification of a stream
  - TCP sessions
  - Virtual connections
- Processing
  - Flow label table
  - Forwarding table
- Routing
  - Algorithms still necessary
  - But not run for every packet!

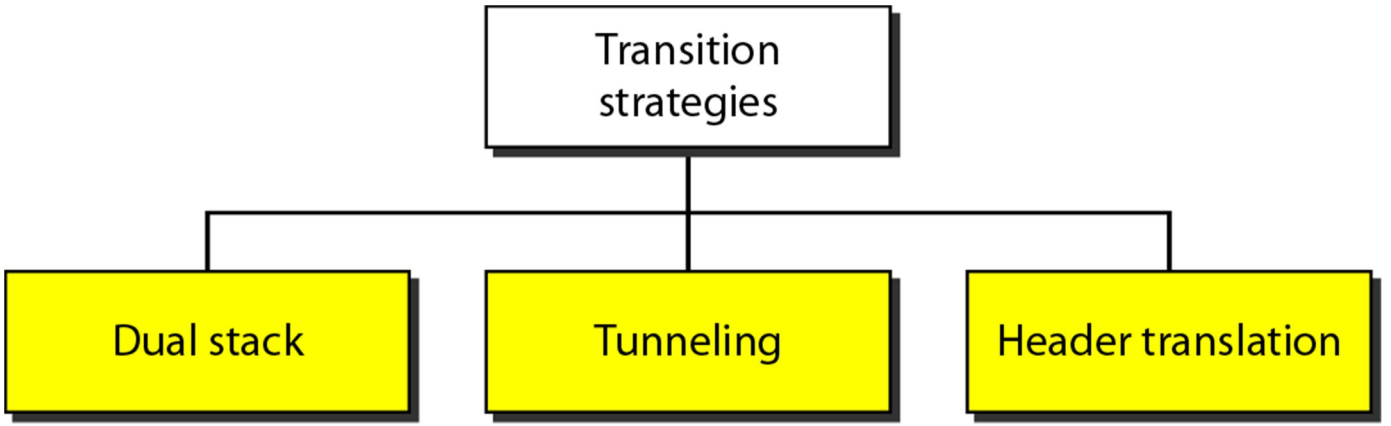



## *Traffic class*

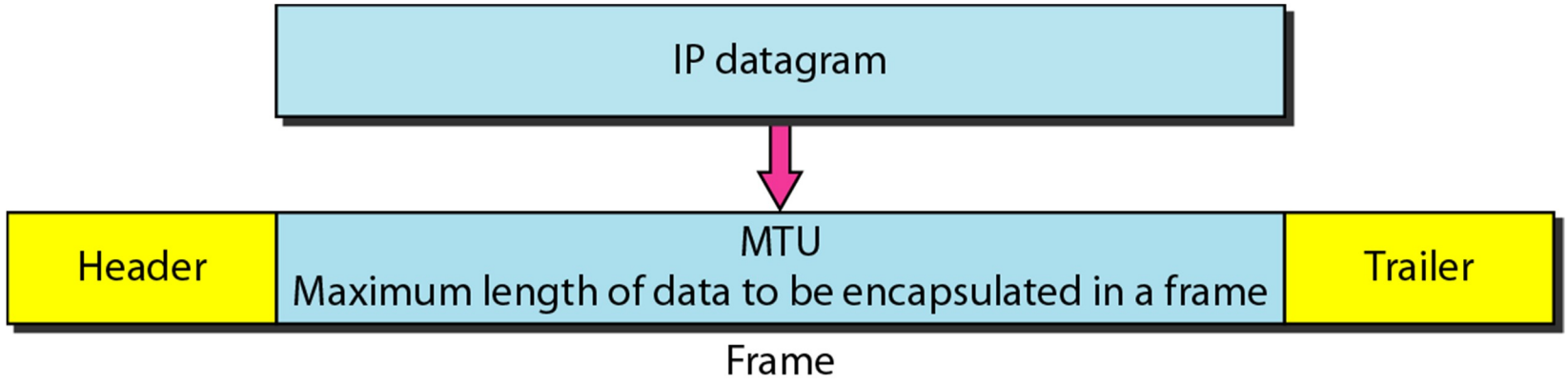
- Classification of packets
  - Queueing schemes
  - Relation to delay
- TCP vs. UDP
  - Congestion-controlled
  - Non-congestion-controlled
- Other protocols
  - RTP
  - RSVP

# Transition: IPv4 → IPv6

- Cannot happen overnight
  - Too many independent systems
  - Economic cost
  - IPv4 address space lasted longer than expected
- Coexisence needed



# Maximum datagram size



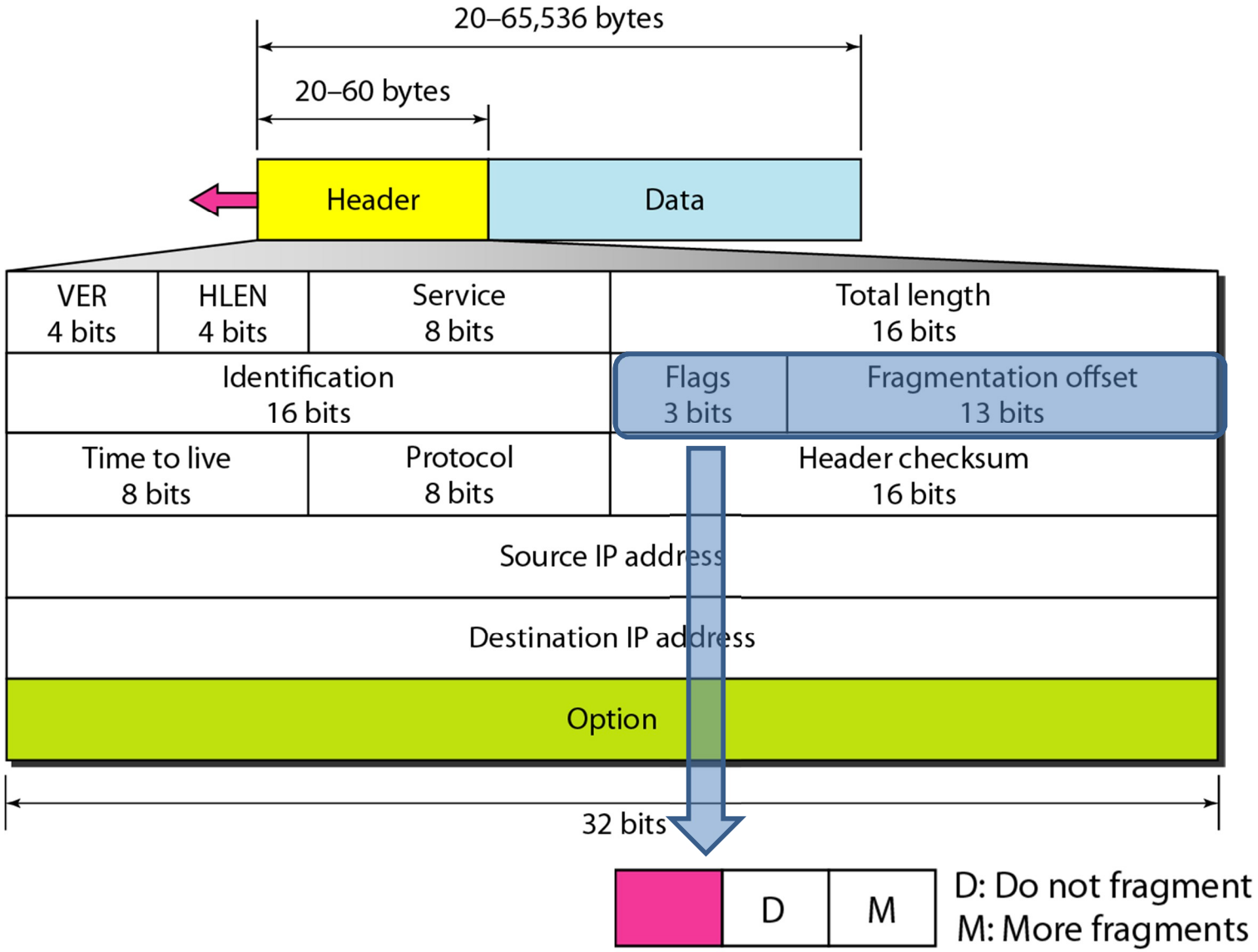

| *Protocol* | *MTU* |
|---|---|
| Hyperchannel | 65,535 |
| Token Ring (16 Mbps) | 17,914 |
| Token Ring (4 Mbps) | 4,464 |
| FDDI | 4,352 |
| Ethernet | 1,500 |
| X.25 | 576 |
| PPP | 296 |

# Fragmentation

- Needed when IP datagram size > MTU
- IPv4
  - Performed by the router meeting the problem
- IPv6
  - Performed by the source router only
- Defragmentation by destination host

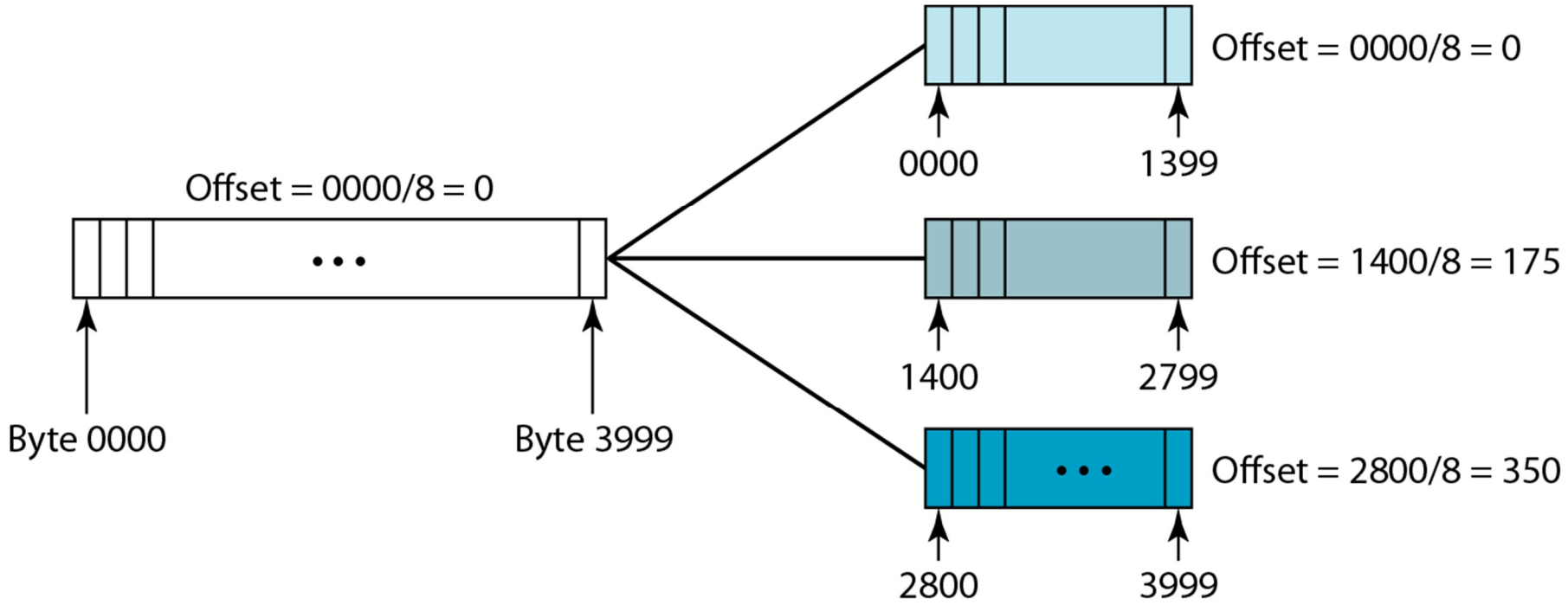

D: Do not fragment
M: More fragments

# Fragmentation field

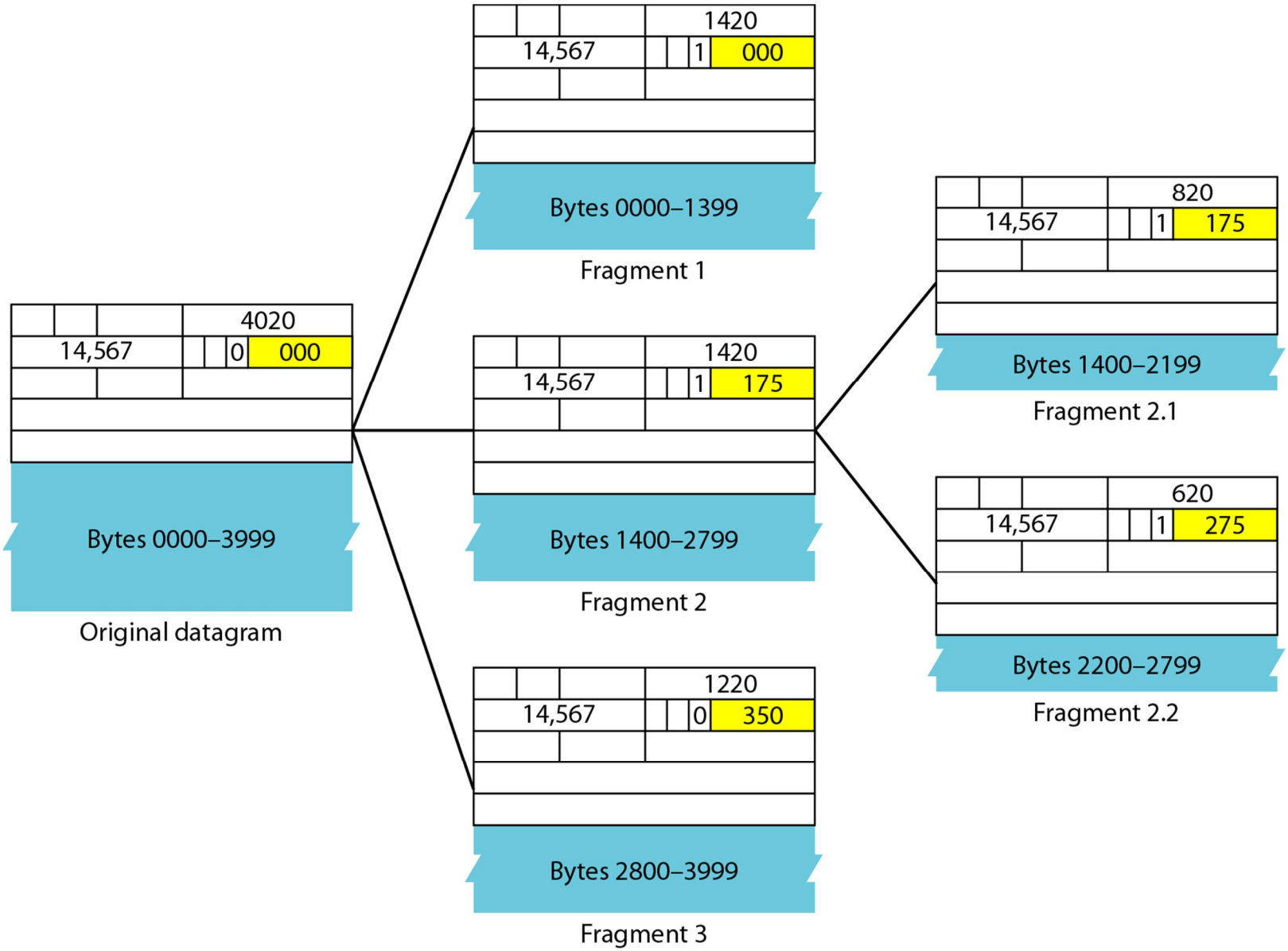


# Fragmentation offset

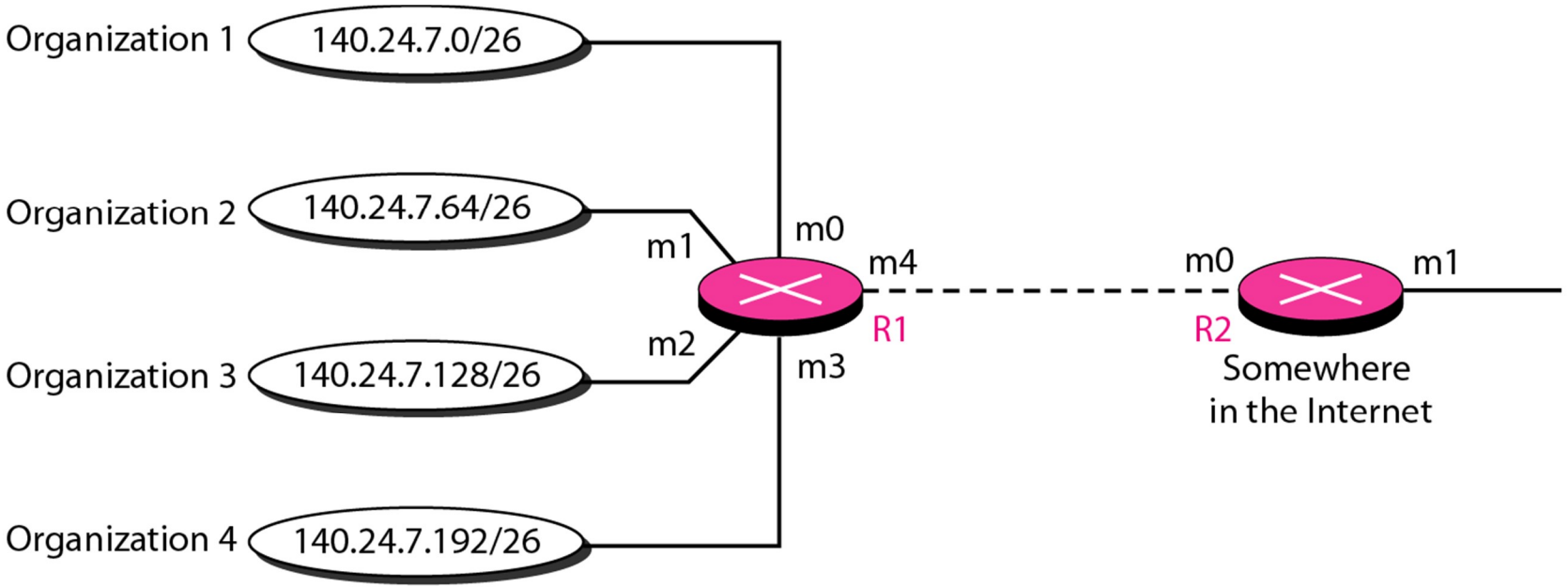
- Relative location of fragments
- 13 bits < 16 bits → /8



# Fragmentation example



# Forwarding: Address aggregation



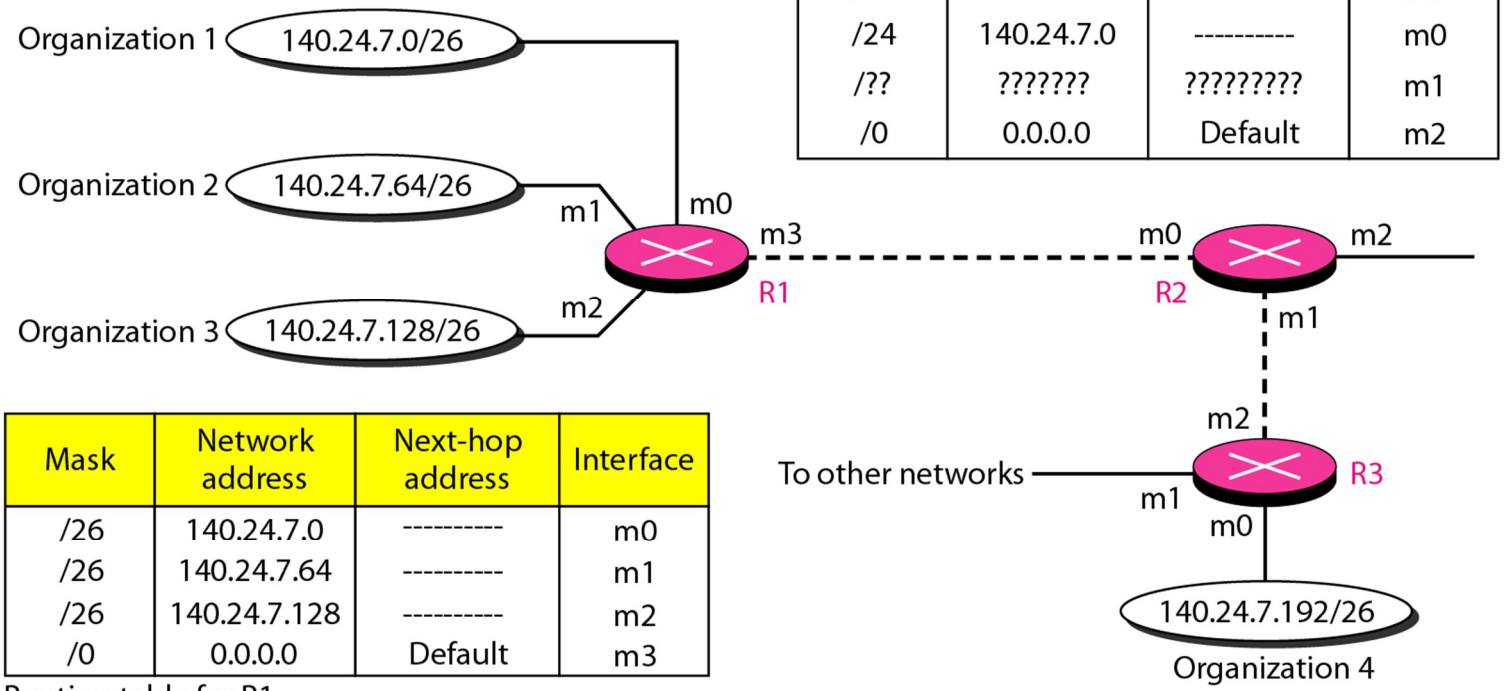

| Mask | Network address | Next-hop address | Interface |
|---|---|---|---|
| /26 | 140.24.7.0 | ---------- | m0 |
| /26 | 140.24.7.64 | ---------- | m1 |
| /26 | 140.24.7.128 | ---------- | m2 |
| /26 | 140.24.7.192 | ---------- | m3 |
| /0 | 0.0.0.0 | Default | m4 |

Routing table for R1

| Mask | Network address | Next-hop address | Interface |
|---|---|---|---|
| /24 | 140.24.7.0 | ---------- | m0 |
| /0 | 0.0.0.0 | Default | m1 |

Routing table for R2

# Forwarding: Longest mask matching




Routing table for R2

| Mask | Network address | Next-hop address | Interface |
|---|---|---|---|
| /26 | 140.24.7.192 | ---------- | m1 |
| /24 | 140.24.7.0 | ---------- | m0 |
| /?? | ??????? | ????????? | m1 |
| /0 | 0.0.0.0 | Default | m2 |

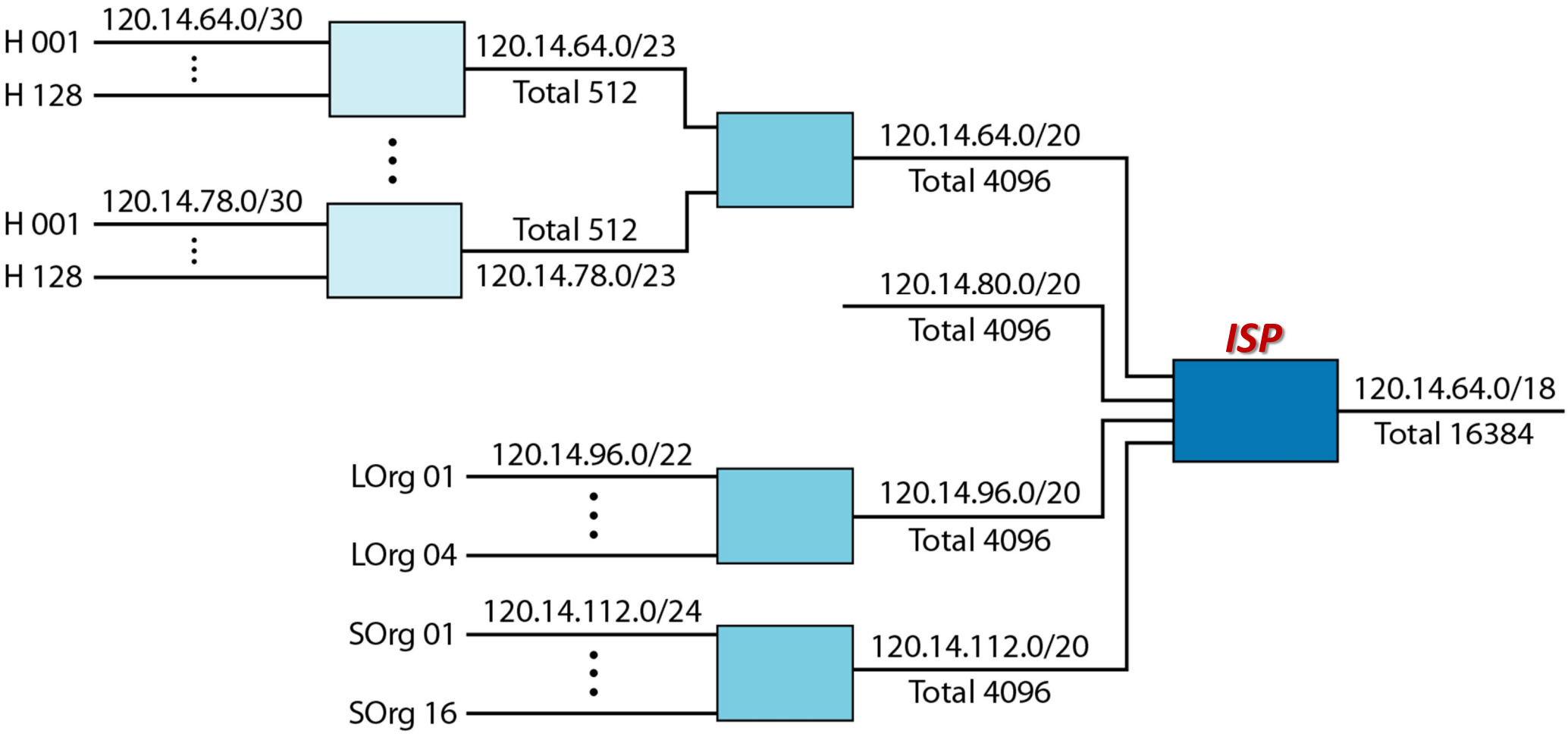
| Mask | Network address | Next-hop address | Interface |
|---|---|---|---|
| /26 | 140.24.7.0 | ---------- | m0 |
| /26 | 140.24.7.64 | ---------- | m1 |
| /26 | 140.24.7.128 | ---------- | m2 |
| /0 | 0.0.0.0 | Default | m3 |

Routing table for R1

| Mask | Network address | Next-hop address | Interface |
|---|---|---|---|
| /26 | 140.24.7.192 | ---------- | m0 |
| /?? | ??????? | ????????? | m1 |
| /0 | 0.0.0.0 | Default | m2 |

Routing table for R3

# Forwarding: Hierarchical routing



# Summary: Internet Protocols (1)

- Internet architecture, internetworking
  - End-to-end principle, routing
- Internet Protocol
  - Addressing, datagram format
  - IPv4 vs IPv6
  - Fragmentation
  - Forwarding