IT 4503
Section 5

Internet Protocols

Section 5.1

Introduction

History of Internet protocol

Some important land marks in internet

- *1966 ARPANET planning starts*
- *1969 ARPANET carries its first packets*
- *1972 Internet Assigned Numbers Authority (IANA) established*
- *1988 OSI Reference Model released*
- *1991 World Wide Web (WWW)*
- *1992 Internet Society (ISOC) established*
- *1995 IPv6 proposed*

History of Internet protocol

Some important land marks in internet Protocols

- *1971 file transfer protocall (FTP)*
- *1974 Telenet packet-switched network*
- *1976 X.25 protocol approved*
- *1982 TCP/IP protocol suite formalized*
- *1982 Simple Mail Transfer Protocol (SMTP)*
- *1983 Domain Name System (DNS)*
- *1989 Border Gateway Protocol (BGP)*
- *1991 World Wide Web (WWW)*
- *1995 IPv6 proposed*

Categorization of Internet protocol

Application Layer

BGP ,DHCP,DNS ,FTP ,HTTP ,IMAP,IRC,LDAP ,MGCP,NNTP ,NTP,POP,RIP,RPC,RTP,SIP,SMTP ,SNMP ,SOCKS,SSH ,Telnet

Transport Layer

TCP,TLS/SSL ,UDP,DCCP,SCTP,RSVP,ECN

Internet Layer

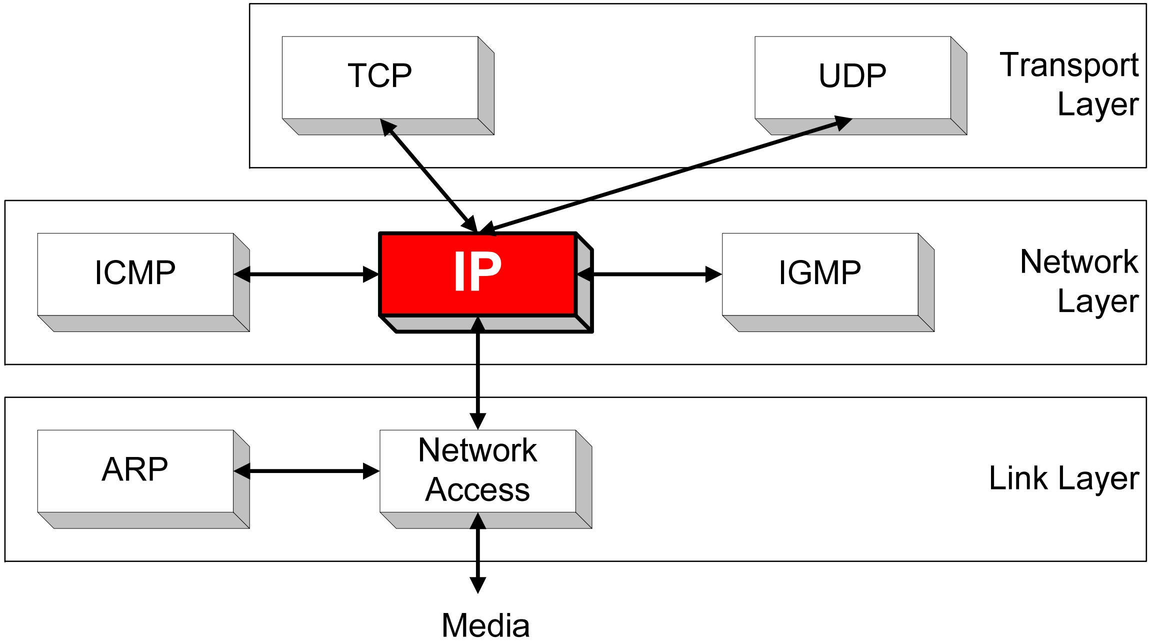
IPv4, IPv6 ,ICMP,ICMPv6,IGMP ,Ipsec

Link Layer

ARP/InARP ,NDP ,OSPF,PPP, Media Access Control (Ethernet, DSL, ISDN, FDDI)

Internet protocol stack

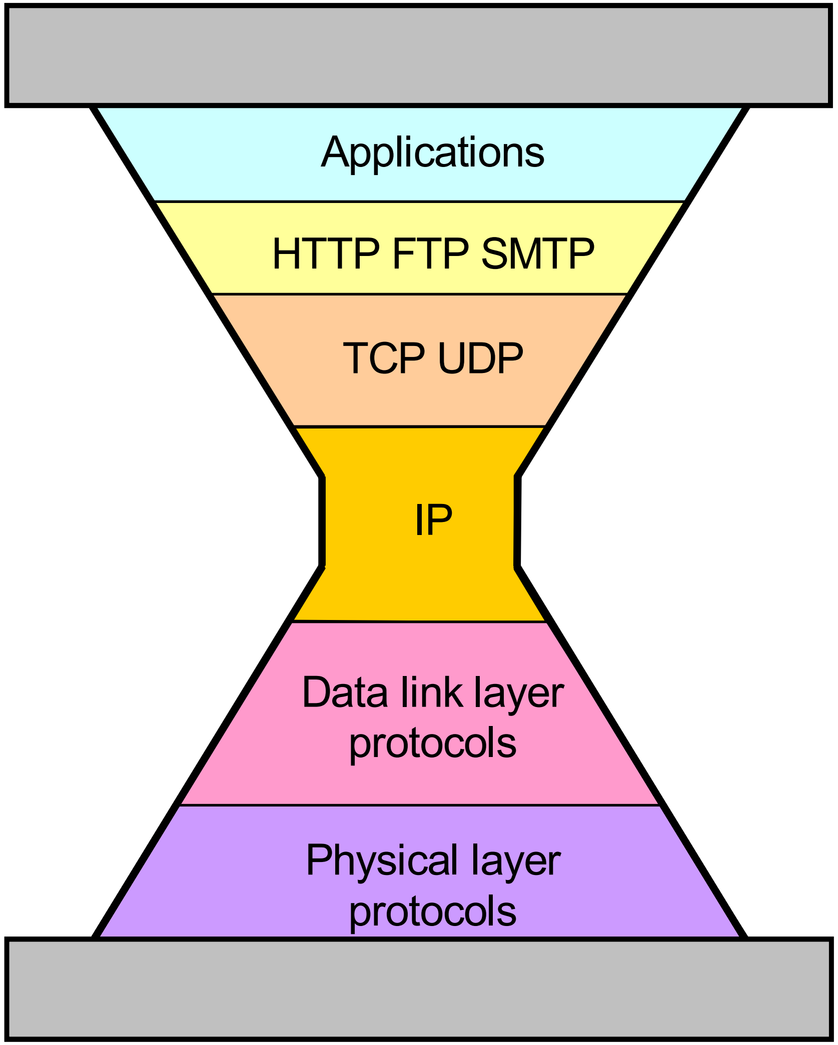
- IP (Internet Protocol) is a Network Layer Protocol.



- IP's current version is Version 4 (IPv4).
  It is specified in RFC 791.

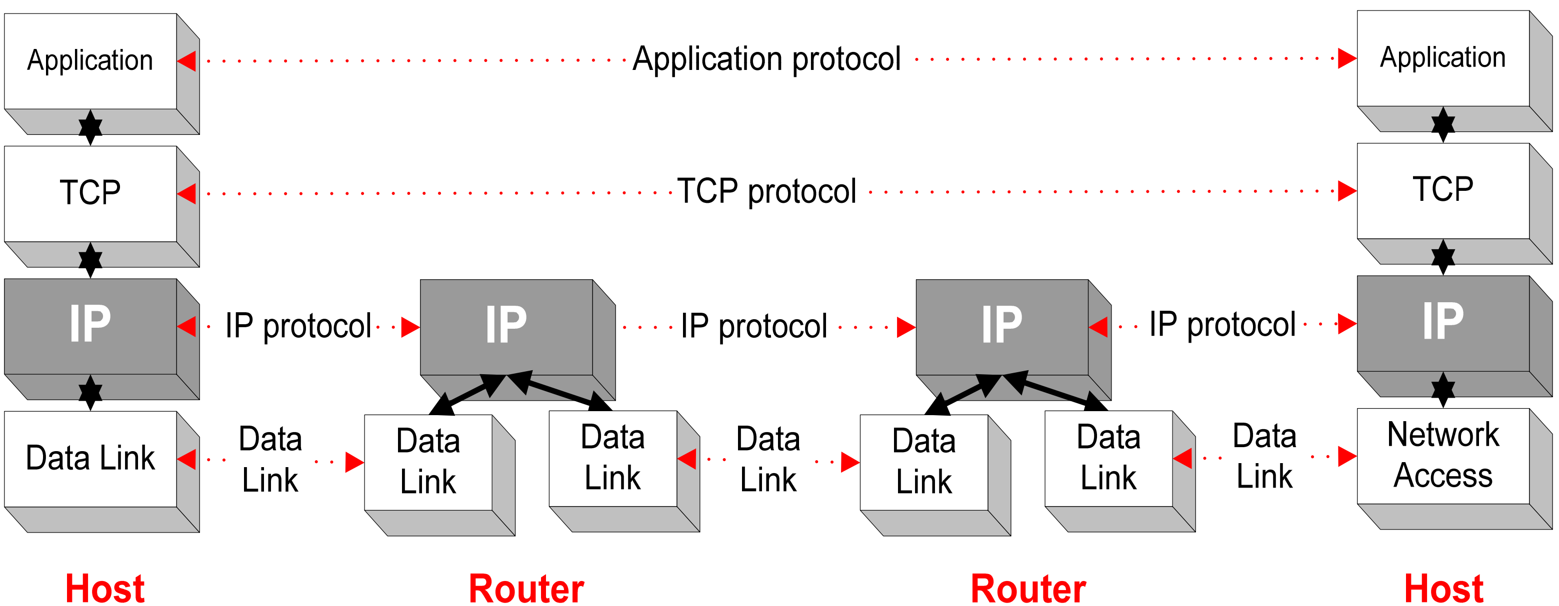
IP: The waist of the hourglass

- **IP is the waist of the hourglass of the Internet protocol architecture**
- Multiple higher-layer protocols
- Multiple lower-layer protocols
- Only one protocol at the network layer.



IP & Routers

- IP is the highest layer protocol which is implemented at both routers and hosts

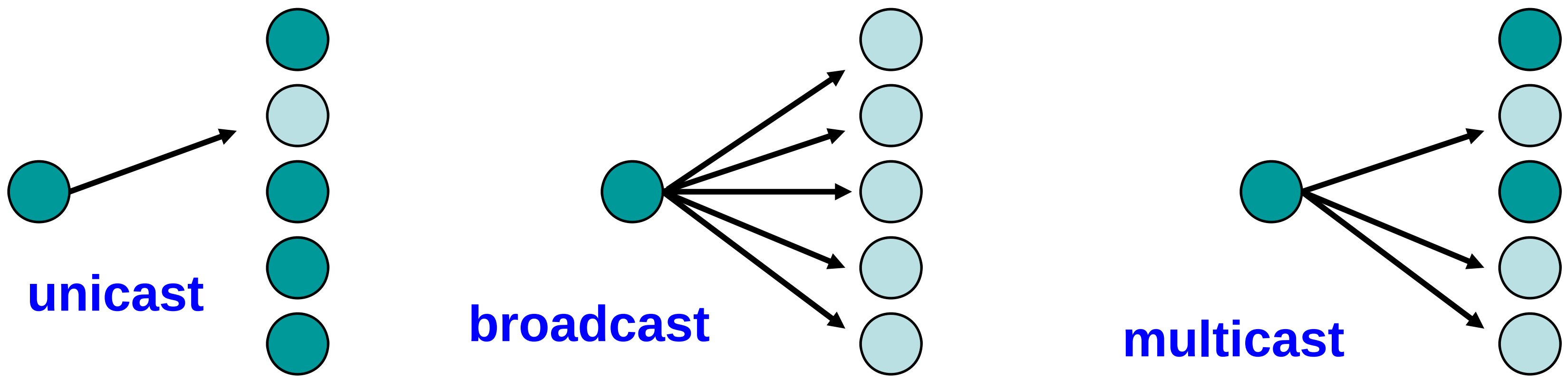


IP Service

- Delivery service of IP is minimal
- IP provides an unreliable connectionless best effort service (also called: “datagram service”).
  - **Unreliable:** IP does not make an attempt to recover lost packets
  - **Connectionless:** Each packet (“datagram”) is handled independently. IP is not aware that packets between hosts may be sent in a logical sequence
  - **Best effort:** IP does not make guarantees on the service (no throughput guarantee, no delay guarantee,…)
- Consequences:
  - Higher layer protocols have to deal with losses or with duplicate packets
  - Packets may be delivered out-of-sequence

IP Service

- IP supports the following services:
  - one-to-one (unicast)
  - one-to-all (broadcast)
  - one-to-several (multicast)

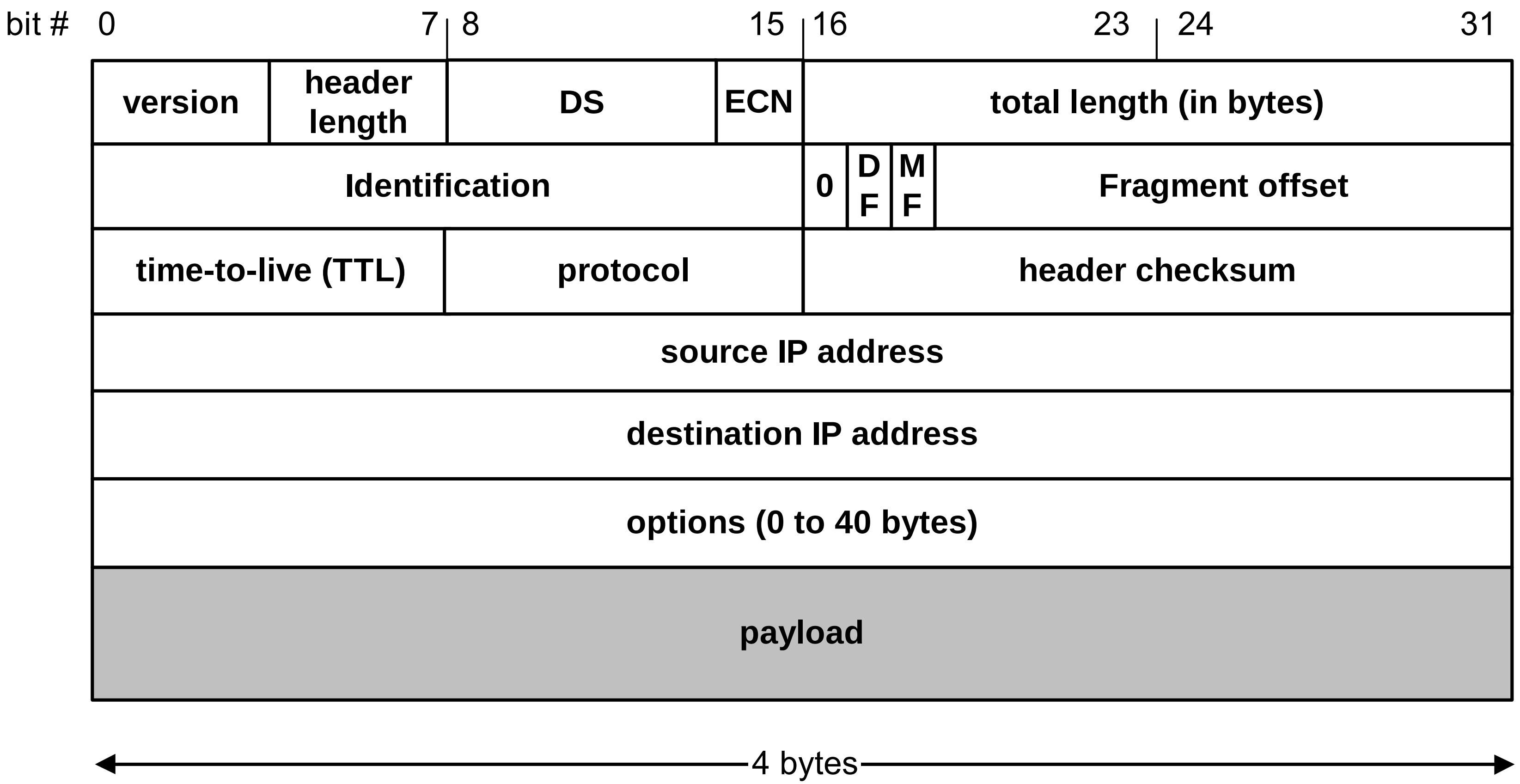


- IP multicast also supports a many-to-many service.
- IP multicast requires support of other protocols (IGMP, multicast routing)

UCSC

BIT

IP Datagram Format



- 20 bytes ≤ Header Size < 2^4 x 4 bytes = 60 bytes
- 20 bytes ≤ Total Length < 2^{16} bytes = 65536 bytes

Fields of the IP Header

- **Version (4 bits)**: current version is 4, next version will be 6.
- **Header length (4 bits)**: length of IP header, in multiples of 4 bytes
- **DS/ECN field (1 byte)**
  - This field was previously called as Type-of-Service (TOS) field. The role of this field has been re-defined, but is "backwards compatible" to TOS interpretation
  - Differentiated Service (DS) (6 bits):
    - Used to specify service level (currently not supported in the Internet)
  - Explicit Congestion Notification (ECN) (2 bits):
    - New feedback mechanism used by TCP

Fields of the IP Header

- **Identification (16 bits):**

  Unique identification of a datagram from a host.
  Incremented whenever a datagram is transmitted

- **Flags (3 bits):**
  - First bit always set to 0
  - DF bit (Do not fragment)
  - MF bit (More fragments)

  Will be explained later→ Fragmentation

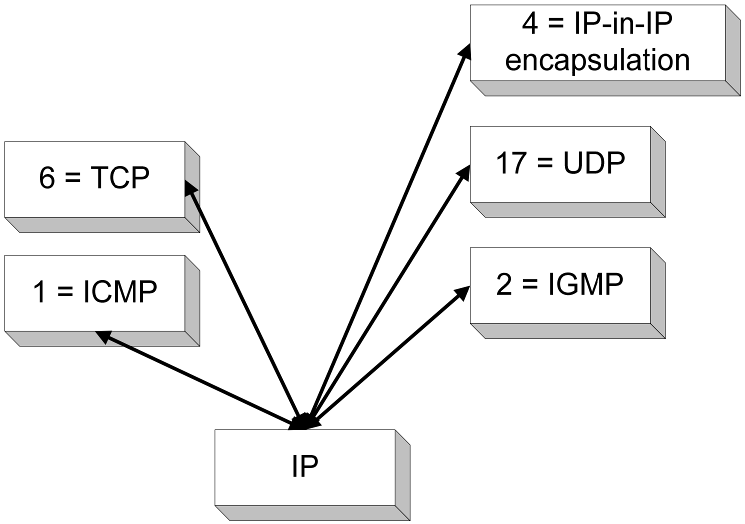
Fields of the IP Header

- **Time To Live (TTL) (1 byte):**
  - Specifies longest paths before datagram is dropped
  - Role of TTL field: Ensure that packet is eventually dropped when a routing loop occurs

  Used as follows:
  - Sender sets the value (e.g., 64)
  - Each router decrements the value by 1
  - When the value reaches 0, the datagram is dropped

Fields of the IP Header

- **Protocol (1 byte):**
  - Specifies the higher-layer protocol.
  - Used for demultiplexing to higher layers.



- **Header checksum (2 bytes):**

  A simple 16-bit long checksum which is computed for the header of the datagram.

Fields of the IP Header

- **Options:**
  - Security restrictions
  - Record Route: each router that processes the packet adds its IP address to the header.
  - Timestamp: each router that processes the packet adds its IP address and time to the header.
  - (loose) Source Routing: specifies a list of routers that must be traversed.
  - (strict) Source Routing: specifies a list of the only routers that can be traversed.
- **Padding:**

  Padding bytes are added to ensure that header ends on a 4-byte boundary

Maximum Transmission Unit

- Maximum size of IP datagram is 65535, but the data link layer protocol generally imposes a limit that is much smaller

  Example:

  - Ethernet frames have a maximum payload of 1500 bytes

    → IP datagrams encapsulated in Ethernet frame cannot be longer than 1500 bytes

- The limit on the maximum IP datagram size, imposed by the data link protocol is called **maximum transmission unit (MTU)**

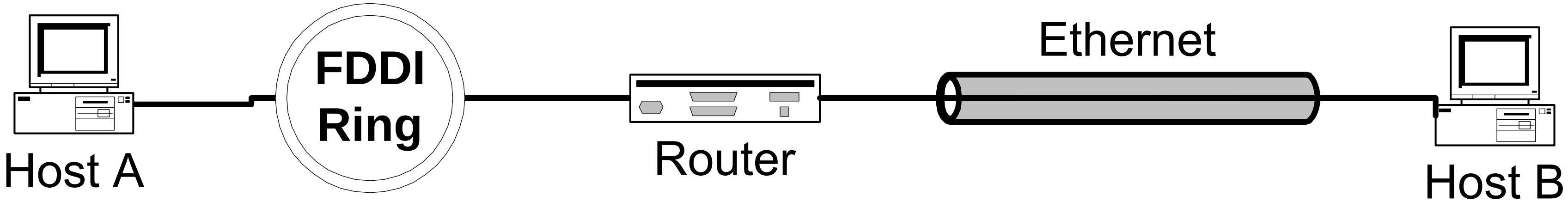
- MTUs for various data link protocols:

Ethernet:	1500	FDDI:	4352
802.3:	1492	ATM AAL5:	9180
802.5:	4464	PPP:	negotiated

IP Fragmentation

- What if the size of an IP datagram exceeds the MTU?
  IP datagram is fragmented into smaller units.

- What if the route contains networks with different MTUs?



MTUs: FDDI: 4352 Ethernet: 1500

- **Fragmentation**:
  - IP router splits the datagram into several datagram
  - Fragments are reassembled at receiver

IP Addresses

- Structure of an IP address
- Classful IP addresses
- Limitations and problems with classful IP addresses
- Subnetting
- CIDR

What is an IP Address?

- An IP address is a unique global address for a network interface
- Exceptions:
  - Dynamically assigned IP addresses
  - IP addresses in private networks
- An IP address:
  - is a **32 bit long** identifier
  - encodes a network number (**network prefix**) and a **host number**

Network prefix and host number

- The network prefix identifies a network and the host number identifies a specific host (actually, interface on the network).

network prefix	host number

- **How do we know how long the network prefix is?**
  - **Before 1993:** The network prefix is implicitly defined (**class-based addressing**)

  or

  - **After 1993:** The network prefix is indicated by a **netmask. (classless inter domain routing)**

Dotted Decimal Notation

- IP addresses are written in a so-called ***dotted decimal* notation**
- Each byte is identified by a decimal number in the range [0..255]
- **Example:**

10000000	10001111	10001001	10010000
1st Byte	2nd Byte	3rd Byte	4th Byte
= 128	**= 143**	**= 137**	**= 144**

128.143.137.144

Example

- **Example**: www.cmb.ac.lk

192.248.16	89

- Network address is: **192.248.16.0 (or 192.248.16)**
- Host number is: **89**
- Netmask is: **255.255.255.0** (or **ffffff00)**
- Prefix or CIDR notation: **192.248.16.89/24**
  - » Network prefix is 24 bits long

Special IP Addresses

- Reserved or (by convention) special addresses:

Loopback interfaces

- all addresses 127.0.0.1-127.255.255.255 are reserved for loopback interfaces
- Most systems use 127.0.0.1 as loopback address
- loopback interface is associated with name "localhost"

IP address of a network

- Host number is set to all zeros

Broadcast address

- Host number is all ones
- Broadcast goes to all hosts on the network
- Often ignored due to security concerns

Special IP Addresses (Cont.)

- **Test / Experimental addresses (Private IPs)**

  Certain address ranges are reserved for "experimental use". Packets should get dropped if they contain this destination address (see RFC 1918):

10.0.0.0	-	10.255.255.255
172.16.0.0	-	172.31.255.255
192.168.0.0	-	192.168.255.255

- **Convention (but not a reserved address)**

  Default gateway has host number set to 'first' or 'last' number: 192.168.100.**1** or 192.168.100.**254**

CIDR - Classless Interdomain Routing

- **Goals:**
  - New interpretation of the IP address space
  - Restructure IP address assignments to increase efficiency
  - Permits route aggregation to minimize route table entries
- CIDR (Classless Interdomain routing)
  - abandons the notion of classes
  - **Key Concept:** The length of the network prefix in the IP addresses is kept arbitrary
  - Consequence: Size of the network prefix must be provided with an IP address

CIDR Notation

- CIDR notation of an IP address:

  192.0.2.0/18

  - "18" is the prefix length. It states that the first 18 bits are the network prefix of the address (and 14 bits are available for specific host addresses)

- CIDR notation can replace the use of subnetmasks (but is more general)

  - IP address 128.143.137.144 and subnetmask 255.255.255.0 becomes 128.143.137.144/24

- CIDR notation allows to drop traling zeros of network addresses:

  192.0.2.0/18 can be written as **192.0.2/18**

CIDR address blocks

- CIDR notation can nicely express blocks of addresses
- Blocks are used when allocating IP addresses for a company and for routing tables (route aggregation)

CIDR Block Prefix	# of Host Addresses
/27	32
/26	64
/25	128
/24	256
/23	512
/22	1,024
/21	2,048
/20	4,096
/19	8,192
/18	16,384
/17	32,768
/16	65,536
/15	131,072
/14	262,144
/13	524,288

CIDR and Address assignments

- Backbone ISPs obtain large block of IP addresses space and then reallocate portions of their address blocks to their customers.

Example:

- Assume that an ISP owns the address block 206.0.64.0/18, which represents 16,384 (2^{14}) IP addresses
- Suppose a client requires 800 host addresses
- With classful addresses: need to assign a class B address (and waste ~64,700 addresses) or four individual Class Cs (and introducing 4 new routes into the global Internet routing tables)
- With CIDR: Assign a /22 block, e.g., 206.0.68.0/22, and allocated a block of 1,024 (2^{10}) IP addresses.

UCSC

BIT

CIDR and Routing

- **Aggregation** of routing table entries:
  - 128.143.0.0/16 and 128.144.0.0/16 are represented as 128.142.0.0/15

- **Longest prefix match**: Routing table lookup finds the routing entry that matches the longest prefix

What is the outgoing interface for 128.143.137.0/24 ?

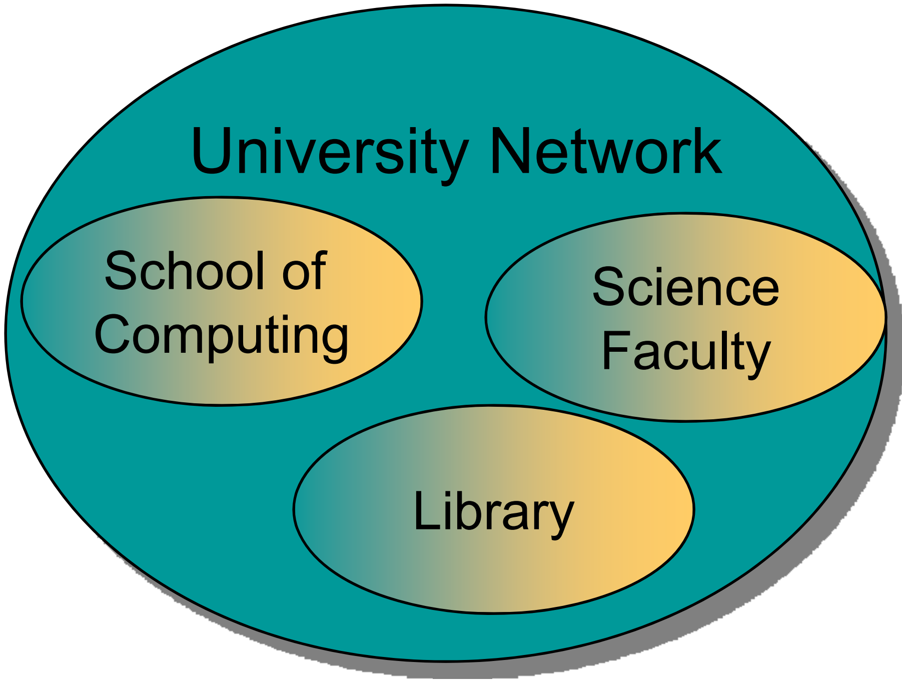
Route aggregation can be exploited when IP address blocks are assigned in an hierarchical fashion

Prefix	Interface
128.0.0.0/4	interface #5
128.128.0.0/9	interface #2
128.143.128.0/17	interface #1

Routing table

Subnetting

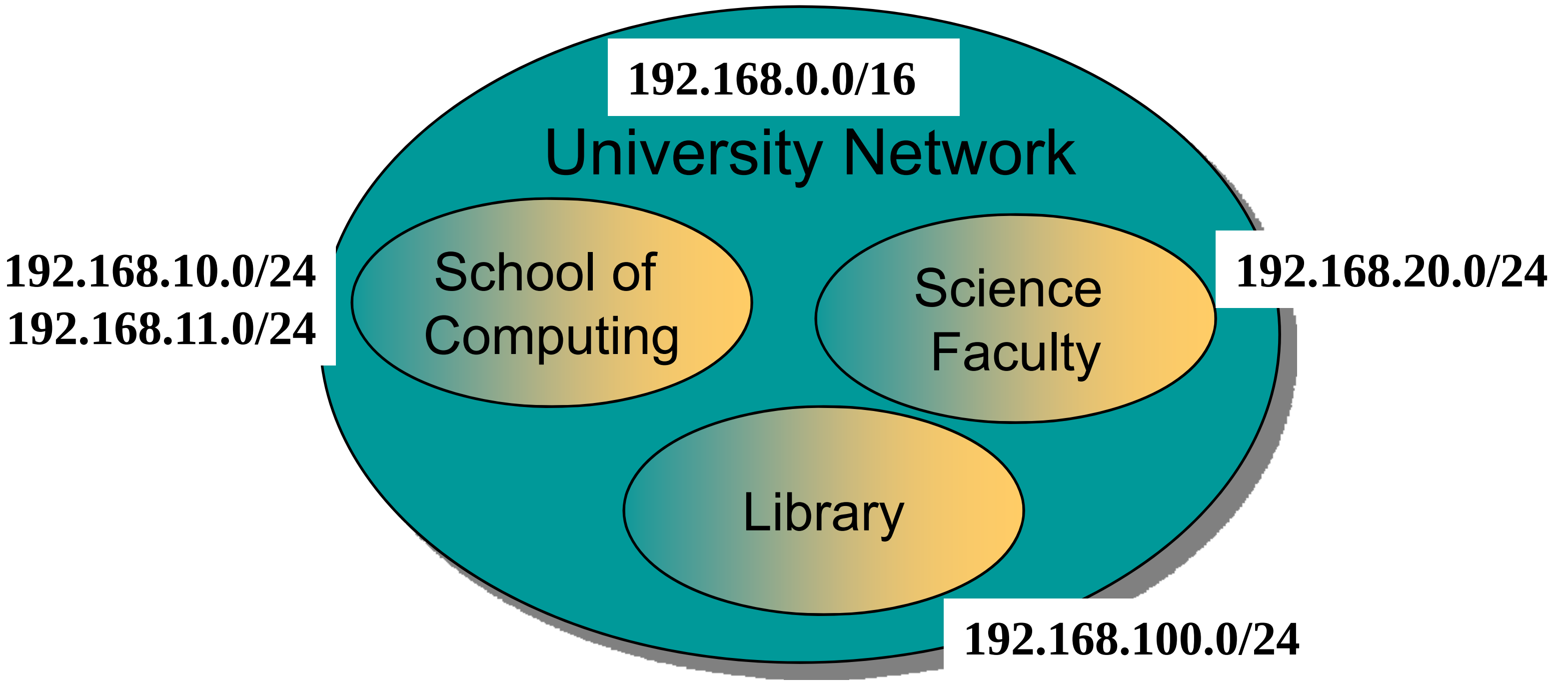
- **Problem**: Organizations have multiple networks which are independently managed
  - **Solution 1:** Allocate a separate network address for each network
    - Difficult to manage
    - From the outside of the organization, each network must be addressable.
  - **Solution 2:** Add another level of hierarchy to the IP addressing structure



→ **Subnetting**

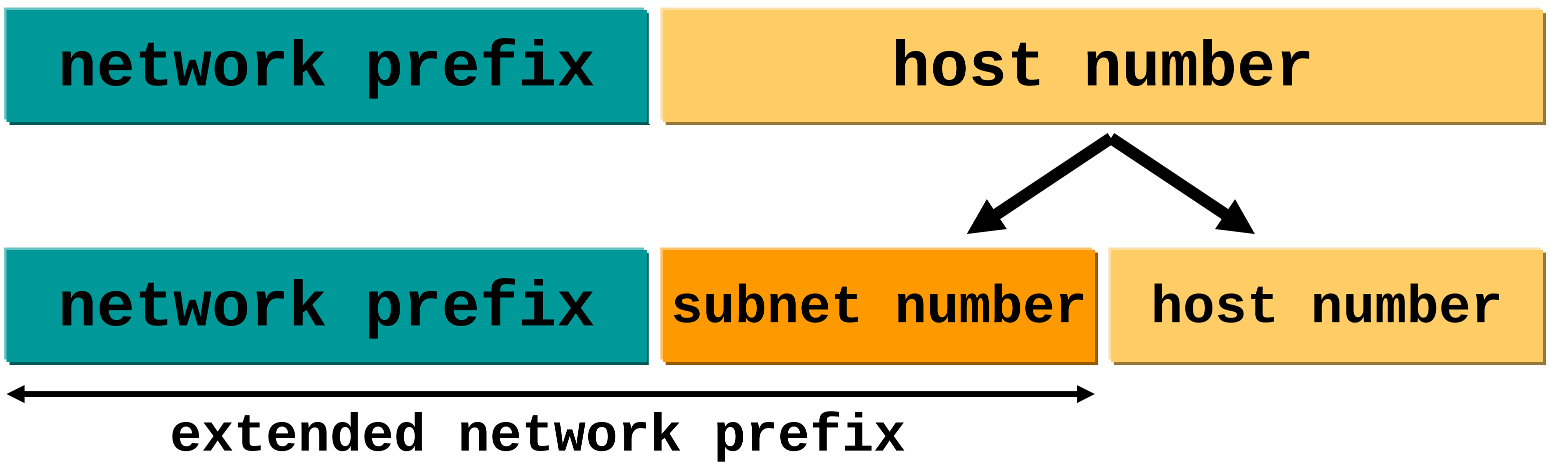
Address assignment with subnetting

- Each part of the organization is allocated a range of IP addresses (subnets or subnetworks)
- Addresses in each subnet can be administered locally



Basic Idea of Subnetting

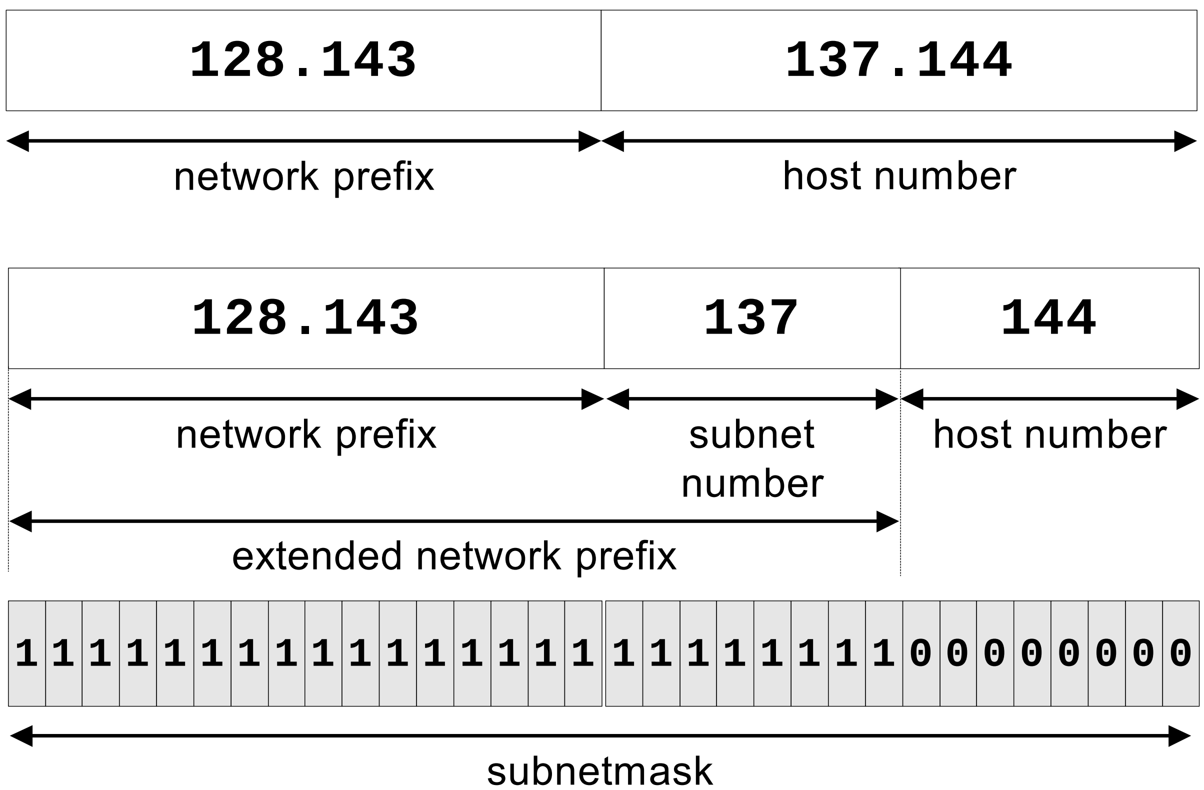
- Split the host number portion of an IP address into a **subnet number** and a (smaller) **host number**.
- Result is a 3-layer hierarchy



- Then:
  - Subnets can be freely assigned within the organization
  - Internally, subnets are treated as separate networks
  - Subnet structure is not visible outside the organization

Subnetmask

- Routers and hosts use an **extended network prefix (subnetmask)** to identify the start of the host numbers



Advantages of Subnetting

- With subnetting, IP addresses use a 3-layer hierarchy:
  - » Network
  - » Subnet
  - » Host
- Reduces router complexity. Since external routers do not know about subnetting, the complexity of routing tables at external routers is reduced.
- Note: Length of the subnet mask need not be identical at all subnetworks.

Variable Length SM (VLSM)

is the process by which we take a major network address and use different subnet masks at different points.

A fixed length mask has the advantage of simplicity. It will be easy for the network staff/users to remember the subnet mask. However, if we have to keep the subnet mask the same we encounter severe problems concerning addressing space.

Some useful tips on VLSM:

- Use as few different masks as possible
- Keep lookup table to figure out the masks for a given subnet
- Make sure not to overlap subnets with VLSM

When do we need to use different subnet masks?

UCSC

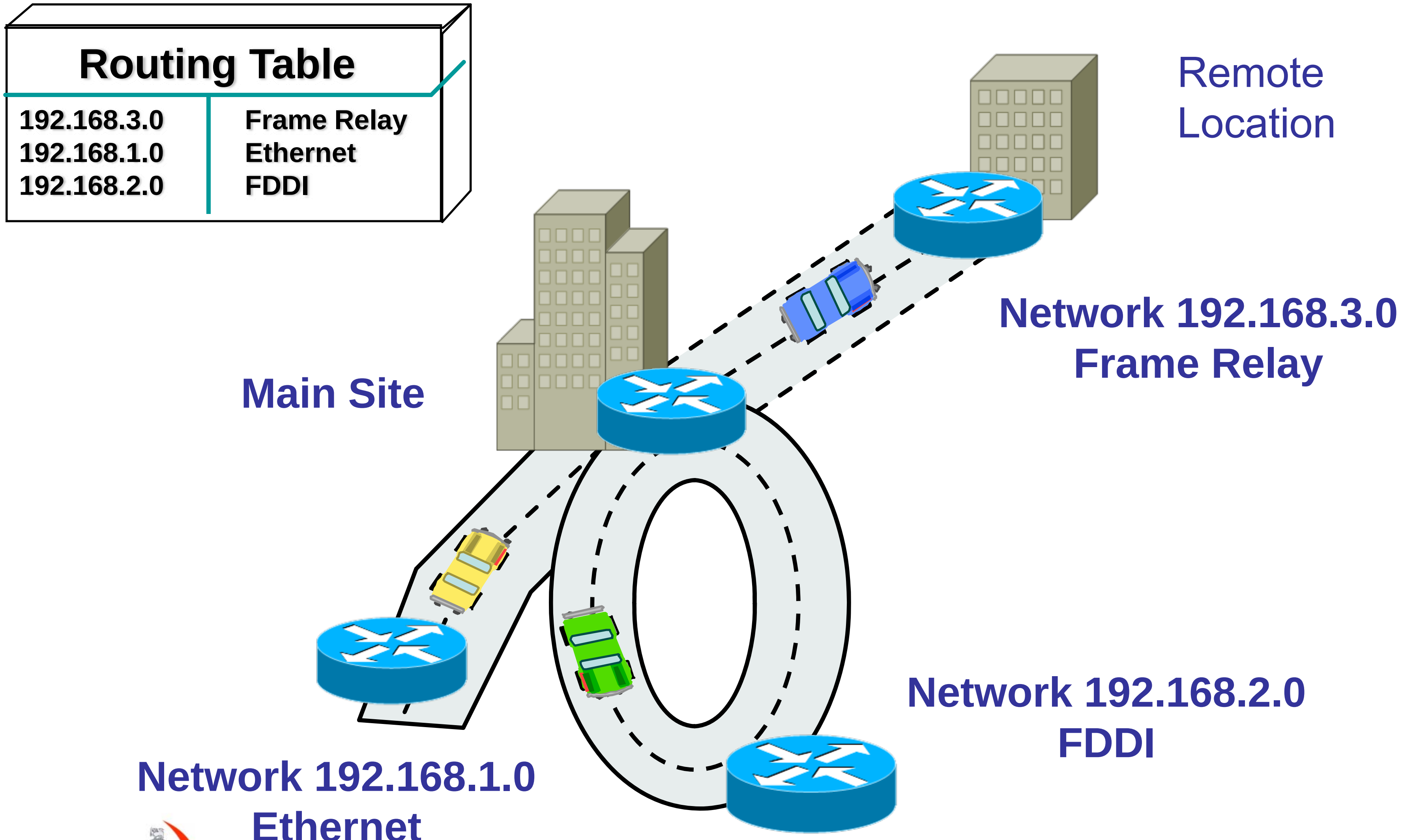
BIT

What is Routing?

- Routing is:
  - Finding a path between a source and destination (path determination)
  - Moving information across an internetwork from a source to a destination (switching)
  - Very complex in large networks because of the many potential intermediate nodes

- Router is:
  - A network layer device that forwards packets from one network to another and determines the optimal path for forwarding network traffic

Router – Layer 3 Device



Routing Algorithms

- Routing algorithms
  - Initialize and maintain routing tables to help with path determination
- Routing algorithms can be grouped in to 2 major classes:
  - ***Non-adaptive Algorithms*** – do not base their routing decisions on measurements or estimates of the current traffic and topology.

    → **Static Routing Algorithms**

  - ***Adaptive Algorithms*** – change their routing decisions to reflect changes in the traffic and the topology

    → **Dynamic Routing Algorithms**

Route information types

- **Destination/next-hop associations**
- **Path desirability**
- **Vary depending on routing algorithm**

UCSC

BIT

Routing Algorithm Goals

- ***Correctness***
- ***Simplicity and low overhead* – efficient routing algorithm functionality with a minimum of software and utilization overhead**
- ***Robustness and stability* – correct performance in the face of unusual or unforeseen circumstances (e.g., high load), reaches equilibrium and stays there**
- ***Rapid convergence* – fast agreement, by all routers, on optimal routes**
- ***Flexibility* – quick and accurate adaptation to changes in router availability, bandwidth, queue size, etc.**
- ***Fairness***
- ***Optimality* – selecting the best route based on metrics and metric weightings used in the calculation**
  - **Minimizing mean packet delay**
  - **Maximizing total network throughput**

Routing Metrics

- Path length
  - Total hop count or sum of cost per network link
- Reliability
  - Dependability (bit error rate) of each network link
- Delay
  - Useful because it depends on bandwidth, queues, network congestion, and physical distance
- Communication cost
  - Operating expenses of links (private versus public)
- Bandwidth and load

UCSC

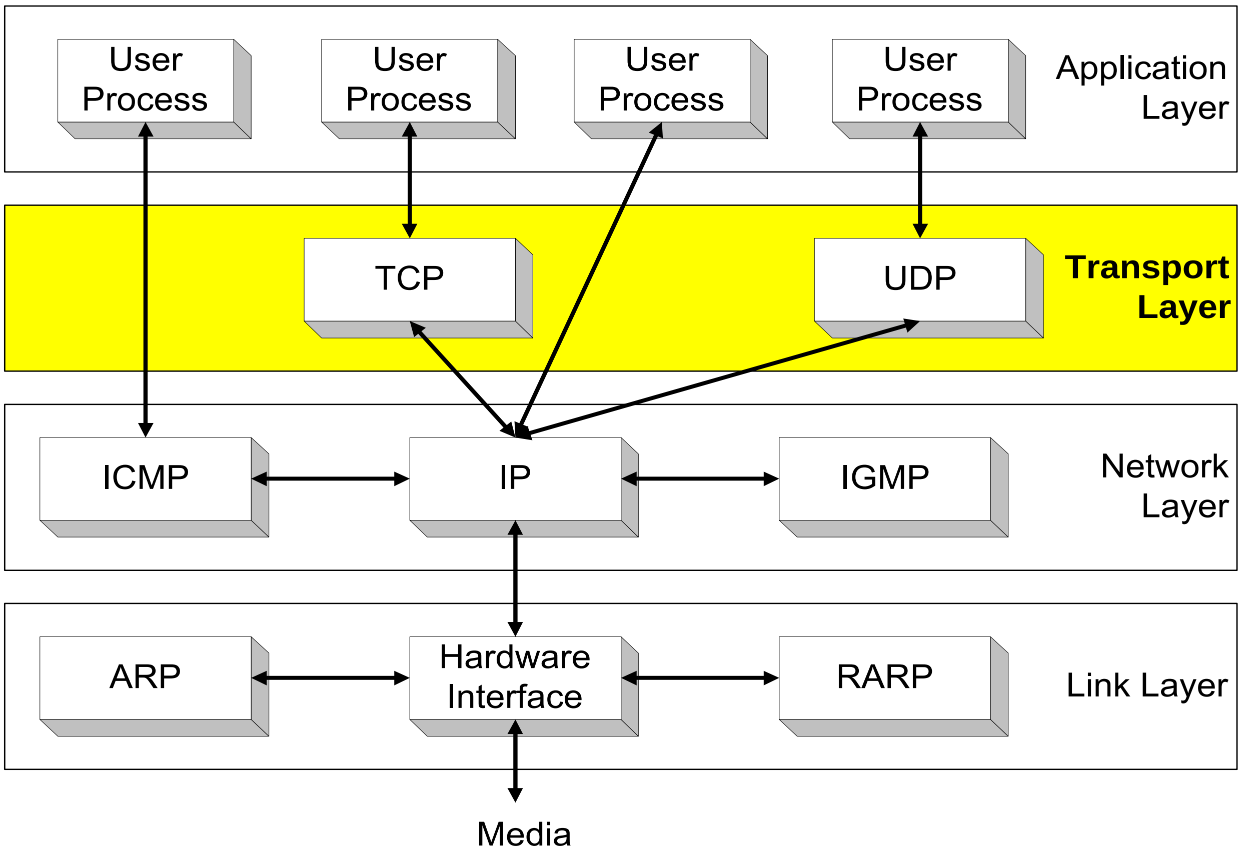
BIT

Section 5.2

Transport Layer protocols

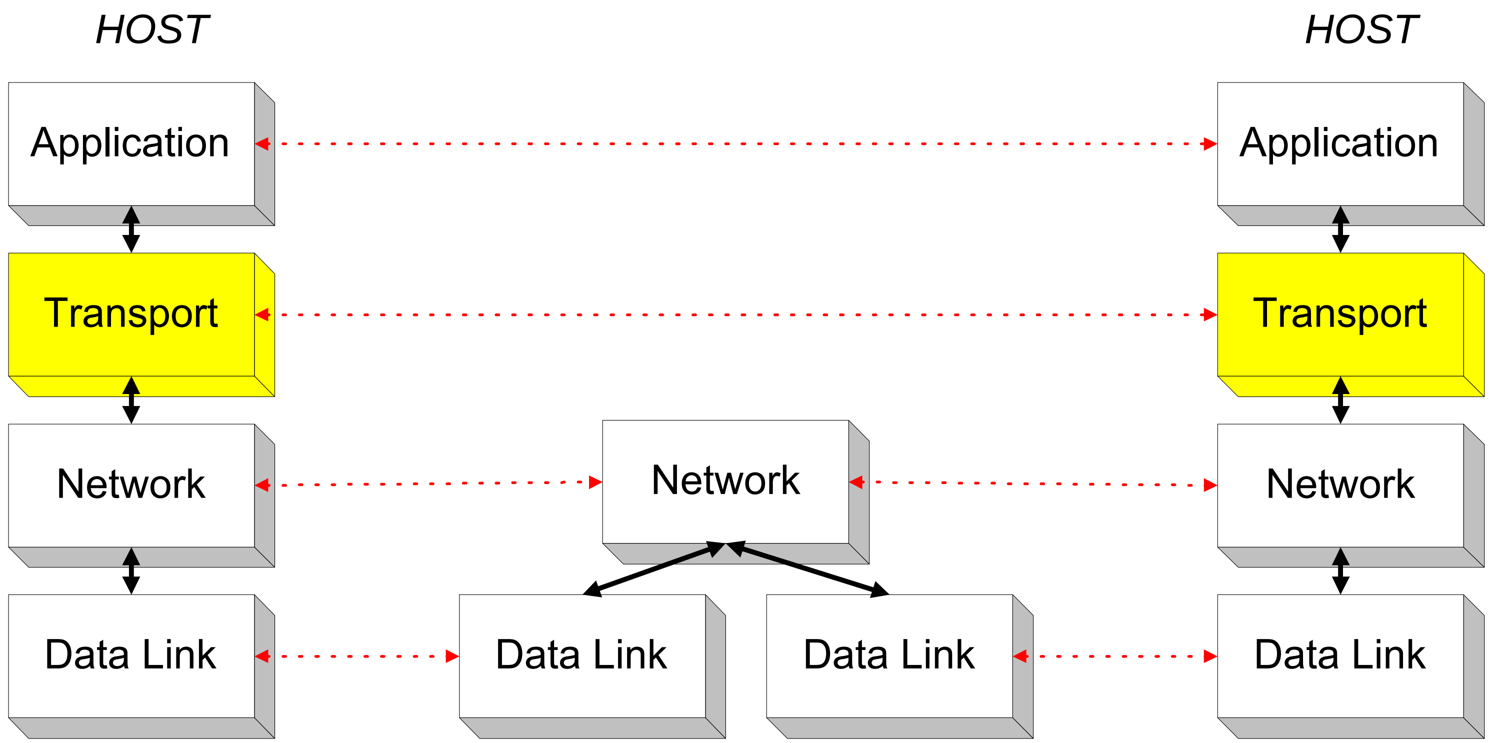
Orientation

- We move one layer up and look at the transport layer.



Orientation

- Transport layer protocols are end-to-end protocols
- They are only implemented at the hosts



Transport Protocols in the Internet

- The Internet supports 2 transport protocols

UDP - User Datagram Protocol

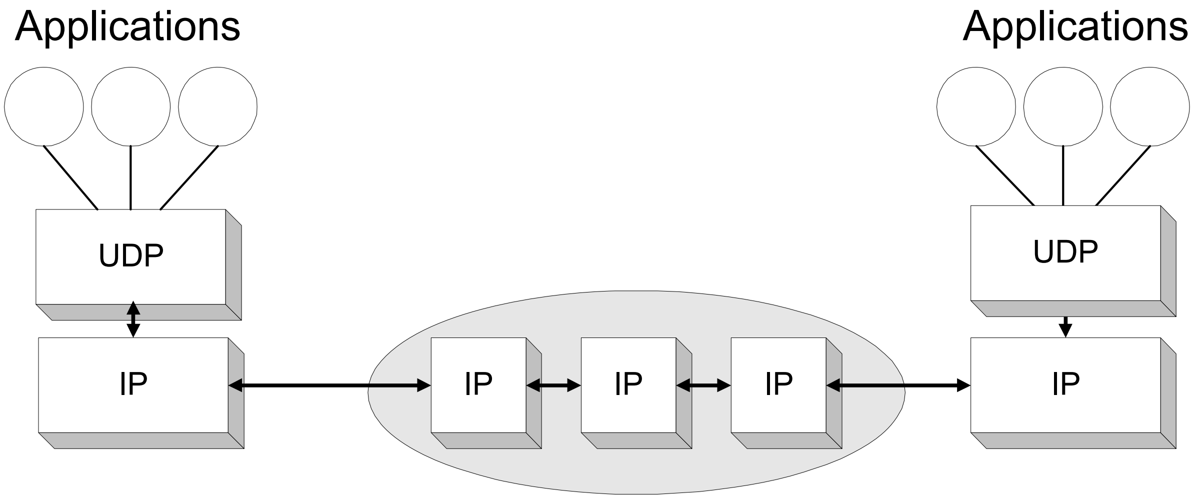
- datagram oriented
- unreliable, connectionless
- simple
- unicast and multicast
- useful only for few applications, e.g., multimedia applications
- used a lot for services
  - network management (SNMP), routing (RIP), naming (DNS), etc.

TCP - Transmission Control Protocol

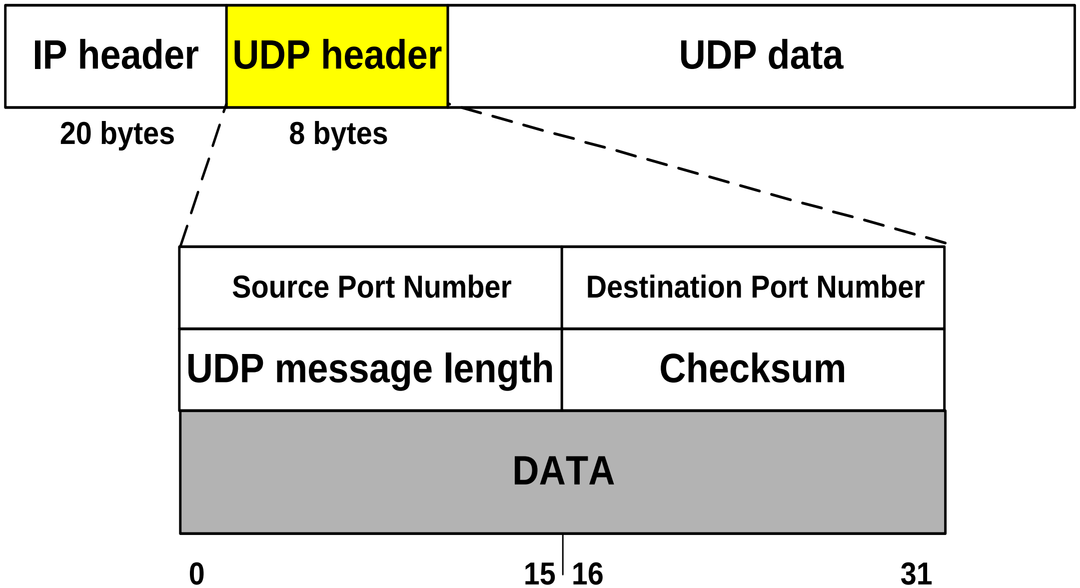
- stream oriented
- reliable, connection-oriented
- complex
- only unicast
- used for most Internet applications:
  - web (http), email (smtp), file transfer (ftp), terminal (telnet), etc.

UDP - User Datagram Protocol

- UDP supports unreliable transmissions of datagrams
- UDP merely extends the host-to-to-host delivery service of IP datagram to an application-to-application service
- The only thing that UDP adds is multiplexing and demultiplexing



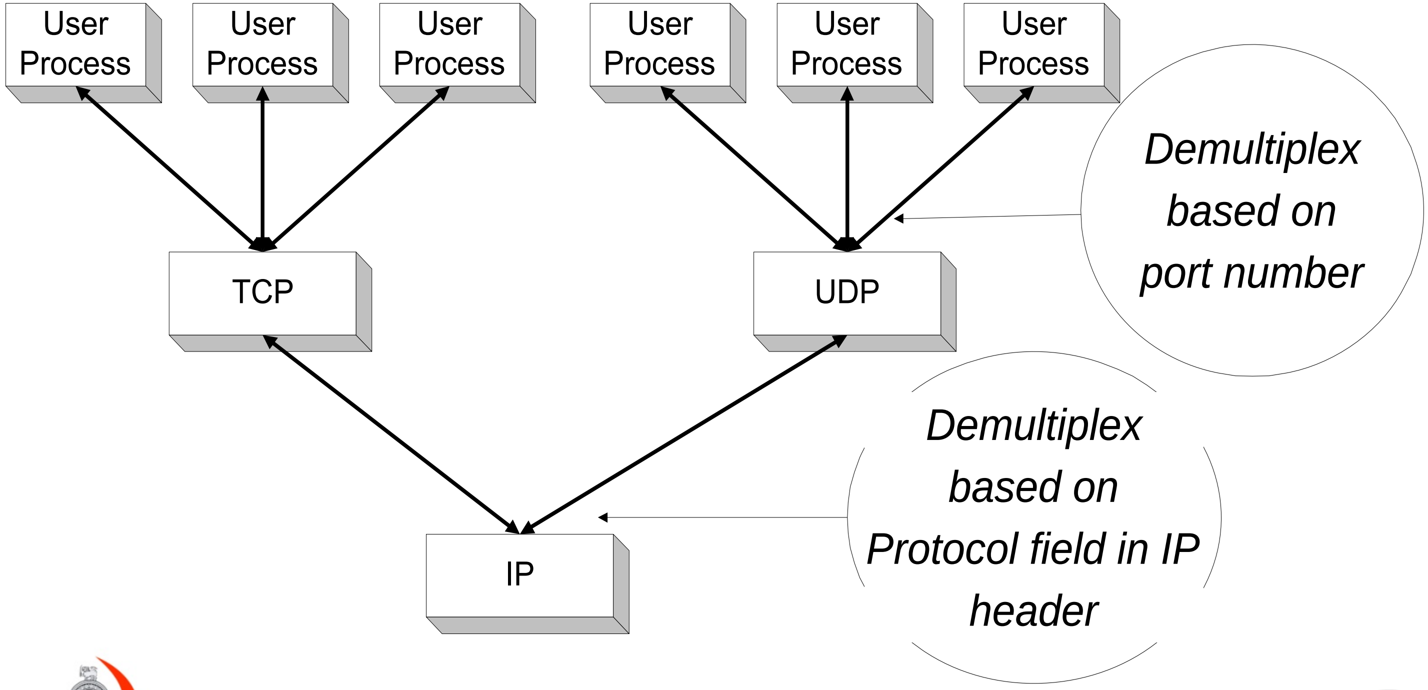
UDP Format



- ❑ ***Port numbers*** **identify sending and receiving applications (processes). Maximum port number is 2^{16}-1= 65,535**
- ❑ ***Message Length*** **is at least 8 bytes (I.e., Data field can be empty) and at most 65,535**
- ❑ ***Checksum*** **is for header (of UDP and some of the IP header fields)**

Port Numbers

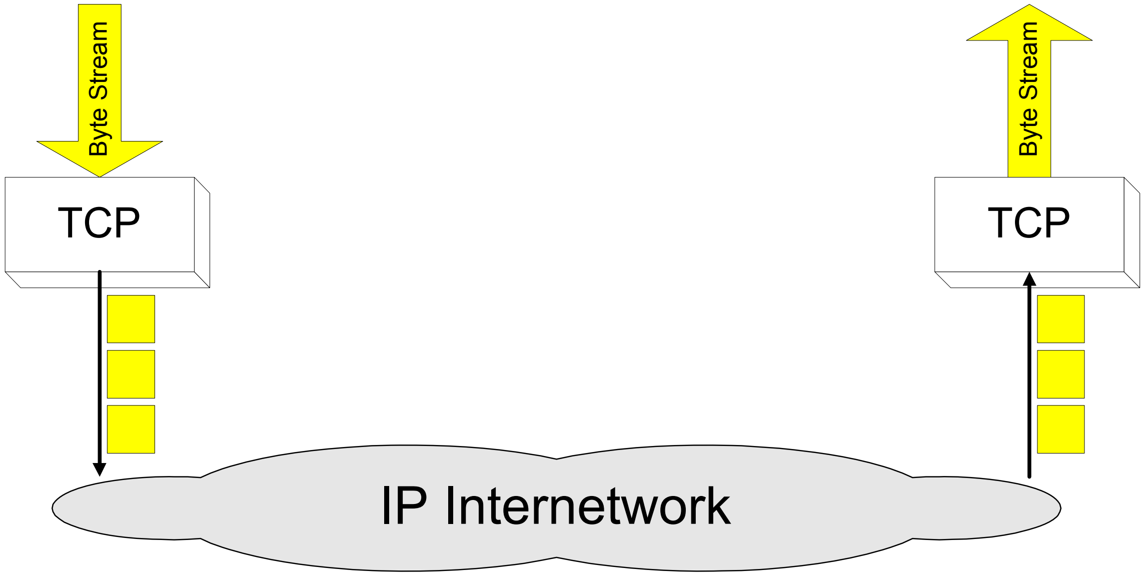
- UDP (and TCP) use port numbers to identify applications
- A globally unique address at the transport layer (for both UDP and TCP) is a tuple **<IP address, port number>**
- There are 65,535 UDP ports per host.



Overview

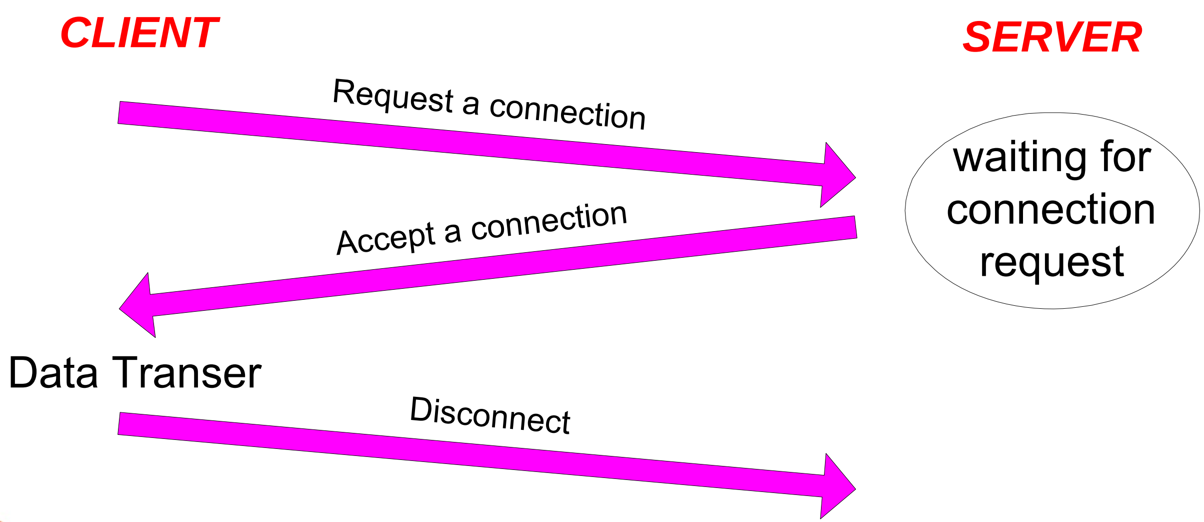
TCP = Transmission Control Protocol

- Connection-oriented protocol
- Provides a reliable unicast end-to-end byte stream over an unreliable internetwork.



Connection-Oriented

- Before any data transfer, TCP establishes a **connection**:
  - One TCP entity is waiting for a connection ("**server**")
  - The other TCP entity ("**client**") contacts the server
- The actual procedure for setting up connections is more complex.
- Each connection is full duplex

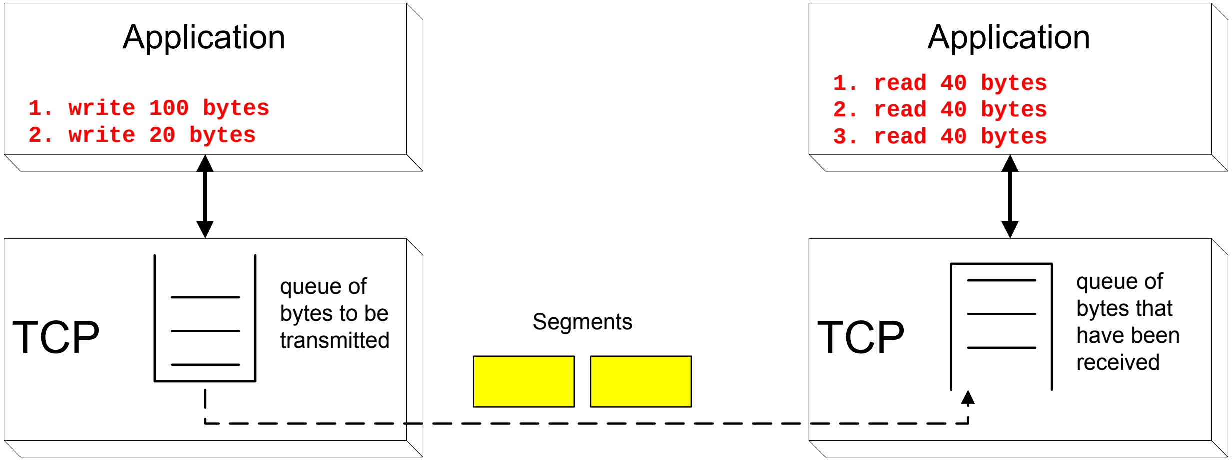


Reliable

- Byte stream is broken up into chunks which are called **segments**
  - Receiver sends acknowledgements (ACKs) for segments
  - TCP maintains a timer. If an ACK is not received in time, the segment is retransmitted
- **Detecting errors:**
  - TCP has checksums for header and data. Segments with invalid checksums are discarded
  - Each byte that is transmitted has a sequence number

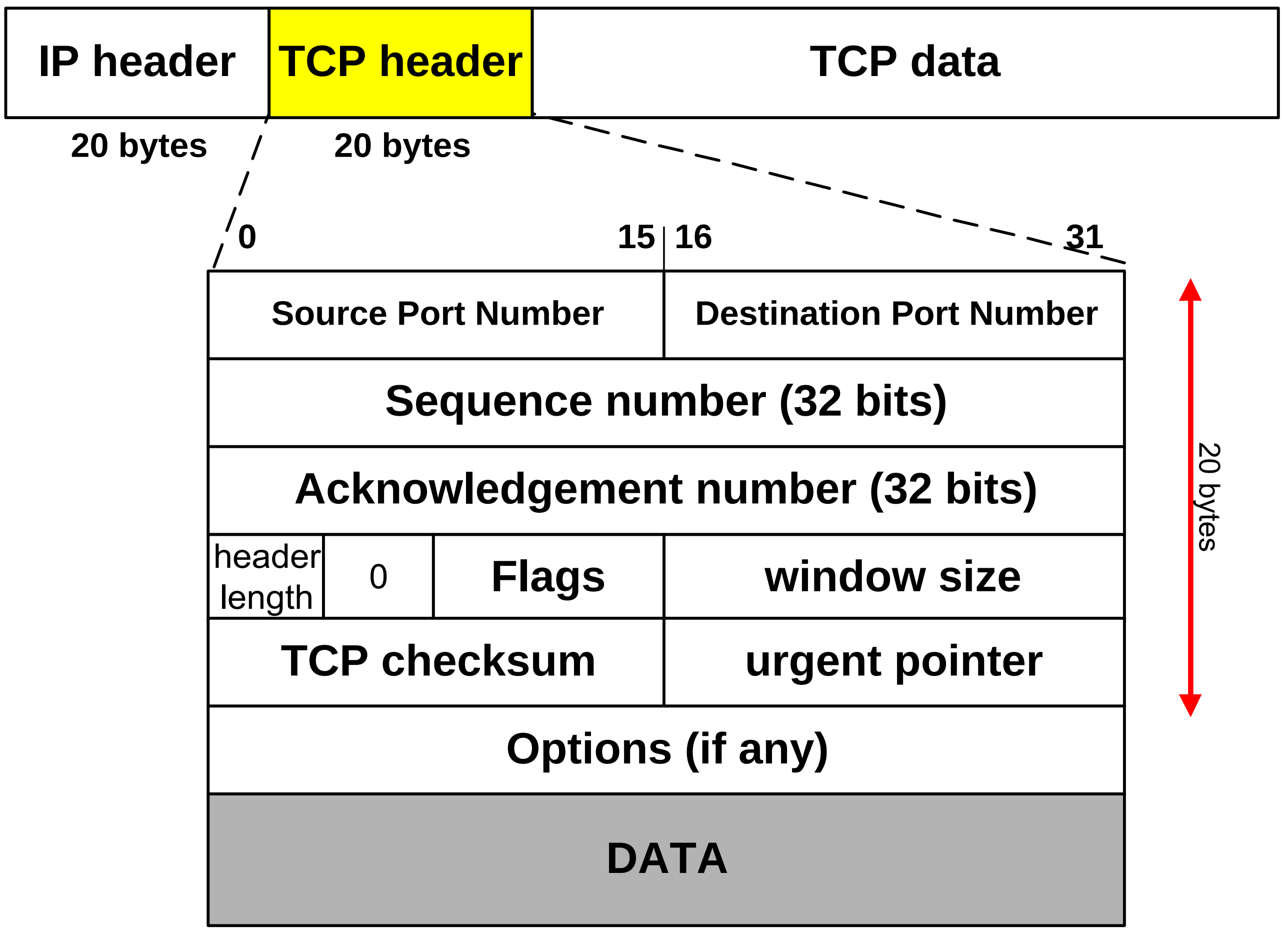
Byte Stream Service

- To the lower layers, TCP handles data in blocks, the segments.
- To the higher layers TCP handles data as a sequence of bytes and does not identify boundaries between bytes
- So: Higher layers do not know about the beginning and end of segments !



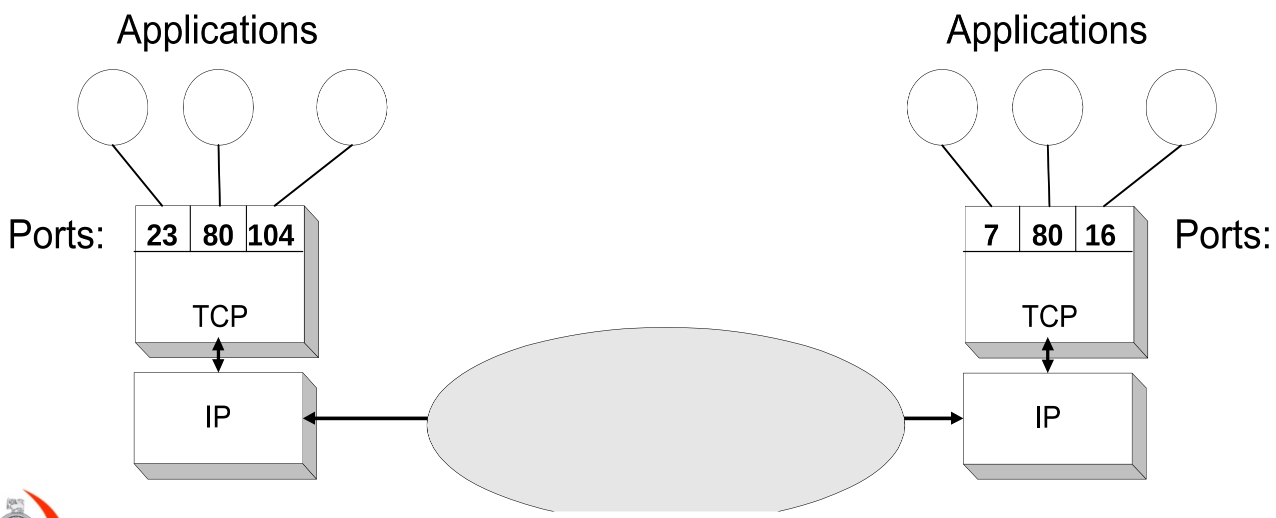
TCP Format

- TCP segments have a 20 byte header with >= 0 bytes of data.



TCP header fields

- **Port Number:**
  - A port number identifies the endpoint of a connection.
  - A pair **`<IP address, port number>`** identifies one endpoint of a connection.
  - Two pairs **`<client IP address, server port number>`** and **`<server IP address, server port number>`** identify a TCP connection.



TCP header fields

- **Sequence Number (SeqNo):**
  - Sequence number is 32 bits long.
  - So the range of SeqNo is
    - o $0 <= SeqNo <= 2^{32} - 1 \approx 4.3$ Gbyte
  - Each sequence number identifies a byte in the byte stream
  - Initial Sequence Number (ISN) of a connection is set during connection establishment

TCP header fields

❑ Acknowledgement Number (AckNo):

- Acknowledgements are piggybacked, I.e
  - o a segment from A -> B can contain an acknowledgement for a data sent in the B -> A direction
- A hosts uses the AckNo field to send acknowledgements. (If a host sends an AckNo in a segment it sets the "**ACK flag**")
- The AckNo contains the next SeqNo that a hosts wants to receive

  Example: The acknowledgement for a segment with sequence numbers 0-1500 is AckNo=1501

UCSC

BIT

TCP header fields

- **Acknowledge Number (cont'd)**
  - TCP uses the sliding window flow protocol to regulate the flow of traffic from sender to receiver
  - TCP uses the following variation of sliding window:
    - o no NACKs (**N**egative **ACK**nowledgement)
    - o only cumulative ACKs

Example:

Assume: Sender sends two segments with "1..1500" and "1501..3000", but receiver only gets the second segment.

In this case, the receiver cannot acknowledge the second packet. It can only send AckNo=1

UCSC

BIT

TCP header fields

- **Header Length (4bits):**
  - Length of header in 32-bit words
  - Note that TCP header has variable length (with minimum 20 bytes)

TCP header fields

- **Flag bits:**
  - **URG: Urgent pointer is valid**
    - If the bit is set, the following bytes contain an urgent message in the range:

      SeqNo <= urgent message <= SeqNo+urgent pointer
  - **ACK: Acknowledgement Number is valid**
  - **PSH: PUSH Flag**
    - Notification from sender to the receiver that the receiver should pass all data that it has to the application.
    - Normally set by sender when the sender’s buffer is empty

TCP header fields

- **Flag bits:**
  - **RST: Reset the connection**
    - The flag causes the receiver to reset the connection
    - Receiver of a RST terminates the connection and indicates higher layer application about the reset
  - **SYN: Synchronize sequence numbers**
    - Sent in the first packet when initiating a connection
  - **FIN: Sender is finished with sending**
    - Used for closing a connection
    - Both sides of a connection must send a **FIN**

TCP header fields

- **Window Size:**
  - Each side of the connection advertises the window size
  - Window size is the maximum number of bytes that a receiver can accept.
  - Maximum window size is 2^{16}-1= 65535 bytes
- **TCP Checksum:**
  - TCP checksum covers over both TCP header **and** TCP data (also covers some parts of the IP header)
- **Urgent Pointer:**
  - Only valid if **URG** flag is set

TCP Connection Establishment

- TCP uses a **three-way handshake** to open a connection:

  (1) ACTIVE OPEN: Client sends a segment with

  - SYN bit set
  - port number of client
  - initial sequence number (ISN) of client

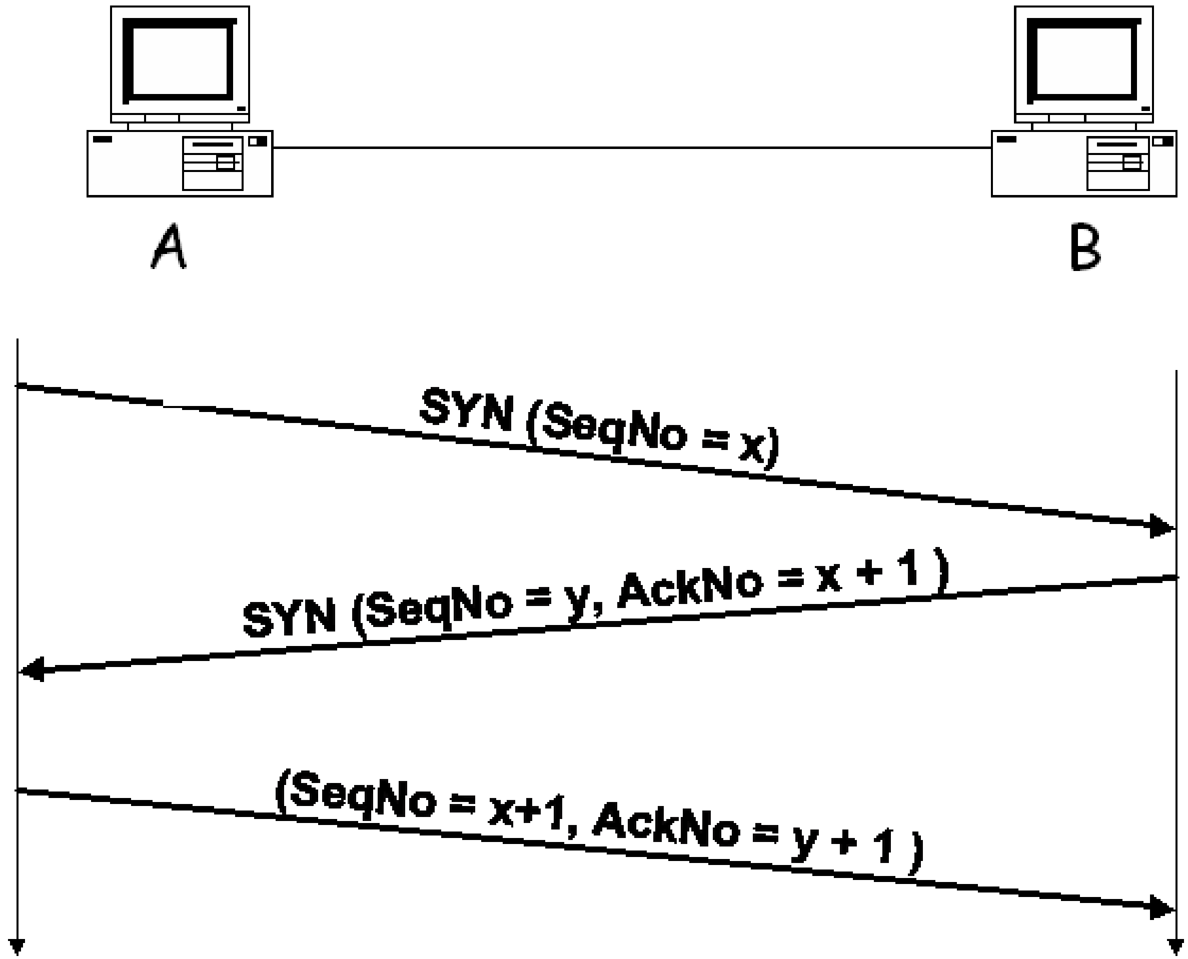
  (2) PASSIVE OPEN: Server responds with a segment with

  - SYN bit set
  - initial sequence number of server
  - ACK for ISN of client

  (3) Client acknowledges by sending a segment with:

  - ACK ISN of server

Three-Way Handshake



TCP Connection Termination

- Each end of the data flow must be shut down independently **(“half-close”)**
- If one end is done it sends a FIN segment. This means that no more data will be sent
- Four steps involved:

  (1) X sends a FIN to Y **(active close)**

  (2) Y ACKs the FIN,

  (at this time: Y can still send data to X)

  (3) and Y sends a FIN to X **(passive close)**

  (4) X ACKs the FIN.

What is Flow/Congestion/Error Control ?

- **Flow Control:** Algorithms to prevent that the sender overruns the receiver with information
- **Error Control:** Algorithms to recover or conceal the effects from packet losses
- **Congestion Control:** Algorithms to prevent that the sender overloads the network

→ The goal of each of the control mechanisms are different.

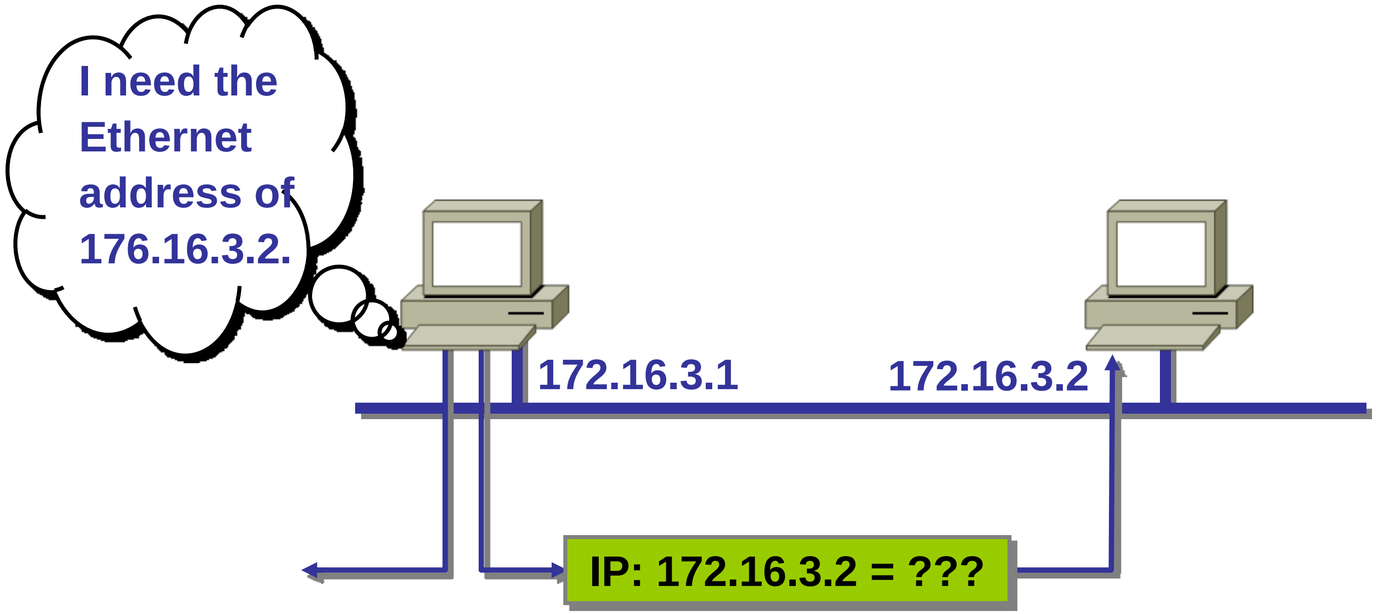
→ In TCP, the implementation of these algorithms is combined

Section 5.3

IP Support Protocols

Address Resolution Protocol (ARP)

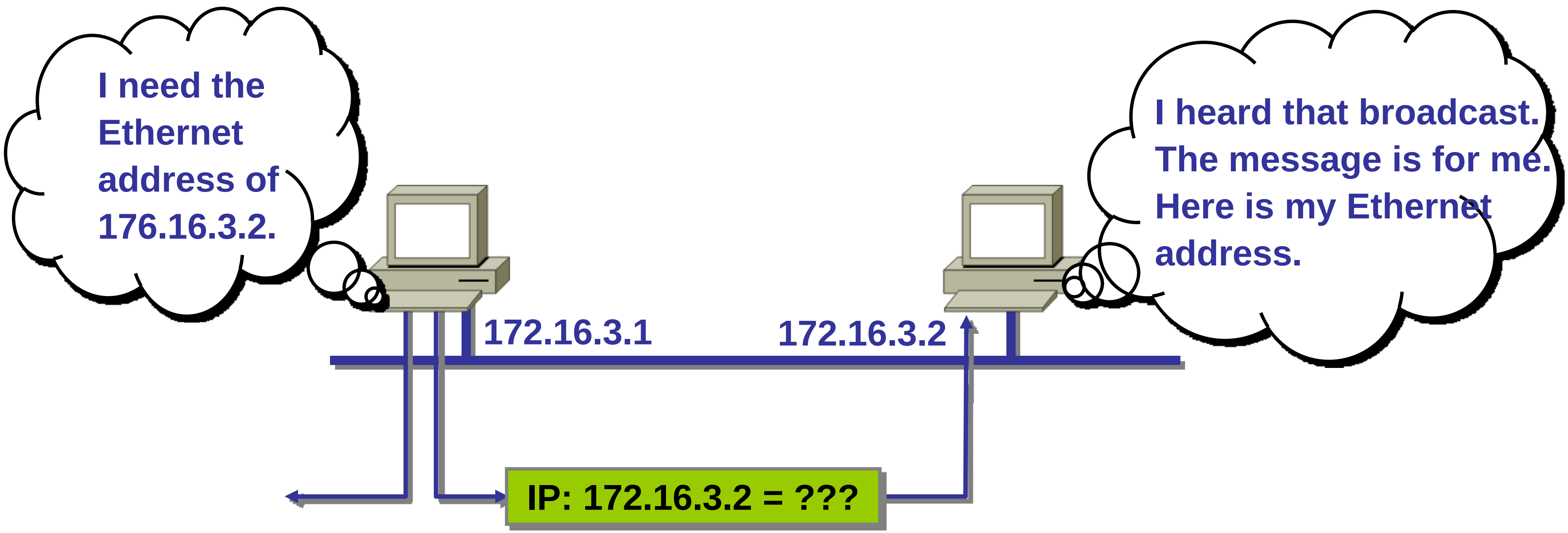
Address Resolution Protocol (ARP)



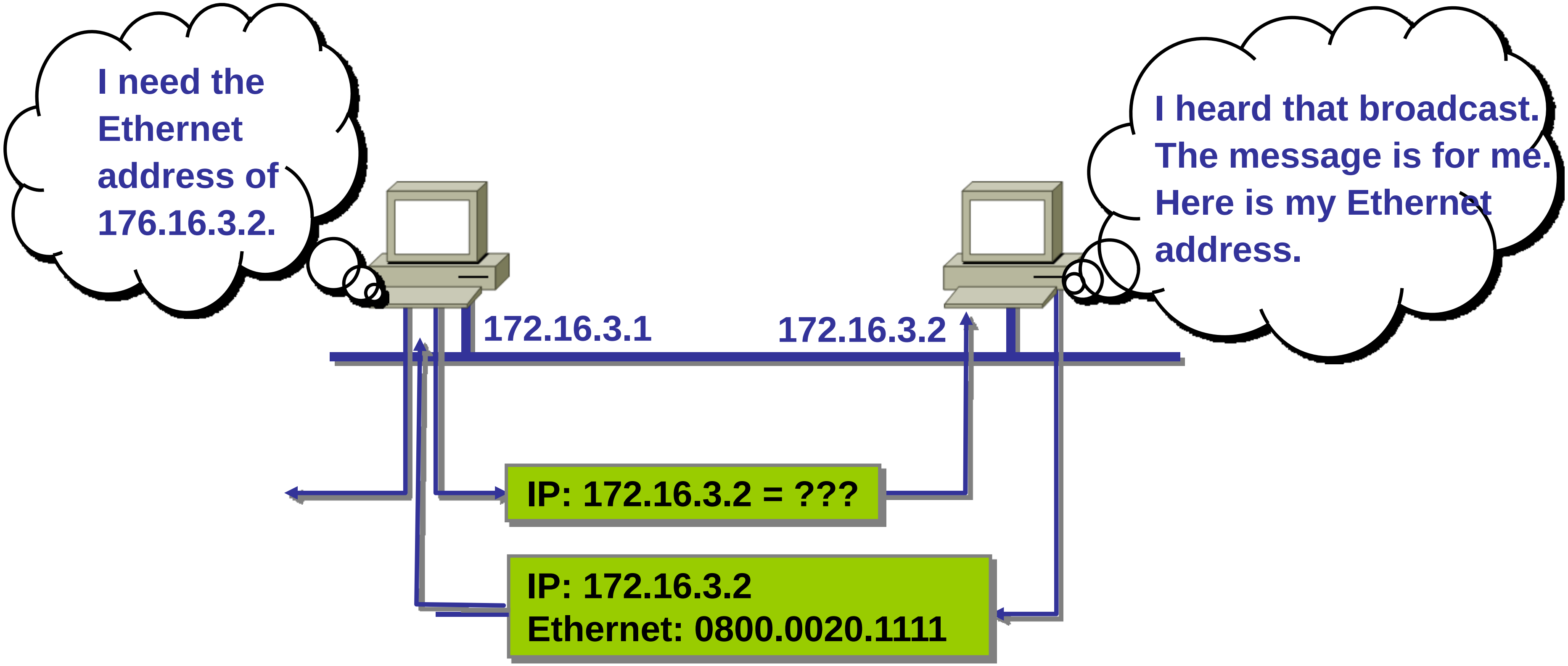
Addressing:

- 48-bit MAC (Ethernet) Address – Flat
- 32-bit Internet Address (IP) – Hierarchical

Address Resolution Protocol (ARP)



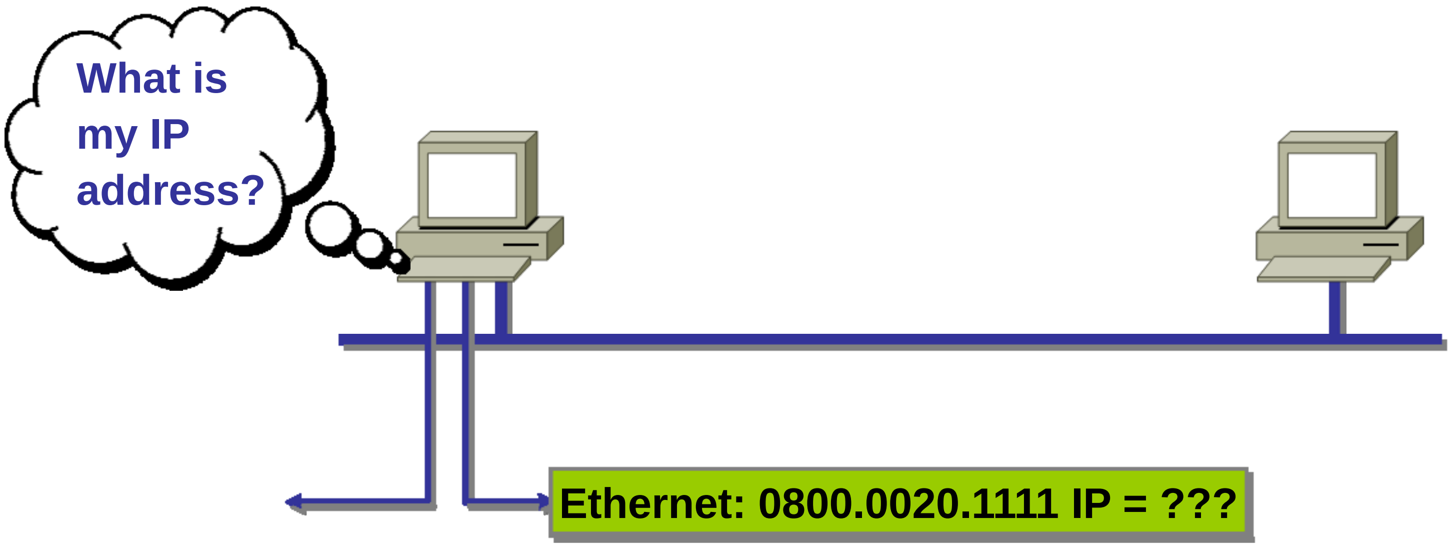
Address Resolution Protocol (ARP)



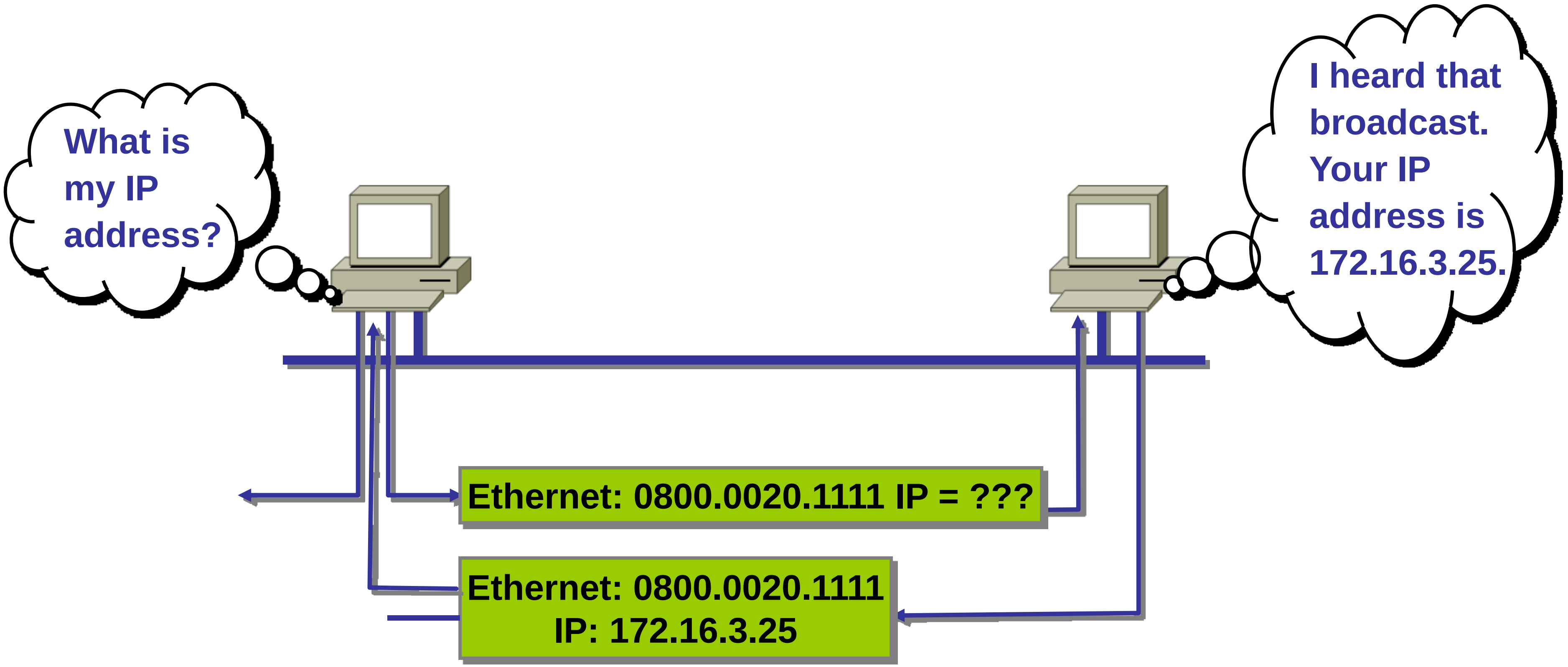
UCSC

BIT

Reverse ARP



Reverse ARP


What is my IP address?
I heard that broadcast. Your IP address is 172.16.3.25.
Ethernet: 0800.0020.1111 IP = ???
Ethernet: 0800.0020.1111
IP: 172.16.3.25

Dynamic Host Configuration Protocol (DHCP)

Dynamic Host Configuration Protocol (DHCP)

- Allows client machines to receive an IP address, DNS information, etc automatically
- Before DHCP users had to type in all this information by hand, which is bad:
  - Easy to mistype something when entering by hand
  - Manually changing network configuration every time you move your laptop is a pain
  - Bootp resolved some of these issues
    - … and DHCP still uses the same port as bootp

DHCP: Basics

- A client leases an IP address from a DHCP server for a given amount of time
- When lease expires, the client must ask DHCP server for a new address (clients attempt to renew lease after 50% of the lease time has expired)
- Typical leases may last for 30 seconds, 24 hours, or longer.

DHCP: Messages Overview

- Several messages are sent back and forth between a client and the DHCP server before it can successfully obtain an IP address

DHCP: DISCOVER

- Hardcoding the addresses of DHCP servers kind of defeats the purpose of automatic configuration
- Solution: A client using DHCP will broadcast a DISCOVER message to all computers on its subnet (address 255.255.255.255) to figure out the IP address of any DHCP servers
- Most routers are configured to pass this request within the campus or enterprise

DHCP: OFFER

- (Optionally) sent from server in response to a DISCOVER
- Contains an IP address, other configuration information as well (subnet mask, DNS servers, default gateway, search domains, etc)
- Note that all DHCP servers that receive a DISCOVER request may send an OFFER; since a client typically does not need > 1 IP address, more messages needed

DHCP: REQUEST

- Sent by client to request a certain IP address
  - Usually the one sent by an OFFER, but also used to renew leases. Also can be sent to try to get same address after a reboot
- This message is broadcast
- Most OSs by default will send a REQUEST for the first OFFER they receive – this means that if there is a rogue DHCP server on your subnet, most clients will *ignore* the OFFERs from the campus DHCP servers (since the OFFER from the rogue server gets to the user's PC first)!

DHCP: ACK/NACK

- Sent by server in response to a REQUEST
- ACK: Request accepted, client can start using the IP it REQUESTed
- NACK: Something is wrong with the client’s REQUEST (for example they requested an IP address they’re not supposed to have)

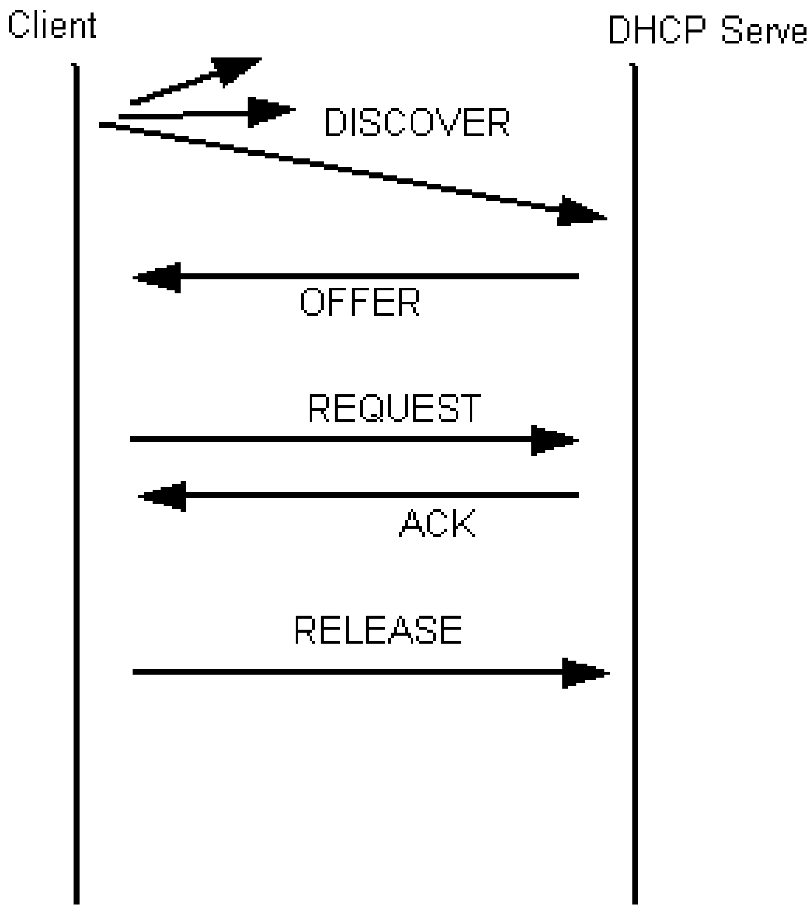
DHCP: RELEASE

- Sent by client to end a lease
- Not strictly required, but is the “polite” thing to do if done with the IP (could just let the lease expire)
- Some clients may not send RELEASEs in an attempt to keep the same IP address for as long as possible

UCSC

BIT

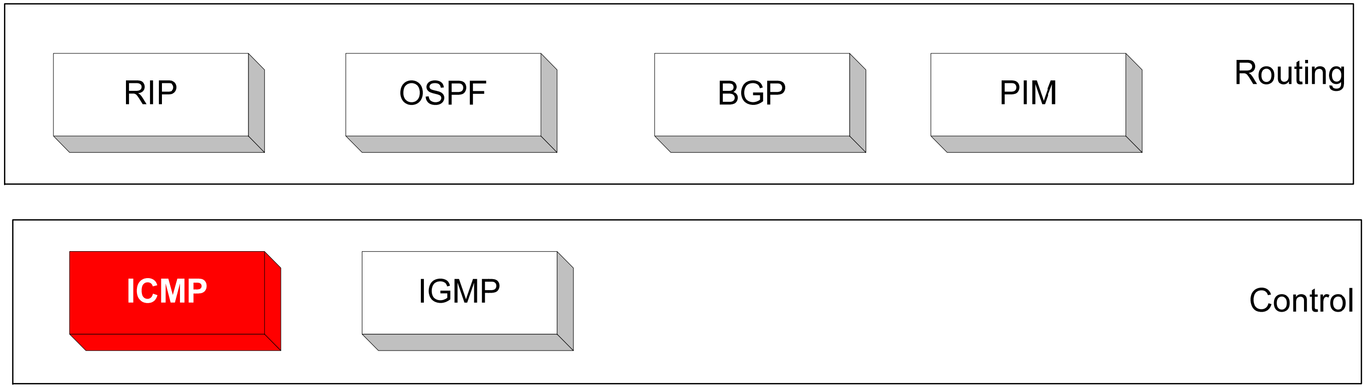
DHCP: Big Picture



Internet Control Message Protocol (ICMP)

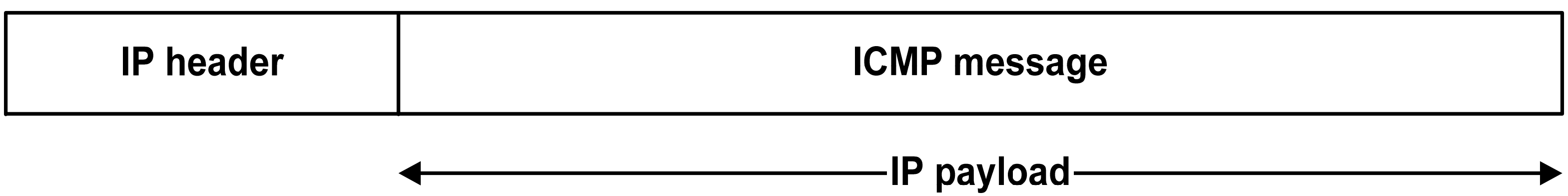
Overview

- The IP (Internet Protocol) relies on several other protocols to perform necessary control and routing functions:
  - Control functions (ICMP)
  - Multicast signaling (IGMP)
  - Setting up routing tables (RIP, OSPF, BGP, PIM, …)



Overview

- The **Internet Control Message Protocol (ICMP)** is a helper protocol that supports IP with facility for
  - Error reporting
  - Simple queries



- ICMP messages are encapsulated as IP datagrams:

ICMP message format

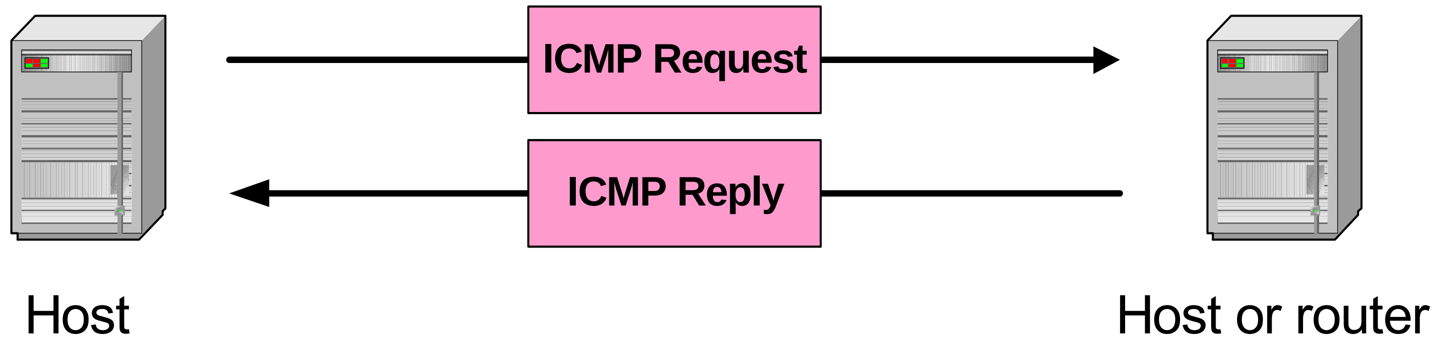
bit # 0 … 7	8 … 15	16 … 23	24 … 31
type	**code**	**checksum**	
additional information or 0x00000000			

4 byte header:

- Type (1 byte): type of ICMP message
- Code (1 byte): subtype of ICMP message
- Checksum (2 bytes): similar to IP header checksum. Checksum is calculated over entire ICMP message

If there is no additional data, there are 4 bytes set to zero.
→ each ICMP messages is at least 8 bytes long

ICMP Query message



ICMP query:

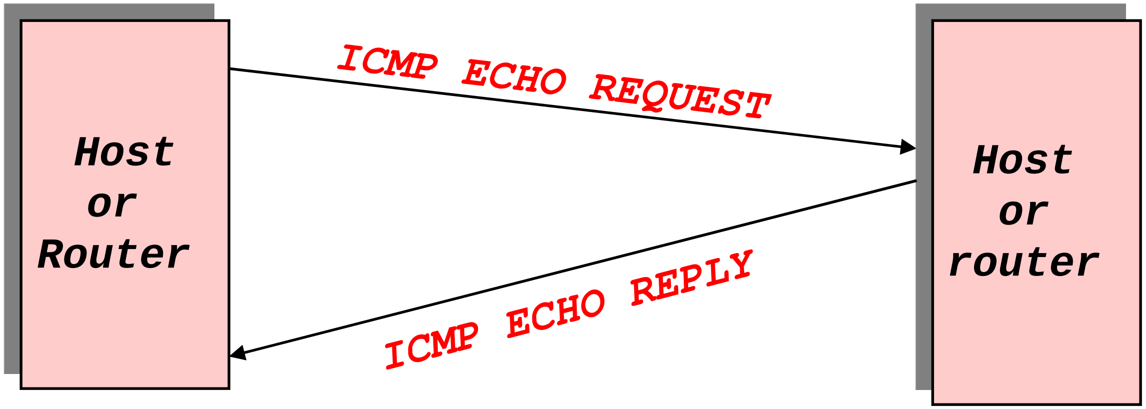
- Request sent by host to a router or host
- Reply sent back to querying host

Example of ICMP Queries

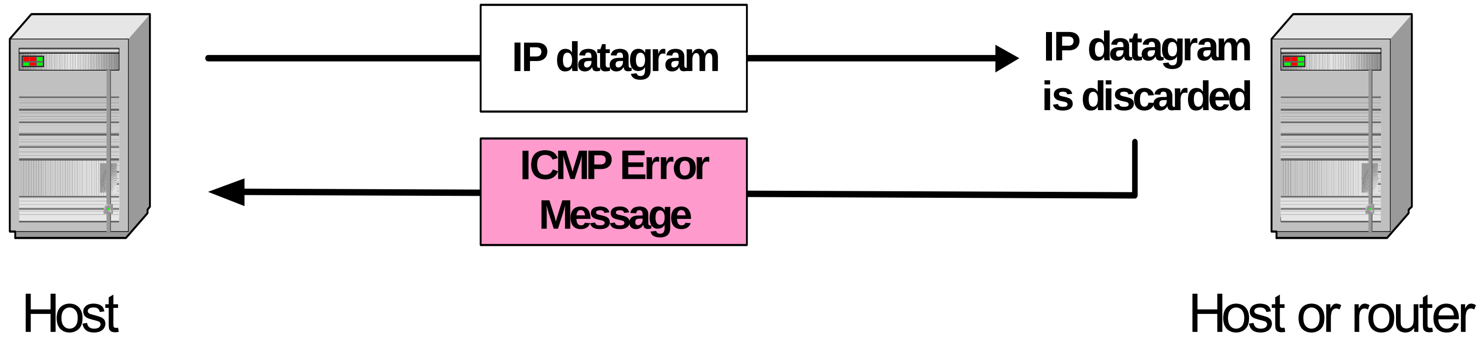
Type/Code:	Description	
8/0	Echo Request	The ping command uses Echo Request/ Echo Reply
0/0	Echo Reply	
13/0	Timestamp Request	
14/0	Timestamp Reply	
10/0	Router Solicitation	
9/0	Router Advertisement	

Example of a Query: Echo Request and Reply

- Ping's are handled directly by the kernel
- Each Ping is translated into an ICMP Echo Request
- The Ping'ed host responds with an ICMP Echo Reply

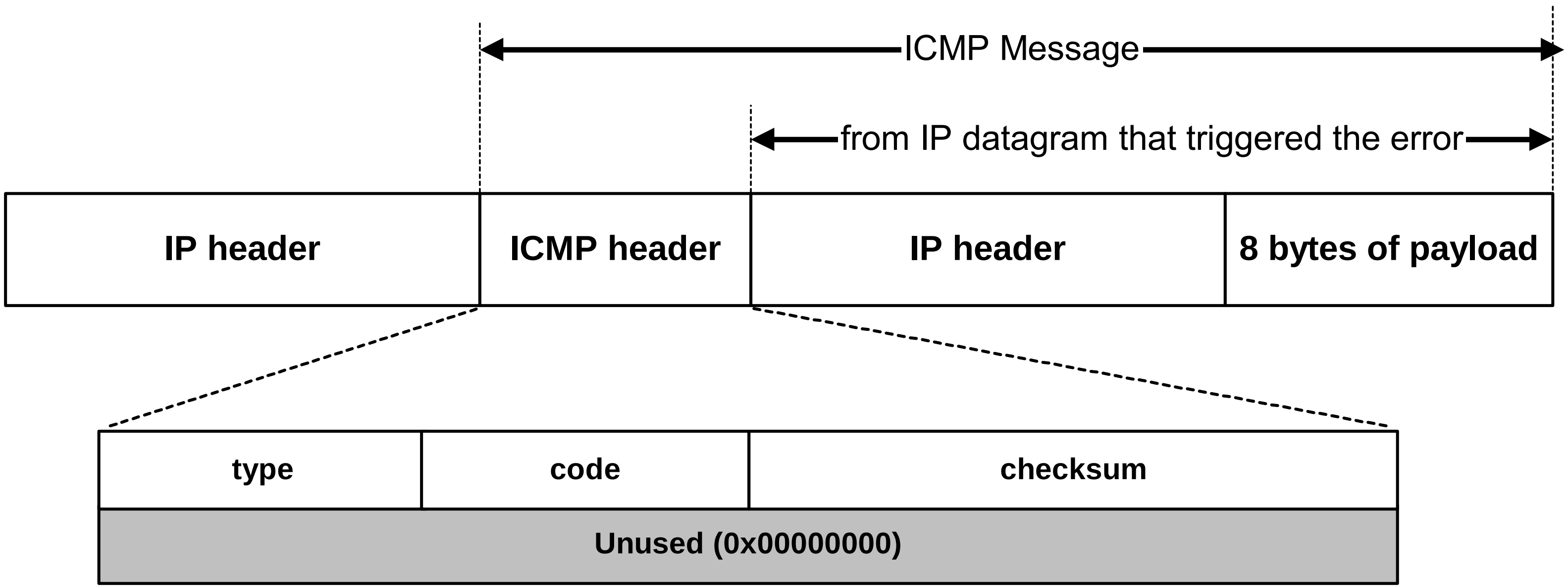


ICMP Error message



- ICMP error messages report error conditions
- Typically sent when a datagram is discarded
- Error message is often passed from ICMP to the application program

ICMP Error message



- ICMP error messages include the complete IP header and the first 8 bytes of the payload (typically: UDP, TCP)

Frequent ICMP Error message

Type	Code	Description	
3	0–15	Destination unreachable	Notification that an IP datagram could not be forwarded and was dropped. The code field contains an explanation.
5	0–3	Redirect	Informs about an alternative route for the datagram and should result in a routing table update. The code field explains the reason for the route change.
11	0, 1	Time exceeded	Sent when the TTL field has reached zero (Code 0) or when there is a timeout for the reassembly of segments (Code 1)
12	0, 1	Parameter problem	Sent when the IP header is invalid (Code 0) or when an IP header option is missing (Code 1)

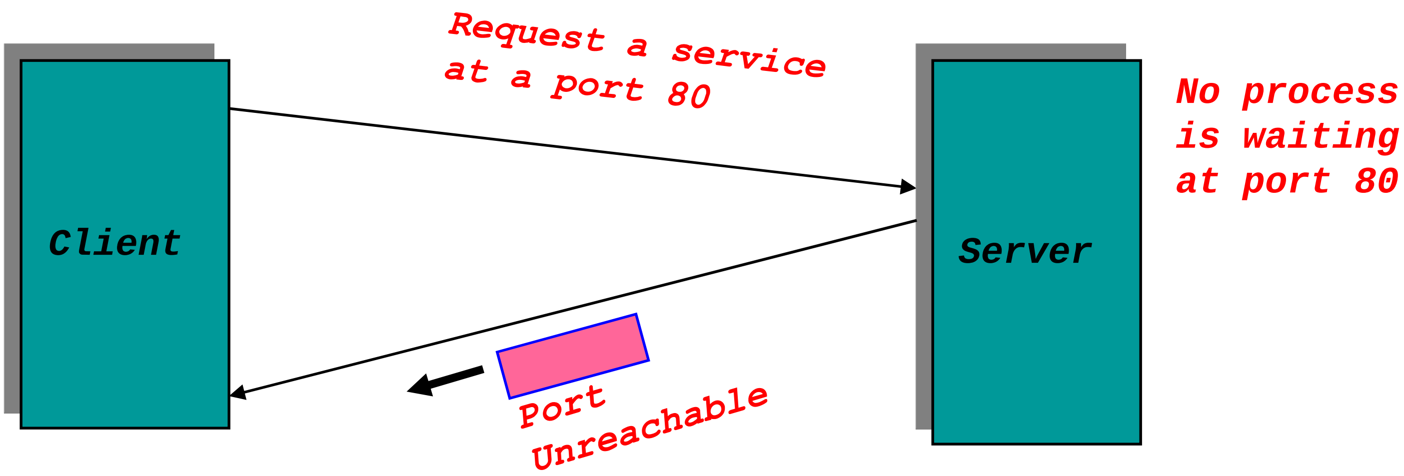
Some subtypes of the “Destination Unreachable”

Code	Description	Reason for Sending
0	Network Unreachable	No routing table entry is available for the destination network.
1	Host Unreachable	Destination host should be directly reachable, but does not respond to ARP Requests.
2	Protocol Unreachable	The protocol in the protocol field of the IP header is not supported at the destination.
3	Port Unreachable	The transport protocol at the destination host cannot pass the datagram to an application.
4	Fragmentation Needed and DF Bit Set	IP datagram must be fragmented, but the DF bit in the IP header is set.

Example: ICMP Port Unreachable

- RFC 792: If, in the destination host, the IP module cannot deliver the datagram because the indicated protocol module or process port is not active, the destination host may send a destination unreachable message to the source host.

- Scenario:



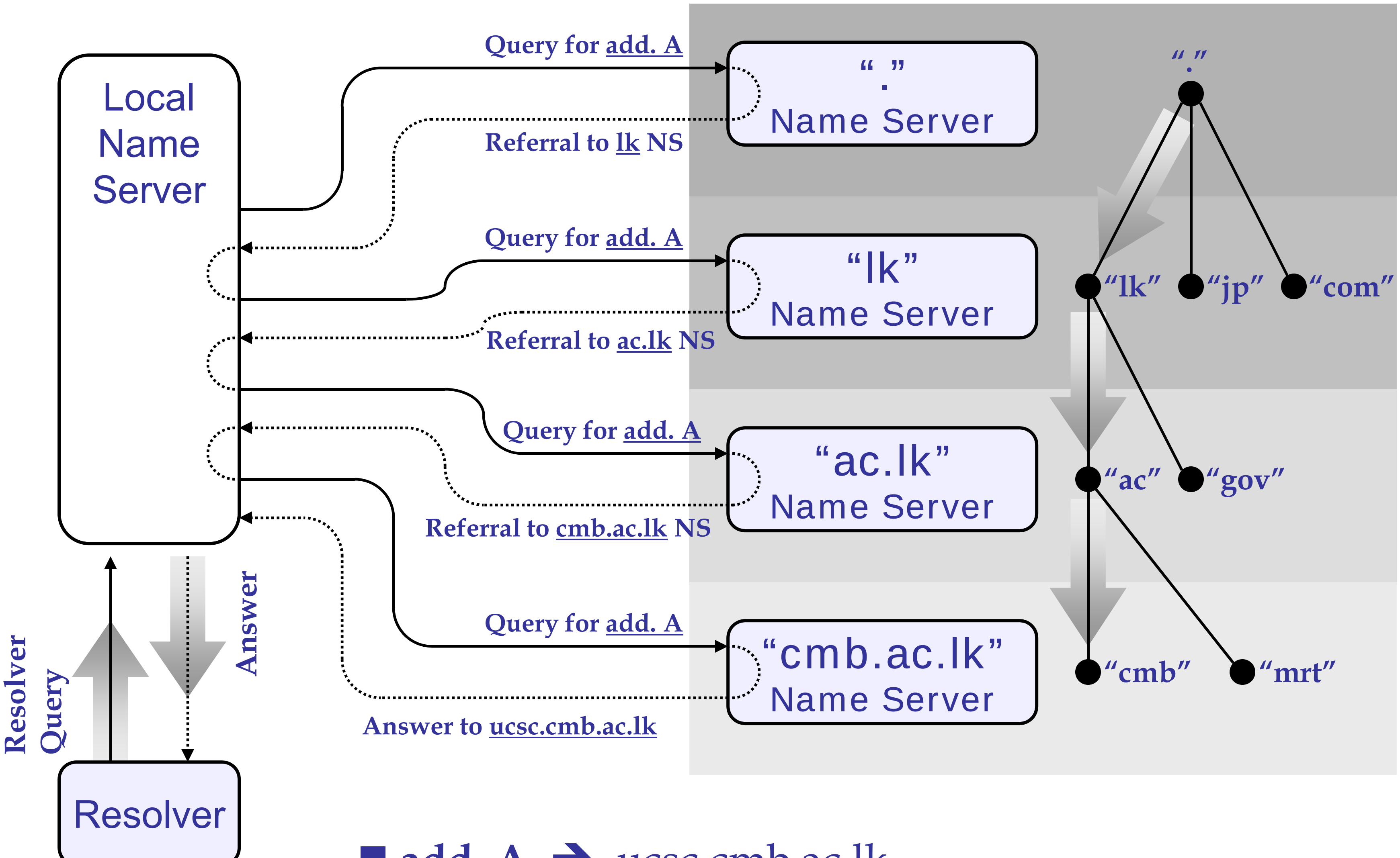
Section 5.4

Application Layer Protocols

What is DNS?

- DNS (Domain Name System)
  - A database that is used by TCP/IP applications to map between hostnames and IP addresses
  - Characteristics of DNS
    - A hierarchical namespace for hosts and IP addresses
    - A host table implemented as a distributed database
    - A Client/Server system
  - Components of DNS
    - Namespace and Resource Record
    - Name Server
    - Resolver (Client)

What is DNS? (con't)



- add. A → ucsc.cmb.ac.lk

What is DNS? (con't)

- Top Level Domains

Domain Suffix	Type of Organization
ARPA	Reverse lookup domain (special Internet function)
COM	Commercial
EDU	Educational
GOV	Government
ORG	Non-commercial organization (such as a nonprofit agency)
NET	Network (such as an ISP)
INT	International Treaty Organization
MIL	U.S. military organization
BIZ	Businesses
INFO	Unrestricted use
AERO	Air-transport industry
COOP	Cooperatives
MUSEUM	Museums
NAME	Individuals
PRO	Professionals (such as doctors, lawyers, and engineers)

What is DNS? (con't)

- Namespace
  - DNS namespace is a tree of "domains"
  - Refers to the actual database of IP addresses and their associated names
  - At the highest level of the hierarchy sit the **root servers**
- Zone
  - A zone is a sub-tree of the DNS database that is administered as a single separate entity. It can consists of one domain or domain with sub-domains

What is DNS? (con't)

- ❑Resource Records (RR)
  - •RRs contain the data associated with domain names
- ❑ Name Server
  - •The server programs that store information about the domain name space
- ❑ Resolver (Client)
  - •The programs that extract information from name servers in response to client requests

DNS: Basics

- Hierarchical namespace
- Distributed system – very few core servers
- Stores other information than simple hostname <-> IP mappings
- Request/response protocol

DNS: Architecture

- DNS servers are responsible for one or more domains of any level
- “Root servers” are maintained throughout the world (one is in Palo Alto) and are responsible for all of the top-level domains
  - When you register a domain, an entry for that domain is added to the appropriate root server
- Owners of each regular domain or subdomain maintain (or outsource) their own DNS servers containing the correct information

DNS: Domain servers

- What kind of records can be requested for a given domain?
  - Address translation
  - Caching information
  - Mail server information
  - Authoritative nameserver information
- How is this data requested?
  - Each record has a type and certain data associated with it – clients request records of a certain type from a server

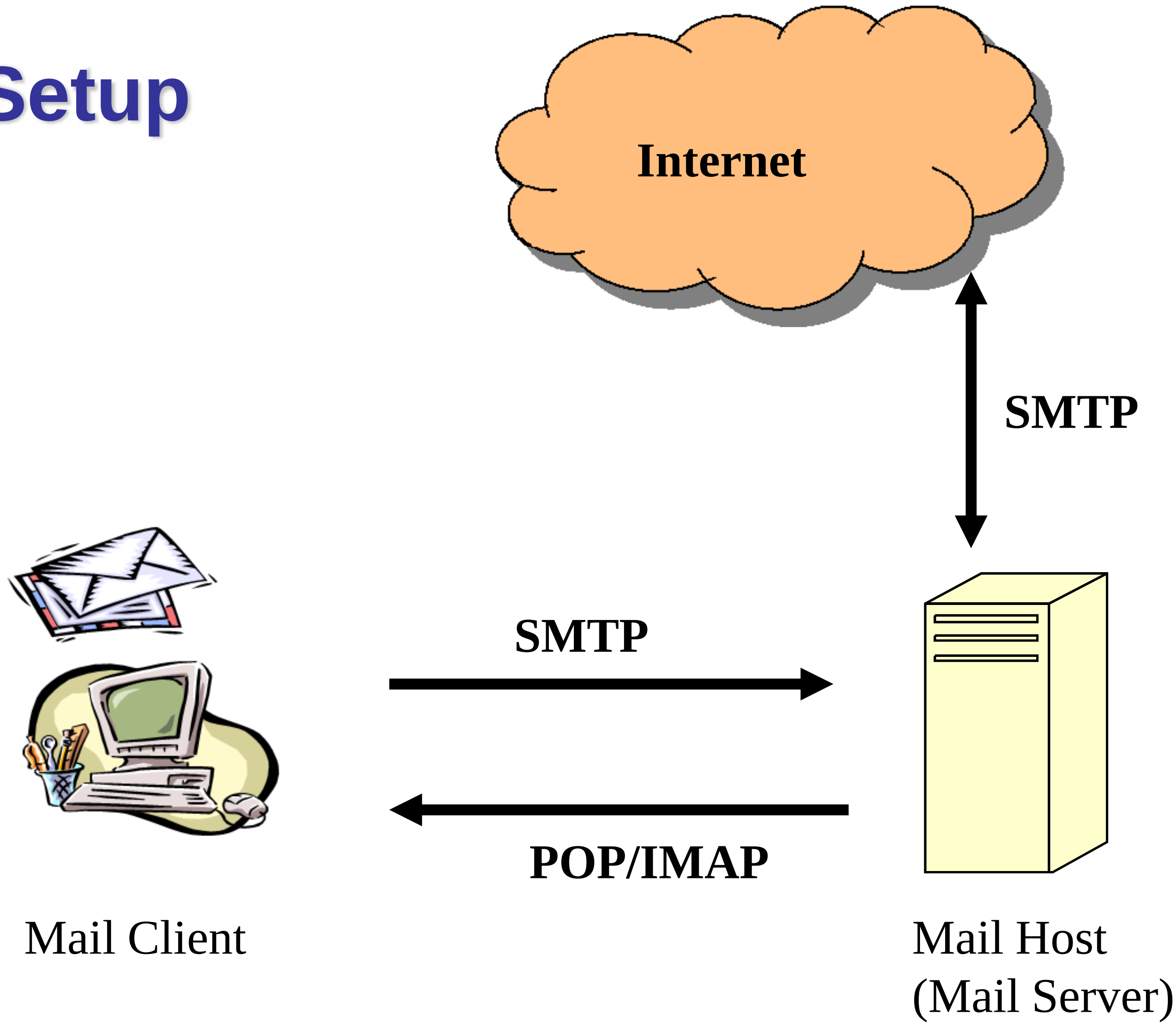
Simple Mail Transfer Protocol

- Basic protocol for email exchange over the Internet
- Fundamental difference between SMTP and FTP/TELNET is that it is NOT an interactive protocol
  - Messages are queued and spooled by SMTP agent
- Users interact with email application
  - E.g. Microsoft Outlook Express!
- Application interfaces with Message Transfer Agent
  - *Sendmail* on UNIX
  - Setup and configured by admins.
- SMTP specifies how MTA's pass email across the Internet
  - Also uses NVT commands

Simple Mail Transfer Protocol

- Client uses email application to construct and send messages
- Message is passed to mail spooler which is part of MTA
  - Application communicates with MTA via email transfer protocol
    - Post Office Protocol (POP3) is common, but not very secure
    - Our department uses IMAP
- MTA's on remote systems listen for incoming mail on well known port (25)
- Messages are delivered in two parts – header and body
  - Header format has exact specification (RFC 822)
  - Body content types are specified by MIME

A Mail Setup


Internet
SMTP
SMTP
POP/IMAP
Mail Client
Mail Host
(Mail Server)

Email Exchange

There are **5** major parts involved in an email exchange

1. The user program
2. The server daemon (MTA)
3. The mailhost
4. A daemon for users to read mail from mailhost (MUA)
5. DNS

Email Exchange (Con't)

- Mail server daemons: **sendmail**, **qmail, postfix**, **exim**, **mmdf, smail**, **zmailer** etc.
- The server daemon usually has 2 function:
  - looks after receiving incoming mail
  - delivers outgoing mail
- The server daemon does not allow you to read your mail. For this you need an additional daemon (**POP**, **IMAP**, etc).
- The DNS and its daemon "**named**" play a large role in the delivery of email.

File Transfer Protocol

- This is the most basic file transfer application in the Internet
  - One of the original client/server applications run on the ARPANET
- Works on both Unix systems as well as non-Unix systems
- Allows for both file transfer and interactive access
- Requires authentication via user name and password
- Requires that a host system run an FTP server
  - Listens for incoming requests on a well known port (21)
  - Anonymous/Guest logins are common
- FTP is a two process model
  - Control process which communicates with peer control process
    - These processes communicate commands/responses as well as port information
  - Data transfer process which actually transfers requested file

File Transfer Protocol Contd.

- Client control process connects to server control process
  - ftp ucsc.cmb.ac.lk
- The client also starts a data transfer process which listens on a local port
  - Communicates this port number to server via control process
- If client requests a file transfer, server initiates connection to client’s data transfer port
  - Server uses well known port for data transfer (20)
- Commands used by FTP are actually a subset of TELNET protocol NVT ASCII

Secure FTP

- SFTP is a program that uses SSH to transfer files.
- SFTP encrypts both commands and data, preventing passwords and sensitive information from being transmitted in the clear over the network.
- It is functionally similar to FTP.
- There are two ways you can use SFTP: graphical SFTP clients and command line SFTP.

Hyper Text Transfer Protocol

- Client can make requests
  - GET for requesting a file from the server
  - POST for submitting information to the server
  - When it makes a request, the client also passes some client side descriptors to the server
- Server responds
  - HTTP headers
  - HTML document
    - or JPEG, or GIF, or…
- Browser implements client side of this service
- Web server implements server side of this service

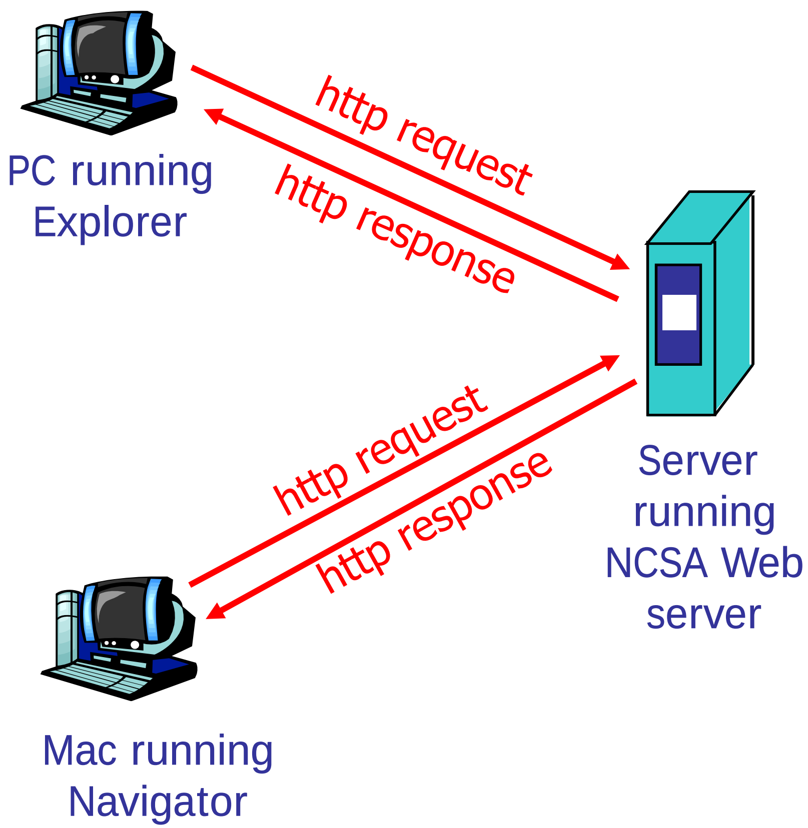
HTTP Request Methods

METHOD	DESCRIPTION
GET	Request to read a web page
HEAD	Request to read a web page’s header
PUT	Request to store web page
POST	Append to a named resource
DELETE	Remove the web page
TRACE	Echo the incoming request
CONNECT	Reserved for future forecast
OPTIONS	Query certain options

The Web: the HTTP protocol

HTTP: hypertext transfer protocol

- Web's application layer protocol
- client/server model
  - *client:* browser that requests, receives, "displays" Web objects
  - *server:* Web server sends objects in response to requests
- http1.0: RFC 1945
- http1.1: RFC 2068



The http protocol: more

http: TCP transport service:

- client initiates TCP connection (creates socket) to server, port 80
- server accepts TCP connection from client
- http messages (application-layer protocol messages) exchanged between browser (http client) and Web server (http server)
- TCP connection closed

http is "stateless"

- server maintains no information about past client requests

Protocols that maintain "state" are complex!

- past history (state) must be maintained
- if server/client crashes, their views of "state" may be inconsistent, must be reconciled

http message format: request

- two types of http messages: *request*, *response*
- http request message:
  - ASCII (human-readable format)

request line (GET, POST, HEAD commands)

header lines

```
GET /somedir/page.html HTTP/1.0
User-agent: Mozilla/4.0
Accept: text/html, image/gif,image/jpeg
Accept-language:fr
```

(extra carriage return, line feed)

Carriage return, line feed indicates end of message

http message format: response

status line
(protocol
status code
status phrase)

```
HTTP/1.0 200 OK
Date: Thu, 06 Aug 1998 12:00:15 GMT
Server: Apache/1.3.0 (Unix)
Last-Modified: Mon, 22 Jun 1998 …...
Content-Length: 6821
Content-Type: text/html

data data data data data ...
```

header
lines

data, e.g.,
requested
html file

http response status codes

200 OK

- request succeeded, requested object later in this message

301 Moved Permanently

- requested object moved, new location specified later in this message (Location:)

400 Bad Request

- request message not understood by server

404 Not Found

- requested document not found on this server

505 HTTP Version Not Supported

What is VoIP?

- VoIP (Voice over Internet Protocol), sometimes referred to as Internet telephony, is a method of digitizing voice, encapsulating the digitized voice into packets and transmitting those packets over a packet switched IP network.

Voice over IP - the basics

- Most implementations use H.323 protocol
  - Same protocol that is used for IP video.
  - Uses TCP for call setup
  - Traffic is actually carried on RTP (Real Time Protocol) which runs on top of UDP.

UCSC

BIT

VoIP Protocols

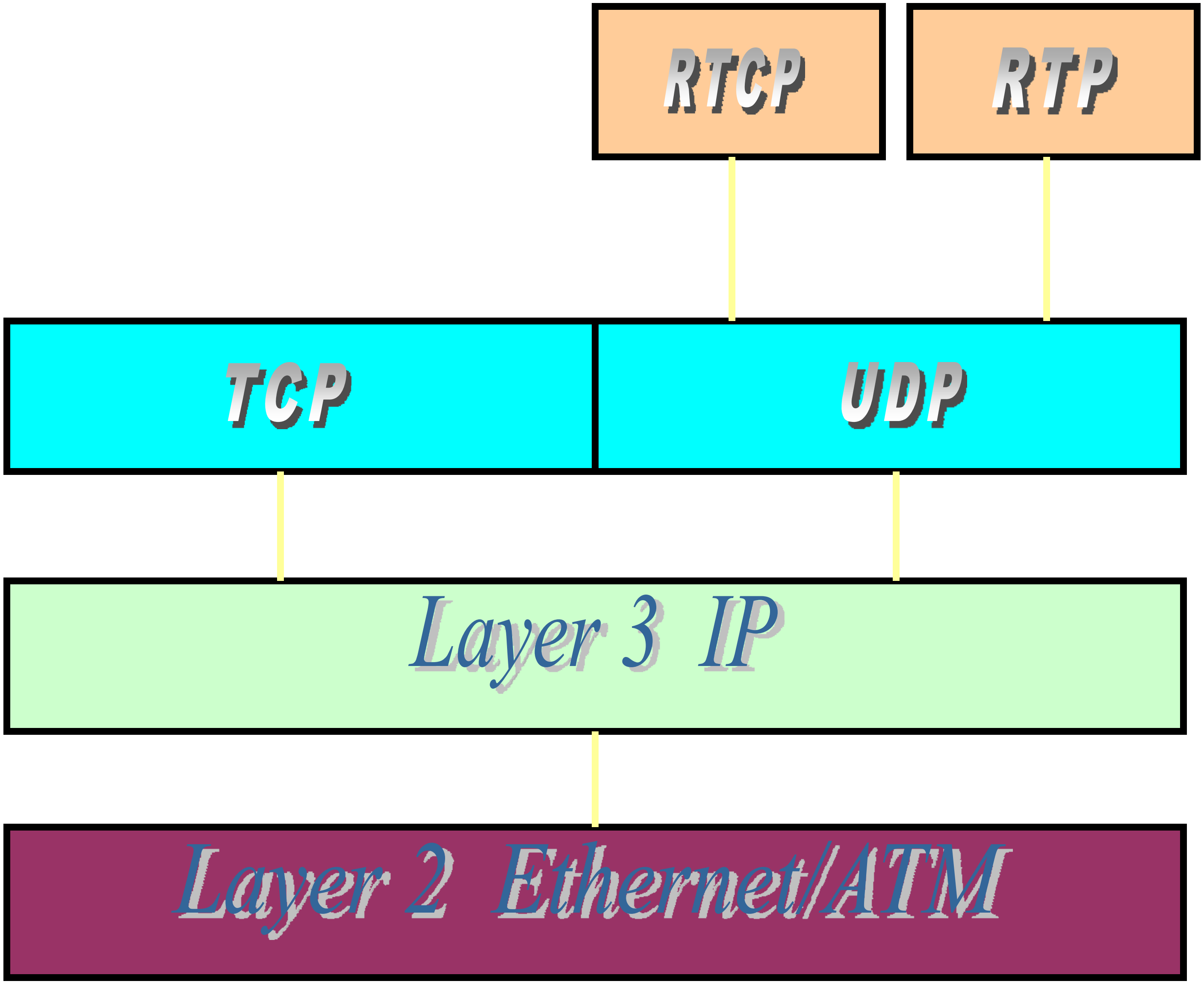
- H.323 Multimedia Standard
  - H.225 RAS - Registration, Admission, Status
  - Q.931 - Call Signaling (Setup & Termination)
  - H.245 - Call Control (Preferences, Flow Control, etc.)
  - Lots of G.7XX CODECS for audio
- SIP – Session Initialization Protocol
  - Covered in next presentation

Here's how it stacks up:

H.323	Multimedia Protocol
H.225	Call setup & Control – RAS (Q.931)
H.235	Security & Authentication
H.245	Call negotiation, capability exchange
H.450	Other supplemental Services
H.246	Circuit Switched Network Interop.
H.332	Conferencing
H.26X	Video CODECS
H.7XX	Audio CODECS

How they fit in: The ISO Model

ISO Model Layer	Protocol or Standard
Presentation	Applications / CODECS
Session	H.323 & SIP
Transport	RTP / UDP / TCP
Network	IP – Non QOS
Data Link	ATM, FR, PPP, Ethernet


RTCP
RTP
TCP
UDP
Layer 3 IP
Layer 2 Ethernet/ATM

Comparison of Packet vs. Circuit Switching

	Circuit	Packet
Call Setup	Database / SS 7 Overlay	H.323 & SIP
Communications Channel	Dedicated	Shared
Addressing	NANP	IPv4 & IPv6

Section 5.5

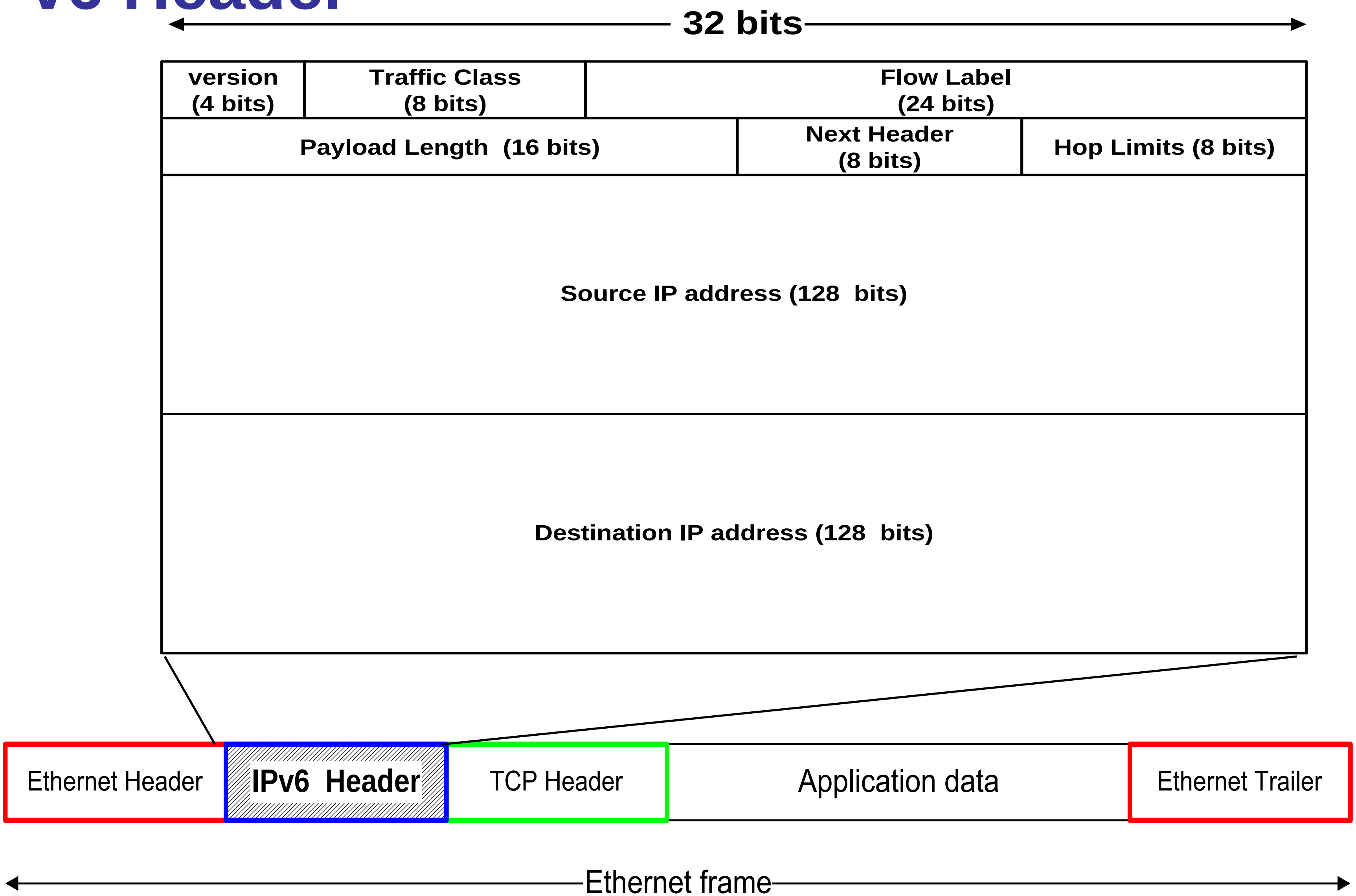
IP version 6 & Multicasting

IPv6 - IP Version 6

- **IP Version 6**
  - Is the successor to the currently used IPv4
  - Specification completed in 1994
  - Makes improvements to IPv4 (no revolutionary changes)

- One (not the only !) feature of IPv6 is a significant increase in of the IP address to **128 bits (16 bytes)**
  - IPv6 will solve – for the foreseeable future – the problems with IP addressing
  - 10^{24} addresses per square inch on the surface of the Earth.

IPv6 Header


32 bits
version
(4 bits)
Traffic Class
(8 bits)
Flow Label
(24 bits)
Payload Length (16 bits)
Next Header
(8 bits)
Hop Limits (8 bits)
Source IP address (128 bits)
Destination IP address (128 bits)
Ethernet Header
IPv6 Header
TCP Header
Application data
Ethernet Trailer
Ethernet frame

IPv6 vs. IPv4: Address Comparison

- **IPv4** has a maximum of
  - $2^{32} \approx$ 4 billion addresses

- **IPv6** has a maximum of
  - **$2^{128} = (2^{32})^4 \approx$ 4 billion x 4 billion x 4 billion x 4 billion addresses**

Notation of IPv6 addresses

- **Convention**: The 128-bit IPv6 address is written as **eight 16-bit integers** (using hexadecimal digits for each integer)

  CEDF:BP76:3245:4464:FACE:2E50:3025:DF12

- **Short notation:**
- Abbreviations of leading zeroes:

  CEDF:BP76:0000:0000:009E:0000:3025:DF12
  → CEDF:BP76:0:0:9E :0:3025:DF12

- ":0000:0000:0000" can be written as "::"

  CEDF:BP76:0:0:FACE:0:3025:DF12

  → CEDF:BP76::FACE:0:3025:DF12

- IPv6 addresses derived from IPv4 addresses have 96 leading zero bits. Convention allows to use IPv4 notation for the last 32 bits.

  ::80:8F:89:90 → ::128.143.137.144

Multicasting applications

- Multimedia
  - Telephony and vedio conference
  - Groupware (CSCW)
  - Internet based Radio/tv broadcast and VoD
  - Games
  - Group VR
- Database Replication- Simultaneous update
- Parallel Computing(GRID)
- Real time news
  - Stock market
  - Conference announcements

Multicasting applications

- Network Control infor exchange
  - Routing protocols eg OSPF
  - Neighbor discovery
  - Routre advertisements/solicitation
- Resource seek
  - DHCP, auto configure,NTP,GK,DNS.

Multicast Address Format

FP (8bits)	Flags (4bits)	Scope (4bits)	Group ID (80+32bits)	
11111111	000T	Lcl/Sit/Gbl	MUST be 0	Locally administered- 32

- flag field
  - low-order bit indicates permanent/transient group
  - (three other flags reserved)
- scope field:
  - 1 - node local
  - 2 - link-local
  - 5 - site-local
  - 8 - organization-local
  - B - community-local
  - E - global
  - (all other values reserved)
- map IPv6 multicast addresses directly into low order 32 bits of the IEEE 802 MAC

Multicast Address Format Unicast-Prefix based

FP (8bits)	Flags (4bits)	Scope (4bits)	reserved (8bits)	plen (8bits)	Network Prefix (64bits)	Group ID (32bits)
11111111	00PT	Lcl/Sit/Gbl	MUST be 0	Locally administered	Unicast prefix	Auto configured

- P = 1 indicates a multicast address that is assigned based on the network prefix
- plen indicates the actual length of the network prefix
- Source-specific multicast addresses is accomplished by setting
- P = 1
- plen = 0
- network prefix = 0

End of Section 5.0