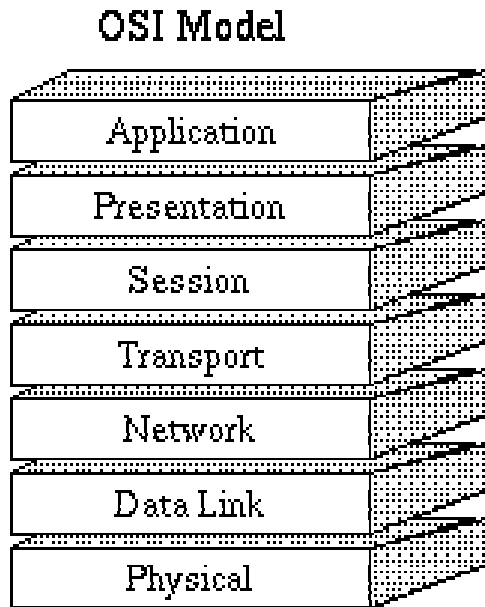# CSC 546: Client/Server Fundamentals

Fall 2000

Major client/server protocols

- OSI 7-layer model
- Microsoft suite: Named pipes + NetBIOS + NetBEUI
- IBM suite: APPC + LU 6.2 + PU 2.1
- Internet suite: RPC/XDR + Sockets + TCP/IP

OSI 7-layer model

OSI Model

| Application |
| Presentation |
| Session |
| Transport |
| Network |
| Data Link |
| Physical |

Open Systems Interconnection model
- developed by the ISO in 1984

- provides an abstract model of networking

    divides the tasks involved in moving info between networked computers into 7 task groups
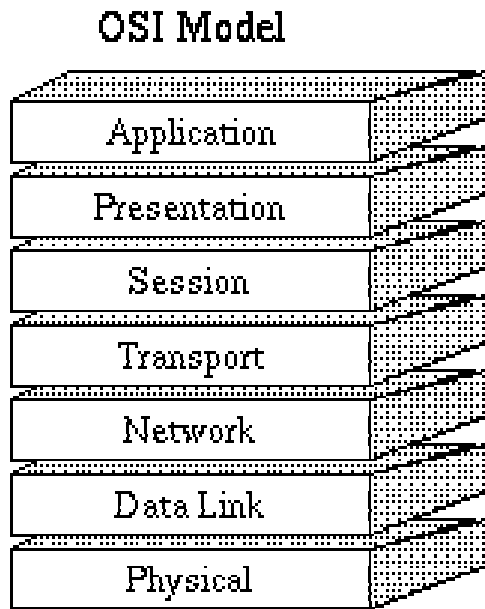
    each task group is assigned a layer

Each layer is reasonably self-contained, so
- can be implemented independently
- changes/updates to a layer need not effect other layers

Upper layers

Top 3 layers deal with application issues

```
OSI Model
┌─────────────┐
│ Application │
├─────────────┤
│ Presentation│
├─────────────┤
│ Session     │
├─────────────┤
│ Transport   │
├─────────────┤
│ Network     │
├─────────────┤
│ Data Link   │
├─────────────┤
│ Physical    │
└─────────────┘
```

- **Application layer**
  describes how applications will communicate
  e.g., HTTP, FTP, Telnet, SMTP

- **Presentation layer**
  describes the form of data being transferred &
  ensures that it will be readable by receiver
  e.g., mapping floating point formats across hosts,
  data compression, encryption

- **Session layer**
  describes the organization of large data sequences
  & manages communication session
  e.g., coordinates requests/responses

Lower layers

Bottom 4 layers deal with data transport issues

OSI Model

- Application
- Presentation
- Session
- Transport
- Network
- Data Link
- Physical
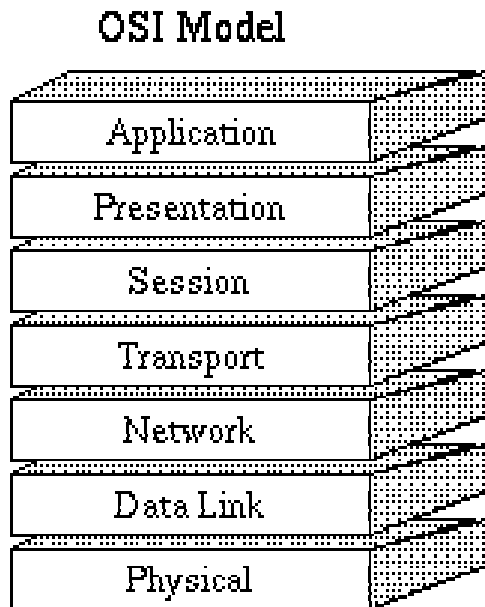
- **Transport layer**
  describes the quality and nature of data delivery
  e.g., how retransmissions are used to ensure delivery

- **Network layer**
  describes how a series of exchanges over various data links can deliver data across a network
  e.g., addressing and routing

- **Data Link layer**
  describes the logical organization of data bits transmitted on a particular medium
  e.g., frame sequencing, error notification

- **Physical layer:**
  describes the physical & electrical properties of the communications media
  e.g., voltage levels, data rates, max distances
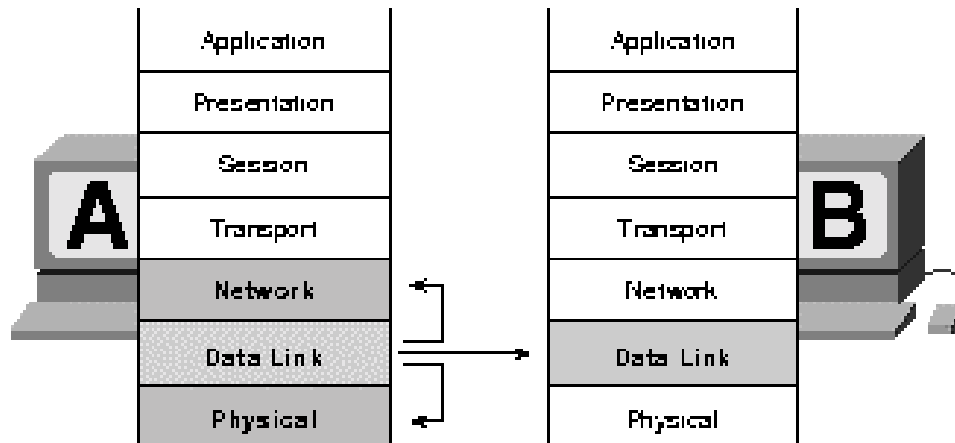
Interaction between layers

Across the network, processes at the same level can communicate
- e.g., Web browser & Web server both run at the application level
  communicate via HTTP protocol

On a machine, each layer can communicate with layer above & below

When a process wants to communicate over the network,
- information must be passed down through the layers
- data actually sent from the physical layer
- when reaches receiver, must get passed up through to the appropriate layer

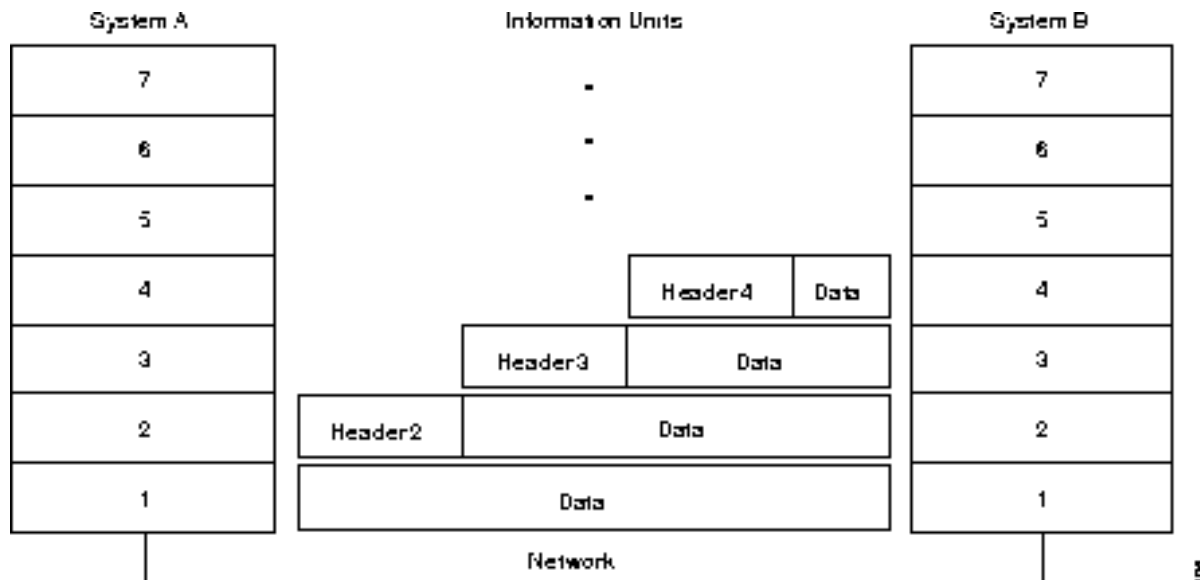| Application	Application
Presentation	Presentation
Session	Session
Transport	Transport
Network	Network
Data Link	Data Link
Physical	Physical

A   B

Layer protocols

each layer has protocols that define communications between adjacent layers

At the source, as data is passed down the layers:

the protocol for each layer adds control information to the data

At the destination, as data is passed up the layers:

the protocol for each layer strips and analyzes the control information for that layer

System A	Information Units	System B
7	.	7
6	.	6
5	.	5
4	Header4 Data	4
3	Header3 Data	3
2	Header2 Data	2
1	Data	1
	Network	

Protocol suites

Microsoft suite: Named Pipes + NetBIOS + NetBEUI

IBM suite: APPC + LU 6.2 + PU 2.1

Internet suite: RPC/XDR + Sockets + TCP/IP

Layer			
Application	Named Pipes	RPC	APPC
Presentation		XDR	
Session	NetBIOS	Sockets	LU 6.2
Transport	NetBEUI	TCP	
Network		IP	PU 2.1
Data Link	IEEE LLC		SDLC
	Token Ring	Ethernet	
Physical	Twisted Pair		Coax

Named pipes

application/presentation protocol for OS/2 and UNIX (a.k.a. FIFOs)

- turns the shared communication channel into a file-like entity
- each process reads & writes to the pipe the same way it would to a file
- instead of writing to disk, data is converted into a message

server creates a named pipe   `DosMakeNmPipe "\PIPE\foo"`

enters LISTEN state   `DosConnectNmPipe`

client opens named pipe   `DosMakeNmPipe "\\Server\PIPE\foo"`

pipe is opened, client & server can read/write to communicate

`DosRead`

`DosWrite`

client can close pipe   `DosClose`

server can close pipe   `DosDisconnectNmPipe`

NetBIOS

session layer protocol for:

Microsoft's MS-Net & LAN Manager, IBM's PC Network, Novell's Netware

- usually associated with NetBEUI, but can also sit on top of TCP/IP, SPC/IPX, …

- offers IPC services that can be used to implement client/server
- offers both connection-oriented and connectionless communications services
- provides a dynamic naming service to facilitate finding a specific process

commands are formatted within a 64 byte Network Control Block (NCB)

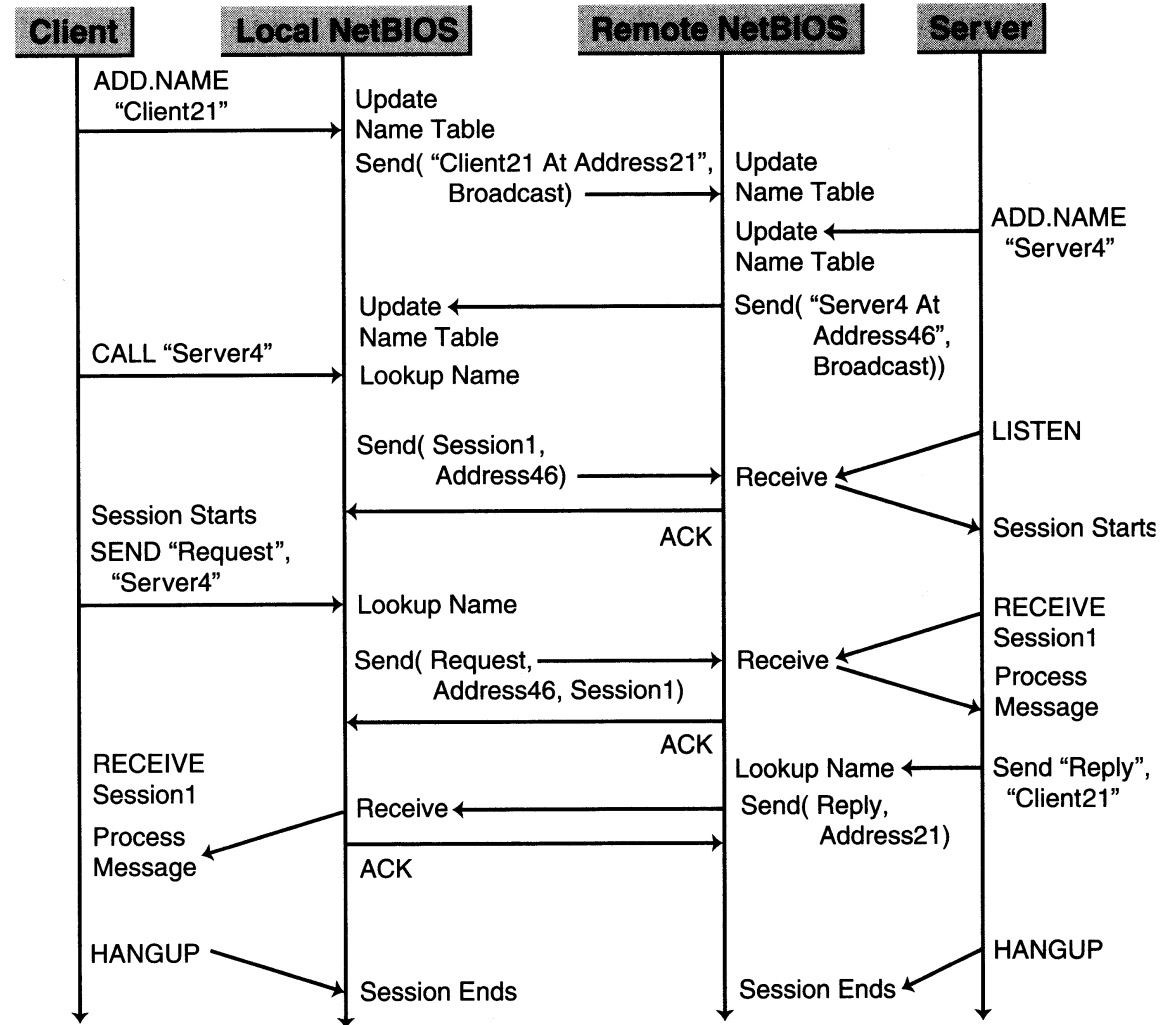|			
1	command code	16	name of Calling process
1	return code	16	local process name
1	session number	1	receive timeout interval
1	name number	1	send timeout interval
4	address of data buffer	4	address of callback
2	length of data buffer	1	completion status
14	reserved		

e.g., blocking SEND has command code 14H, nonblocking SEND has code 94H

Example NetBIOS session

connection-oriented
  (datastream) session:

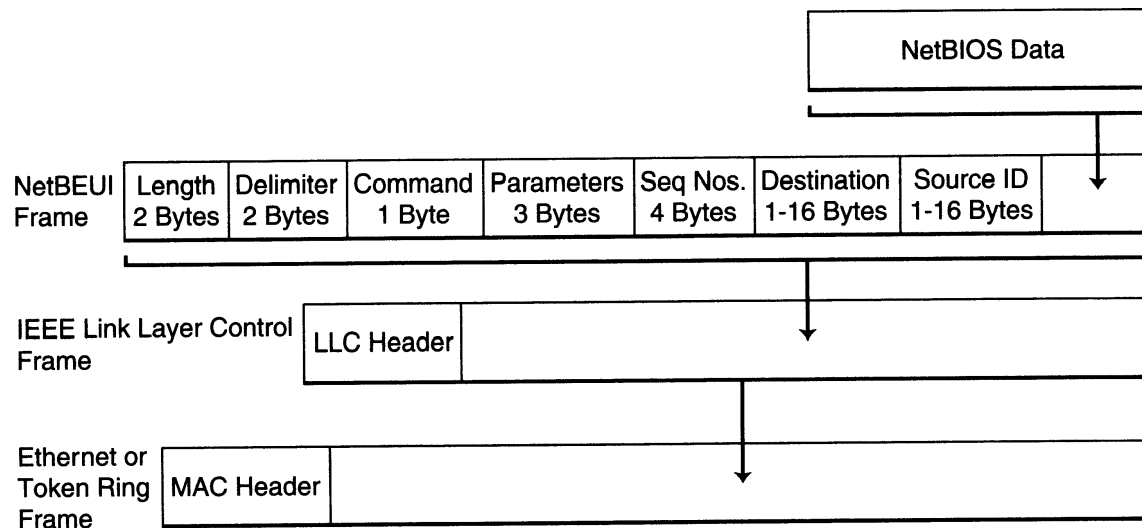for connection-less
  (datagram) session:

- no session number
- no CALL
- no LISTEN
- no HANGUP

Client	Local NetBIOS	Remote NetBIOS	Server

ADD.NAME
"Client21"

Update
Name Table

Send("Client21 At Address21",
  Broadcast)

Update
Name Table

Update
Name Table

ADD.NAME
"Server4"

Update
Name Table

Send("Server4 At
  Address46",
  Broadcast))

CALL "Server4"

Lookup Name

Send(Session1,
  Address46)

Receive

LISTEN

Session Starts

Session Starts

ACK

SEND "Request",
  "Server4"

Lookup Name

RECEIVE
Session1

Send(Request,
  Address46, Session1)

Receive

Process
Message

ACK

RECEIVE
Session1

Lookup Name

Send "Reply",
  "Client21"

Process
Message

Receive

Send(Reply,
  Address21)

ACK

HANGUP

Session Ends

Session Ends

HANGUP

NetBEUI

NetBIOS Extended User Interface:

- transport/network layer protocol to encapsulate NetBIOS commands

- numbered information frames provide sequenced, reliable datastream
- unnumbered information frames provide datagrams

	NetBIOS Data

NetBEUI Frame	Length 2 Bytes	Delimiter 2 Bytes	Command 1 Byte	Parameters 3 Bytes	Seq Nos. 4 Bytes	Destination 1-16 Bytes	Source ID 1-16 Bytes	

IEEE Link Layer Control Frame	LLC Header	

Ethernet or Token Ring Frame	MAC Header	

IBM's SNA protocols

IBM developed Systems Network Architecture (SNA) in 1974

- consolidated different communication products used with IBM equipment

Network layer:

a PU is an address used to administer physical devices

PU 2.1 node types can communicate with other nodes without host involvement

Transport/Session/Presentation layers:

an LU corresponds to a process to which Transaction Programs (TPs) can attach

TPs conduct *conversations* with each other using connection-oriented services

Application level:

Advanced Program to Program Communications (APPC) Protocol

provides the set of operations that TPs can use to talk to each other

similar to NetBIOS, uses a common control block structure for commands (verbs)
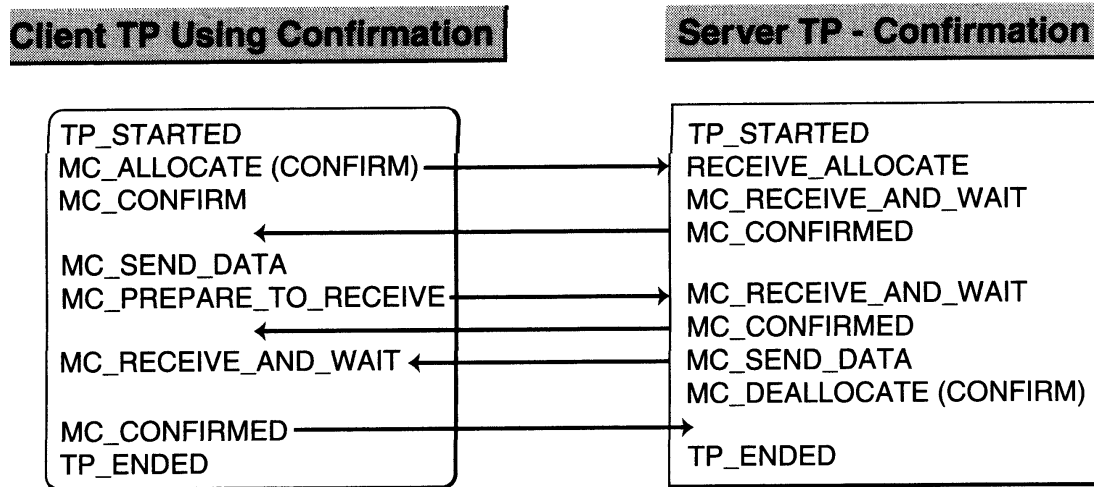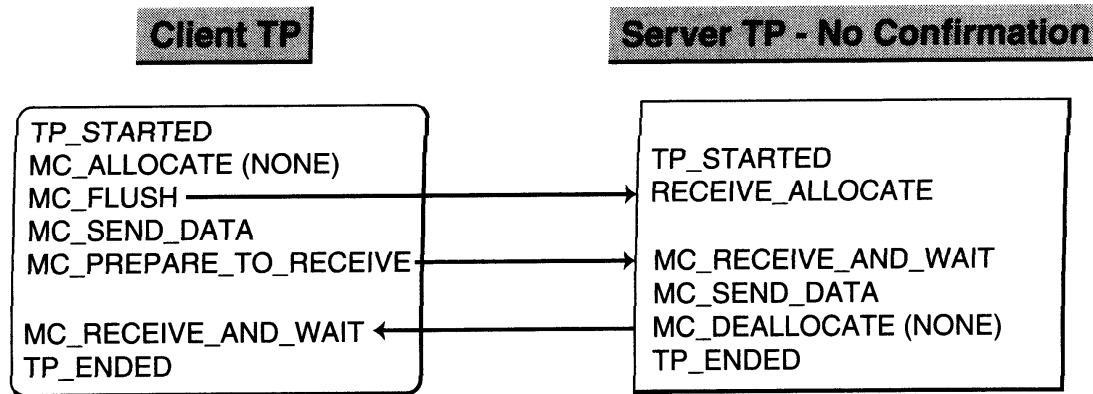
APPC

APPC verbs are either:

- control verbs, used to start or end a TP

  e.g., `TP_STARTED`   `TP_ENDED`

- conversation verbs, used to exchange messages between TPs

  e.g., `MC_ALLOCATE`   `RECEIVE_ALLOCATE`

  `MC_SEND_DATA`   `MC_RECEIVE_AND_WAIT`

- operator verbs, used by system administrators to manage sessions

  e.g., `MC_ALLOCATE`   `RECEIVE_ALLOCATE`

APPC is optimized for supporting long conversations

- high overhead in creating sessions and allocating conversations
- messages are internally buffered to optimize use of the network
- uses half-duplex style

APPC example conversations

Client TP

```
TP_STARTED
MC_ALLOCATE (NONE)
MC_FLUSH
MC_SEND_DATA
MC_PREPARE_TO_RECEIVE

MC_RECEIVE_AND_WAIT
TP_ENDED
```

Server TP - No Confirmation

```
TP_STARTED
RECEIVE_ALLOCATE

MC_RECEIVE_AND_WAIT
MC_SEND_DATA
MC_DEALLOCATE (NONE)
TP_ENDED
```

Client TP Using Confirmation

```
TP_STARTED
MC_ALLOCATE (CONFIRM)
MC_CONFIRM

MC_SEND_DATA
MC_PREPARE_TO_RECEIVE

MC_RECEIVE_AND_WAIT

MC_CONFIRMED
TP_ENDED
```

Server TP - Confirmation

```
TP_STARTED
RECEIVE_ALLOCATE
MC_RECEIVE_AND_WAIT
MC_CONFIRMED

MC_RECEIVE_AND_WAIT
MC_CONFIRMED
MC_SEND_DATA
MC_DEALLOCATE (CONFIRM)

TP_ENDED
```

Internet protocols

Network layer: Internet Protocol (IP)

- provides generalized packet network interface
- handles routing through the Internet
- connectionless and unreliable

Transport layer: Transmission Control Protocol (TCP)

- provides a virtual circuit over which two processes can communicate
- supplies logic to give reliable, connection-oriented session
- FTP (file transfer) and HTTP are built on top of TCP

Transport layer: User Datagram Protocol (UDP)

- fast, unreliable, connectionless alternative to TCP
- SMTP (email) and Telnet are built on top of UDP

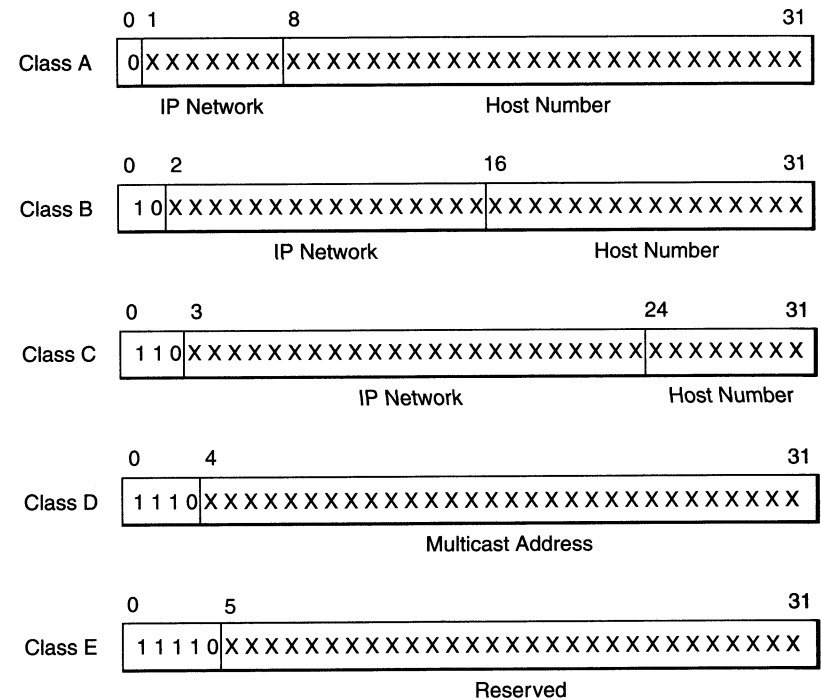# IP addresses

IP addresses are 32 bits long

10010011, 10000110, 00000010, 00010100

↓ *written as a dotted sequence*

147.134.2.20

divided into 5 classes

- class A: start with 0
  some reserved by InterNic

- class B: start with 10
  up to 65,535 host in subnetwork

- class C: start with 110
  up to 255 host in subnetwork

- class D: start with 1110
  used for multicasting

- class E: start with 11110
  reserved for future use

IPng (a.k.a. IPv6)

IP next generation
- address size increases from 32 to 128 bits
- colon-separated hex addresses replace dot-separated decimal
  e.g.,   FEAB:BC98:AB14:DB:1080::0:26
  can mix modes: 0:0:0:0:0:FFFF:147.134.2.20
- flexible address format replaces rigid class structure
- address allocation is decentralized
- IPng headers are simplified
- new extensions provide support for improved security

prefix determines the type of address
  e.g., FE: local use addresses,   FF: multicast addresses

IPng packet headers

```
 4 bits   version #6
28        traffic flow label
16        payload length
 8        header type
 8        hop limit
128       source address
128       destination address
```

Next week…

Internet communication protocols

- RPC/XDR
- Sockets
- TCP/IP

Review Chapter 10, read online articles

As always, be prepared for a short quiz