IP Address Assignment

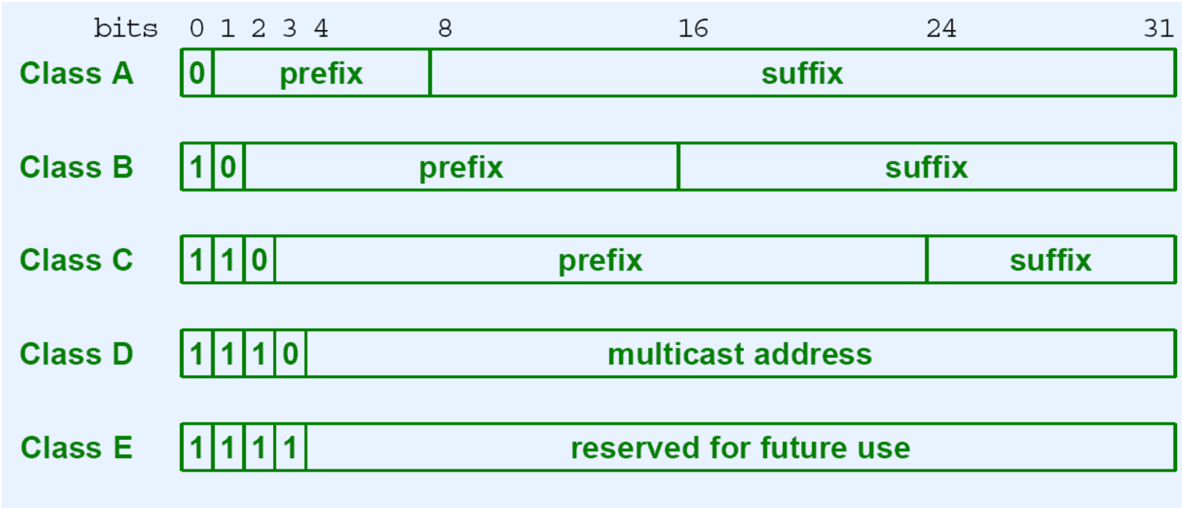
An IP address does not identify a specific computer. Instead, each IP address identifies a connection between a computer and a network. A computer with multiple network connections (e.g., a router) must be assigned one IP address for each connection.

IP Address Details

The user's local computer could be a nearby host but today is nearly always a desktop computer dedicated to a single user. A host is a multi tasking multi user computer to which various users connect using a terminal or terminal emulation program. All computing is performed inside the host and only keystrokes and characters to be displayed on the user's terminal screen are sent back and forth between the terminal and host. Today, dedicated hardware terminals (dumb terminals) are increasingly rare, having largely been replaced by terminal emulation programs running on desktop computers. A terminal emulation program uses the general purpose computer on which it is running, to act like a dedicated hardware terminal so that the host cannot detect the difference (dumb terminal or terminal emulation program.).

- Divided into two parts
  - Prefix identifies network
  - Suffix identifies host
- Global authority assigns unique prefix to network
- Local administrator assigns unique suffix to host

Original Classes Of Addresses



0 bit is msb

- Initial bits determine class
  (covers large range of network # and # of computers attached → trade off)
- Class determines boundary between prefix and suffix

Dotted Decimal Notation

- Shorthand for IP address
- Allows humans to avoid binary
- Represents each octet in decimal separated by dots
- NOT the same as names like www.somewhere.com

Example Of Dotted Decimal Notation

32-bit Binary Number	Equivalent Dotted Decimal
10000001 00110100 00000110 00000000	129 . 52 . 6 . 0
11000000 00000101 00110000 00000011	192 . 5 . 48 . 3
00001010 00000010 00000000 00100101	10 . 2 . 0 . 37
10000000 00001010 00000010 00000011	128 . 10 . 2 . 3
10000000 10000000 11111111 00000000	128 . 128 . 255 . 0

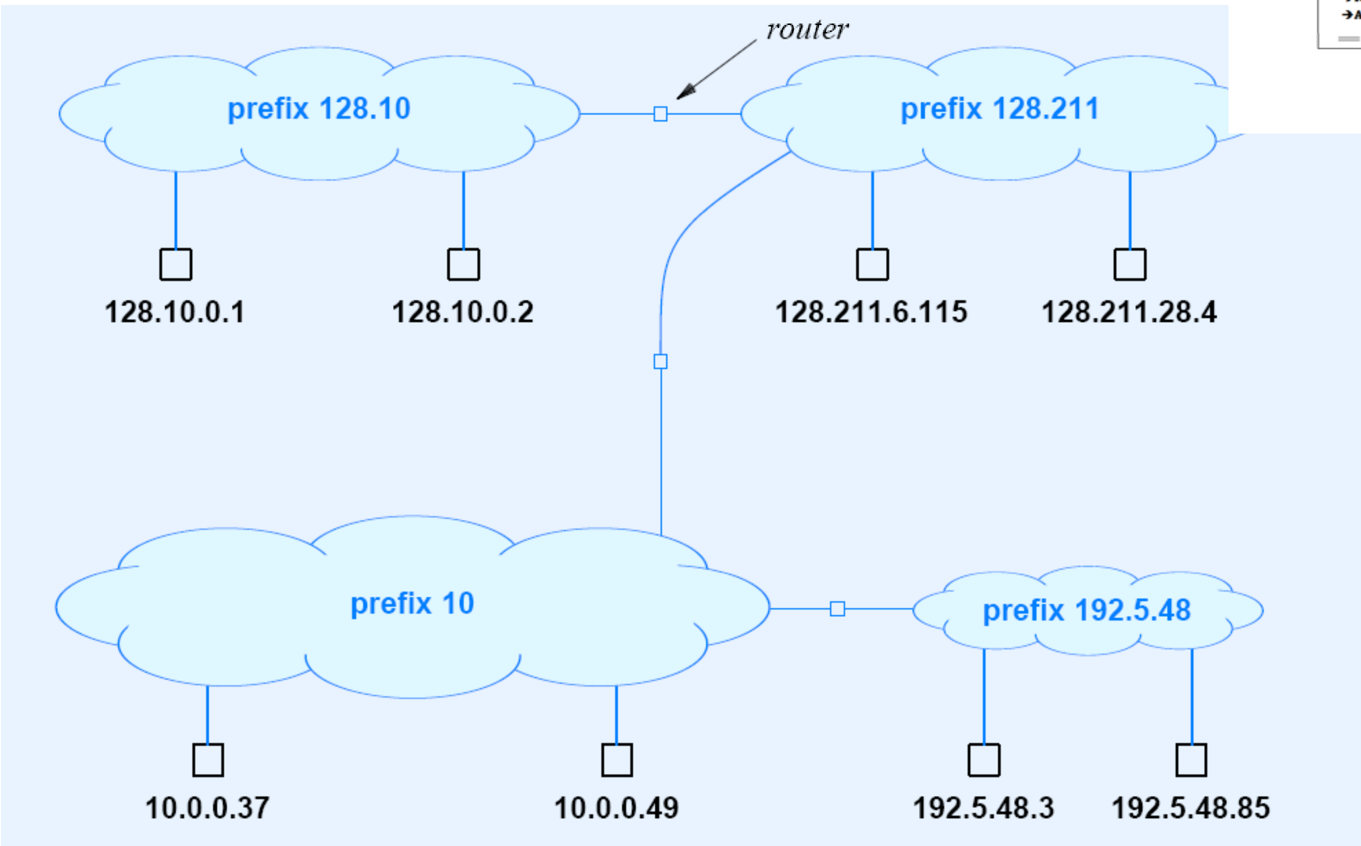
- Four decimal values per 32-bit address
- Each decimal number
  - Represents eight bits
  - Is between 0 and 255

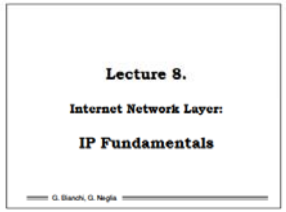
Classful Addresses And Network Sizes

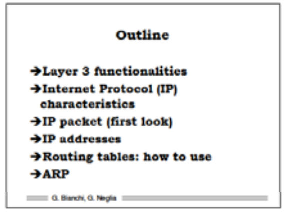
Address Class	Prefix Bits	Max Nets	Suffix Bits	Max Hosts Per Net
A	7	128	24	16777216
B	14	16384	16	65536
C	21	2097152	8	256

- Maximum network size determined by class of address
- Class A large
- Class B medium
- Class C small

Addressing Example







Subnet And Classless Addressing

- Not part of original scheme
- Invented to prevent address exhaustion
- Allow boundary between prefix and suffix to occur on arbitrary bit boundary
- Require auxiliary information to identify boundary
- Subnet = Classless addressing

Read Text (pp.294)

Address Mask

- Accompanies IP address
- 32 bit binary value
- Specifies prefix / suffix boundary
  - 1 bits cover prefix
  - 0 bits cover suffix
- Example: class B mask is

255 . 255 . 0 . 0

Subnet Addressing

- Goal: extend address space
- Invented in 1980s
- Works within a site
- Technique
  - Assign single network prefix to site
  - Divide suffix into two parts: network at **(within)** site (→ subnet) and host
    → (site – subnet – host)
- Typical use: divide class B address

Example Of Subnet Addressing

- Single Class B number such as 128.10.0.0 assigned to site
- Site chooses subnet boundary such as 24 bits

From msb to lsb, 0 → 31

- Routers and hosts configured with corresponding subnet mask

$$M = 255.255.255.0$$

- Given destination address, D, extract prefix with "logical and" operation

$$D \,\&\, M$$

Subnet is defined in the host address space

The subnet-host number t radeoff

Here's a table that let's you see at a glance the trade off between the number of subnets and hosts with different subnet masks for both Class B and Class C addresses. We've already subtracted two from the results in the last two columns to take the reserved network and host numbers into account.

Class B Subnetting:

# Mask Bits	Subnet Mask	# Subnets	# Hosts
2	255.255.192.0	2	16382
3	255.255.224.0	6	8190
4	255.255.240.0	14	4094
5	255.255.248.0	30	2046
6	255.255.252.0	62	1022
7	255.255.254.0	126	510
8	255.255.255.0	254	254
9	255.255.255.128	510	126
10	255.255.255.192	1022	62
11	255.255.255.224	2046	30
12	255.255.255.240	4094	
	14		
13	255.255.255.248	8190	6
14	255.255.255.252	16382	2

Class C Subnetting:

# Mask Bits	Subnet Mask	# Subnets	# Hosts
2	255.255.255.192	2	62
3	255.255.255.224	6	30
4	255.255.255.240	14	14
5	255.255.255.248	30	6
6	255.255.255.252	62	2

subnet mask

A mask used to determine what subnet an IP address belongs to. An IP address has two components, the network address and the host address. For example, consider the IP address 150.215.017.009. Assuming this is part of a Class B network, the first two numbers (150.215) represent the Class B network address, and the second two numbers (017.009) identify a particular host on this network.

Subnetting enables the network administrator to further divide the host part of the address into two or more subnets. In this case, a part of the host address is reserved to identify the particular subnet. This is easier to see if we show the IP address in binary format. The full address is:

10010110.11010111.00010001.00001001

The Class B network part is:

10010110.11010111

and the host address is

00010001.00001001

If this network is divided into 14 subnets, however, then the first 4 bits of the host address (0001) are reserved for identifying the subnet.

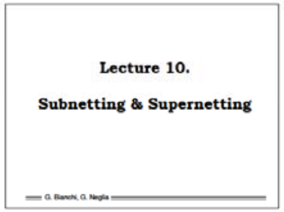
The subnet mask is the network address plus the bits reserved for identifying the subnetwork. (By convention, the bits for the network address are all set to 1, though it would also work if the bits were set exactly as in the network address.) In this case, therefore, the subnet mask would be 11111111.11111111.11110000.00000000. It's called a *mask* because it can be used to identify the subnet to which an IP address belongs by performing a bitwise AND operation on the mask and the IP address. The result is the subnetwork address:

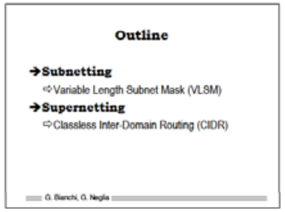
Subnet Mask	255.255.240.000	11111111.11111111.11110000.00000000
IP Address	150.215.017.009	10010110.11010111.00010001.00001001
Subnet Address	150.215.016.000	10010110.11010111.00010000.00000000

The subnet address, therefore, is 150.215.016.000.

Classless Addressing

- Goal: extend address space
- Invented in 1990s
- Works throughout Internet
- Accommodates
  - Original classful addresses
  - Subnet addresses
  - Other forms





Classless Addressing (continued)

- Technique
  - Allow arbitrary prefix size
  - Represent network address as pair
    (address, mask size)
- Known as Classless Inter-Domain Routing (CIDR)

CIDR Example in the TEXT(pp.295-296)

CIDR

- Uses slash notation (→ Appendix 3-pp.686)
- Example

Address Mask: 255.255.128.0

128.211.0.0/17

means that the boundary between prefix and suffix occurs after the first 17 bits.

- Each network can be as large or small as needed (power of two)

Special Addresses

IP address
@start up
as source address

No host
only network
itself

Prefix	Suffix	Address Type	Purpose
all-0s	all-0s	this computer	bootstrap
network	all-0s	network	network ID
network	all-1s	directed bcast	bcast on specified net
all-1s	all-1s	limited bcast	bcast on local net
127	any	loopback	testing

System start up

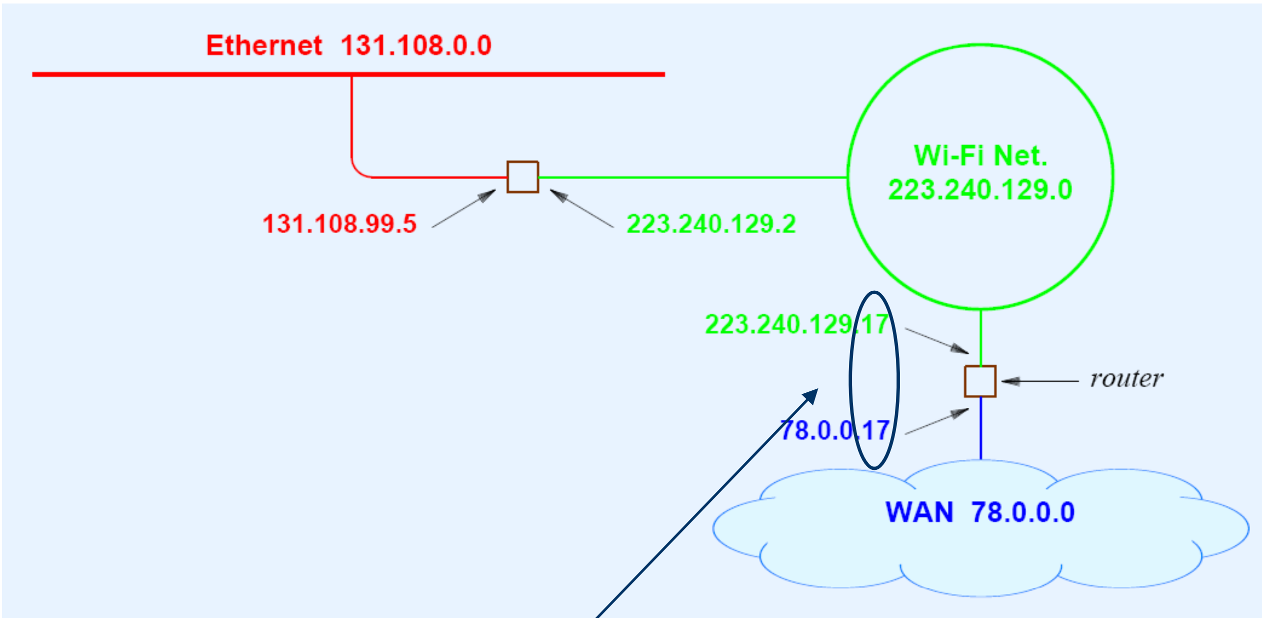
Broadcasting
without network No.

- Network address not used in packets
- Loopback never leaves local computer

(Two applications (one in local and the other in remote) run in local computer, The packet communicate through IP layer in local host.)

(network prefix 127/8 for loop back, 127.0.0.1 → host computer loopback)

Illustration Of Router Addresses



- Address prefix identifies network
- Need one router address per connection

(The suffix need not be the same but identical suffix is recommended.)

PART XII

Internetworking Part 4

(Transport Protocols, UDP and TCP, Protocol Port Numbers)

Transport Protocol

- Separate layer of protocol stack
- IP can't distinguish multiple application program running → Transport Protocol serves an application as an end point
- Conceptually between
  - Applications
  - IP

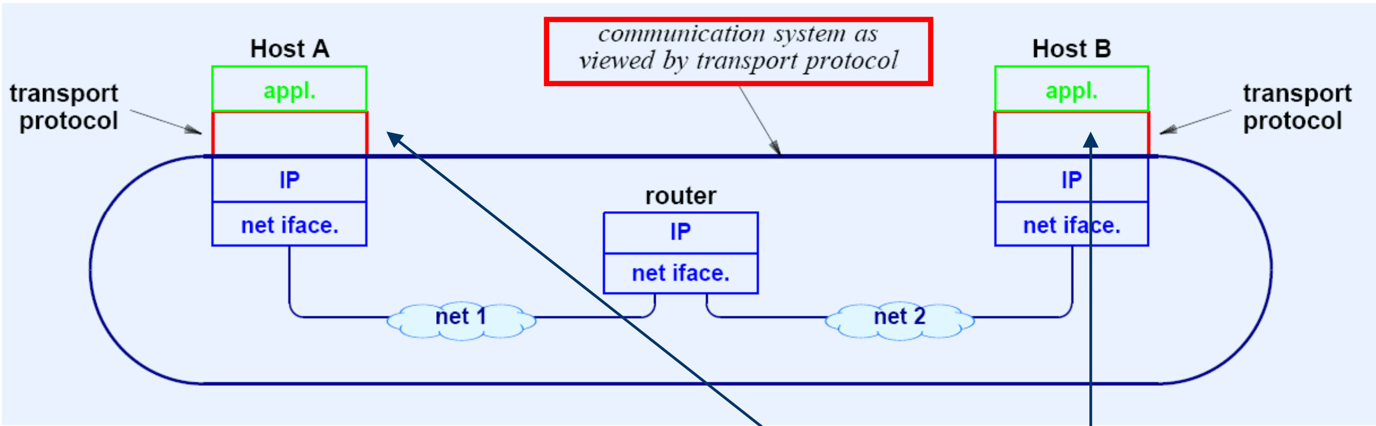
Terminology

- IP
  - Provides computer-to-computer communication
  - Source and destination addresses are computers
  - Called machine-to-machine
- Transport protocols
  - Provide application-to-application communication
  - Need extended addressing mechanism to identify applications
  - Called end-to-end (protocol)

Transport Protocol Functionality

- Identify sending and receiving applications
- Optionally provide
  - Reliability
  - Flow control
  - Congestion control
- Note: not all transport protocols provide above facilities

Relationship Between Transport Protocols And Other Protocols



- Transport protocols are end-to-end
- Transport protocol on one computer uses IP to communicate with transport protocol on another computer

Two Transport Protocols Available

- Transmission Control Protocol (TCP)
- User Datagram Protocol (UDP)
- Major differences
  - Interface provided to applications
  - Underlying functionality

User Datagram Protocol

- Lightweight transport
- Becoming more popular (IP telephony)
- Best-effort delivery

UDP Features

- End to End

  (Protocol Port No. → locally mapped to program identifier local OS uses.)

- Connectionless service

  (No need to pre-establish or terminate network, arbitrarily long time between messages, Low control message – Low overhead)

- Arbitrary interaction

  (1-to-1, 1-to-many, Many-to-1, Many-to-many)

UDP Features (Continued)

- Message-oriented interface

  (doesn’t divide message for transmission nor combine message for delivery, UDP preserves data boundary)
- Best-effort semantics (= IP best effort)
- Each message encapsulated in IP datagram
- Uses protocol ports to identify applications

  (doesn’t depend on identifiers used by the local OS)

UDP Details

- Accepts and delivers messages
  - Message received is exactly same as message sent
  - Boundaries preserved
- Maximum message size approximately 64K octets
  - No connection overhead
  - No state information maintained

UDP Semantics

- Same best-effort semantics as IP (i.e., unreliable transfer)
- Message can be
  - Lost
  - Duplicated
  - Delayed
  - Delivered out of order
- Works best in LAN applications

Interaction With UDP

- UDP allows communication that is
  - 1-to-1
  - 1-to-many
  - Many-to-1
  - Many-to-many
- Application programmer chooses

Packet Delivery

- UDP can support
  - Unicast
  - Multicast
  - Broadcast

(By IP multicast or broadcast)

Def. of Datagram

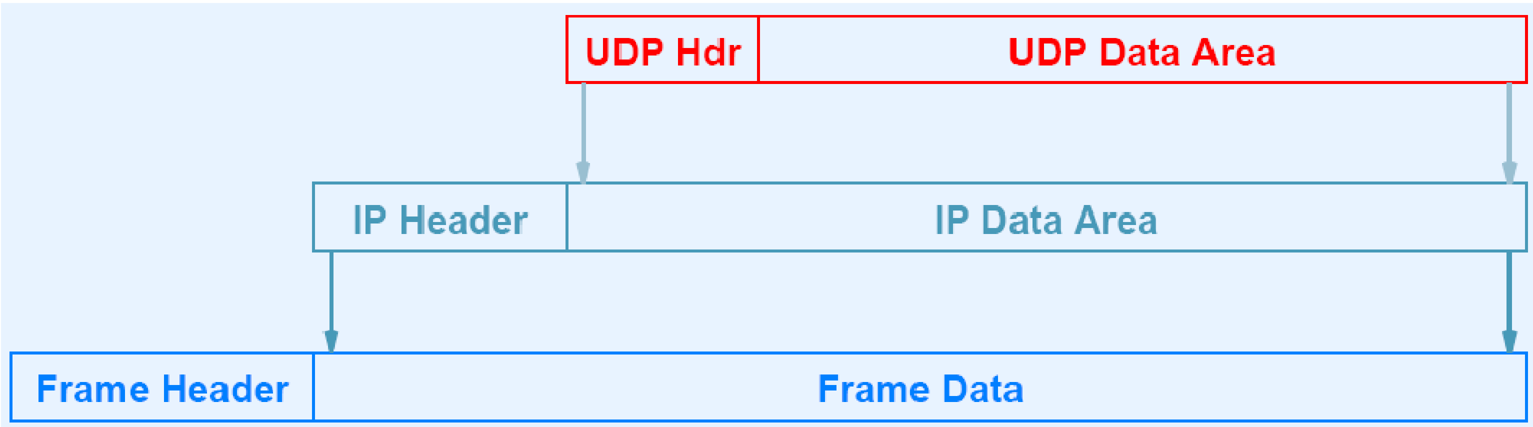
- A datagram is, to quote the Internet's Request for Comments (RFC) 1594, "a self-contained, independent entity of data carrying sufficient information to be routed from the source to the destination computer without reliance on earlier exchanges between this source and destination computer and the transporting network."

User Datagram Format

0	16	31
UDP SOURCE PORT	UDP DESTINATION PORT	
UDP MESSAGE LENGTH	UDP CHECKSUM	
DATA		
...		

- Extremely small header (called thin layer)
  (UDP source port: protocol port number of sending application)
- Checksum optional
  (Checksum can contain pseudo header – source, destination, type IP field to make sure no possibility of transmission error by omitting the above IP field)

UDP Encapsulation



- Two levels of encapsulation
- UDP datagram size cannot exceed maximum IP payload

Identifying An Application

- Cannot extend IP address
  - No unused bits
- Cannot use OS-dependent quantity
  - Process ID
  - Task number
  - Job name
- Must work on all computer systems

Identifying An Application (continued)

- Invent new abstraction
  - Called <u>protocol port number</u>
  - Used to identify sending or receiving application unambiguously
  - <u>Independent of underlying operating system</u>
  - Used only with TCP/IP protocols

Protocol Port Numbers

- Server
  - Follows standard
  - Always uses same port number
  - Uses low port numbers
- Client
  - Obtains unused port from protocol software
  - Uses high port numbers

DNS: an Internet service that translates ***domain names*** into IP addresses

Protocol Port Example

- Domain name server (DNS) application is assigned port 53
- Application using DNS obtains port 28900
- UDP datagram sent from application to DNS server has
  - Source port number 28900
  - Destination port number 53
- When DNS server replies, UDP datagram has
  - Source port number 53
  - Destination port number 28900

Transmission Control Protocol (TCP)

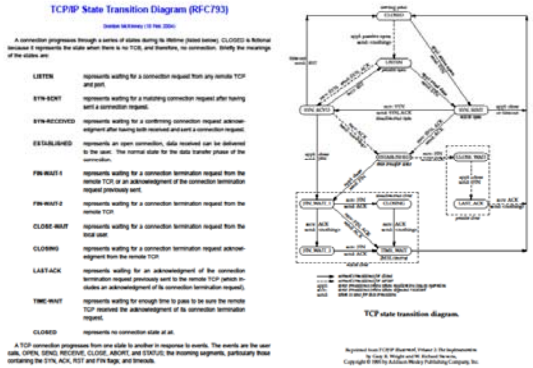
- Major transport protocol used in Internet
- Heavily used
- Completely reliable transfer

TCP Features

Streaming: The client browser or plug-in can start displaying the data before the entire file has been transmitted. For streaming to work, the client side receiving the data must be able to collect the data and send it as a steady stream to the application that is processing the data and converting it to sound or pictures. This means that if the streaming client receives the data more quickly than required, it needs to save the excess data in a buffer. If the data doesn't come quickly enough, however, the presentation of the data will not be smooth.

- Connection-oriented service
  (Request then use the connection)
- Point-to-point
- Complete Reliability
- Full-duplex communication
- Stream interface
- Stream divided into segments for transmission
- Each segment encapsulated in IP datagram
- Uses protocol ports to identify applications
- Reliable Connection Set Up
  (No duplicate packet in the previous connection)
- Graceful Connection Shutdown
  (Deliver all before shutdown)

TCP Feature Summary



TCP provides a completely reliable (no data duplication or loss), connection-oriented, full-duplex stream transport service that allows two application programs to form a connection, send data in either direction, and then terminate the connection.

Apparent Contradiction

- IP offers best-effort (unreliable) delivery
- TCP uses IP
- TCP provides completely reliable transfer
- How is this possible?

Achieving Reliability

- Reliable connection startup
- Reliable data transmission
- Graceful connection shutdown

(Challenge:
Duplicate packet from old connection,
Computer Reboot)