TCP/IP

IP Addressing

minimum requirements (addressing) for communicating locally: (Physical segment/ same)

(IP Address)

Network ID	Host ID

+

[(subnet mask =) identify which portion of IP Address is the network ID and which portion is Host ID]

+

[(default gateway =) IP address of Router on the same physical segment)]

for remote access (different physical segments / subnet)

* All devices on the same physical segment share a common network ID.
* each physical segment has a unique network ID.

— IP addressing modes —

public:
- registered internet IP addresses
- can connect directly to the internet
- in the current organization, ARIN to ISP, ISP to companies

private:
- non-registered internet IP addresses
- RFC 1918 - 10.0.0.0
- cannot connect directly to internet
- need Network Address Translation (NAT) or proxy services to connect to internet (included in some routers, proxy servers or firewalls)

— IP address = — 32-bit binary number

11000000.10101000.01110001.00010011

11000000.10101000.01110001.00010011 (divided into 4, 8-bit octets)

192 . 168 . 113 . 19 (converted to decimal numbers)

(decimal range of an octet: 0-255)

— it contains the device's: Network ID and Host ID

ex: 192.176.11 . 201

No.

Date

Classes of IP addressing :—

Class A = (Network ID . ~~Network~~ Host ID . Host ID . Host ID) = (1/127.---.---.---)

In Binary, any address that starts with a "0" in the first bit!

{ first address: 00000001.00000000.0000 0000.00000000 = 1.0.0.0
{ last address: 01111111. 00000000. 00000000. 00000000 = 127.0.0.0 (Loopback address)

✓ 1-126 = 126 possible class A Network IDs (because the last 127 is reserved for broadcast purposes)

✓ 256 * 256 * 256 = 16,777,216 (minus 2) ~~for~~ = 16777214 (number of host IDs)

(minus 2) ≡ { host IDs can not be all 1's (reserved for broadcast address)
" " " " " " 0's (" " "this network only" address

example:

33.0.0.1 — 33.0.0.255 (255 addresses)
33.0.1.0 — 33.0.1.255 (256 ")
33.0.255.0 — 33.0.255.255 (256 ") } A total of 65535 Addresses
33.1.0.0 — 33.1.255.255 (65536 addresses)
33.2.0.0 — 33.2.255.255 (65536 ")
33.255.0.0 — 33.255.255.254 (65535 addresses)

Class B = (Network ID . Network ID . Host ID . Host ID) = (128/191 .---.---.---)

In binary, any address that starts with a "10" in the first two bits of first octet

{ first address: 10000000. 00000000. 00000000. 00000000 = 128.0.0.0
{ last address: 10111111. 11111111. 00000000. 00000000 = 191.255.0.0

Class C = (Network ID . Network ID . Network ID . Host ID) = (192 - 223 . ...)

In binary, any address that start with "110" in first three bits of first octet.

first address: 110 00000 . 0000 0000 . 00000000 . 0000 0000 = 192.0.0.0

last address: 110 11111 . 1111 1111 . 11111111 . 11111111 = 223.255.255.0

Class D & E : (reserved)

Class D = { used by Multicast applications
shared addresses
224.0.0.0 — 239.255.255.255 }

Class E = { experimental
240.0.0.0+ }

= What makes up a Subnet Mask (SNM)?

In binary, 1's represent what portion of IP address is the Network ID.

0's represent what portion of IP address is the Host ID.

for example:

207.23.106.99 (class C address)

⇒ Network. Network. Network. Host

⇒ 11111111. 11111111. 11111111. 00000000

⇒ 255. 255. 255. 0 (subnet mask)

similarly, class B subnet mask = 255.255.0.0

class A " " = 255.0.0.0

= ARP (Address Resolution Protocol)

example. RFC 826

- Maps a MAC (Hardware) (Physical) Address to a destination Host's or router's IP Address.

Source (computer) Host retrieves — (obtains) MAC address from cache or with a local broadcast.

- ~~MAC address places in the~~
- Host places MAC address in the address field of data-link frame (Ethernet, TokenRing, etc.)

* MAC/Physical/Hardware address is a unique address, 6 byte address (in most cases), burnt into network interface card by the manufacturer of network interface card

= local or Remote algorithm =

Before the data can be transferred over the network, a TCP connection must be established/setup -

- IP address of destination computer is needed and that is obtained by (DNS, WINS, Broadcast, etc.) server/site.
- once IP address is obtained, it is analyzed to determine if it is local or remote, by source host
  - ≒ comparing destination's network ID to its (source Host) network ID with subnet mask

if (destination host's network ID = Source Host's network ID)
then [it is local (same physical segment/subnet)]
and [further, it checks whether destination host's MAC address is in its cache. If not, it starts a local broadcast to obtain it and cached it for reuse.]

else it is remote
source host checks its local route table for a special route to the network of destination host, if there is, the source computer checks its ARP cache for a cached MAC address of the router associated with the router in local route table.
and if router's MAC address is not in cache, a local broadcast is initiated to obtain routers MAC address and it is cached for reuse.

No.

Date

to obtain MAC address from remote host—

if there is no special route in the source host's local route table, source computer checks its ARP cache for a cached MAC address of the default gateway.

if it is not there, a local broadcast is initiated to obtain and cache it for reuse.

= Why custom subnet masks?

- All devices/hosts on the same physical segment must have the same network ID.

to correctly address multiple physical segments, we must divide the single network ID into sub-networks or "subnets", and these are created with custom subnet masks.

((a subnet is a portion or subdivision of the IP addresses that are associated with an assigned network ID.))

((The range of IP addresses included in a subnet is determined by the subnet mask.))

= Why subnets?

- hardware specifications (standard, vendor, ...)
- network performance
- geographic
- different topologies
- routers

= Creation of Subnet IDs =

example: (Network ID . Network ID . Host ID . Host ID) → default SNM

⇒ Network . Network . SN-ID . Host) → custom SNM

(subnet ID)

- each physical segment of network has a unique subnet-ID and the subnet ID is common to all hosts on a physical segment
- each host on the network has a host ID unique to its subnet ID
- choose (custom) subnet mask that creates the number of:
  - subnet-IDs >= physical segments
  - Host IDs per subnet-ID >= Hosts per physical segment

example: 152.77.0.0 { with 5 physical segments that each have a maximum of 5000 host on each segment

Net . Net . Host . Host

↳ 11111111 . 11111111 . 00000000 . 00000000

↳ Net . Net . SNID Host . Host → Custom SNM

			SNM	#of SN-IDs	#of Hosts/SN ID
11111111 .	11111111 .	10000000 .	00000000	SNM	#of SN-IDs	#of Hosts/SN ID
		10000000	00000000	128	2-2=0	32768-2
		11000000	00000000	192	4-2=2	16384-2
OK →		11100000	00000000	224	8-2=6	8192-2
		11110000	00000000	240	16-2=14	4092-2
		11111000	00000000	248	32-2=30	2048-2

use a custom subnet mask of

⇒ 255 . 255 . 224 . 0 (all devices)

No.
Date

== One bits in a subnet mask =

- the 'one' bits must be contiguous, left to right in SNM, no embedded zeros

ex:' 1111 1111. 11111111. 1111 0000. 00000000 (OK)
1111 1111. 11111111. 1100 1100. 00000000 (wrong)

- subnet masks are expressed as a "/20" or some other "/number" alternatively.

ex:- for (152.77.32.0/20) ~~=~~

here, the "/20" = the number of '1' bits in SNM

so, in SNM, it will be (11111111.11111111.1111 0000.0000 0000) ⇒
(20 '1's)

⇒ ⇒ (255.255. 240. 0)

= (Minus 2) ! = why ? =

- { * the first host address is "zeros" i.e., all address is zeros.
* the last " " "ones" i.e., all " " ones. }

(both are invalid.)

- number of subnets

[don't (minus 2) if all hardware & software supports RFC 1812]

(All current stuffs (devices & S/w) have RFC 1812 now.

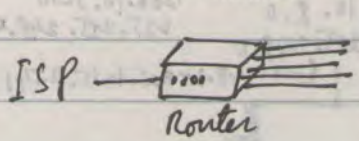
ex:

= determining (Subnet IDs) SNM = 224 ⇒

152.77.0.0 Network Address

255.255.224.0 Custom SNM

1110 0000. 00000000 (Subnet mask) SNID/Host ID (last 2 octets-Binary)	Subnet ID (decimal conversion)
000 00000. 0000 0000 (invalid)	152.77.0.0
001 00000. 0000 0000 (First SNID)	152.77.32.0
010 . . . . . (Second ")	152.77.64.0
011 - - - - - - - - (III "	152.77.96.0
100 - - - - - (IV ")	152.77.128.0
101 - - - - - (V ")	152.77.160.0
110 - - - - - (VI ")	152.77.192.0
111 - - - - - ( ) invalid	152.77.224.0



152.77.0.0 / 255.255.224.0

255.255.1110 0000.00000000	subnet mask	IP Address
152.77.00100000.00000000	First subnet ID	152.77.32.0
152.77.00100000.00000001	First Host ID	152.77.32.1
152.77.00100000.00000010	second Host ID	152.77.32.2
. . . . .	. . . . .	. . . . .
152.77.00111111.11111110	Last Host ID	152.77.63.254
152.77.01000000.00000000	second subnet ID	152.77.64.0
152.77.01000000.00000001	First Host ID	152.77.64.1
. . . . .	. . . . .	. . . . .
152.77.01011111.11111110	Last Host ID	152.77.95.254
	Third subnet ID	

No.

Date

CIDR (Classless Interdomain Routing) is used —

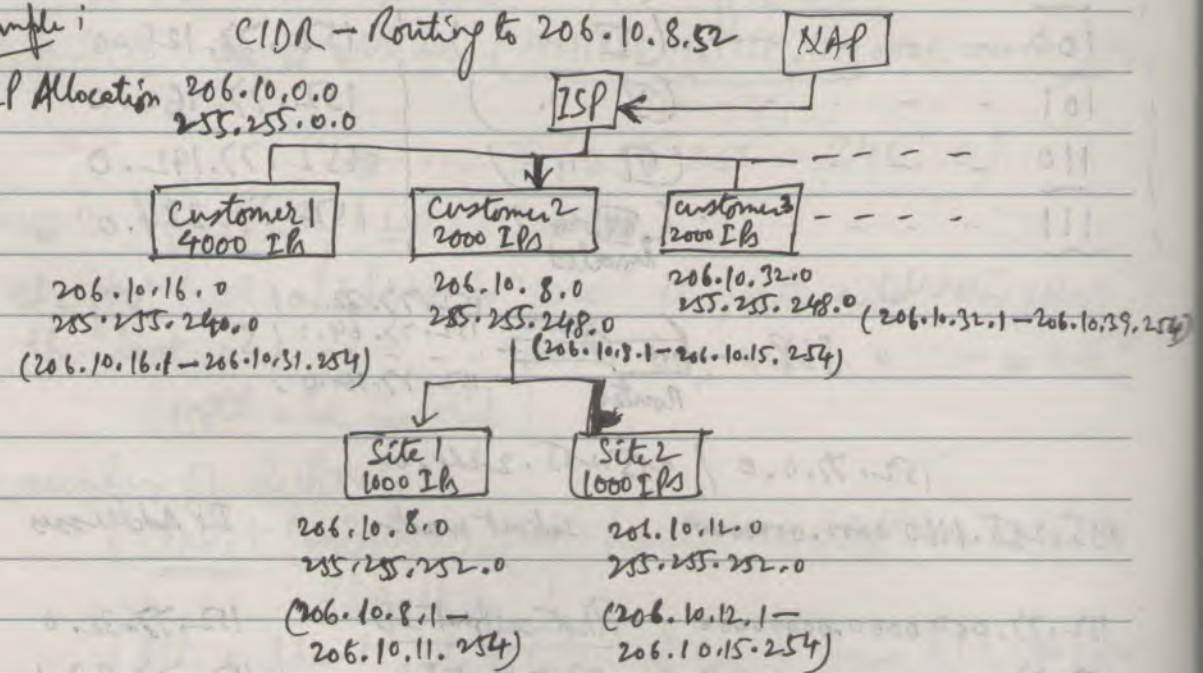
→ due to inefficient use of IP address space, proliferation of routes

~~forgets~~ * A large number of networks are represented/summarized as a single route.

= Classless numbering =

- forgets address classes (A, B, C)
- everything depends on the subnet mask (subnetting)
- one hop at a time

example:

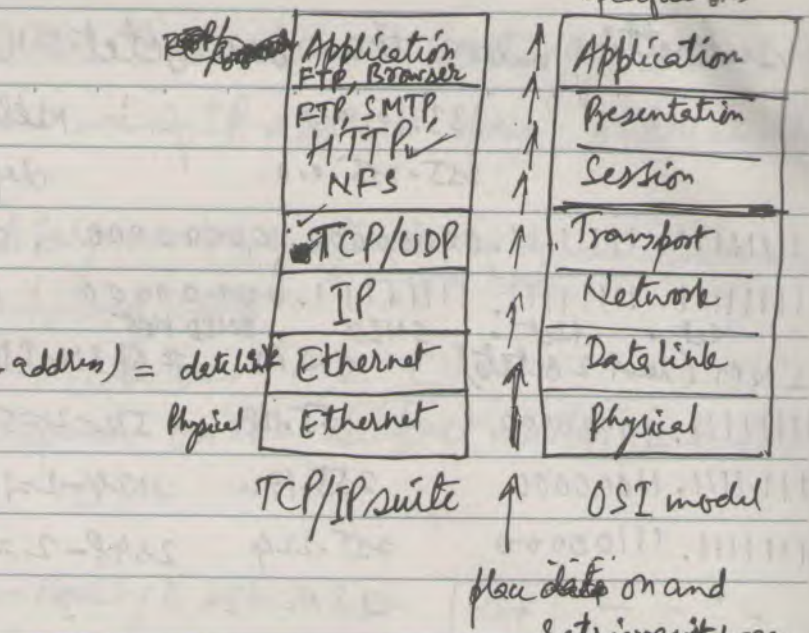


= subnetting more than one octet: Class B chart = (example)

133.10.0.0	Network address
255.255.0.0	default SNM
11111111.11111111.00000000.00000000	SNM (Binary)
11111111.11111111.11111111.00000000	Custom SNM

SNM (last 2 octets)	Net. Net. SNID SNID Host SNM	# of SN-IDs	# of Hosts per SN-ID
11111111.10000000	255.128	512-2=510	128-2=126
11111111.11000000	255.192	1024-2=1022	64-2=62
11111111.11100000	255.224	2048-2=2046	32-2=30
. . .	. . .	. . .	. . .
11111111.11111110	255.254	invalid	
11111111.11111111	255.255	invalid	

Open systems Interconnect (OSI) model - ISO specifications



✓ physical layer: specification for physical components of network (i.e. cabling, interconnection methods, distances)

✓ data link layer:- media Access control (MAC / hardware) and logical link control (LLC),
- address network devices (MAC address)
- place data in a data frame (packet)
- gives the data bits to NIC card (MAC)
- controls the NIC's access to the media (MAC)
- detects transmission errors (LLC)

example: NIC card and driver

network layer:- network addressing
- move packets between different networks - routing

example: TCP/IP software

Transport layer:- receives data from application and segments it ✓
- can provide transmission error detection and correction

example: → TCP/IP software

Session layer:- manage and track user sessions (ongoing communication)

Presentation layer:- make sure the receiving station can read the transferred data (data formatting)

Application layer:- gives end-user application access to network resources

(TCP/IP) Internet suite of protocols

Application
Transport
Internet
Network

internet/DOD model

protocols = - ARP (address resolution protocol) (used to discover MAC address)
- IP (internet protocol) (provides addressing and routing), (connectionless)
- TCP (transmission control protocol) provides (three way handshake) (agreement/acknowledgement) connection-oriented, reliable data delivery
- UDP (user datagram protocol) provids connection-less data delivery
- DNS (domain naming servers) protocol, retrieves IP address when user enter URLs.
- HTTP (hyper text transfer protocol) provides web browsing
- HTML (protocol provides page formatting)
- others: ICMP, SMTP, POP3, DHCP, FTP, SNMP, BGP, RIP, etc.

{Ethernet / (IEEE 802.3)} = (LAN technology for transferring data 10 Mbs to 1 Gbs.) = (OSI model's physical and data link layers specifications)
- connectionless

(calling & connection) x Baseband = 10 Mbs
eg. 10 Base 5 = 10 Mbs Base 5

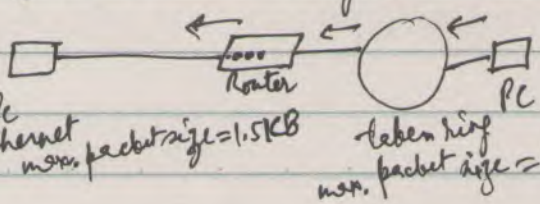
ethernet interconnection methods: Bus (cheap, but not fault tolerance)

ethernet interconnection method: star (more expensive but fault tolerant uses hubs

ethernet addressing (MAC/hardware/NIC addressing)
- 6 byte address:
  - first 3 bytes are organization unique identifier (OUI) assigned by to manufacturer

---

IP packet fragmentation = a single IP packet can be subdivided by a router into multiple packets during data transmission to accommodate destination network (which is determined by data link layer)



No.

Date

Ports – a method for applications to identify themselves to the TCP and UDP programs.

(multiple applications use the same TCP program)
e.g., DNS, FTP, webserver, etc...

✓ (each application is assigned a unique port number by TCP)

(possible 65536 port numbers)

Server side ports = (RFC1700 reserves the first 1000 port numbers for server side applications)

e.g.: webserver port 80
DNS port 53
SMPT port 25

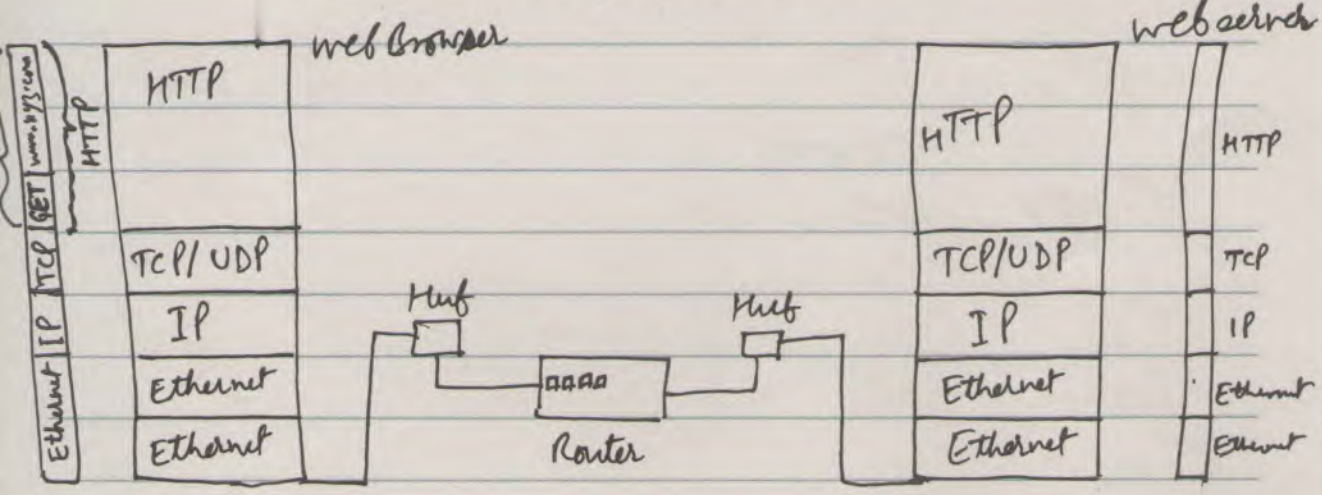
client side ports = (numbers assigned to client network applications as they initialize)

– (starting at 1024)

OSI Computer : Building packets

client	OSI	TCP/IP	Server
web browser (HTTP/HTML)	APPLICATION	HTTP/HTML	web server (HTTP/HTML)
	Presentation		
	Session		
TCP/IP Software	Transport	TCP/IP	TCP/IP Software
TCP/IP Software	Network	IP	"
NIC Card & Driver	data link	ethernet	NIC card & Driver
" "	physical	ethernet	NIC card driver

end to end communication



Network commands =

① ipconfig /all

② tracert 142.58, 224.88

(tracing route to ...)

Web Server (Serving dynamic web pages)