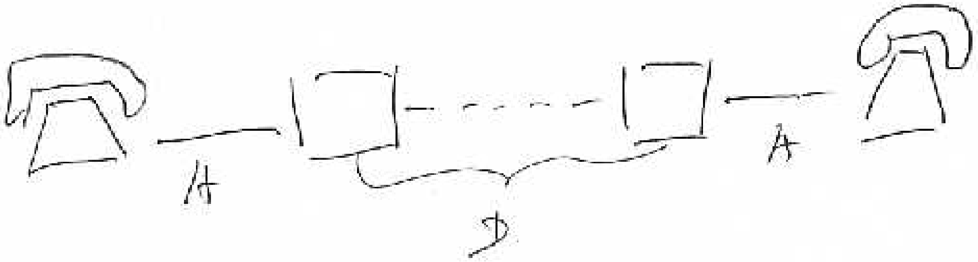
Old fashioned telephony
(POTS, PSTN)



Circuit.
Set-up,
tear-down.

D: Digital. PCM
64 kb/s. (56 kb/sec).

<u>bit</u> synchronized.
64 kb (56 kb) whether you use it
or not.

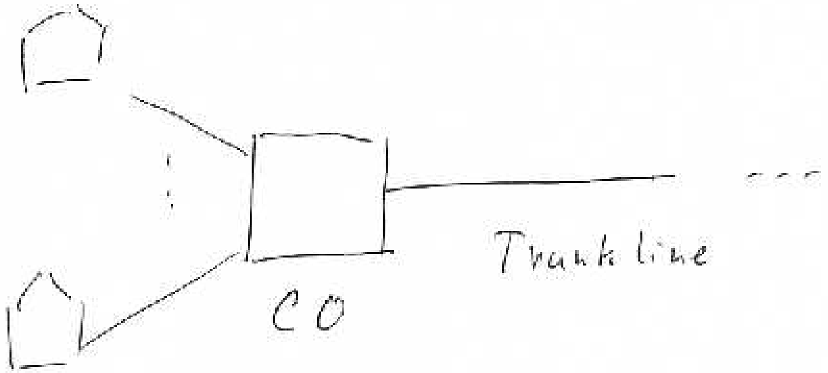
ATM. <u>cells</u>. Ci: (virtual) circuit.
set-up. tear down

But Rate is <u>variable</u>

Each VC: max rate,
but below: variable.
send data "only when something to send".
<u>Statistical</u> multiplexing.

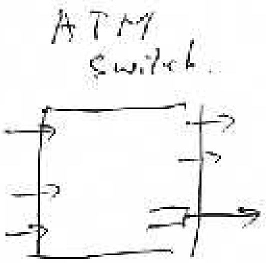
"Statistical Multiplexing":

(1) Telephony.



The number of calls that want to use a trunkline "varies statistically".
It is "engineered" to be able to take all comers "most of the time". ("No Buffer").
Occasionally: "Fast Busy".

(2) ATM



VPs, VCs that uses output port varies statistically.
Each: "load" varies statistically.
In addition: Buffer. | SVC can be refused.
Rate Control per VC/VP.
Last: drop from buffer.

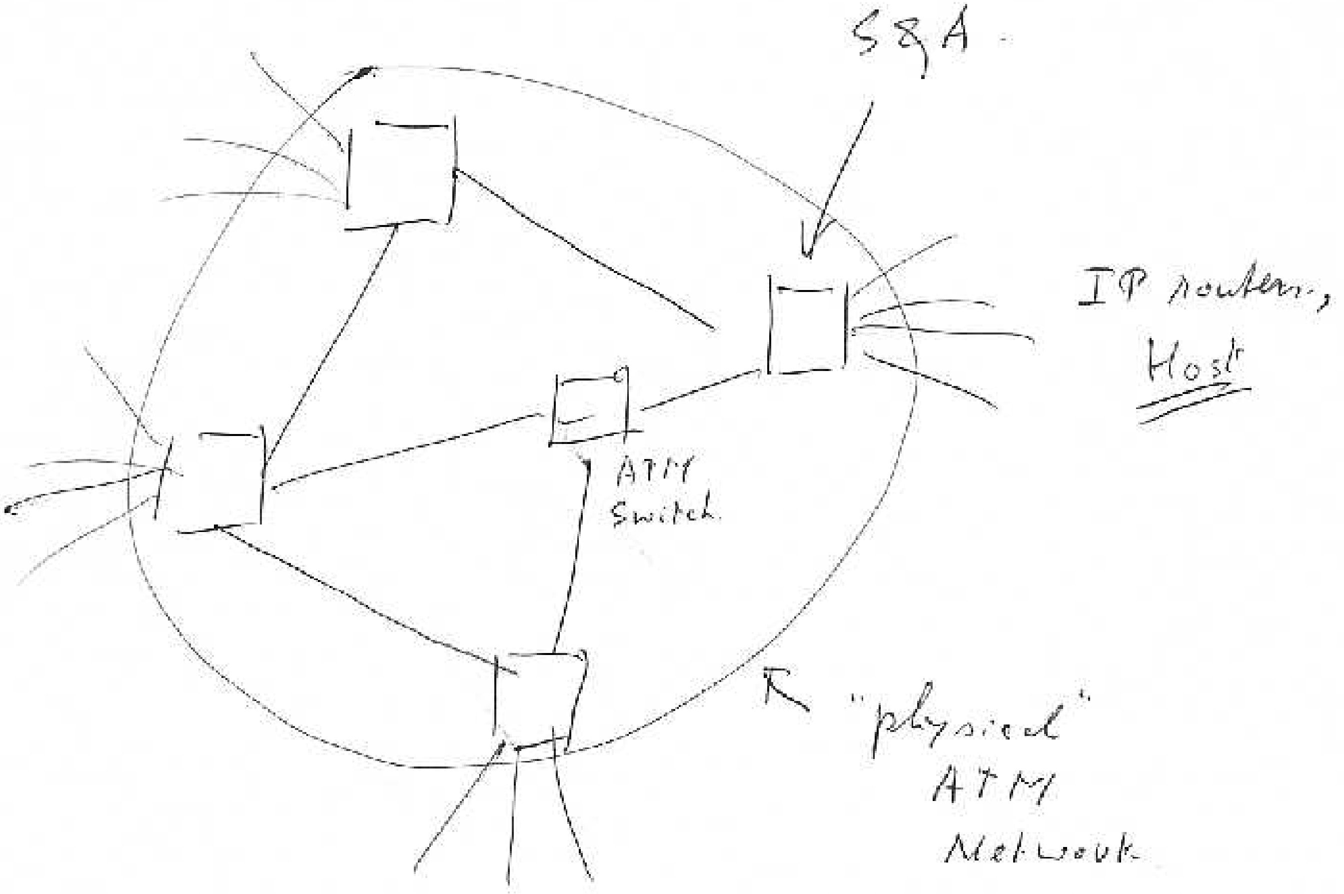
~~[illegible]~~ IP over ATM.

IP packet: <u>segmented</u> a into "swallow sized pieces".
(bite sized,
not byte-sized!).

AAL-5: up to 48 ~~[illegible]~~ bytes per ATM cell.

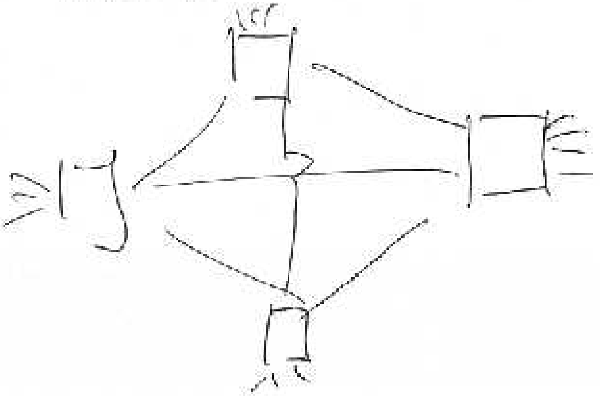
(~~Non~~
AAL: ATM Adaptation Layer.

other AALs: some of the ATM-data cells are "stolen" for control.



S&A : Segmentation and Re-assembly.

~~The AT~~
To IP, the ATM network looks like a fully connected network



From the point of view of the (IP) ~~xxx~~ routers and hosts:

The ATM network looks like "one giant LAN" (WAN, MAN).

"Like" ethernet. But:

No ~~bra~~ broadcast is possible.

ATM is an NBMA. Network.
Non-Broadcast Multiple Access

ATM has a number of "Adaptation Layers", to accomodate other services.

ATM Adaptation Layer. AAL.

AAL 1: CBR. "like Telephony".

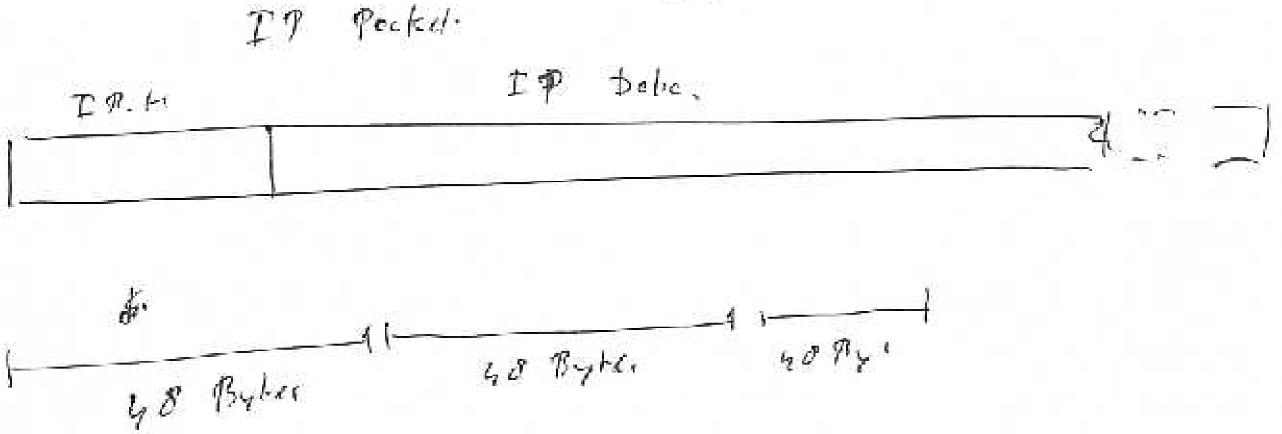
AAL 2: for VBR, "synchronous".
standard never completed.

AAL 3/4: for VBR, "asynchronous".
(could be used for IP over ATM)

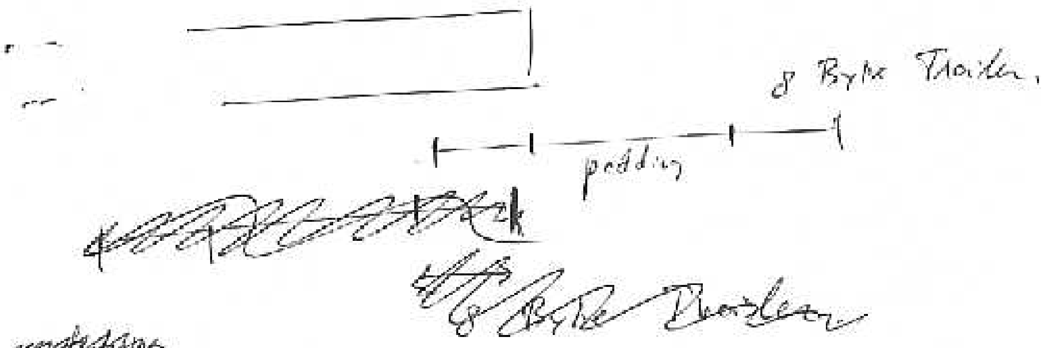
AAL5: "Classic" IP over ATM
LAN emulation. (LANE).

IP over ATM with AAL 5:

IP over ATM with ~~IP~~ AAL5.



This is one of the two AAL5 methods.



The other uses the first 8 bytes of each cell for "bookkeeping":
LLC & SNAP.
LLC: logical link control
SNAP: Subnetwork Attachment Point
contains Type ~ ethernet type

8 Byte Trailer: gives length. (in bytes) of "IP packet" (or other payload).

Comer p. 360

Trailer

8 bit UU	8 bit CPI	16 bit length	32 bit CRC

Type

UU: any value
CPI: must be zero

16 bit length.
$0 \leq \cdots \leq 2^{16}-1 = 65535$

But

240 Bis
after 240.

Use of "SNAP Header".

(optional).

If more than one protocol (IP, ~~A~~ IPX, ..) can be sent over on VC, the protocol must be identified.

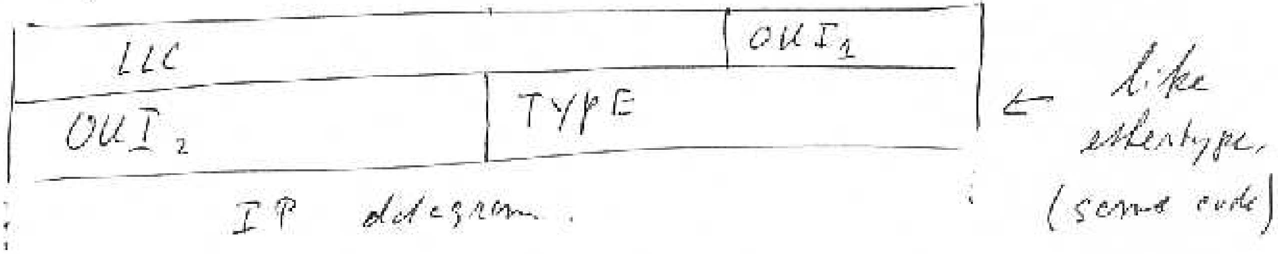
SNAP header.

or: LLC + SNAP

LLC: Logical Link Control

SNAP: Sub Network Attachment Point.

in front of Packet



TYPE for IP v4: 0800 (etc.)

With SNAP header, a K-byte IP packet becomes $K + \underset{\underset{\text{SNAP}}{\uparrow}}{8} + \underset{\underset{\text{Trailer}}{\uparrow}}{8} = K+16$ Bytes.

A "clean TCP ACK": 40 Bytes becomes 56.
needs 2 ATM cells!

~~57~~
241.

ATM has an MTU of 9180.

(Total ~~size~~ length ≤ 9180).

(not $2^{13} = 8192$ (!))

~~SAR Segmentation & Re-assembly.~~

SAR Segmentation & Re-assembly.

Last bit of "Payload Type" in ATM header is set to 1 for "end of block".

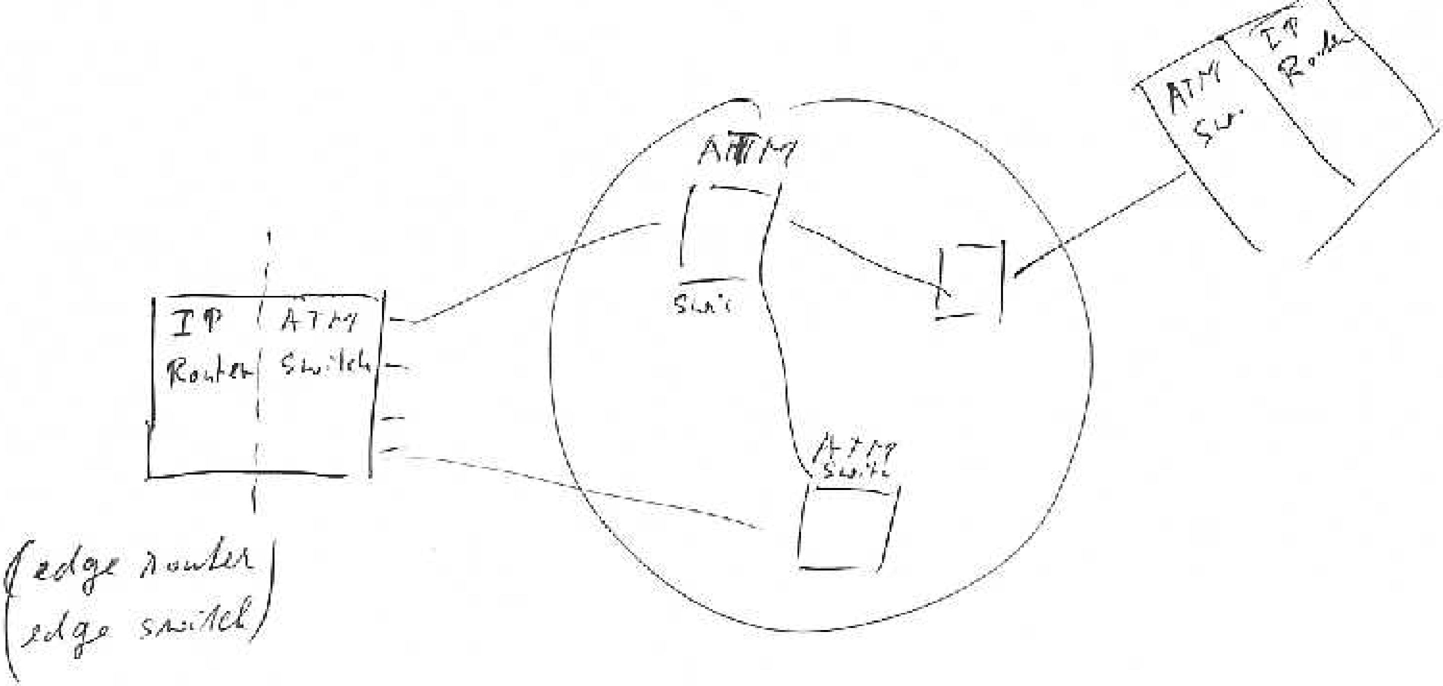
block: need not be IP packet.

The "dest" keeps previous packets, computes padding (must be all zeros), regenerates packet.

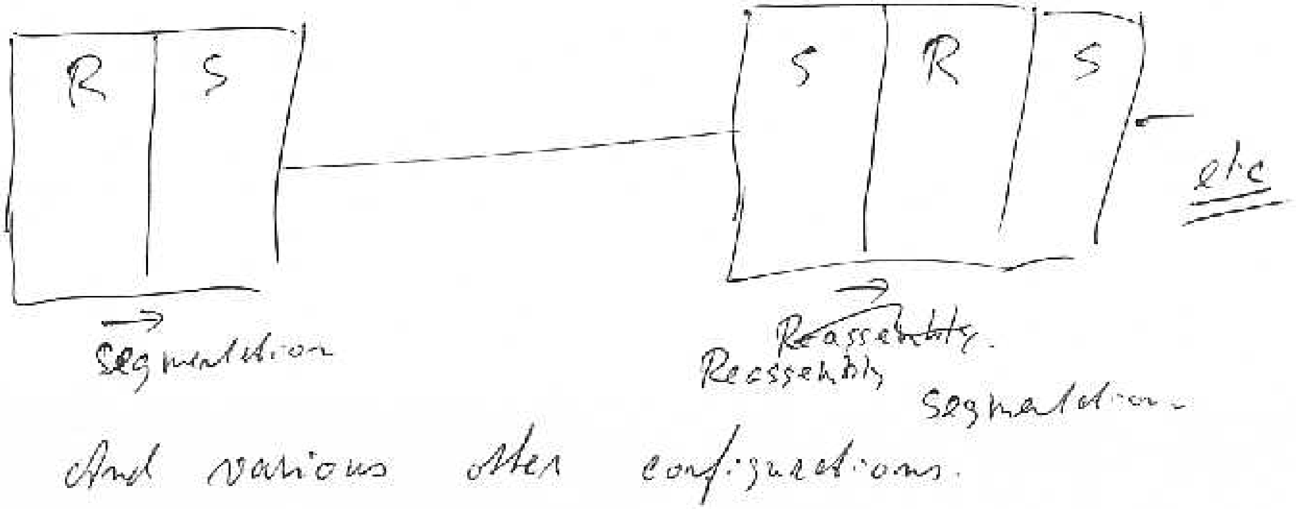
~~P~~ IP packet > 9180 bytes:

first IP fragmentation to 9180 bytes.

then AAL 5 on each fragment.



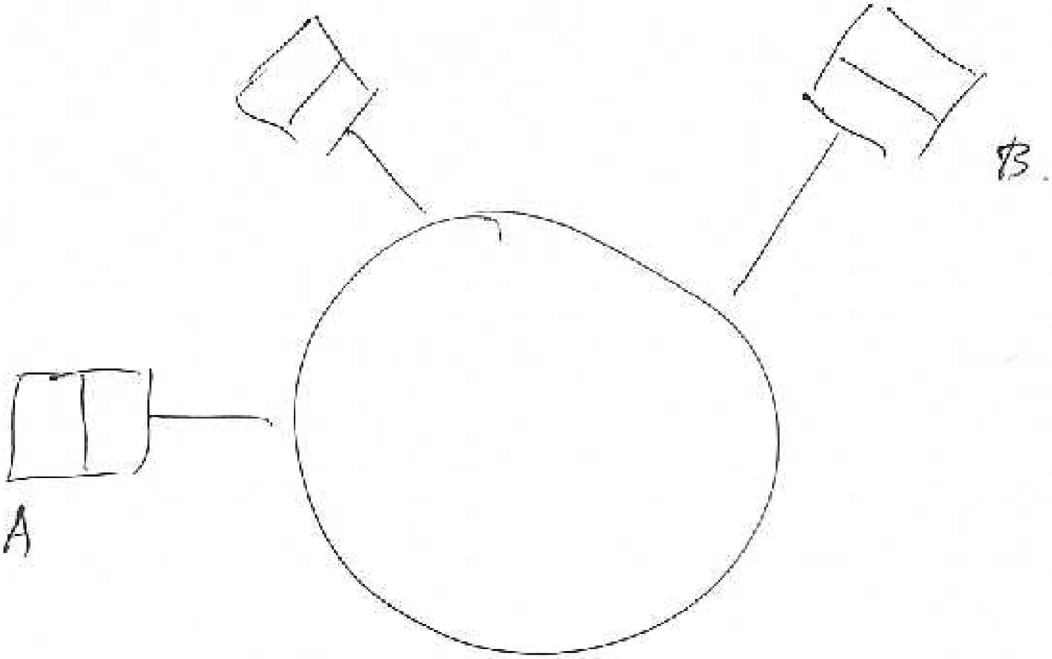
This is a nice way of <u>thinking</u> about it, although actually most (all?) the IP routers have ATM capability:



And various other configurations.

Let's think of the top figure p. 242 (notes).

How does address binding work?



If Host A knows the IP address of host B, how does it translate that into a physical address?

ATM: physical addresses are either 8 octets or 20 octets.

Let's gloss: always 20 octets.

A knows IP address of B.
How does it find the physical address?

IP over ethernet: we had ARP.
because of ethernet has <u>physical broadcast</u>.

ATM can talk over existing VC, only.

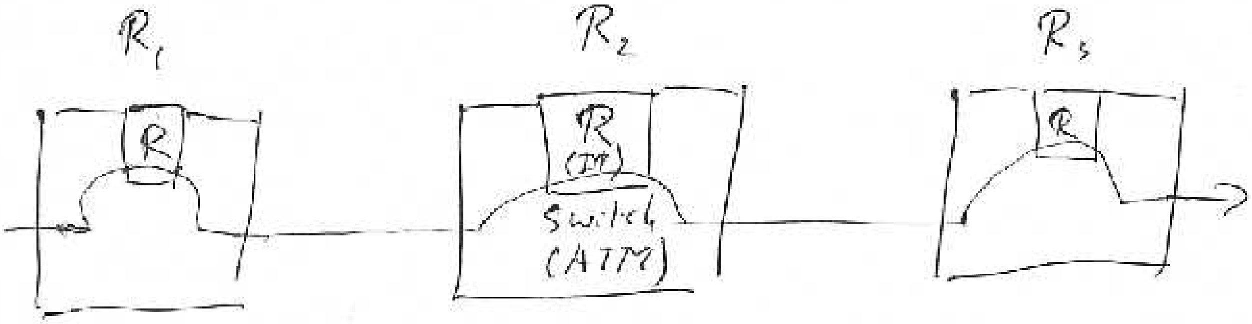
Mechanisms:

① "Find out who is at other end of existing VC".

② Special <u>Server</u> in ATM network.

ATMARP protocol.
Comer pp 366 - 371.

LIS Logical IP Subnet.
"like" VLAN with ethernet
A number of Hosts who "can reach each other". (see Comer p 364)



~~If R_1 sees a sig~~

Base-level: ~~SAR~~

every unit: ~~AB~~ Reassembly,
Routing,
Segmentation.

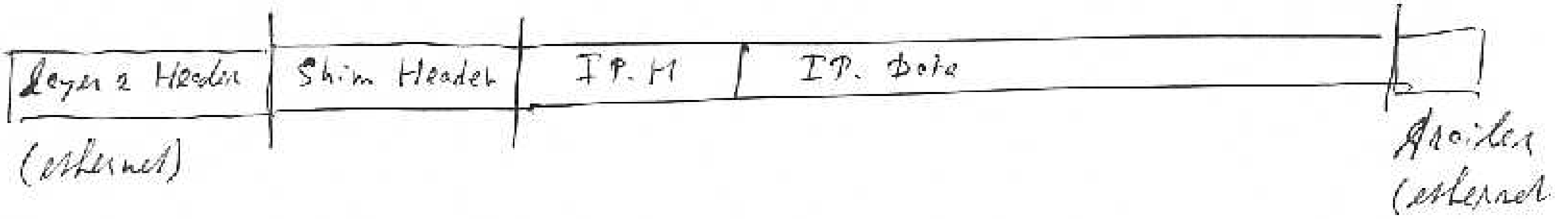
If a "unit" sees a lot of traffic for a specific "flow",
it can create a ~~[illegible]~~ VC that bypasses SAR,
tells predecessor "use this VC for this flow".

<u>Relievers</u> : <u>Routing Unit</u>.

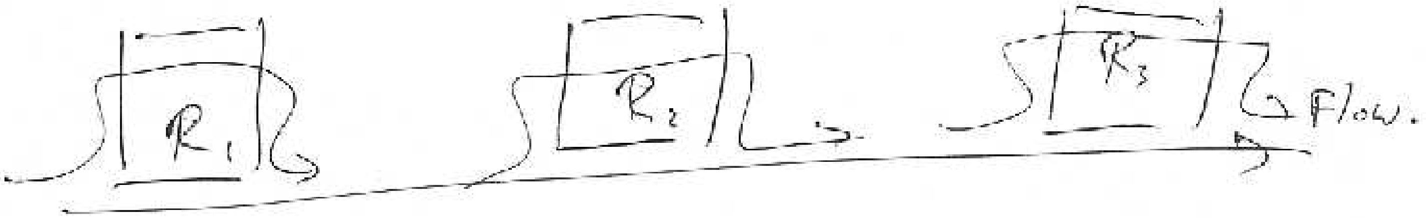
This <u>was</u> the motivation for

Tag-Switching,
MPLS Multi-Protocol Label Switching.

MPLS now (also without ATM)



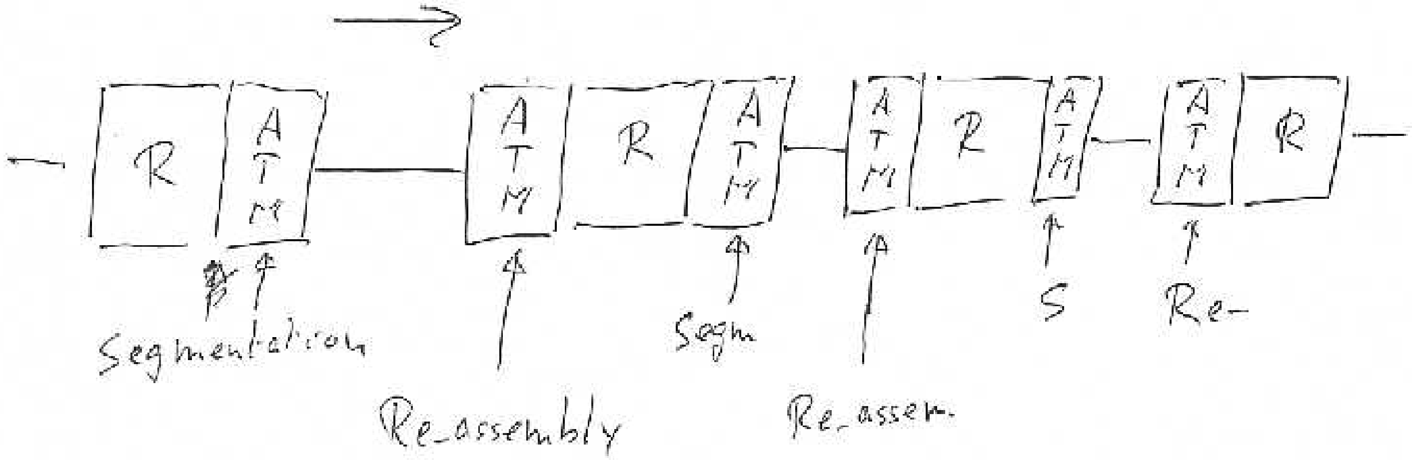
Shim Header or Label



If R_2 sees that a "flow" has a lot of traffic, it can tell R_1:
For this flow, put a shim header on top.
Then I do not need to route the packet: just use label to forward.

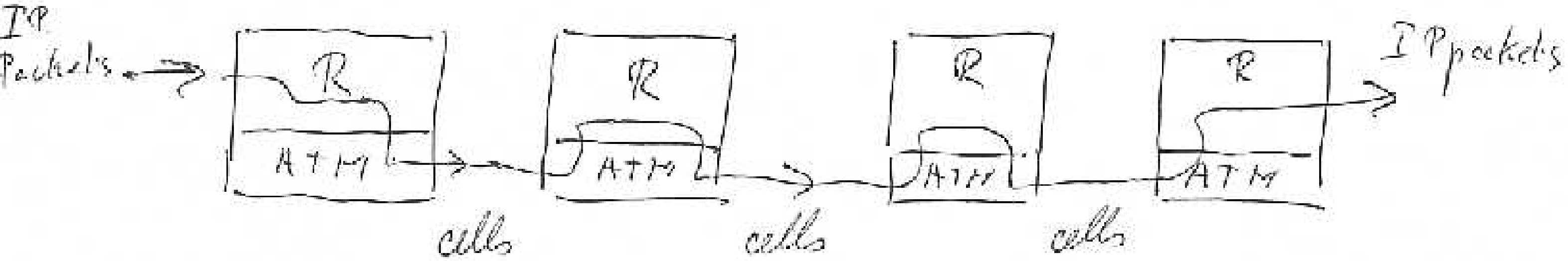
to here 04/23/2004

~~Start here 04/27/2004~~
yes, start here 04/27/2004. Before ~~243~~ 247.
IP over ATM.



SAR

Alternative drawing:



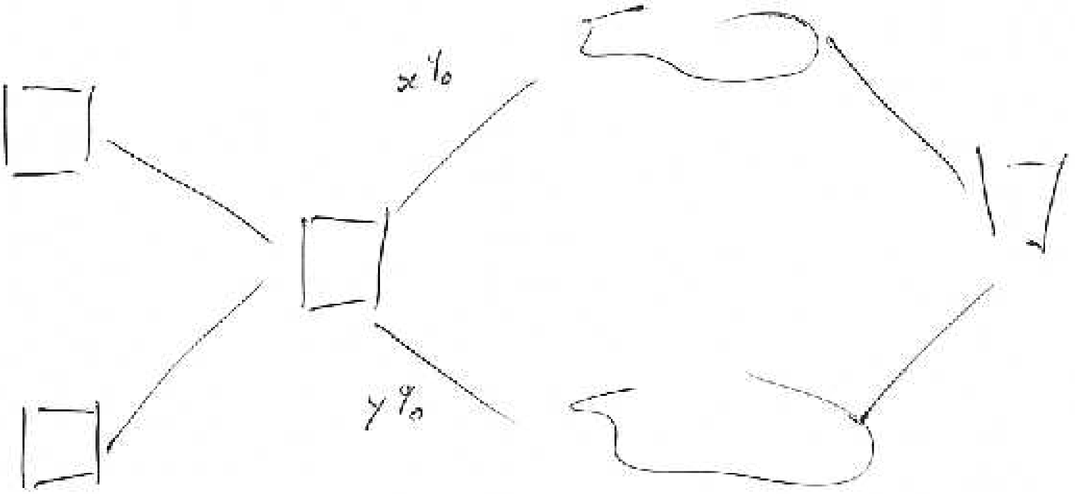
MPLS: Multi-protocol label Switching.
Shim-header. 4 Bytes (32 bits)
of which 20 bits is the label.

Where is label switching / MPLS / tag switching going?

Not clear.

Originally:
(Ipsilon, Peter Newman)
Increase capacity of routers.

Later: Traffic Engineering
Recovery ← ?



Achieve any distr. x%, y%.

Now: The bridge between IP and optical switching?

Current: MPLS to implement VPNs.
Virtual Private Networks.

The concept of "flow".

Originally:
a stream of packets with same

IP dest address,
IP source address,
dest port
source port
protocol

Sometimes (MPLS)
"A flow is what we call a flow".

Comer p. 373.

Usual nomenclature:
Routing: layer 3 and up.
Switching: ATM, and layer 2 and down.

ATM *now* is considered a layer ~~3~~ 2 entity.

(But people talk about "layer 4 switching", etc).

BOOTP and DHCP

Comer Ch. 23.

A computer's configuration file contains:

its name

its IP address(es), network mask(s)

name / address of router

name / address of DNS server.

What about the famous "diskless workstation"?

RARP helped it get its IP address.

BOOTP: helps it to get "everything".

BOOTP: BOOT Protocol.

So there is a BOOTP server.

RARP: implemented in kernel.
(frametype 0806).

BOOTP: implemented at user level.
(UDP, ports 67, 68).

DHCP

Dynamic Host Configuration Protocol.

~~Extension~~ of BOOTP.
Successor.

Important capability:
hands out temporary IP addresses.

My laptop:

"Accept IP address".

"Accept DNS server" (actually, 2 or 3)

⋮

At Univ:
Log in by wireless. Get "DHCP address" from "wireless router".

At home:
Log in by either wireless or fixed.
Either case: get "DHCP address" from linksys router.

In lab: Mr Jain often resets to ~~some~~ some fixed IP address, DNS servers.
Have to set it back!

On my workstation:

```
arp -a.
(lots of entries including)

hme0   dhcp34-190   255.255.255.255.   00:06:5b:01:45:d0
```

```
arp -an

~~h~~
hme0   128.235.34.190   255.255.255.255   00:06:5b:01:45:d0
```

Name : dhcp34-190 } temporary!
Address : 128.235.34.190 }

The name/addr is assigned (*leased*) for a fixed amount of time.

Before the lease *expires* it must be renewed.

There is a DHCP Server.

Reached by multicast: 224.0.0.12.

"All DHCP servers and DHCP Relay Agents".

Comer, p. 324

DHCP:

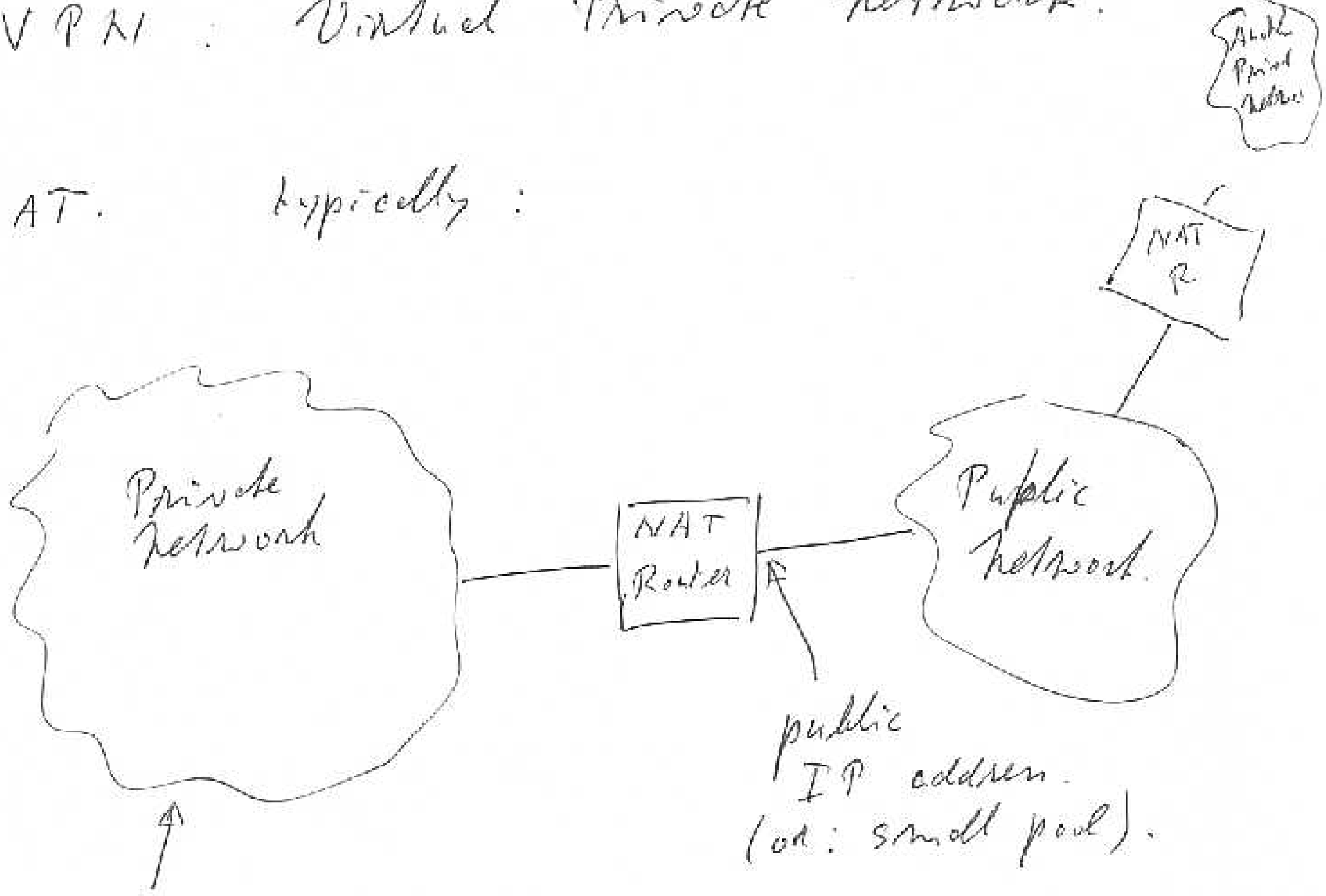
- Saves IP addresses.
  If you have a pool of 100 customers
  but at most 50 have their computer
  on ~~at the~~ at a time:
  need only 60 addresses.
- Mobility. etc.
  - (i) Laptop at home, at Univ,
    at Starbuck, ...
  - (ii) True Mobile IP.

NAT, VPN. Comer Ch. 20.
p 389 etc.

NAT: Network Address Translation.

VPN: Virtual Private Network.

NAT. typically:



Private Addresses.
Comer p. 132.

10/8	10.0.0.0/8
172.16/12	172.16.0.0/12
192.168/16	192.168.0.0/16
169.254/16	169.254.0.0/16

My lab: 10.0.0.0/8.

Home: a subset of
192.168.0.0/16

actually, subset of
192.168.1.0/24

How does communication work?

inside one private network: obvious.

inside public network: old hat.

private - private : hard, often impossible.

private - ~~private~~ public.

<u>Usual</u> situation:

Must be initiated from private side.

NAT also is firewall, state full.

If inside → outside there are packets

s. IP, d. IP, s Port, d Port, Prot.

then it allows

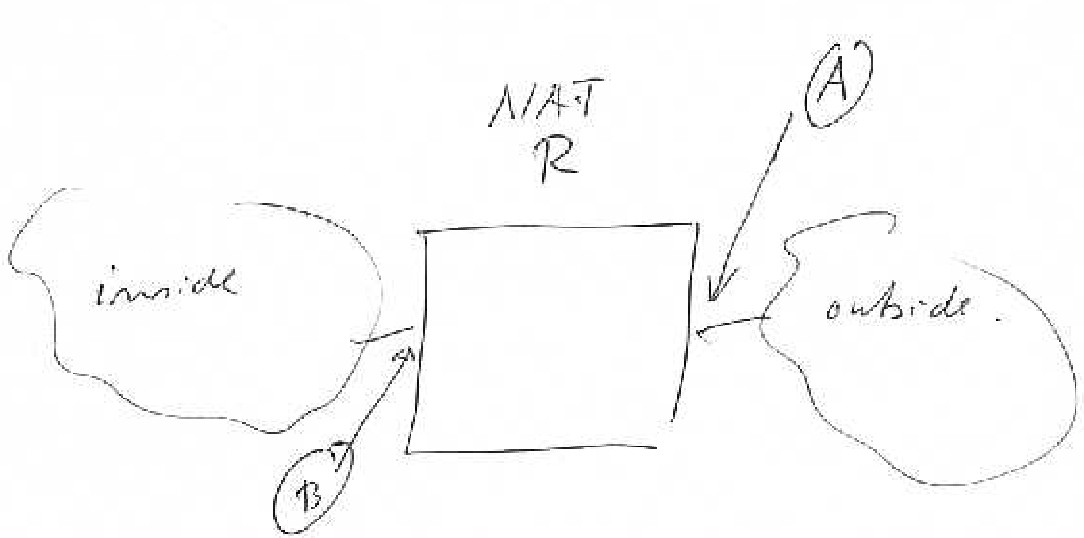
outside → inside

the obvious complement.

even for UDP (!).

The NAT router "knows" protocols.

"As long as it is an answer to something that was initiated ~~at~~ inside".



~~s. I~~

A: "public IP address".

B: "private IP address".

packet →

s. IP, d. IP, s. P, d. P, Prot.

translation:

$$\underset{\substack{\uparrow\\ \text{changed}\\ \text{(simple)}}}{A}\ ,\ \overset{\substack{\text{same}\\ \downarrow}}{d.\,IP},\ \underset{\substack{\uparrow\\ \text{changed}\\ \text{(complicated)}}}{s.P^{*}}\ ,\ \overset{\substack{\text{same}\\ \downarrow}}{d.P.}\ ,\ \overset{\substack{\text{same . "as before."}\\ \downarrow}}{Prot}.$$

The NAT router remembers the translation

(s. IP, s. P) → (A, s. P*).

(possibly a different map for every d. IP, Prot).

A: either 1 address,
or small set of addresses.
let's say one.

map (s. IP, s. P) → s. P*
must be 1-1. (inverse exists).

→ If (Addr1, Addr2, Port1, Port 2, Prot)
maps in (A, Addr2, P*, Port 2, Prot)
then
← ~~(A, Add~~ (Addr 2, A, Port 2, P*, Prot)
maps into
(Addr2, Addr 1, Port 2, Port 1, Prot).

Important! P* must be chosen such, that this inverse map is unique! This limits number of Flows.

NAT is a bit of a kludge.
Has become indispensable.

Sometimes port numbers, IP addresses must be changed in data: complicated!

Sometimes certain combinations of IP address, port number are preset for a given translation.

To allow specific services to be implemented inside, accessible from outside

---

Now I should do VPN.
Comer pp 389 - 394

But first: IPv6.
It exists!

To here 04/27/2004.

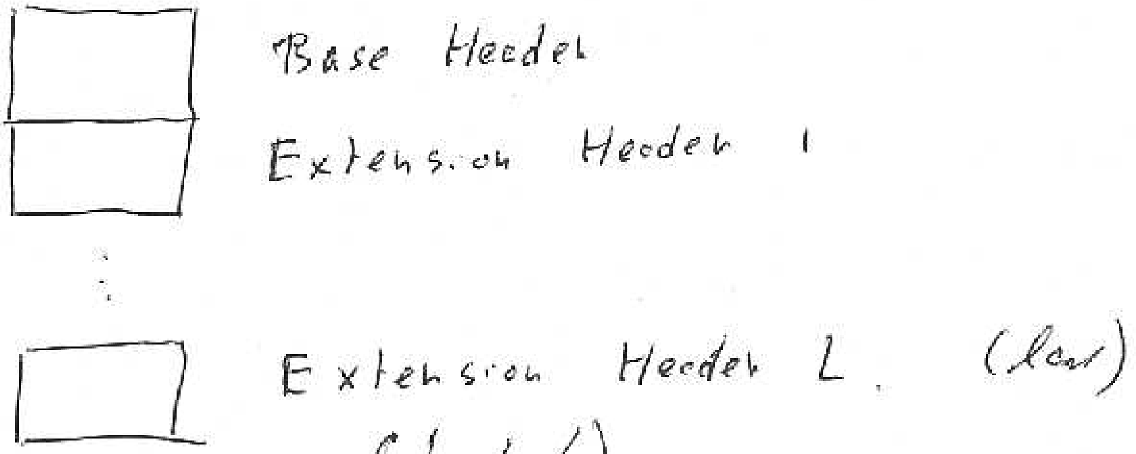
#7 IPv6 IP version 6

IPng IP next generation.

Differences with IPv4:

① 128 bit addresses
(from 32 to 128: increase by <u>factor</u> 2^{96})
$2^{32} \sim 4*10^{9}$ $2^{96} \sim 64*10^{27}$.
More on addresses later.

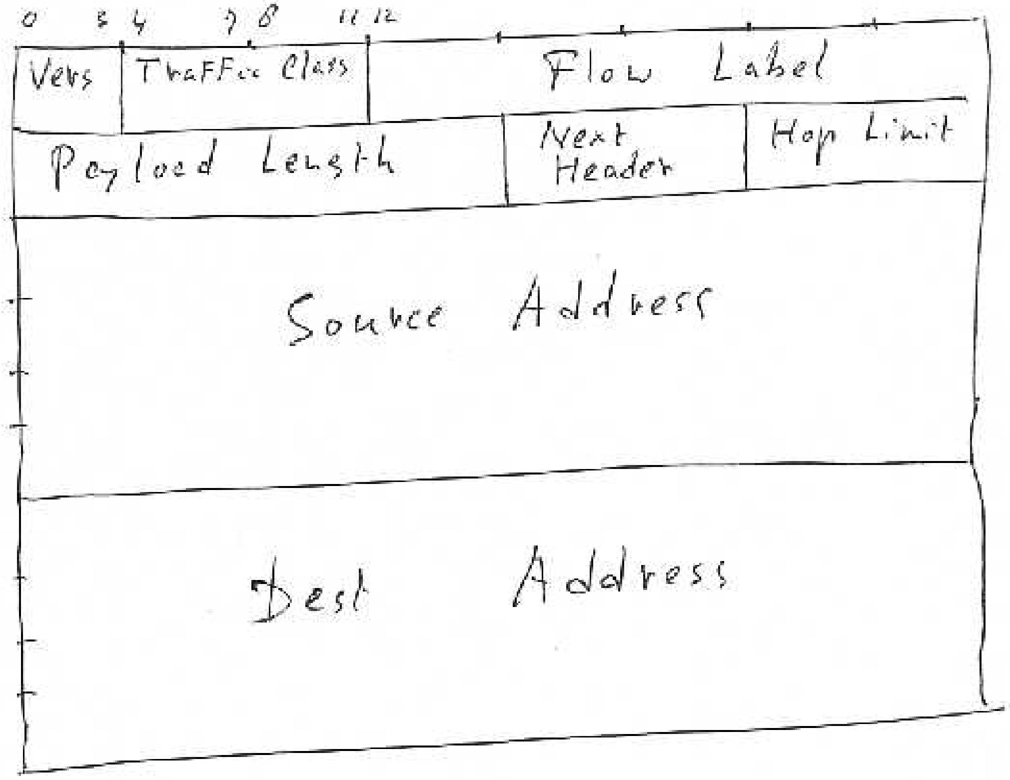
② Instead of <u>options</u>, <u>extension headers</u>.



($L \leq 6$).

Each of these headers contains a "Next Header" Field, that gives the type of the next extension header. ("Type" is wrong word).
The "Next Header" field in the last extension header functions as "Protocol" field.
(~~2~~6 for TCP, 17 for UDP, etc.).

The Base-Header.



Version: 6 1010.

Traffic Class: "like ToS" in IPv4.
DSCP and ECN ?

Flow Label: "like" the label in MPLS.

Payload length: in bytes.
Excluding Baseheader.
My understanding: including
Extension Headers.

Next: describes (type! kind!) of next header.

Hop limit: TTL. now really hops.

Source Address,
Dest Address:
each 128 bits.

Next header : (8 bits, same as PROT in IPv4).

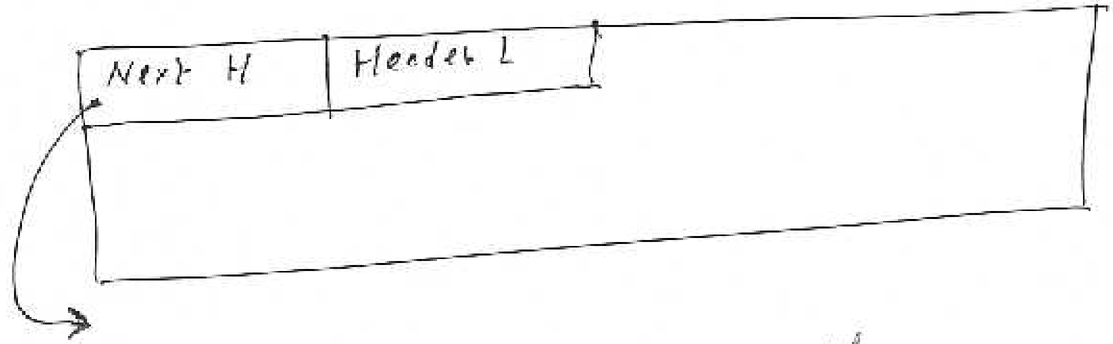
0 : "Hop by Hop".
2 : ICMP (ICMPv6). } (!)
6 : TCP (old).
17 : UDP (old).
43 : Source Route
44 : Fragmentation
51 59 : Null. (No Next Header).
60 : Destination Option. (Related to Source Route).

51 Authentication.

See IANA, Protocol numbers.

In IPv6, no more IGMP !

IPv6 Extension Headers..
Format.



(unless TCP, UDP, etc. These are as before).

<u>Inside</u> an extension header there may be a number of options.

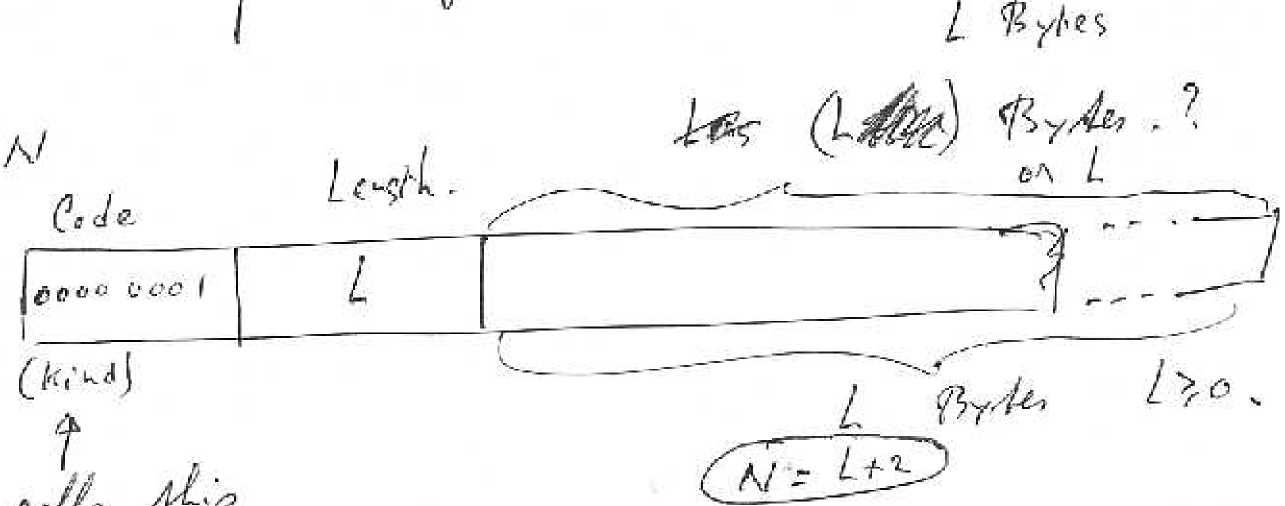
For the "Hop by Hop" extension header:

Options:

Pad 1 : 0000 0000. (~~a~~ 1 octet of zeros).

P. "Like EoO". padding.

Pad N



Comer calls this "Type". I think "Code" is more generally accepted.

Pad 2 :

0000 0001	0000 0000

~~or 0000 0010 or~~ ?

Pad 3 :

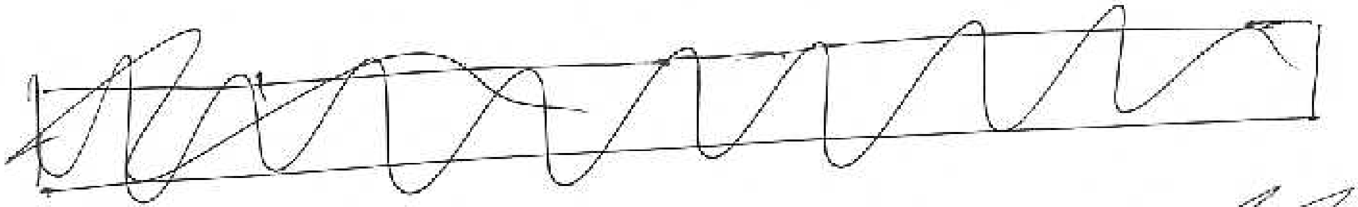
0000 0001	0000 0 0 01	0000 0000

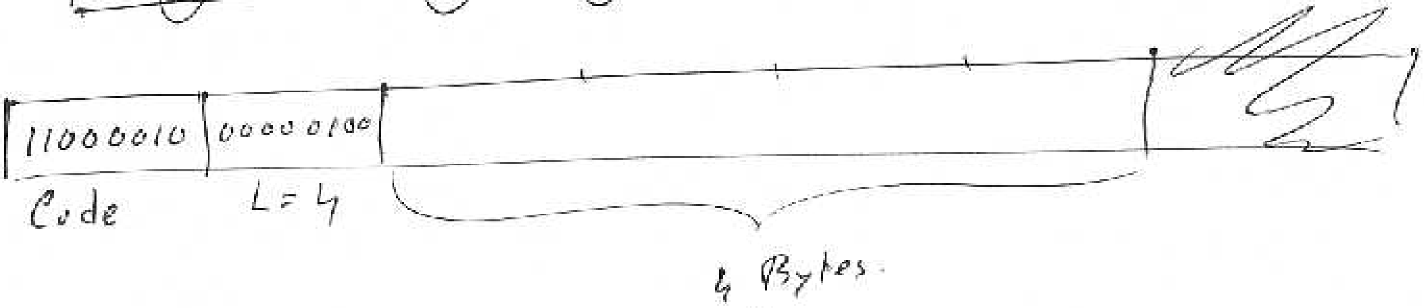
or 0000 0001 ?

Padding Bytes.

Jumbo Payload.

(In case of payload > 65535).

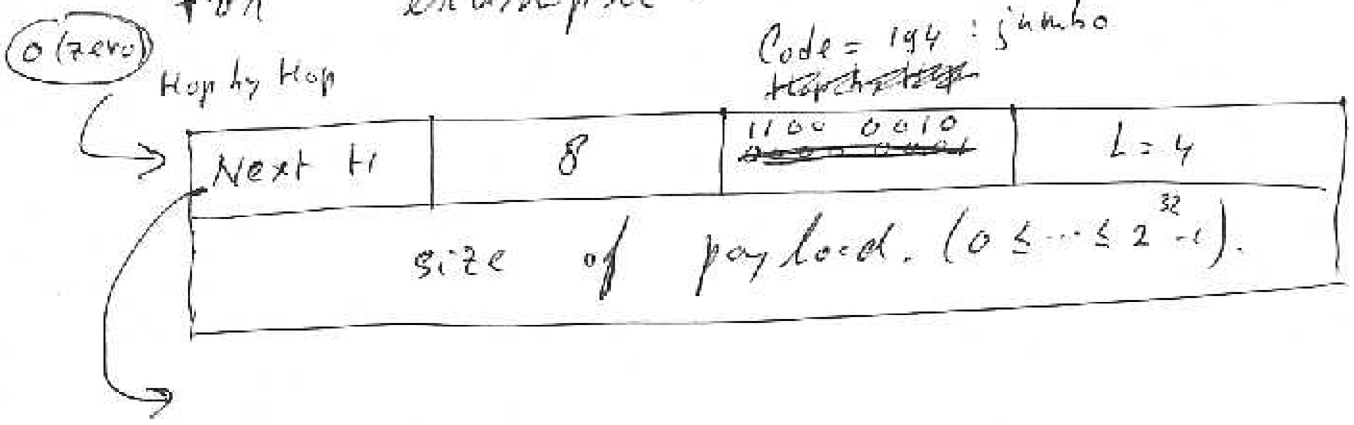




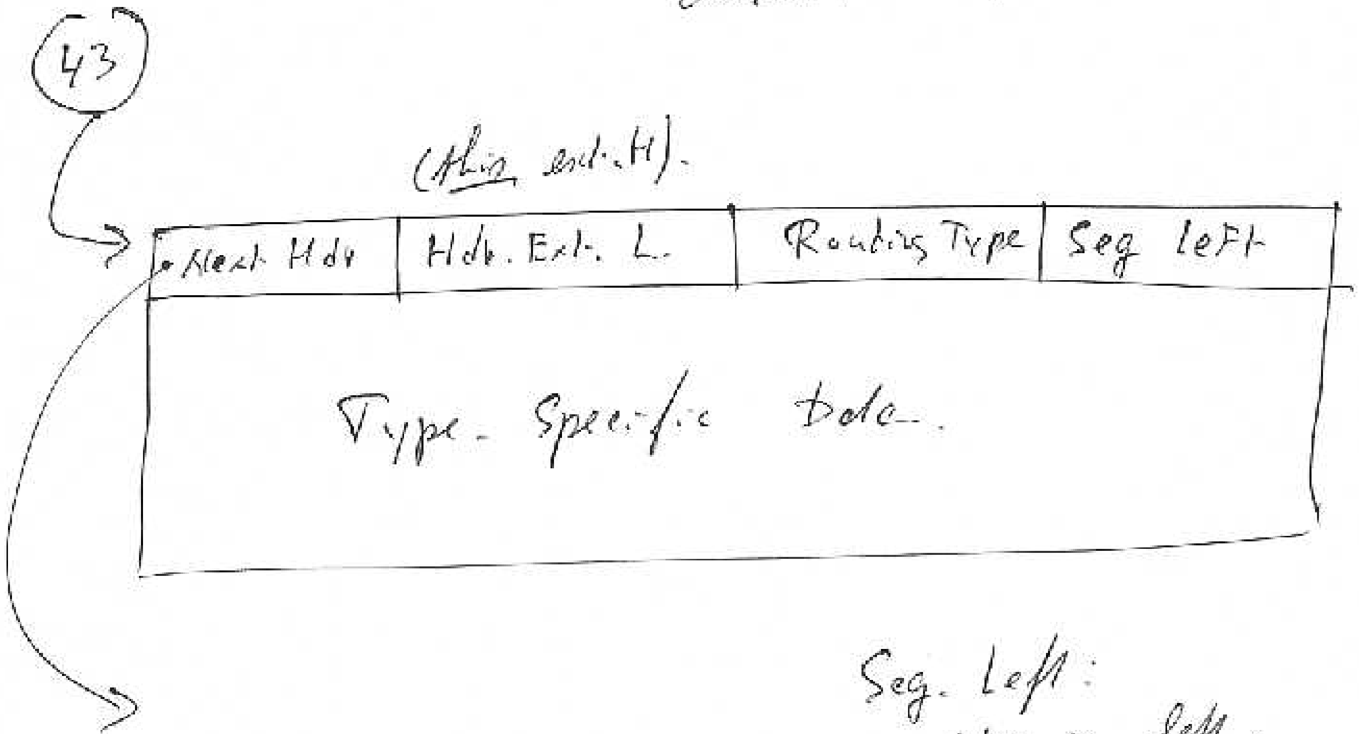
Payload size.
(up to $2^{32}-1$) (!)

Over-ride max of $2^{16}-1$.

For example.



For the IPv6 Source Routing Extension header.



Seg. Left:
\# addresses left.

Currently: only *one* Routing Type: 0 (zero). "Loose Source Route".

IPv6 Addresses.

Usual method:

xxxx:xxxx:xxxx:xxxx:xxxx:xxxx:xxxx:xxxx

Hexadec.

There are various methods to abbreviate. See Comer pp 610-611.

Address Structure.

Comer is out-dated.

(Comer pp 614-619).

IPv6 Addresses.

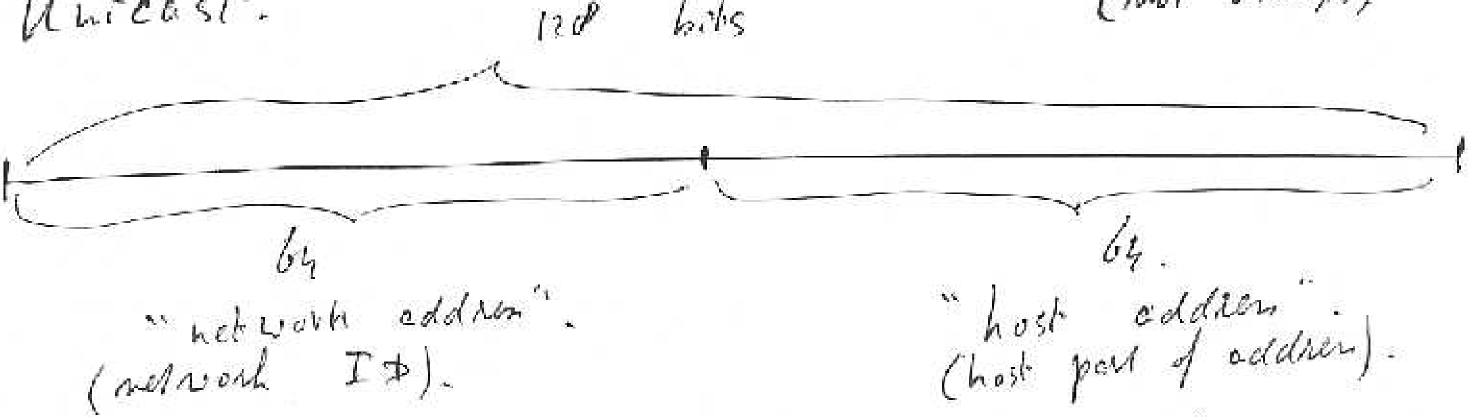
① Unicast (an interface)

② Multicast (group of interfaces)

③ Anycast
"Any one from this group".
Preferably the best. e.g., closest.

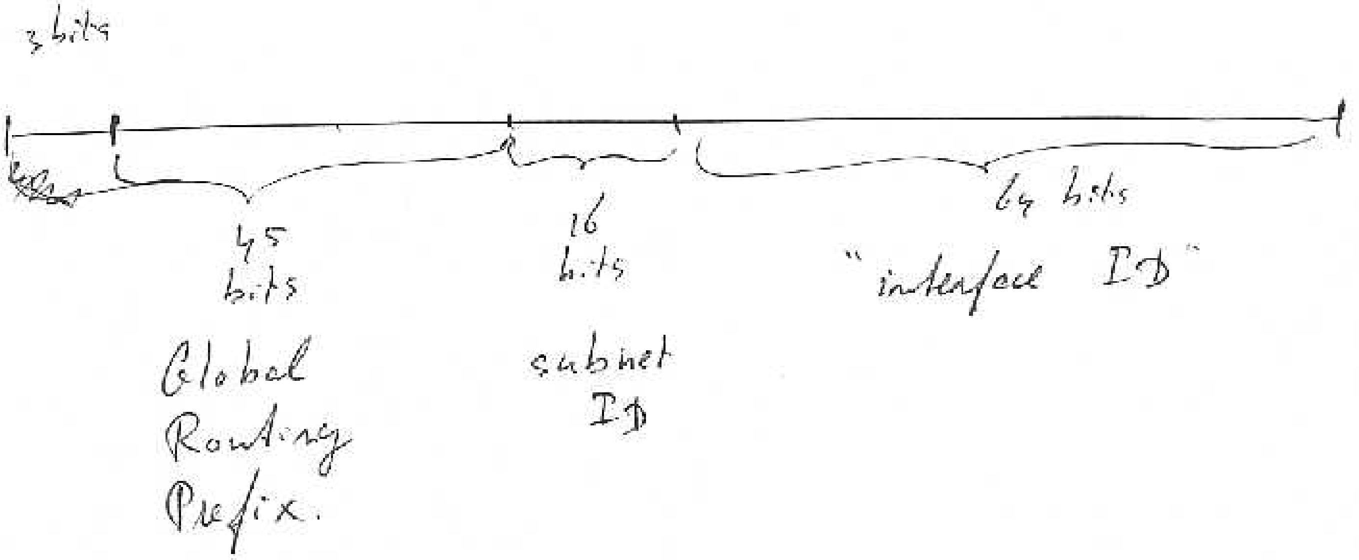
Unicast.

Mostly
(not always)



"Usually" the "host address" is already globally unique.
E.g. derived from ethernet address.
or from EUI-64 address
(Another IEEE unique identifier).

Often:



If the first 3 bits are 001,
this is address space controlled
("delegated") by IANA.

2000::/3

(shorthand : 2000:: ~~there are all~~
the addresses that starts with ~~for~~ 2000, further (Hex!)
only zeros.
2000::/3 : all ~~addition~~ addresses that have
the same first 3 bits as 2000::

RFC 3587.

RFC 3513

There is a way to encode IPv4 addresses in IPv6:

$$\underbrace{0\ \cdots\cdots\ 0}_{80\text{ bits}}\ \Big|\ \underset{16\text{ bits}}{0000}\ \Big|\ \underbrace{\text{IPv4 address}}_{32\text{ bits.}}\ \Big|$$

used for interface that also has (full). IPv6 address

$$\underbrace{0\ \cdots\cdots\ 0}_{80\text{ bits}}\ \Big|\ \underbrace{\text{FFFF}}_{16\text{ bits}}\ \Big|\ \underbrace{\text{IPv4 address}}_{32\text{ bits}}\ \Big|$$

used for interface that only has IPv4 address: Tunneling IPv4 over IPv6.
(Next Header = 4 = 00000100).

Link-Local and
Site-local addresses.
RFC 3513.

$$\underbrace{1111\ 1110\ 10}_{\text{10 bits}} \Big| \underbrace{0 - - - 0}_{\text{54 bits}} \Big| \underbrace{x - - - - x}_{\text{64 bits}} \Big|$$

interface ID:

"Link-Local".
Is *not* allowed to go outside "Subnet".

$$1111\ 1110\ 11 \Big| \underbrace{\text{Subnet ID}}_{\text{54 bits}} \Big| \underbrace{\text{interface ID}}_{\text{64 bits.}} \Big|$$

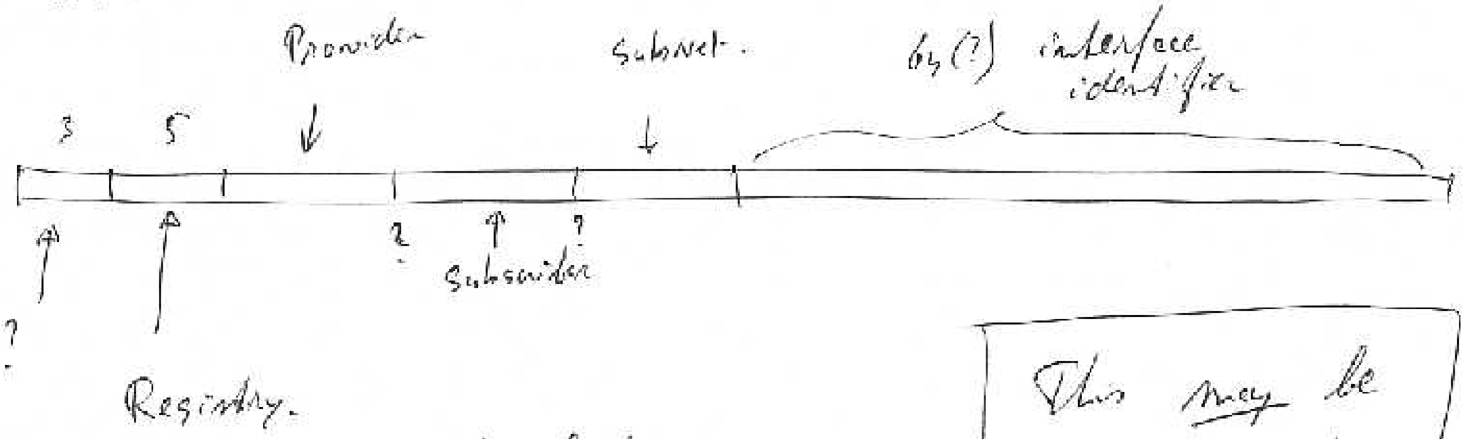
Is not allowed to go ~~"off site"~~ "off site".

(site = company). (emulate Private Networks?)

An interface thus can have multiple addresses!
with subnet ID & interface ID: *must* be globally unique.

interface ID alone: please globally unique.
(not compulsory).

Last on Address Structure



(?) Internic 11000 ← North America
RIPNIC 01000 ← Europe
APNIC 10100 ← Asia & Pacific

This *may* be out of date. But gives the spirit.

Provider: ISP (ATT, ...)
Subscriber: Company. IBM, Dupont, GM.
SubNet: probably 16 bits?
Interface: as before.

An "old" book gave 010 as the typical value of the first three bits. (Forouzan, 2003)

RFC 3587 says that if the first three bits are 001, it is IANA domain.

Still some flux?

Why IPv6?

① Address Space

IPv4 has only ~ 2^{32} ~ $4 * 10^9$ addresses.

Jan 2004: ~ $2.5 * 10^8$ hosts on Internet.

(How is this obtained?)

Only factor 16 left.

② Since [illegible] 1985:

Computers a factor 100 (?) faster.

Communication a factor 100 (?) faster.

Different Trade-offs.

③ New applications, with different requirements.

---

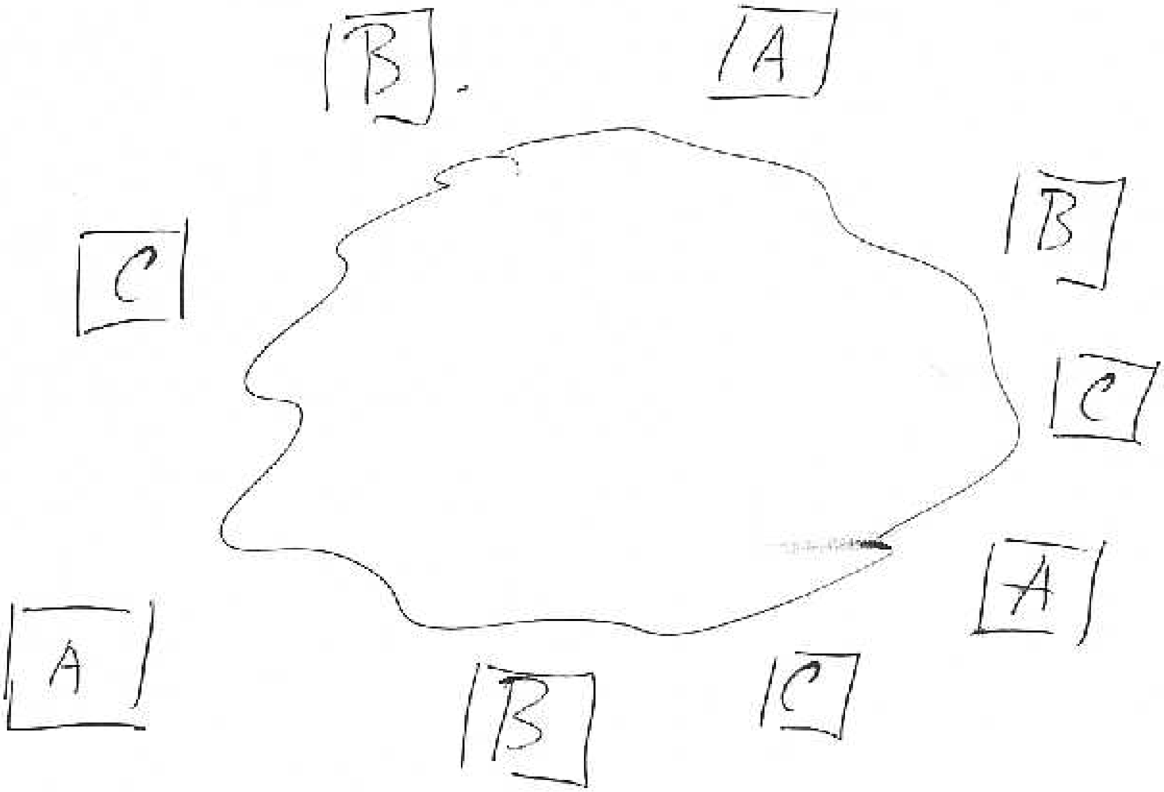
④ Internet larger.

In 1990 - 1995 it doubled every ~ 6 months.

Slower now, but still growing.

VPN Comer Ch. 20
pp 389 - on.

Virtual Private Network.



One public internet
Several companies that want a private internet.
Each could use private addresses.
e.g. 10.0.0.0/8.
Different companies might use the same address!

How do we implement this?

(1) Leased Line.

Works. Expensive.

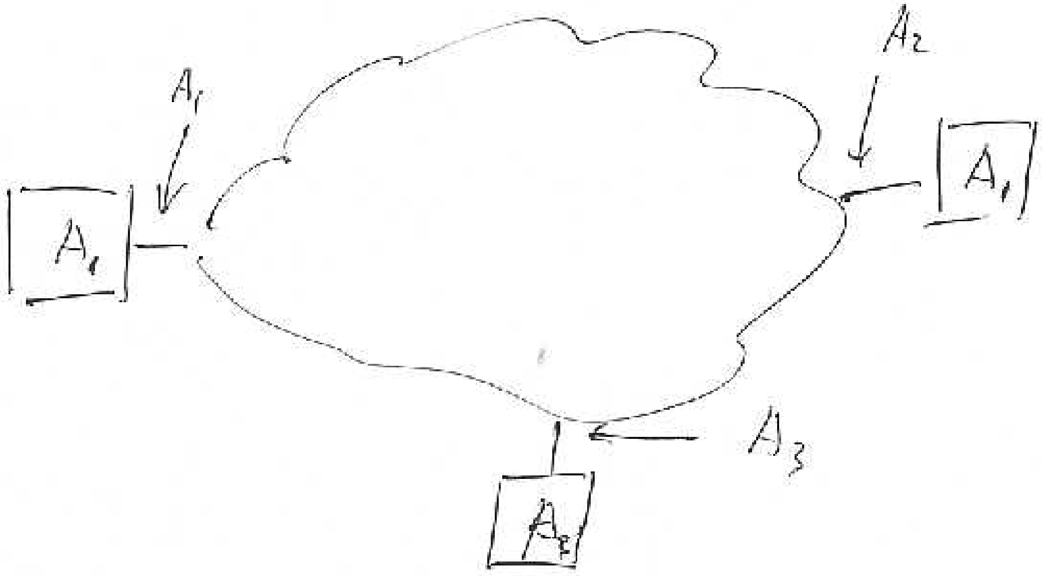
This ~~a~~ is a "Real" Private Network.

(2) with a VPN

Virtual Private Network.

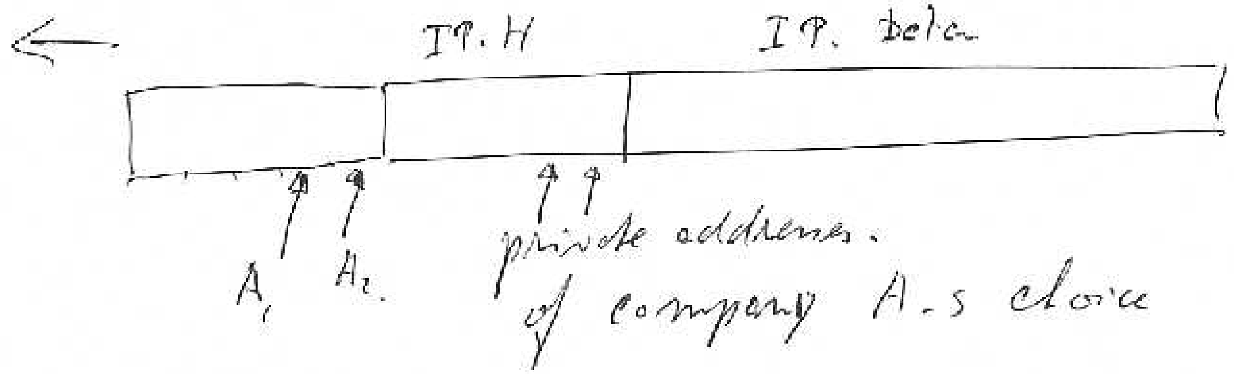
How?

2A : IP tunneling.



A_1, A_2, A_3 : <u>public</u> IP addresses.

Packet $A_1 \rightarrow A_2$.



private addresses.
of company A-s choice

Tunneling. IP over IP.

we can even encrypt the whole inner packet! Including "inner header".
(prevents "Traffic Analysis").

---

2 B. MPLS.

construct a MPLS paths from A_1 to A_2, etc.

put "shim header" on top.
packet itself could be encrypted!

---

2 C. ATM VC. (Virtual Path).

Same idea.