OSPF

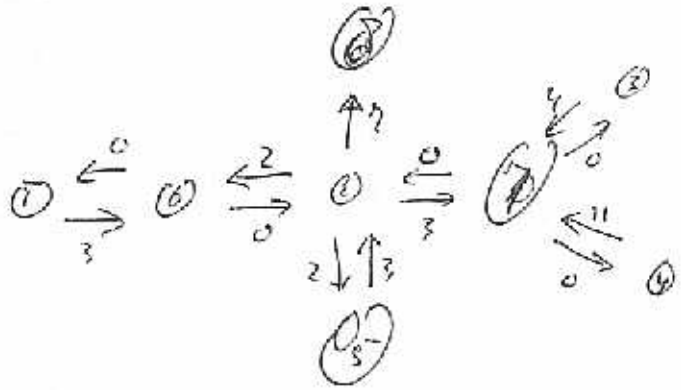
Dijkstra's Algorithm.

Matrix:

to \ from	1	2	3	4	5	6	7	8
1						0		
2					3	0	0	
3							0	
4							0	
5		2						
6	3	2						
7		5	4	11				
8		7						

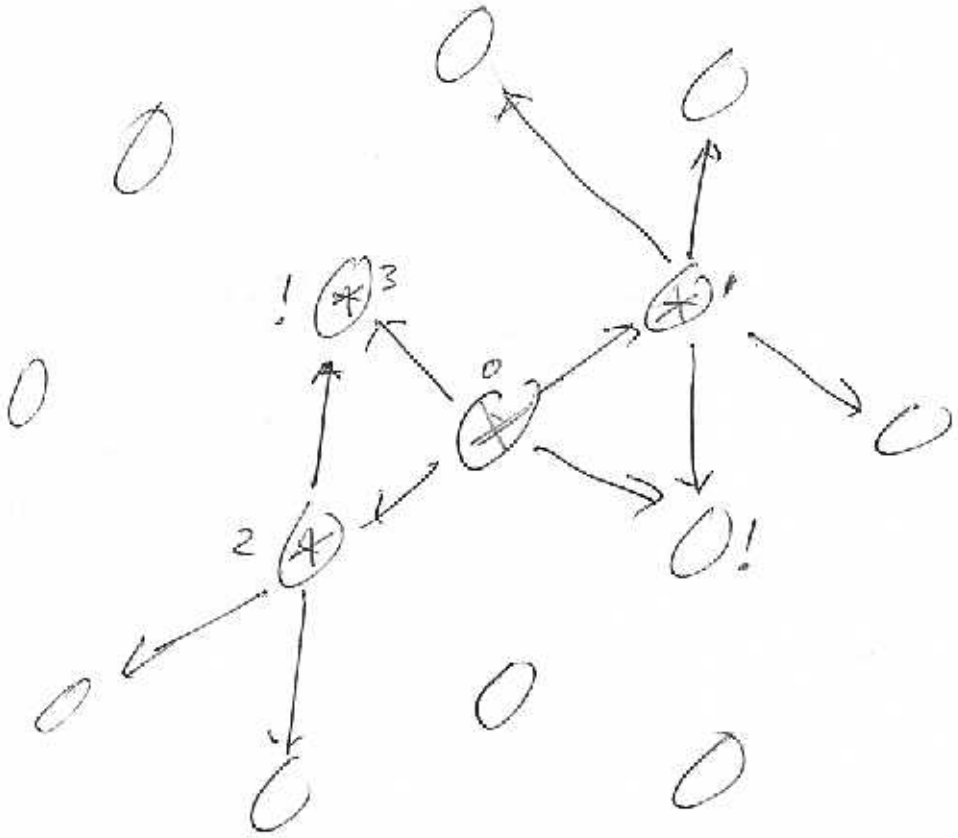
Translate into picture ("graph").

directed graph



Dijkstra:

Shortest path from here to every other else:



etc

Implementation: Use a

Special kind of binary tree

(Heap).

Complexity: E log N.

RIP 1, RIP 2: IGPs distance vector

OSPF, ISIS: IGPs link state.

BGP: Border Gateway Protocol.
~~A little bit of both.~~ not link state at all

Formally: more like "distance vector".
but the route description is not a single number (cost) but a complicated story.

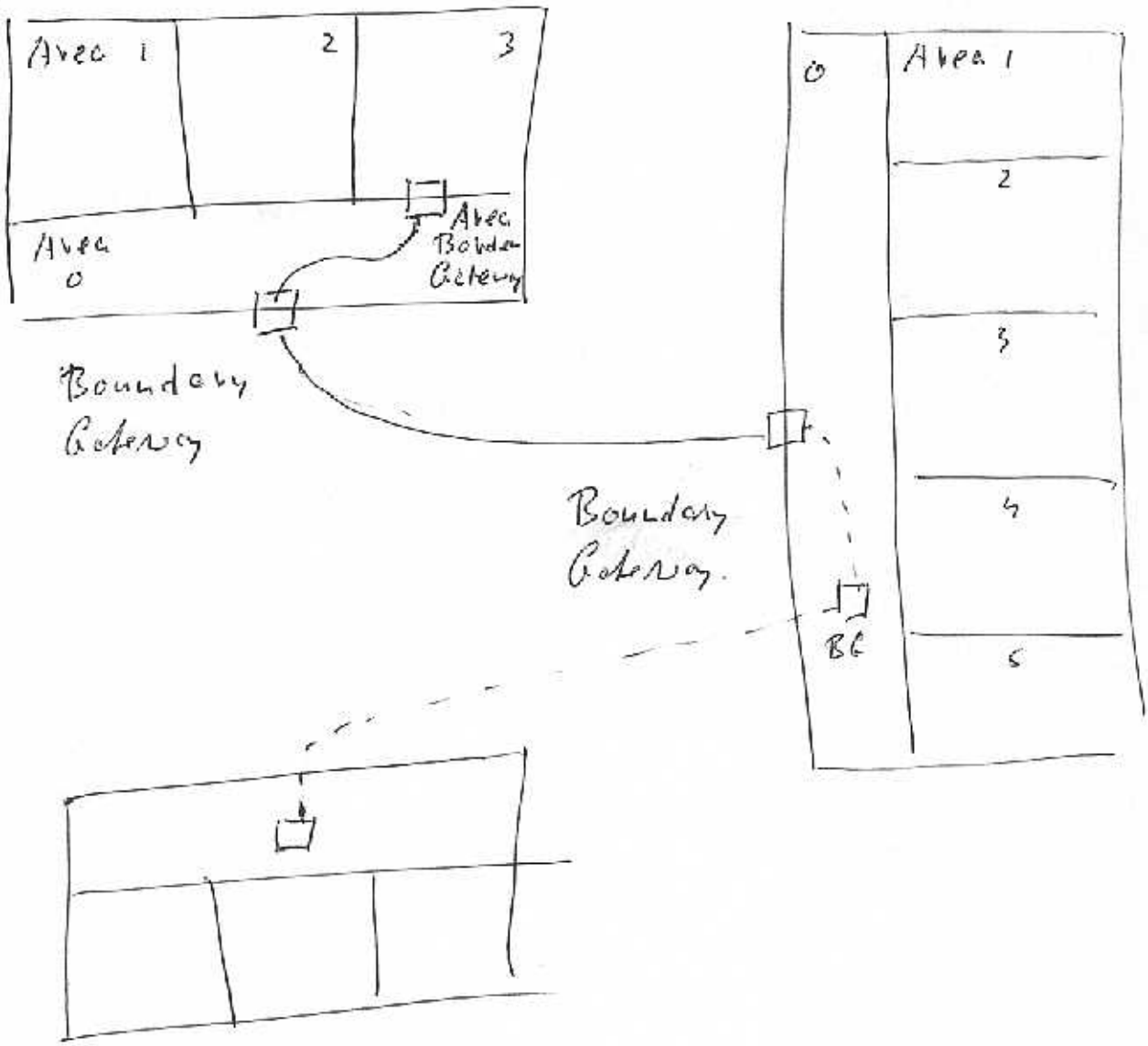
Also: BGP does not have an algorithm behind it.

BGP: routers have attributes.

Attributes ~~are~~ can be very complicated.

AS 1

AS 2



BGP: in Boundary Gateways.

BGP advertisement:

"I can reach network N.
The route (path) has the following
attributes ... (Long story).

//

List of attributes:
includes list of ASs the route passes
through.

"I know of a route to network N. The
route from me to network N passes
through to ASs:
AS1, AS2, ...
(4 more attributes!).

There is a tendency to choose the shortest
list of ASs.
Result: if a BGP router prefers others not
to use a specific route, it may "doubly list"
ASs: $AS_0, AS_1, AS_1, AS_1, AS_2, \ldots$.

BGP:

"Initially" a complete list.
then: "deltas" (corrections, additions).

BGP: "in between" ASs.

RIP, OSPF: inside Areas.

both (in particular OSPF):
Also in between Areas in one AS.

BGP: in between As.

For example:

RIP or OSPF inside area.

OSPF for coordination between Areas in an AS.

BGP for coordination between ASs.

ATT network: many ASs.

smaller companies: one AS
(for ASs)

ATT uses BGP (only?).
Ask Dr Gottlieb.

Contracts between transport companies:

I'll take traffic from you, as long as:

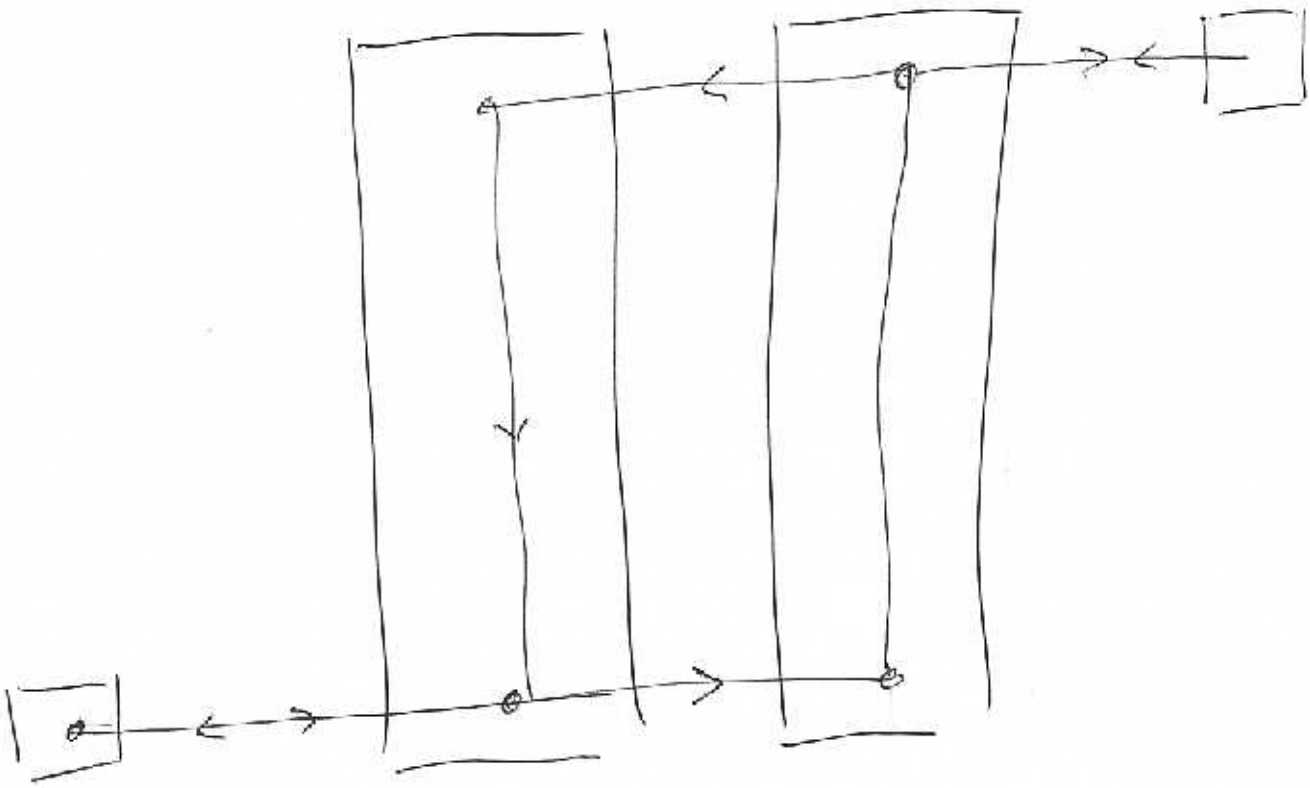
either:

- it is for one of my customers

or

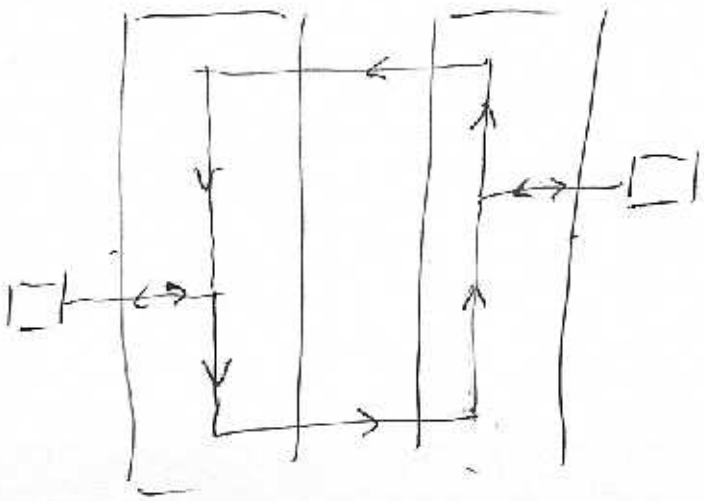
- you pay me

Hot potato routing



Asymmetry!

even



Multicast.

Comer, Ch. 17 pp 319 ⇒

First, Physical layer (ethernet!)

A. Unicast. (We know).

B. Broadcast (FF:FF:FF:FF:FF:FF).
(we know)

C. Multicast. (New).

When is an ethernet address a multicast address?

Not entirely sure. I think:
if the leading octet is odd.

binary: xxxx xxx1 : yyyy ...

Very often (hexadec) 01:xx:xx:xx:xy:xx
(hexadec)

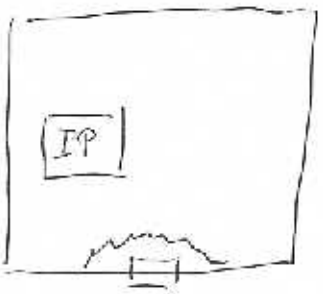
For example: Starting with
01 :
03 :
09 :
33 :
AB :
CF :

Go to Google.
look for ethernet multicast.
www.cavebear.com/Cavebear/Ethernet/multicast.html

Example of multicast address
(ethernet!) 01:00:5E:00:00:01.

This is the "all systems this LAN"
ethernet multicast address.

functionally the same as FF:FF:FF:FF:FF:FF.
But les different role.



An ethernet multicast frame can contain
an IP packet.
(Can contain other stuff!).

In case of IP:

0 driver checks: ~~is it~~
A. is this "my" physical address? if yes:
→ IP.
B. if not: is this FF...:FF? if yes:
→ IP.
C. if not: is this one of the ethernet
multicast addresses I was told to watch
out for? if yes: → IP
D ~~if multicast~~ else: ↓

Then IP checks again!

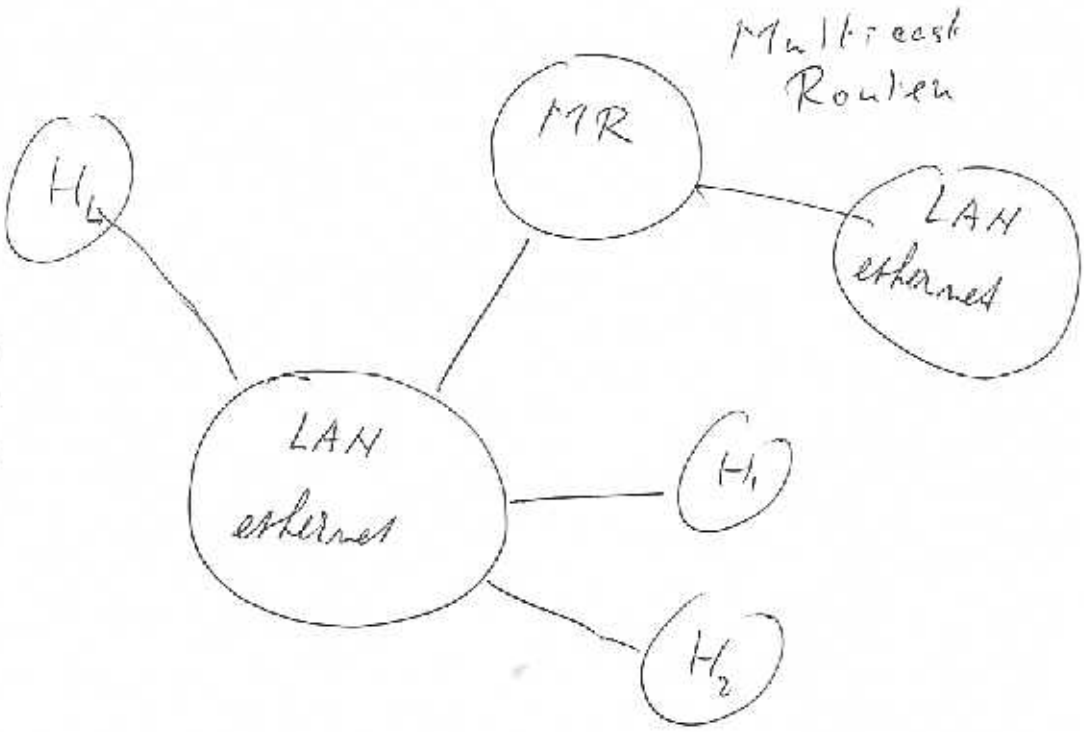
IP multicast. | RFC 1112, 2236 etc.

Recall: 1110 ... means multicast.

128+64+32 = 224 224.0.0.0/4

IP Multicast Addresses.

These are called "group addresses".
"Address" of a multicast group.



(Usually) Multicast Router is needed to get the packet to the network.

Who is the customer?

a Process in a Host.

Dist: & Application Process.

The OS knows, for each IP group address,

" I do / do not have a customer.

The OS tells the MR

I do have a customer for this group.

Each group, on "each" LAN, has an MR responsible for that group.
(Often one MR per LAN, there may be more. Each group assigned to one).

Each MR, for each of its LANs, for each of its groups, knows:

There is / is no customer in this LAN.

If there is:

The multicast IP packet is put inside a multicast ethernet packet and sent.

How does the address map?

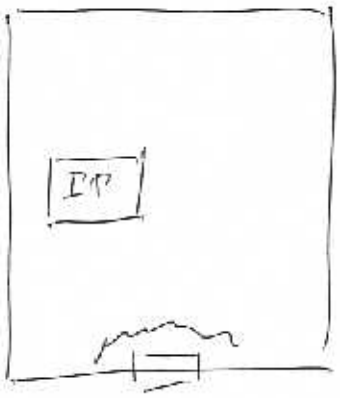
IP multicast: 1110 xxxx : xYYY YYYY ... YYYY

maps into ↓ some.

0000 0001 : 0000 0000 : 0101 1110 : 0YYY YYYY : ... YYYY

ethernet multicast (!)

potentially 32 different IP multicast addresses could map in some ethernet multicast address. (unlikely).

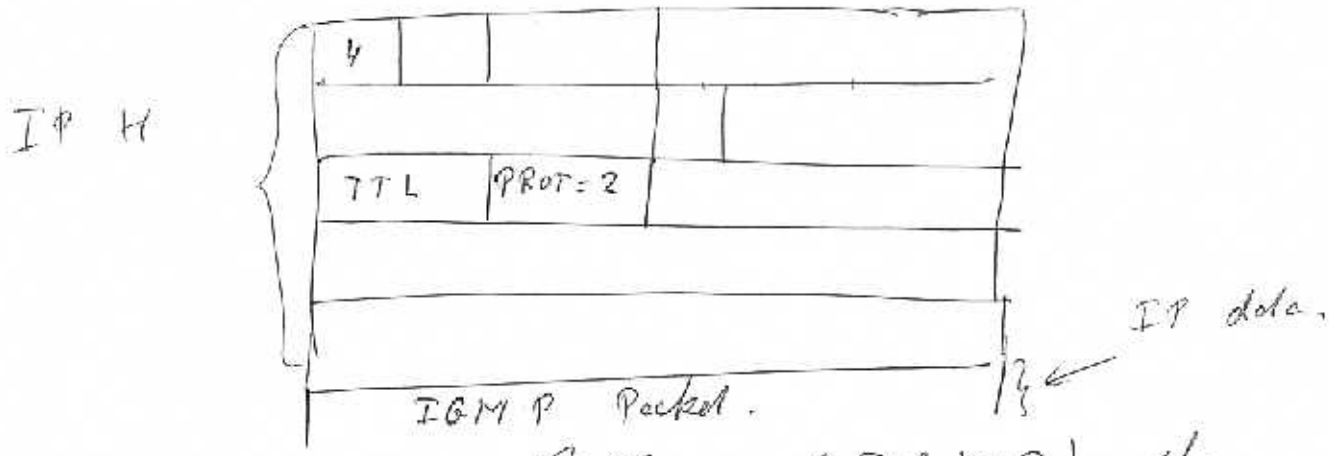


Interface / driver checks:
This is one of the physical multicast address I was told to report to IP!

IP checks: Is this indeed one of the IP multicast addresses I was told to watch out for? If yes: to appl. (to customer).

How does the IMR know there are customers in the LAN?

IGMP.
Internet Group Management Protocol.
Directly under IP: PROT = 2.



// PROT = 2 (IGMP) then very often (always?) the dest. address is an IP multicast address.

"IGMP uses multicast"

("Bootstrapping")

How does the process (customer) know the group address?

(1) "Comer p 324". (Permanent multicast addr)

(2) "NY Times"

e.g. IETF,

Or "Radio Locator"

(Internet Radio).

IGMP.

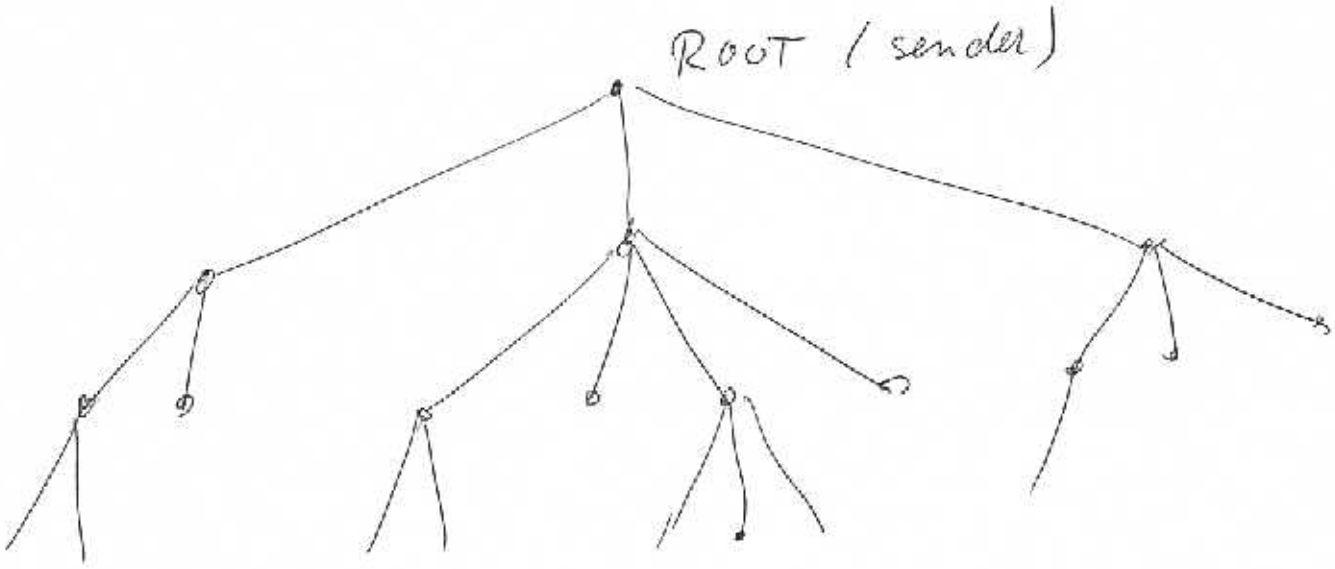
Comer pp 329 - 331

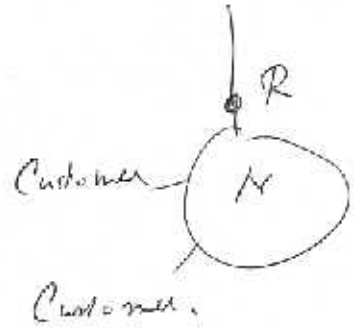
Skip.

How does the packet get to the MR?

"Multicast Routing".

Multicast Tree.





Info must go to every "customer".

Multicast: one copy of packet on every link of tree.

Could be implemented as "multiple unicast".
Why is that not desirable?

(1) Too many copies on links close to Root.

(2) Root must know all customers!

Customers may join, leave.
(Internet Radio).

How do we choose a tree?

① (Dumb).

Flought experiment.

ROOT has list of customers (!)

For each (unicorn)

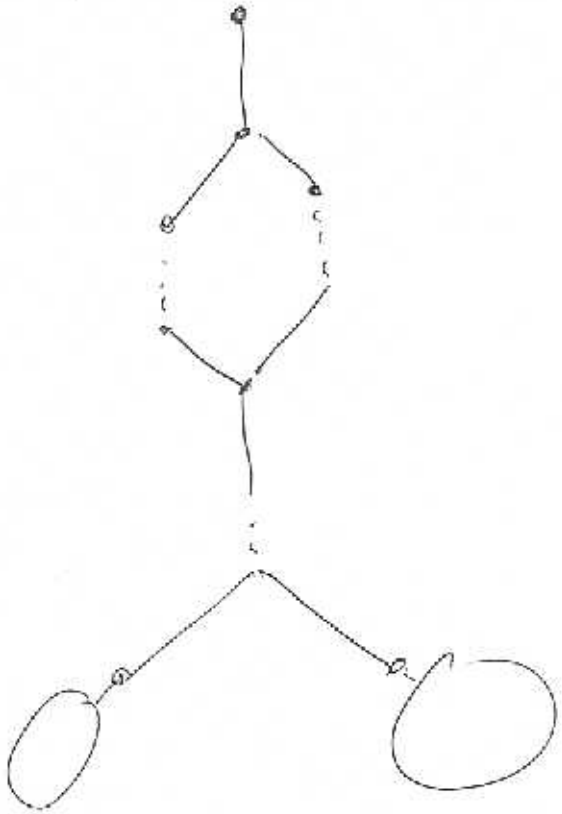
Output Interface? Next Hop?

Select set of "next hops" etc.

What is wrong?

A: ROOT does not have list.

B:



Result need not be tree!

(2) Better:

(Idea of "Reverse Path Forwarding").
(Idea only! see p 225 of these notes)

Set-up:

Each "customer" sends set-up packet to "ROOT".

each router on the way:

~~(A) If I already get the packets~~

A If I do not yet get the packets:

Remember "downlink" Interface, Next Hop (Router)

Send up toward Root.

B. If I already get packets (already in Tree):

Remember downlink interface, send Top.

Start sending them.

~~This was the router of~~

C. Bell of Whistle: Prune & Graft.

If a client disappears: MR finds out.

any more left? if none left: stop sending, tell ancestor. etc. Prune.

If new client appears: graft.

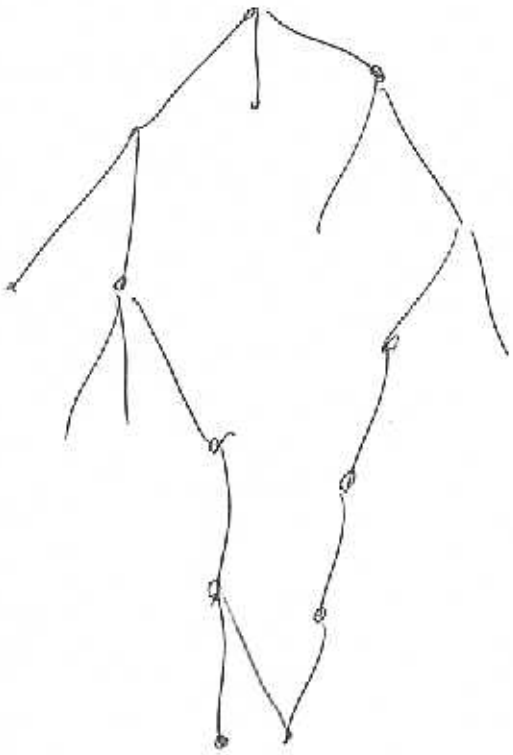
Counter intuitive nomenclature:

Trees obtained as above (roughly!)
are called "shared trees".

These are not optimal.

What is optimal? Minimize $\sum_{s \in Tree} c_s$

c_s: "cost of link s".



("Shortest Path" for every client.)

Find optimal tree:

(1) Set of "must have" nodes.
(Root, networks with clients)

(2) Set of "optional" nodes.
(Other Routers, Networks).

Find "least cost tree" that spans all "must have" nodes.

This is a Steiner Tree problem.

NP complete. (≡ hard).

And worse: dynamic.

~~Such~~
Only heuristics.

Such trees are called
"Shortest Path Trees".

Problem: Asymmetrical Routing.

There is no standard for Multicast.

There are some implementations that "kind of" work.

Best known: MBone.

Multicast Backbone.

Big in mid-nineties.

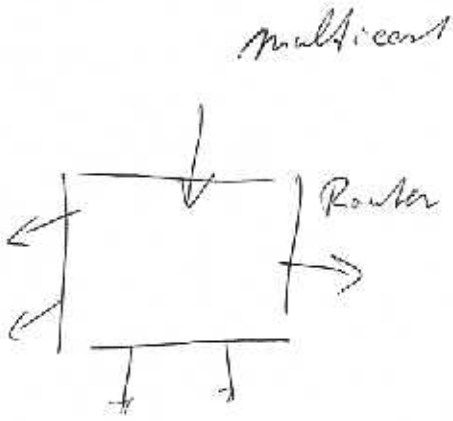
& have not kept up.

Used to multicast IETF meetings.

Currently: Internet Radio

& assume it is multicast.

Reverse Path Forwarding
R P F.



Multicast packet comes in, from Root.

Question:
"If I were to send to root, would this be the interface?".

if no: drop packet.

if yes: <u>broadcast</u> on all other interfaces.

Assuming the <u>unicast</u> tables do not have loops: packets will not loop.

But "multiplication" is possible

Problem with asymmetrical routing.

Truncated Reverse Path Forwarding
" " " Broadcasting

T R P F
T R P B .

"like" R P F, but with "pruning" and "grafting".

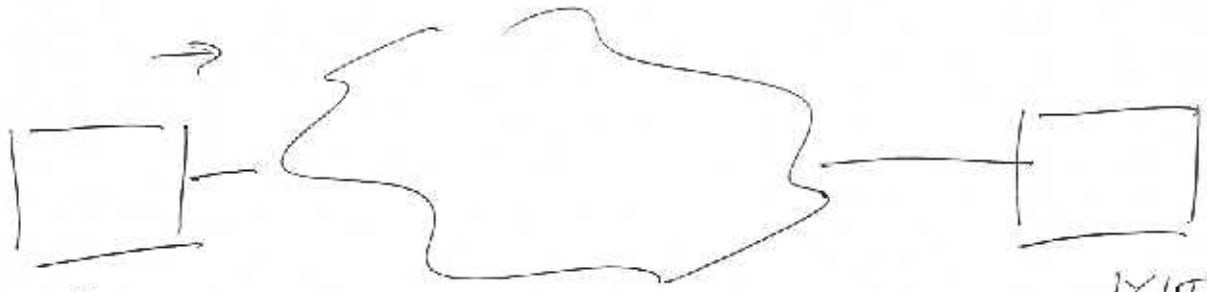
Still problem with asymmetrical routing.

Partial Solution:
Core Based ~~Routing~~ ~~C B R~~ CBT.
Trees

~~PIM~~ P I M - S M
(Protocol Independent Multicast, Sparse Mode).

Divide Internet into "Areas".
Each "Area" has a "Core Router" (CBT)
"Rendez-vous Point"
(RP)
that acts as ~~the~~ "Root" in the area.

Using IP Tunnels in Multicast.



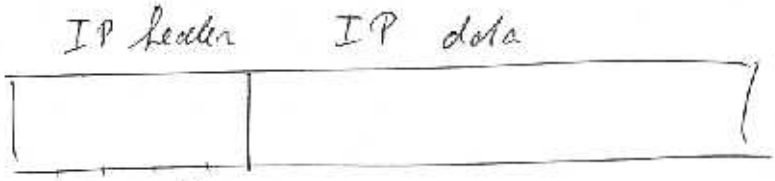
MR1

Multicast-Capable Router. (understands multicast addressing, forwarding etc)

MR2

Network Only Non-MR. (Multicast Ignorant).

Solution: Tunneling



PROT = 4.

IP over IP (IPv4 over IPv4)

Under Way: only outer TTL decreases!

Reliable Multicast.

(E.g. Software distribution?
Nuclear Arms Inspection?).

Reliable Multicast "does not exist".

(Tree would be constant). (Special "Annexes" to Routers, or not in Routers at all)

"ACK implosion".

There are hacks.

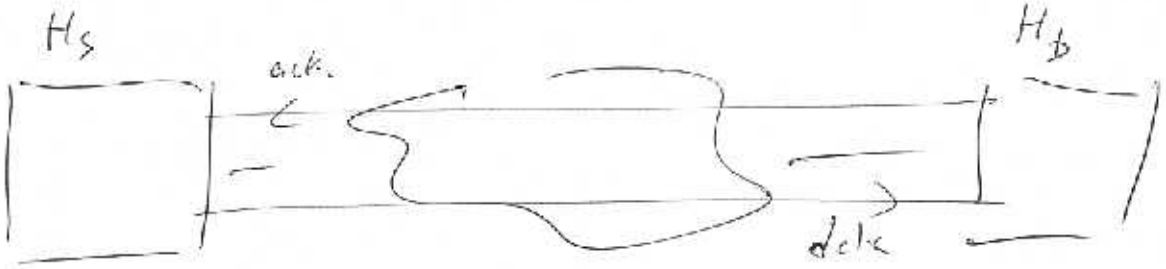
PGM Pretty Good Multicast
(Pragmatic General Multicast)
RFC 3208.

SCTP Simple Control Transmission Protocol
(Stream Control Transmission Protocol).
RFC 3708, ..., 2960

PGP Pretty Good Privacy
RFC 1991, 2015, 2226, 3156
(Name was not changed!)

(Content change).

Why not VoIP over TCP?



Reve BUFF

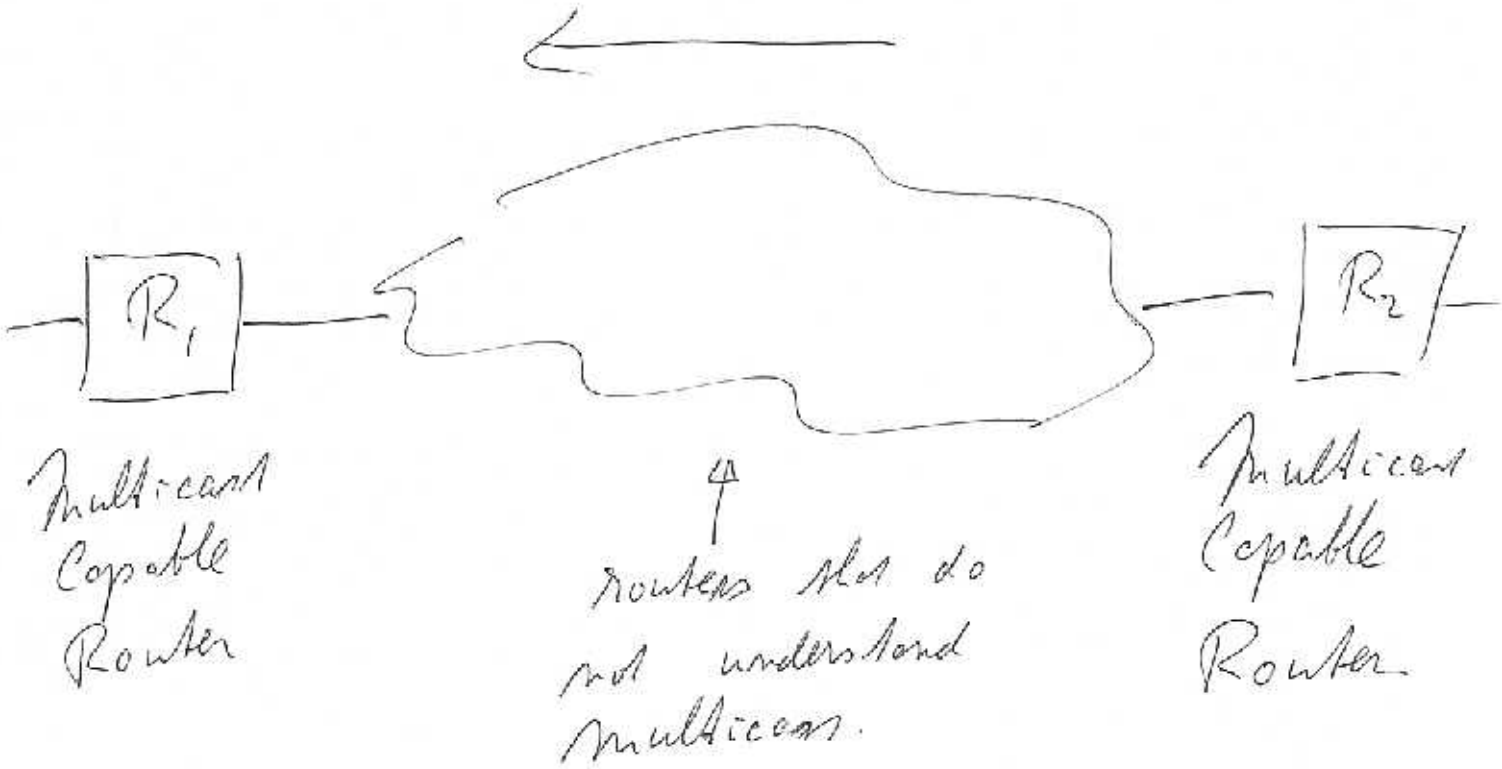
miss.ng

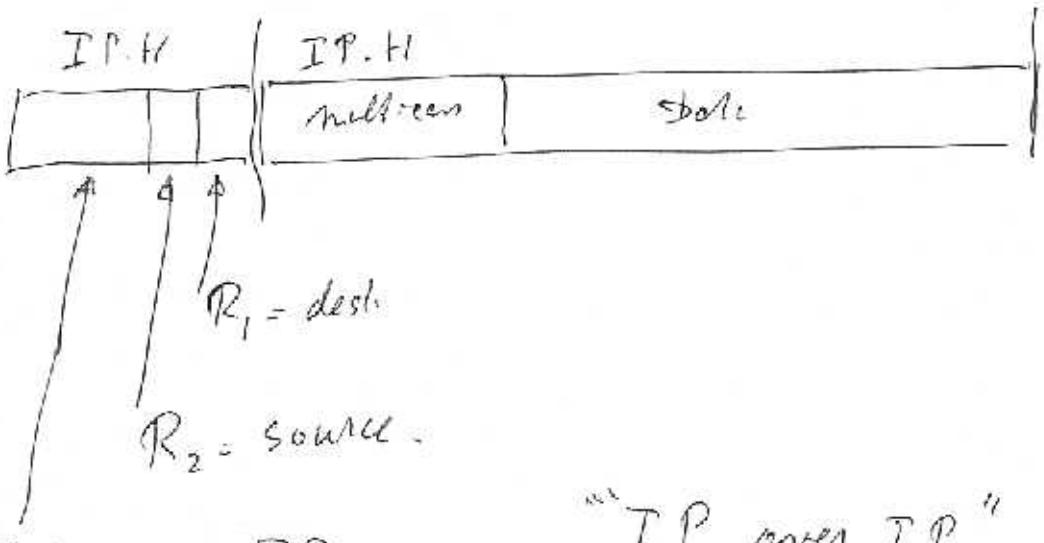
not to ~~destination~~ dest appl. until missing data are there.

Delay!

Better to "fake it".

Did I discuss IP tunneling when talking about multicast? Before 230



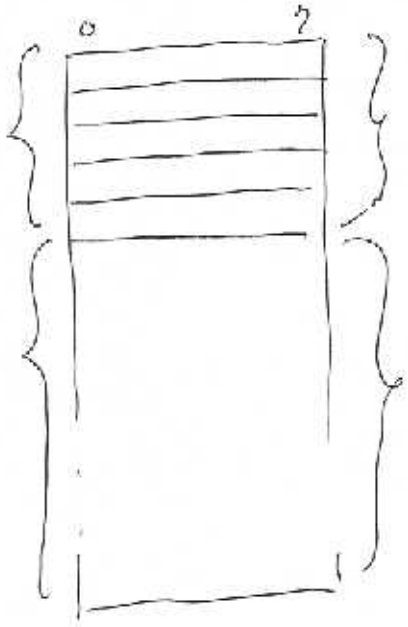


"IP over IP"

ATM.

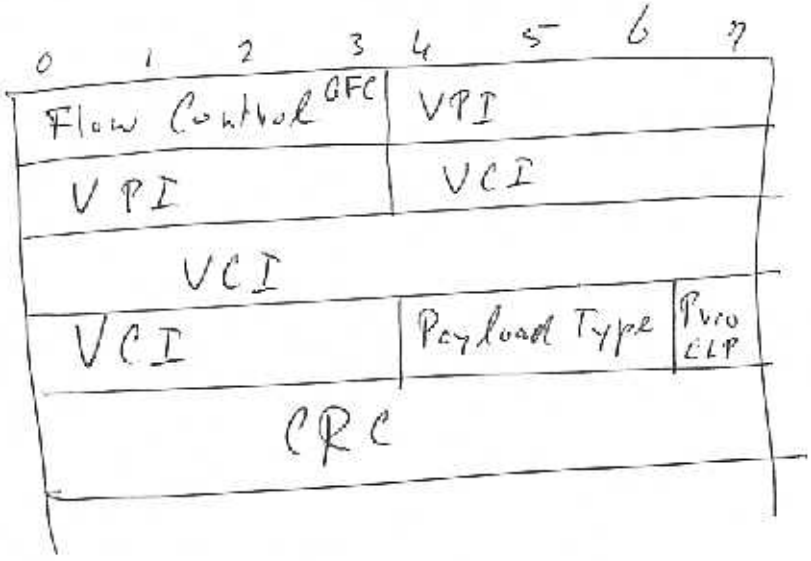
Not packets, but <u>cells</u>.

Each cell: 48 data bytes
5 header bytes.



48: Compromise between 32 and 64.
(most possible?).

ATM cell header



This is the UNI format. (User - Network Interface).

There also is an NNI (Network - Network Interface) (no GFC, added to VPI).

ATM uses "Virtual Circuits". VCs.

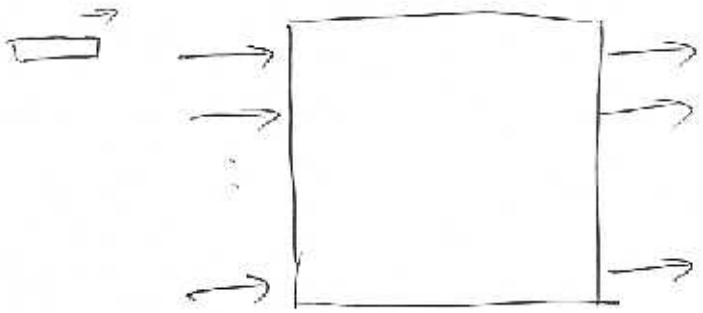
either { Permanent ("provisioned")
Switched ("like phone call").

Anyhow: ATM is <u>circuit oriented</u>.

But with "Random (Varying) Intensity".

Payload Type: "Like"

Switch



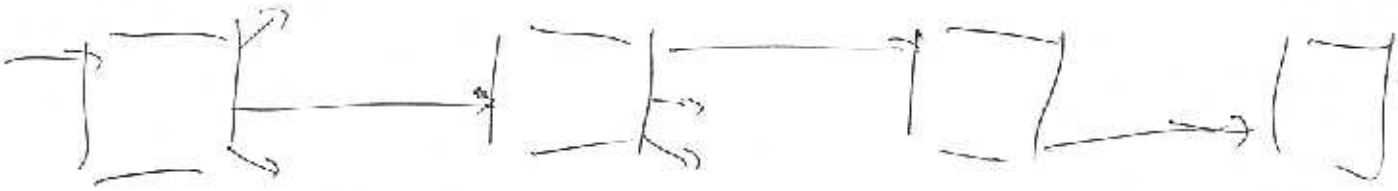
Switch looks at VPI/VCI.

VPI/VCI <u>with</u> input port

determines output port & new VPI/VCI.

So the packet may have a different VPI/VCI on every "leg" (link)

Why is that so?



etc.

There are only $2^8 = 256$ different VPI number

There are $2^{16} = 65536$ different VCI number.

There might not be a single one free on all links. (and then: find it?). ⇐

But: every link almost certainly has at least one free.

ATM: circuit oriented.

virtual circuits must be set-up.