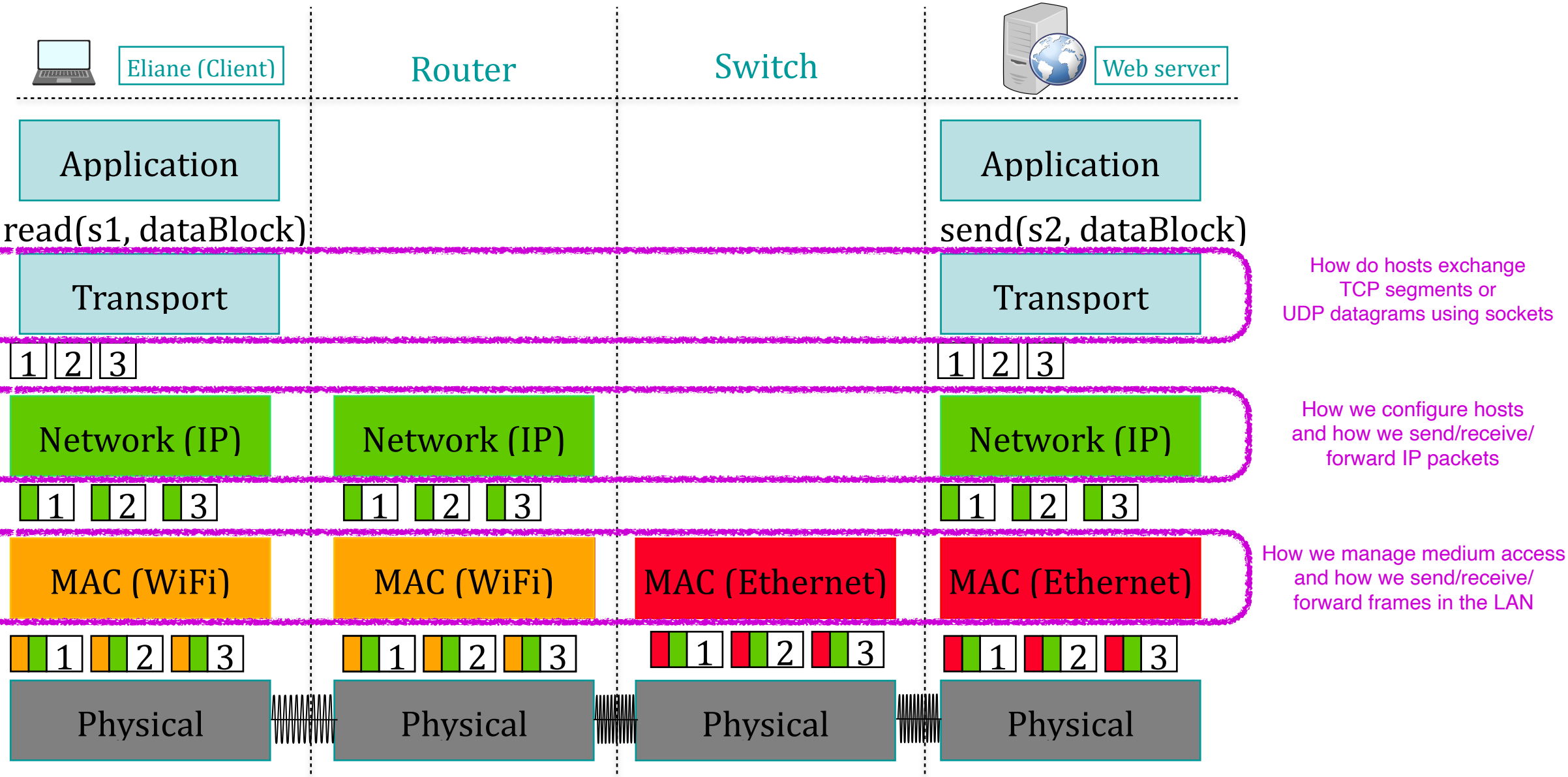
Midterm recap: basic network operation

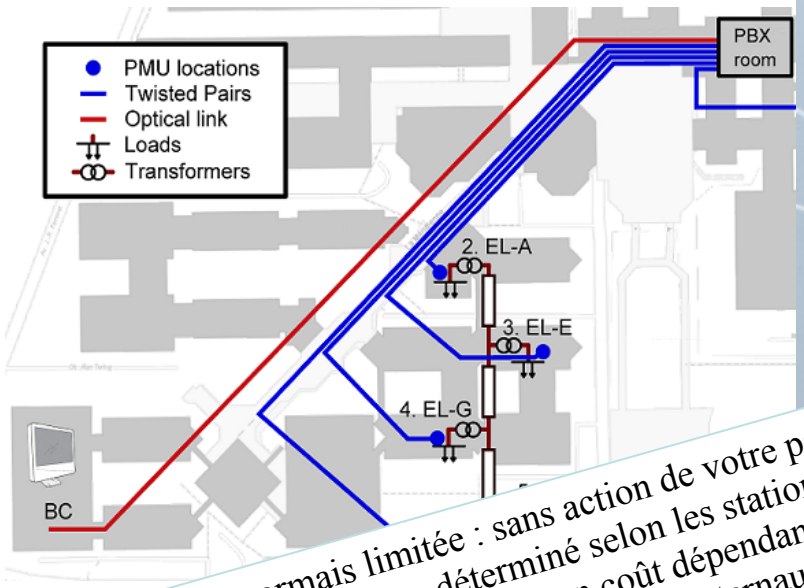


Next: how to address practical issues…

Same layers, but more complicated/interesting concepts:

- How to do “live tv broadcasting” by also reducing the stress at the source
  - use a special case of IP forwarding —> multicast
- How do IP routers populate their forwarding tables?
  - do they all use the same routing protocol/policies, even if they belong to different operators?
  - IGP, BGP
- Can we avoid congestion?
  - can TCP sources use all the available capacity, without suffering significant losses?
  - also, can we ensure that all sources take a “fair share” of the available capacity?

IP Multicast



EPFL

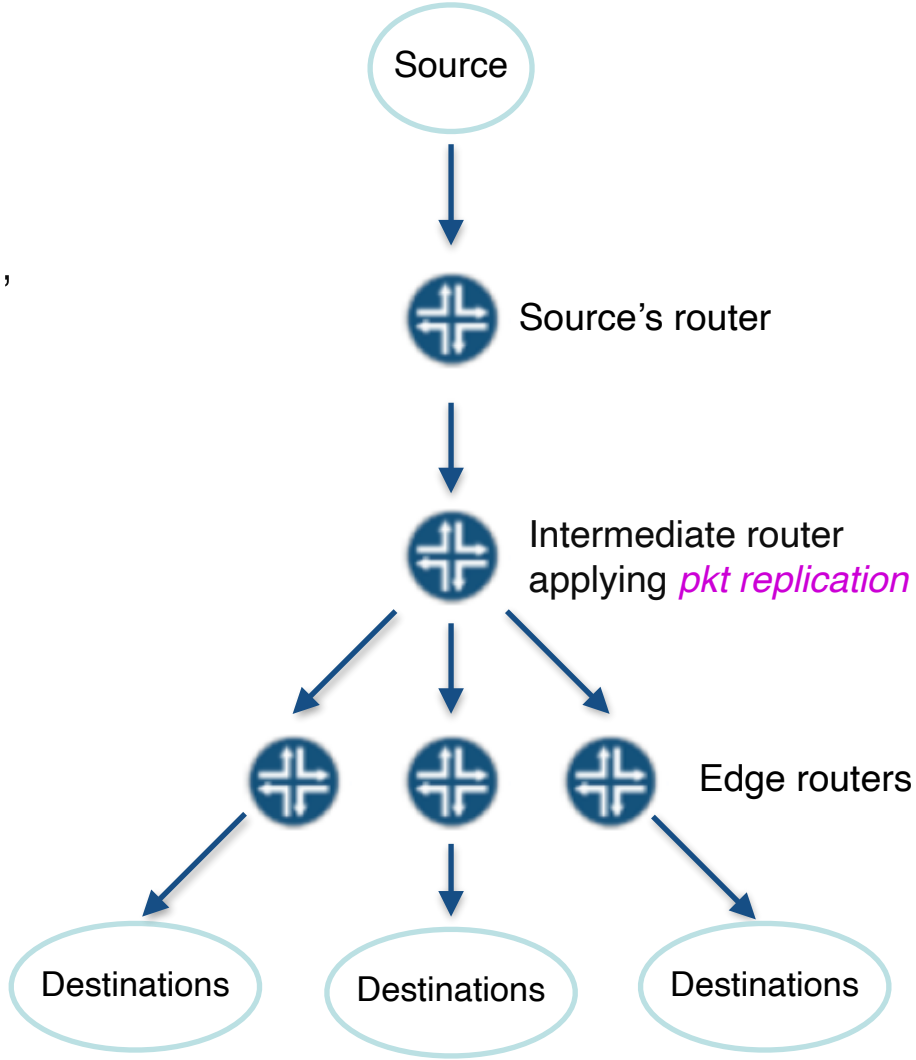
La durée d'écoute est désormais limitée : sans action de votre part (un simple clic), la diffusion s'arrête au bout d'un temps déterminé selon les stations. En effet, pour nous, diffuseurs, les technologies actuelles imposent un coût dépendant de la durée et du nombre d'auditeurs. Plusieurs éléments nous indiquent que les internautes ayant accès à l'internet illimité ne coupent pas l'écoute, lorsqu'ils quittent leur ordinateur allumé. Radio France ne peut continuer à financer pour celui qui n'écoute pas. C'est pourquoi nous avons mis en place ce système de confirmation, un peu contraignant, mais qui nous permet de mieux contrôler les coûts de diffusion.

http://viphttp.vacast.net/V4/radiofrance/fip_bd.m3u

IP Multicast

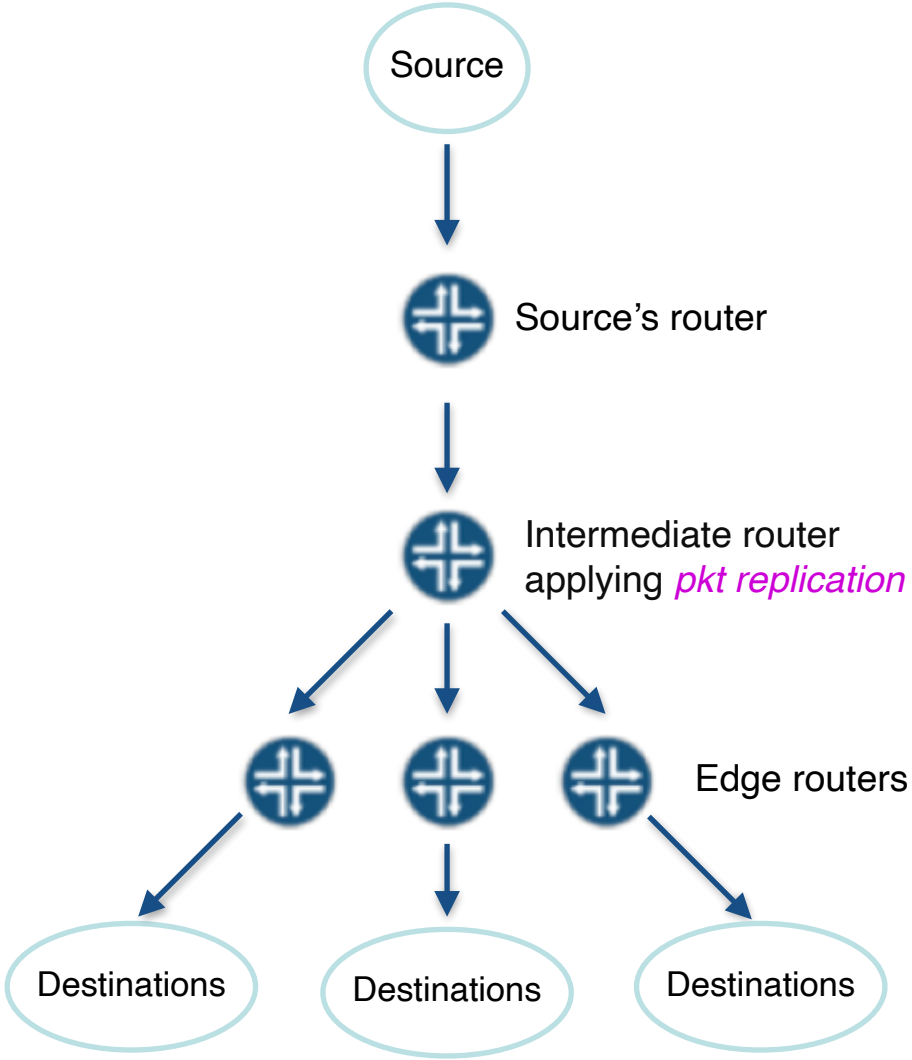
- Recall:
  - Unicast = source sends data to a *unicast* IP address, and a *single* destination receives it
  - Multicast = source sends data to a *multicast* IP address, but *multiple* destinations receive it

- Multicast delivers packets to multiple recipients *without* sending multiple copies from the source
- Traffic is *replicated* at the *network layer*
- Used only:
  - within a *single domain* (= big network under a single administrative entity, composed of multiple subnets)
  - with *UDP* (cannot be done with TCP!)



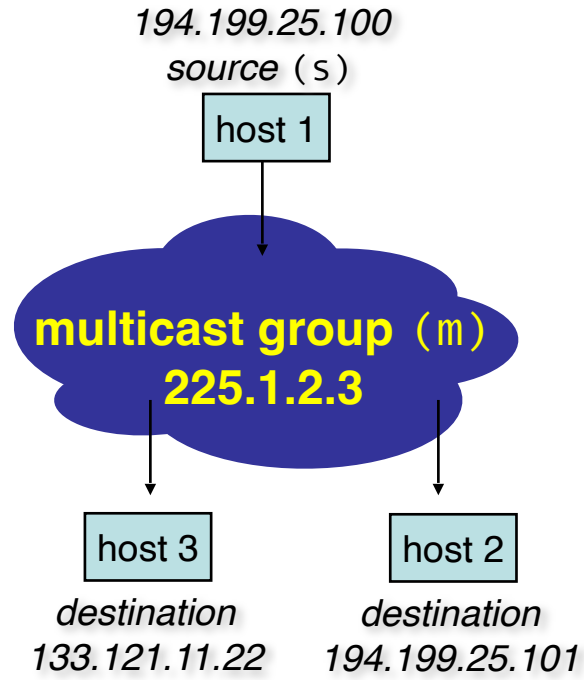
Typical process

- Destinations *subscribe to* a multicast group by sending *join* messages to their routers
  - IGMP (Internet Group Management Protocol, in IPv4) or
  - MLD (Multicast Listener Discovery, in IPv6)
- Routers either build distribution tree via a *reverse path forwarding* (PIM) or use a *source-routing* approach (BIER) [see next slides]
- Source simply sends UDP packets to multicast address
- Intermediate IP routers between source and destinations *replicate traffic*



Multicast addresses and groups

- Reserved IP address spaces:
  - IPv4: **224.0.0.0/4** (i.e. 224.0.0.0 to 239.255.255.255)
  - IPv6: **ff00::/8**, bits 13-16 determine the "scope":
    ff02/16 = same subnet,
    ff05/16 = same domain



- *Any Source Multicast* (ASM) group:
  - the group is identified by the multicast address;
  - *any* source can send to this group
- *Source Specific Multicast* (SSM) group:
  - the group is identified by `(s,m)`
    where `m` is a multicast address and `s` is a (unicast) source IP address;
  - *only* `s` can send to this group
  - default SSM addresses: 232.0.0.0/8 and ff3x::/96 (x=scope bits; e.g. ff35::/96 = site-local) [RFC7371 for more details]

PIM: Protocol Independent Multicast (typical example)

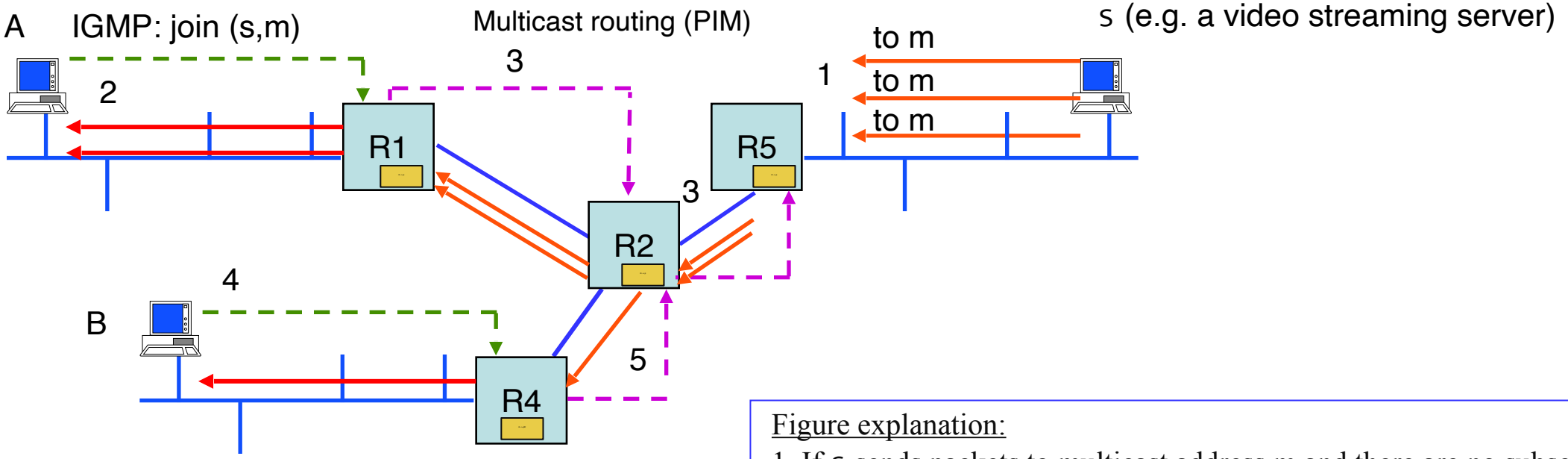


Figure explanation:

1. If s sends packets to multicast address m and there are no subscribed destinations yet, the data is dropped at router R5.

So, we use the following steps:

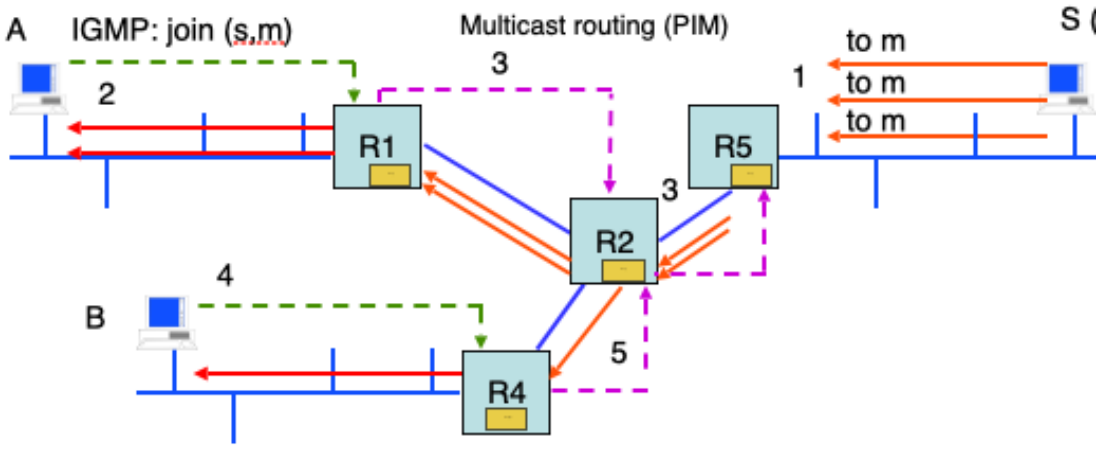
2. A joins the SSM group (s , m).
3. R1 informs the rest of the network that (s , m) has a member at R1 by using e.g. the multicast routing protocol PIM-SM; this results in a tree being built. Data sent by s now reach A.
4. B joins the multicast address m.
5. R4 informs the rest of the network that m has a member at R4; the multicast routing protocol adds branches to the tree. Data sent by s now reach both A and B.

PIM: Protocol Independent Multicast

supports ASM and SSM

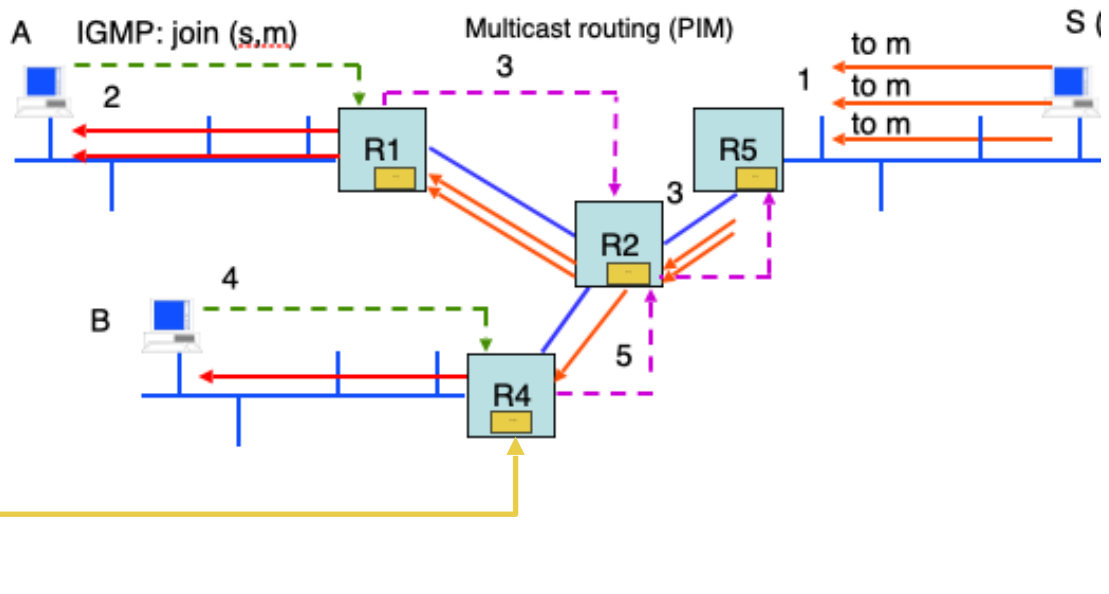
exists in 2 versions

- PIM-DM (*Dense* Mode) makes heavy use of broadcast and can be used only in small, tightly controlled networks
- PIM-SM (*Sparse* Mode) is more reasonable and is used e.g. for TV distribution
  - When used with SSM, PIM-SM uses *reverse path forwarding (RPF)*: when a router (such as R1) needs to add a receiver, it sends a PIM/JOIN router message towards the source, using unicast routing
  - This creates the distribution tree *on the fly*
  - [PIM-SM for ASM is more complicated; it uses one multicast router as Rendez-vous Point (RP): destination routers create a tree from RP, using RPF; router closest to source sends source packets to RP; if there exists an interested receiver in the domain, RP creates a tree from source (using RPF) otherwise drops; destinations create trees from sources, using RPF.]



PIM-enabled routers must keep per-flow state information

In addition to longest prefix match, a multicast IP router does *exact match* for multicast groups



Multicast *state information* is dynamically kept in router for every known multicast group:

(s, m) or (,m)* // id of group
valid incoming interfaces // for security
outgoing interfaces // this is the routing info
other information required by multicast routing protocol

This *per-flow state information* cannot be aggregated based on prefix: *scalability issues*

BIER (Bit Index Explicit Replication)

Why ? Multicast routing requires routers to keep per-flow state (dynamic, depends on who listens to the group) and apply exact match

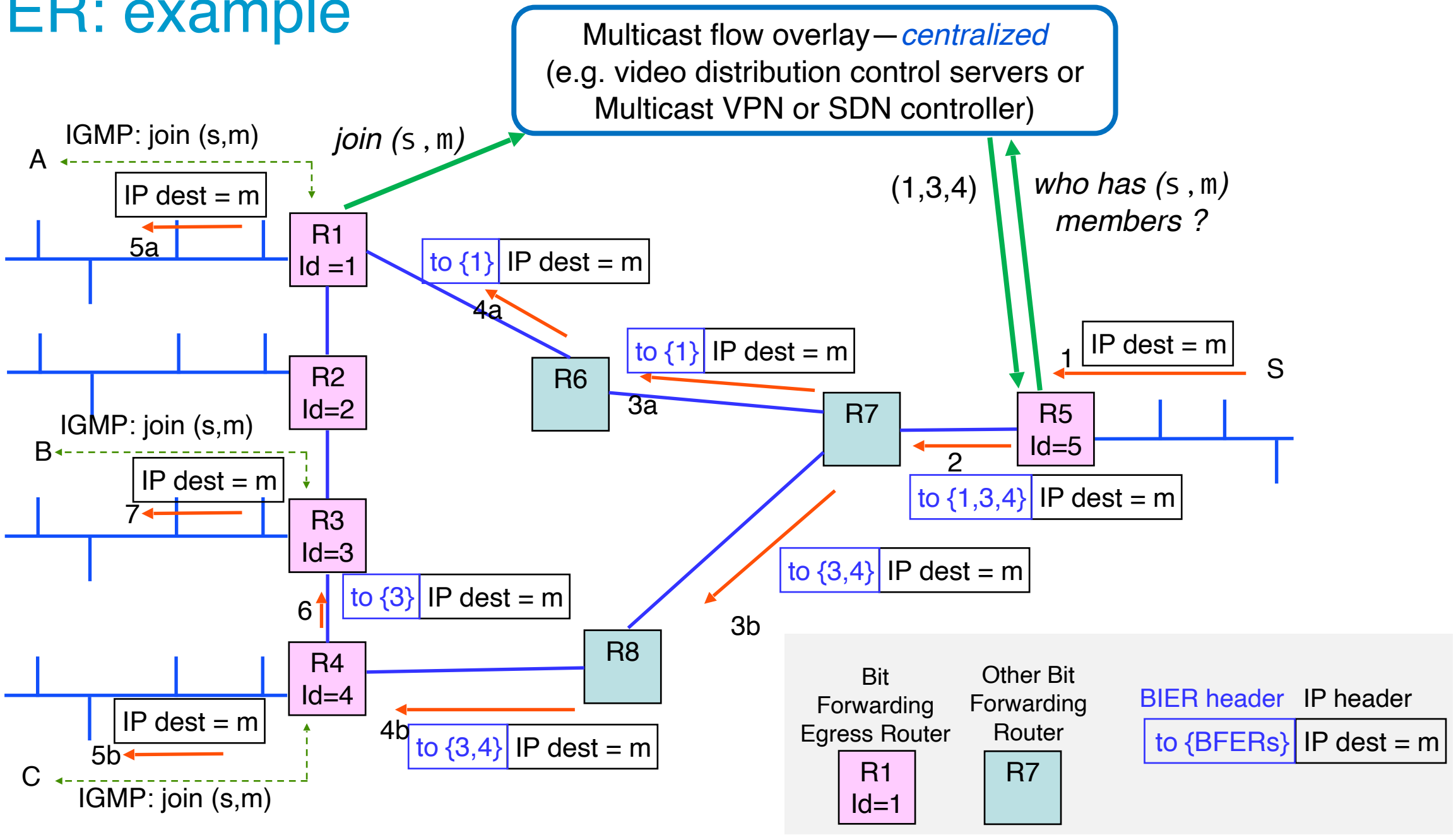
➡ This causes a stress to *backbone* routers; so, we need an alternative that *scales better*

How ?

BIER uses a *centralized* entity and an *extension* header:

- A multicast packet has a *BIER header* that contains (roughly speaking) the list of destination BIER routers (Bit-Forwarding Egress Routers, BFERs)
- BIER routers rely on unicast routing in order to determine how to deliver a packet to the set of BFERs indicated in the BIER header
  - If several BFERs are reached via *different* next-hops, BIER routers *replicate* packet
  - Then send *only a single* copy to each next hop for *all* BFERs that are reached through this next hop
  - In this copy, the BIER header is modified accordingly

BIER: example



Group membership information is distributed by an external infrastructure called “multicast flow overlay”, for example: a special set of servers used to control the distribution of the video content, or a system to manage multicast virtual private networks using MPLS and BGP (see later in MPLS lecture) or a central network controller (SDN).

All routers on the figure are BIER routers (Bit Forwarding Routers, BFRs). Routers R1-R5 are egress routers (they need to forward multicast packets to the outside); such routers have a BFR-id, for example R1’s BIER-id is 1.

Router R5 learns from the multicast flow overlay that the group (s,m) has members in routers with BFR-ids 1,3 and 4.

1. Router R5 has an IP multicast packet to send to m, from s. R5 knows that it should send the packet to R1, R3 and R4. From unicast routing (classic IP forwarding table), R5 finds that R1, R3 and R4 are reached via the same next-hop (R7) therefore R5 sends one packet with BIER header (1,3,4) to R7.

2. From unicast routing, R7 finds that R3 and R4 are reached via the same next-hop (R8) but R1 requires a different next-hop. Therefore, R7 duplicates the packet and creates 2 packets, one with BIER header (1), sent to R6, and one with BIER header (3,4), sent to R8.

3a. R6 forwards the packet to R1.

4a, 5a. R1 belongs to the destination list contained in the BIER header, therefore R1 knows it should forward the packet using classic multicast. The BIER header is removed and the packet is forwarded to the west LAN interface where A can receive it.

3b. From its forwarding table, R8 finds that R3 and R4 are reached via the same next-hop (R4). Therefore, R8 sends the packet to R4.

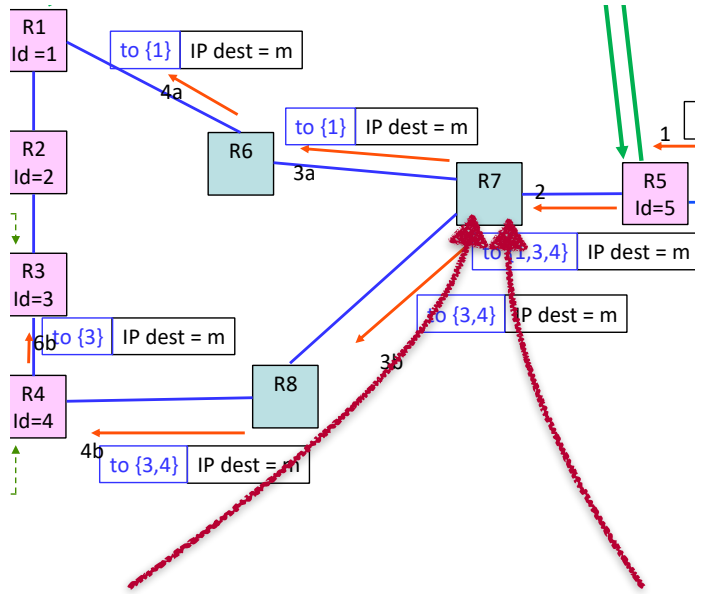
4b, 5b. R4 belongs to the destination list contained in the BIER header, therefore R4 knows it should forward the packet using native multicast. The BIER header becomes (3). 5b. The BIER header is removed and the packet is forwarded to R4’s west LAN interface where C can receive it.

6b. R4 sends one copy of the packet to R3.

7b. The BIER header is removed and the packet is forwarded to R3’s west LAN interface where B can receive it.

BIER: packet replication

- For each destination BFER, BIER router pre-computes (based on the IP forwarding table) a "forwarding bit mask" that indicates the set of destination BFERs that are reached by the same next-hop.
- To forward a packet, with destination set S, BIER router runs:
  1. Send a copy to 1st destination in S with destination set = $S \cap S_1$, where S_1 is the forwarding bit mask of 1st destination.
  2. If $S \backslash S_1 \neq \emptyset$, duplicate the packet but with destination $S \backslash S_1$ and goto 1; else break.



Example at router R7:

R7 receives packet with destination set $S = \{1,3,4\}$:

1. First destination in S is 1, R7 uses Bier Index Forwarding table and finds that the forwarding bit mask for destination 1 is $S_1 = \{1,2\}$: sends a copy of packet to next-hop (R6), with destination set $S' = S \cap S_1 = \{1\}$
2. $S'' = S \backslash \{1,2\} = \{3,4\}$ is not empty; R7 duplicates packet with destination S'' and applies the same: first destination is 3, so R7 sends a copy to next-hop (R8) with destination set $\{3,4\}$.
3. Now $\{3,4\} \backslash \{3,4\} = \emptyset$, so it breaks.

BIER Index Forwarding Table at R7

Dest. BFER	Forwarding bit mask	Next-Hop
1	{1,2}	R6
2	{1,2}	R6
3	{3,4}	R8
4	{3,4}	R8
5	{5}	R5

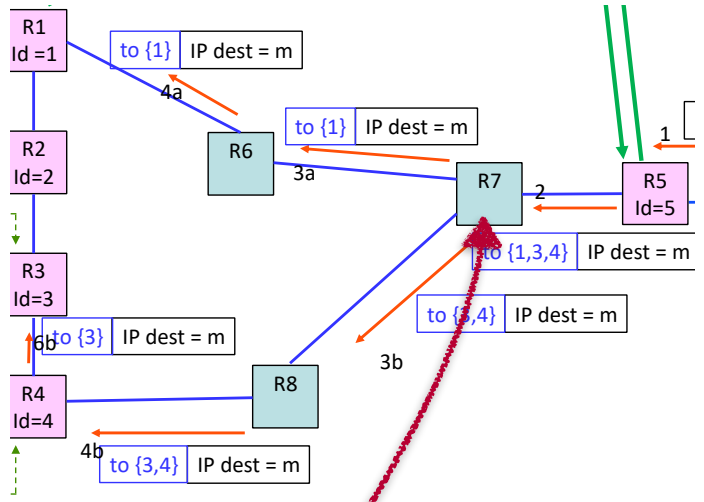
IP Forwarding Table at R7

Dest. IP addr	Next-Hop
R1	R6
R2	R6
R3	R8
R4	R8
R5	R5

How to optimize processing at BIER Routers?

- For each destination BFER, BIER router encodes the forwarding bit mask in a *bitstring*
  - example for 5 possible BFERs:

    Destination BFERs = {1,3,4} —> bitstring = 01101
    Destination BFERs = {3,4} —> bitstring = 01100
    Destination BFERs = {1} —> bitstring = 00001
- Set intersection becomes a *bitwise AND with mask*
- Set difference becomes a *bitwise AND with the bit-inverted mask*
- More complicated mechanisms exist, if the #of BFERs is large [RFC 8279]



Example of BIER forwarding at router R7:

R7 receives packet with (bitstring) BIER header = 01101

1. Least significant destination is 1. R7 looks into Bier Index Forwarding table and finds the corresponding mask; then applies an **AND of the BIER header with the mask** 00011 and sends a copy of packet to next-hop (R6) with header bitstring = 00001; R7 computes the remaining bitstring (by **AND-ing the BIER header with inverse of mask**) as 01100.
2. The outcome is non-zero; so R7 processes the remaining bistring = 01100 by applying the same: least significant destination is 3; so R7 sends a copy to next-hop (R8) with header 01100.
3. Now, the remaining bitstring is all-zeros (00000); so it breaks.

Bier Index Forwarding Table at R7

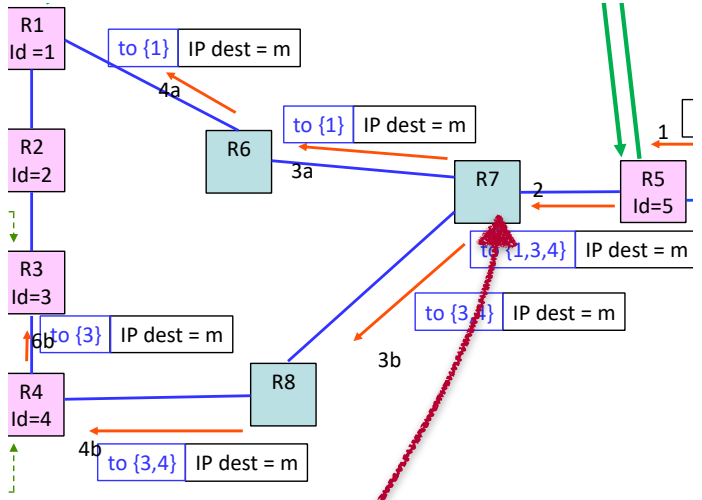
Dest. BFER	Forwarding bit mask	Next-Hop
1	**0 0011**	R6
2	**0 0011**	R6
3	**0 1100**	R8
4	**0 1100**	R8
5	**1 0000**	R5

BIER Table contains static info, no per-flow state

BIER Routers — recap

- In addition to IP unicast forwarding table (longest prefix match), a BIER router does bitstring processing using a bit forwarding table
- A Bit Forwarding Ingress Router (BFIR, such as R5) must map destination multicast address to a BIER header
  - requires *out of band* mechanism
  - this is the *only* (dynamic) per-flow information cached by BIER
- BIER forwarding table, called Bit Index forwarding table, is automatically derived from router's unicast IP forwarding table.
- Inside a BIER domain, multicast packets have an additional header and the IP destination address is not used (*tunneling*).

Multiple BIER domains can be interconnected by BFIR-BFER interconnection.



Bier Index Forwarding Table at R7

Dest. BFER	Forwarding bit mask	Next-Hop
1	**0 0011**	R6
2	**0 0011**	R6
3	**0 1100**	R8
4	**0 1100**	R8
5	**1 0000**	R5

BIER Table contains static info, no per-flow state

Is there Multicast ARP (address resolution protocol)?

- *No,* multicast MAC address is *algorithmically* derived from multicast IP address:
  - Last 23 bits of IPv4 multicast address are used in MAC address
  - Last 32 bits of IPv6 multicast address are used in MAC address

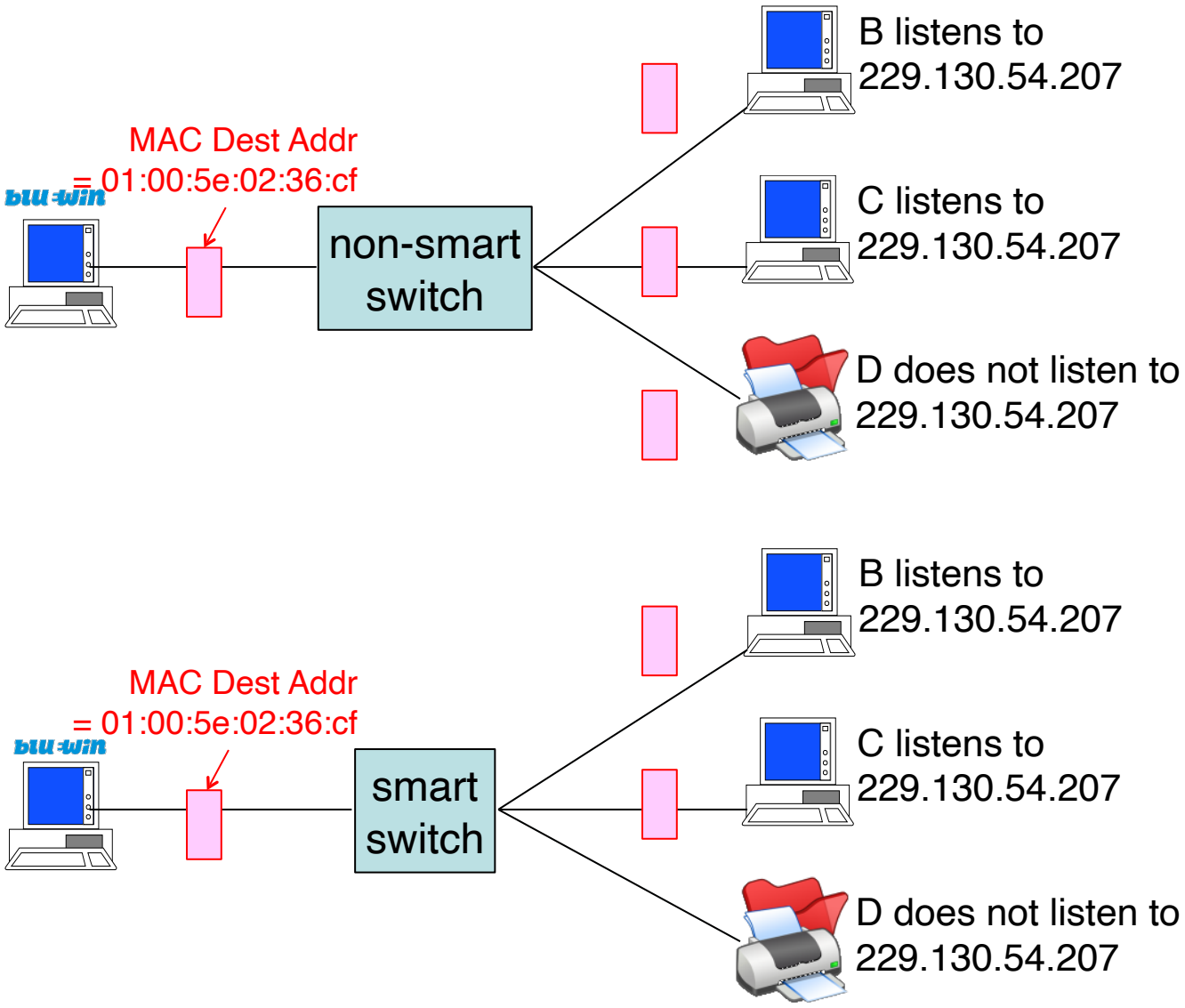
- Note:
  - Multicast MAC address depends only on multicast IP address `m`, not on source address `s`, even if `m` is an SSM address
  - Several multicast IP addresses may yield the same MAC address
    - packets received unnecessarily at the MAC layer are removed by the OS; hopefully this happens rarely

1st bit of hextet is 0

MAC multicast addr.	*Used for*
01-00-5e-YX-XX-XX	IPv4 multicast
33-33-XX-XX-XX-XX	IPv6 multicast

IP dest address	229.130.54.207
IP dest address (hexa)	e5-82-36-cf
IP dest address (bin)	…-10000010-…
Keep last 23 bits (bin)	…-00000010-…
Keep last 23 bits (hexa)	02-36-cf
MAC address	01-00-5e-02-36-cf

MAC Multicast: how do switches handle multicast frames?



Some (non smart) switches simply treat multicast frames as broadcast.

Some smarter switches:

- listen to IGMP/MLD subscription messages and overhear who listens
- deliver only to intended recipients (IGMP or MLD snooping)
- but do not distinguish SSM from ASM.

Security of IP Multicast

IP multicast w/ or w/o BIER makes attacks easier (e.g. *Denial of Service*, witty worm)

mitigations: limit multicast rate and number of groups;
control which multicast group is allowed (*access lists*)

SSM is safer as routers and destination can reject unwanted sources

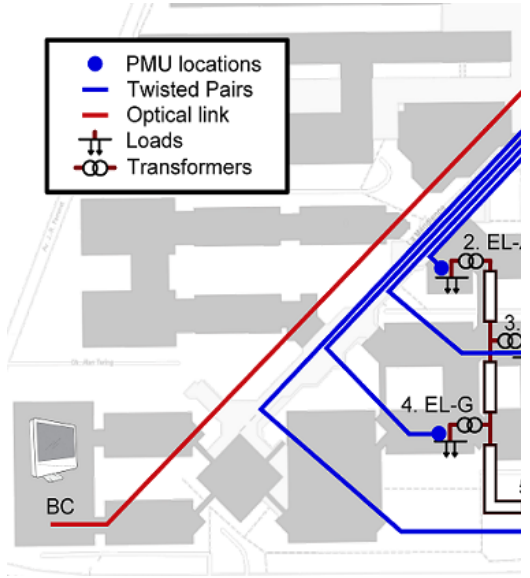
IGMP/MLD is not secure and has the same problems as ARP/NDP

mitigated by same mechanisms: sniffing switches observe all traffic and implement *access control*

Multicast-capable networks must deploy exhaustive *filtering* and *monitoring* tools to limit potential damage.

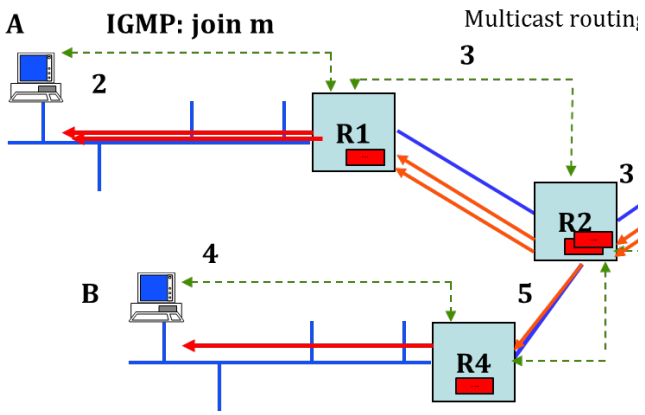
Multicast in Practice

- Multicast is good for sources : one packet sent for *n* destinations -- replication is done O*(log(n))* times
- Multicast is not supported everywhere, but is used in:
  - Internet TV distribution (PIM-SM / BIER)
  - EPFL and other academic networks (PIM-SM)
  - Data Center Virtualization Services (BIER)
  - some corporate networks for news, sensor streaming, time synchronization, large videoconferences etc…
  - industrial networks (smart grids, factory automation)
- Works only with UDP, not with TCP (not easy to handle TCP acks from multiple receivers?)
  - need to handle errors, usually via redundancy (extra traffic)



Say what is true

A. A
B. B
C. C
D. A and B
E. A and C
F. B and C
G. All
H. None
I. I don't know



Go to web.speakup.info or download speakup app

Join room 46045

A. In order to send to a multicast group a system must first join the group with IGMP or MLD
B. In order to receive from a multicast group a system must first join the group with IGMP or MLD
C. A system can know whether a packet is multicast by analyzing the IP destination address.

Solution

F

R is a backbone router used for multicast distribution. In which case must R keep per-flow information ?

A. If R uses PIM-SM as multicast routing protocol
B. If R uses BIER as multicast routing protocol
C. In both cases
D. In neither case
E. I don’t know

Go to web.speakup.info or download speakup app

Join room 46045

R is an *ingress* edge router used for multicast distribution. In which case must R cache per-flow information ?

A. If R uses PIM-SM as multicast routing protocol
B. If R uses BIER as multicast routing protocol
C. In both cases
D. In neither case
E. I don’t know

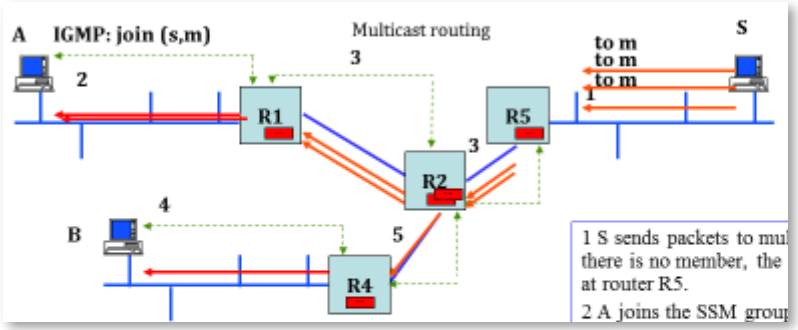
Go to web.speakup.info or download speakup app

Join room 46045

Solution

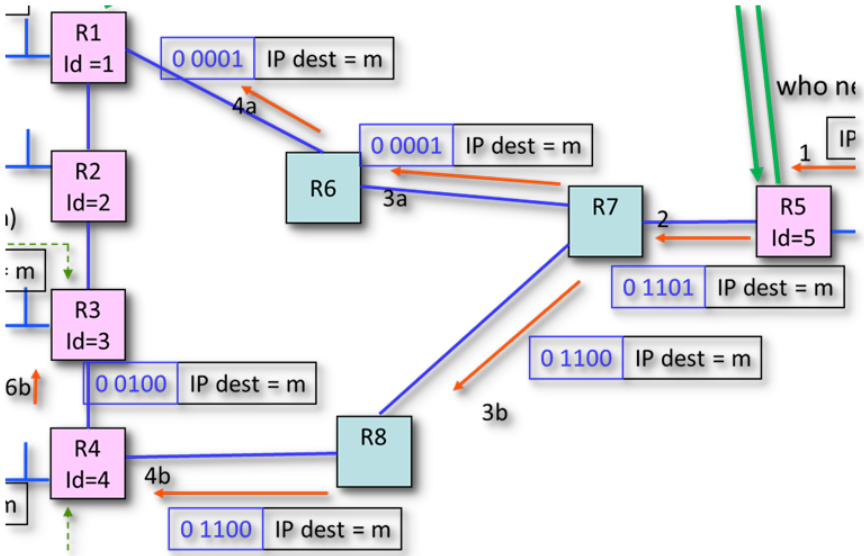
Answer to first question: A

Answer to second question: C (A and B)



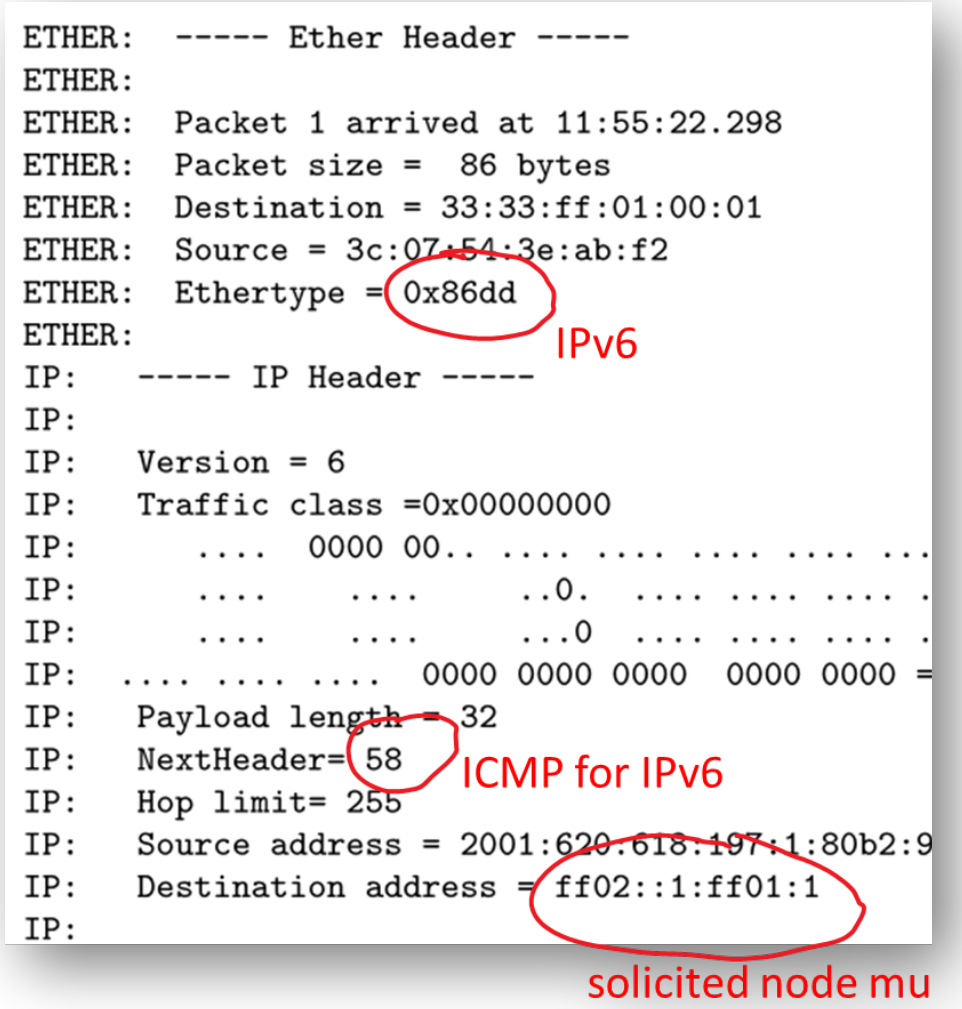
With PIM-SM, all routers keep per-flow information.

With BIER, only ingress routers (such as R5 on the figure) need to *cache* per-flow information.



The destination MAC address is…

A. A group address derived from the last 23 bits of the IPv6 target address
B. A group address derived from the last 24 bits of the IPv6 target address
C. A group address derived from the last 32 bits of the IPv6 target address
D. A broadcast address
E. The MAC address of an ARP server
F. I don't know



Solution

Answer C

For multicast, the destination MAC address is automatically derived from the IP destination MAC address, by taking the last 32 bits (IPv6) or the last 23 bits (IPv4).

The IPv6 destination address is ff02::1:ff01:1
The last 32 bits correspond to the last 8 hexadecimal symbols, i.e. ff01:1; in uncompressed form, the 8 hexadecimal symbols are ff01:0001. The MAC destination address is therefore 33:33:ff:01:00:01

Summary

- IP multicast came as an after-thought and uses a different principle than IP unicast (exact match versus longest prefix match)—deployed only in specific network domains

- IP multicast addresses cannot be aggregated
- Traditional multicast (PIM) requires per-flow state in routers —> does not scale well
- BIER avoids this problem

- BIER uses a different forwarding principle, based on bistrings (which represent sets of destinations).