IK1350 Protocols in Computer Networks/ Protokoll i datornätverk Spring 2008, Period 3

Module 9: Multicasting and RSVP

Lecture notes of G. Q. Maguire Jr.

KTH Information and Communication Technology

For use in conjunction with *TCP/IP Protocol Suite*, by Behrouz A. Forouzan, 3rd Edition, McGraw-Hill, 2006.

For this lecture: Chapters 10, 15

Outline

- Multicast
- IGMP
- RSVP

Multicast and IGMP

Broadcast and Multicast

Traditionally the Internet was designed for unicast communication (one sender and one receiver) communication.

Increasing use of multimedia (video and audio) on the Internet

- **One-to-many** and **many-to-many** communication is increasing
- In order to support these in a *scalable* fashion we use **multicasting**.
- Replicating UDP packets **where** paths diverge (i.e., split)

MBONE was an experimental multicast network which operated for a number of years. (see for example *http://www-mice.cs.ucl.ac.uk/multimedia/software/* and *http://www.ripe.net/ripe/wg/mbone/home.html*)

Multicasting is useful for:

- Delivery to multiple recipients
  - reduces traffic, otherwise each would have to be sent its own copy ("internet radio/TV")
- Solicitation of service (service/server discovery)
  - Not doing a broadcast saves interrupting many clients

Filtering up the protocol stack

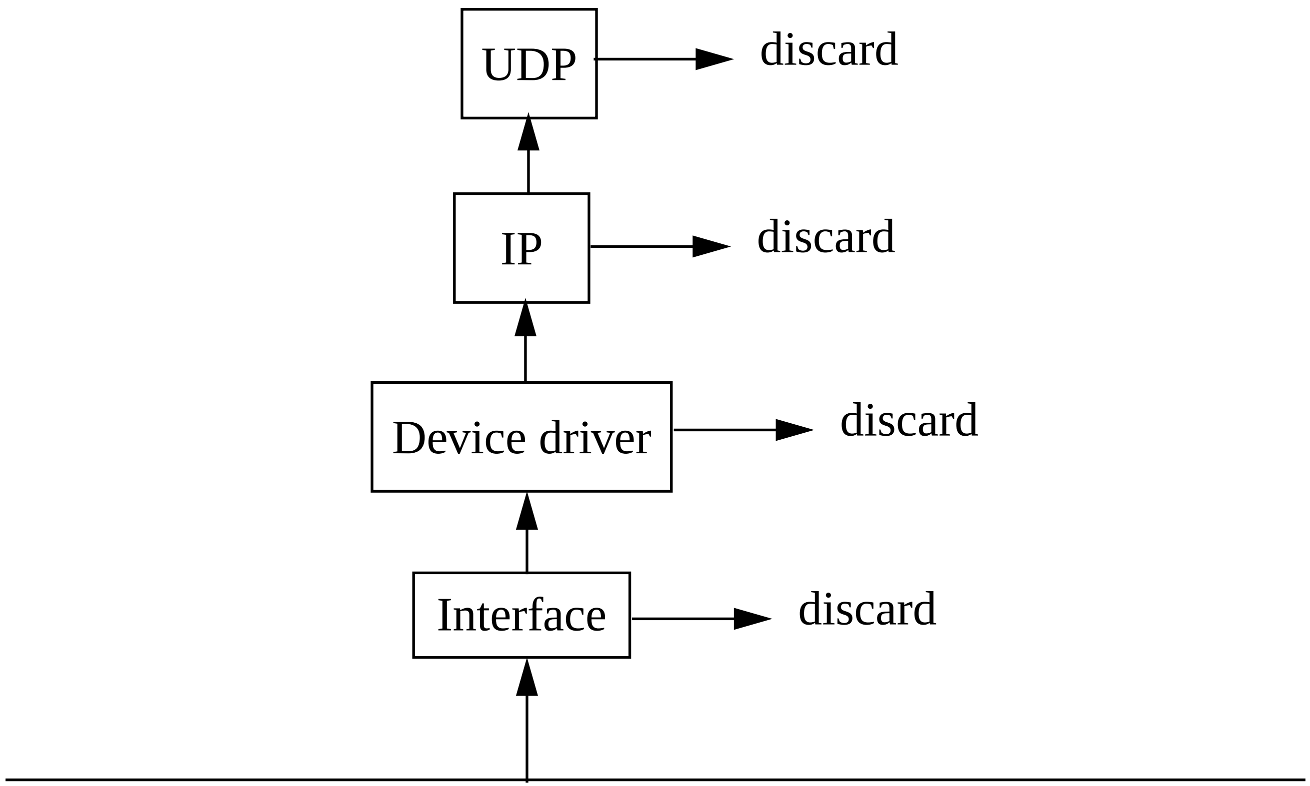


Figure 91: Filtering which takes place as you go up the TCP/IP stack
(see Stevens, Volume 1, figure 12.1, pg. 170)

We would like to filter as soon as possible to avoid load on the machine.

Broadcasting

- Limited Broadcast
  - IP address: 255.255.255.255
  - never forwarded by routers
  - What if you are multihomed? (i.e., attached to several networks)
    - Most BSD systems just send on first configured interface
    - routed and rwhod - determine all interfaces on host and send a copy on each (which is capable of broadcasting)
- Net-directed Broadcast
  - IP address: netid.255.255.255 or net.id.255.255 or net.i.d.255 (depending on the class of the network)
  - routers must forward
- Subnet-Directed Broadcast
  - IP address: netid | subnetid | hostID, where hostID = all ones
- All-subnets-directed Broadcast
  - IP address: netid | subnetid | hostID, where hostID = all ones and subnetID = all ones
  - generally regarded as obsolete!

To send a UDP datagram to a broadcast address set SO_BROADCAST

Other approaches to One-to-Many and Many-to-Many communication

Connection oriented approaches have problems:

- large user burden
- have to know other participants
- have to order links in advance
- poor scaling, worst case $O(N^2)$

Alternative centralized model

CU-SeeME uses another model - a Reflector (a centralized model)

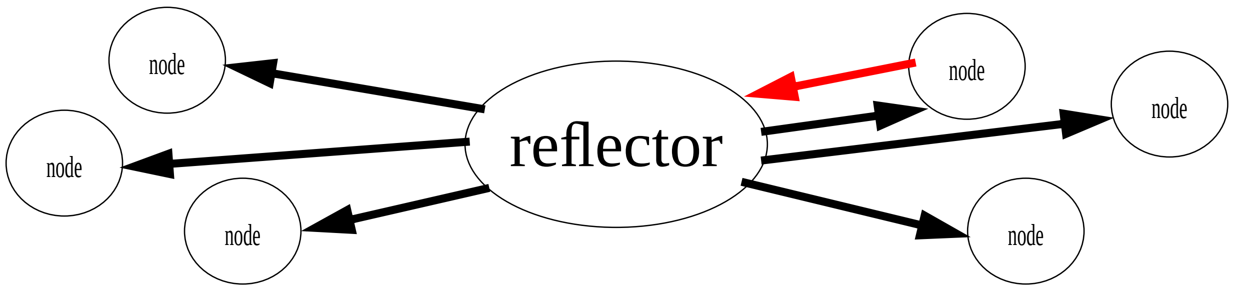


Figure 92: Reflector

- All sites send to one site (the reflector) overcomes the N^2 problems
- The reflector sends copies to all sites

Problems:

- Does **not** scale well
- Multiple copies sent over the same link
- Central site must know all who participate

Behavior could be changed by explicitly building a tree of reflectors - but then you are moving over to Steve Deering's model.

Multicast Backbone (MBONE)

Expanding multicasting across WANs

World-wide, IP-based, real-time conferencing over the Internet (via the MBONE) in daily use for several years with more than 20,000 users in more than 1,500 networks in events carrier to 30 countries.

For a nice paper examining multicast traffic see: “Measurements and Observations of IP Multicast Traffic” by Bruce A. Mah <bmah@CS.Berkeley.EDU>, The Tenet Group, University of California at Berkeley, and International Computer Science Institute, CSD-94-858, 1994,12 pages:

http://www.kitchenlab.org/www/bmah/Papers/Ipmcast-TechReport.pdf/

IP Multicast scales well

- End-nodes know nothing about topology
  - Dynamically changes of topology possible, hosts join and leave as they wish
- Routers know nothing about “conversations”
  - changes can be done without global coordination
  - no end-to-end state to move around

Participants view of Multicast

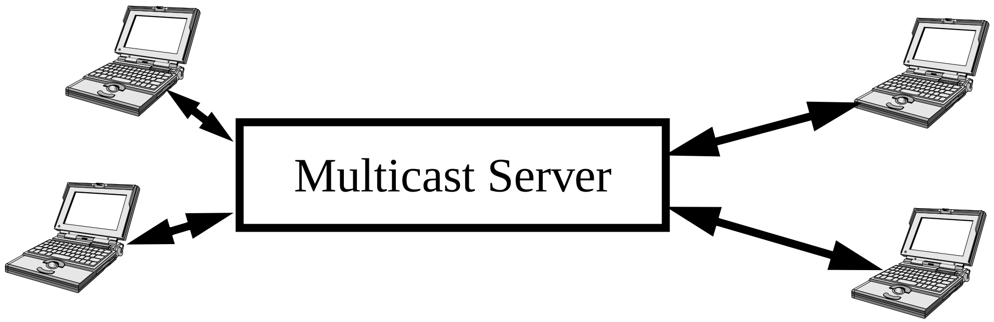


Figure 93: MBONE behaves as if there were a multicast server, but this functionality is distributed not centralized.

Core Problem

How to do efficient multipoint distribution (i.e., at most one copy of a packet crossing any particular link) without exposing topology to end-nodes

Applications

- Conference calls (without sending N copies sent for N recipients)
- Dissemination of information (stock prices, "radio stations", …)
- Dissemination of one result for many similar requests (boot information, video)
- Unix tools:
  - nv - network video
  - vat - visual audio tool
  - wb - whiteboard
  - sd - session directory
  - …

Steve Deering's Multicast

Dynamically constructs efficient delivery trees from sender(s) to receiver(s)

- Key is to compute a spanning tree of multicast routers

Simple service model:

- receivers announce interest in some multicast address
- senders just send to that address
- routers conspire to deliver sender's data to all interested receivers
  - so the real work falls once again to the routers, not the end nodes
  - Note that the assumption here is that it is worth loading the routers with this extra work, because it reduces the traffic which has to be carried.

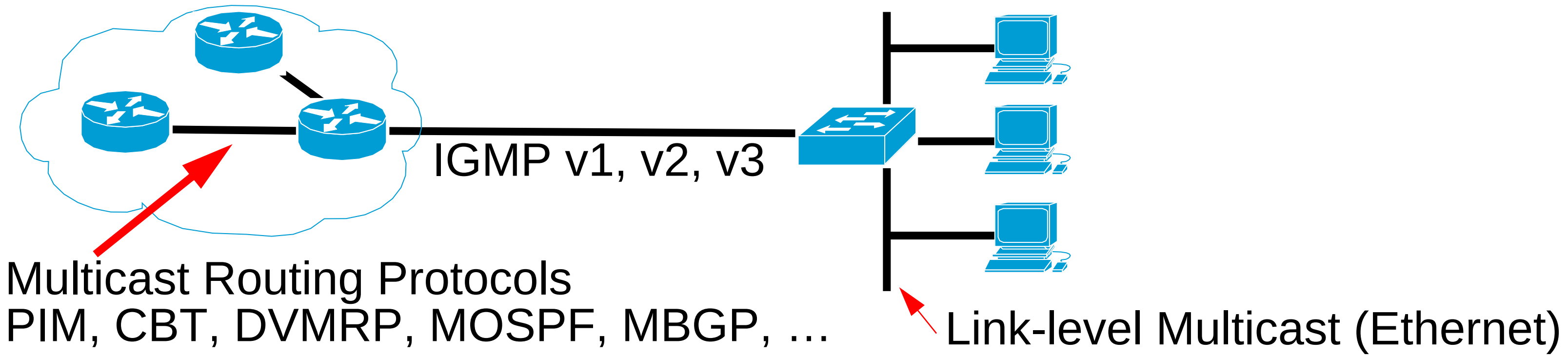


Figure 94: IP Multicast Service Model

IP WAN Multicast Requirements

- Convention for recognizing IP multicast
- Convention for mapping IP to LAN address
- Protocol for end nodes to inform their adjacent routers,
- Protocol for routers to inform neighbor routers
- Algorithm to calculate a spanning tree for message flow
- Transmit data packets along this tree

Multicasting IP addresses

Multicast Group Addresses - “Class D” IP address

- High 4 bits are 0x1110; which corresponds to the range 224.0.0.0 through 239.255.255.255
- host group ≡ set of hosts listening to a given address
  - membership is dynamic - hosts can enter and leave at will
  - no restriction on the number of hosts in a host group
  - a host need not belong in order to send to a given host group
  - permanent host groups - assigned well know addresses by IANA
    - 224.0.0.1 - all systems on this subnet
    - 224.0.0.2 - all routers on this subnet
    - 224.0.0.4 - DVMRP routers
    - 224.0.0.9 - RIP-2 routers
    - 224.0.1.1 - Network Time Protocol (NTP) - see RFC 1305 and RFC 1769 (SNTP)
    - 224.0.1.2 - SGI’s dogfight application

Internet Multicast Addresses

http://www.iana.org/assignments/multicast-addresses listed in DNS under MCAST.NET and 224.IN-ADDR.ARPA.

- 224.0.0.0 - 224.0.0.255 (224.0.0/24) Local Network Control Block
- 224.0.1.0 - 224.0.1.255 (224.0.1/24) Internetwork Control Block
- 224.0.2.0 - 224.0.255.0 AD-HOC Block
- 224.1.0.0 - 224.1.255.255 (224.1/16) ST Multicast Groups
- 224.2.0.0 - 224.2.255.255 (224.2/16) SDP/SAP Block
- 224.3.0.0 - 224.251.255.255 Reserved
- 239.0.0.0/8 Administratively Scoped
  - 239.000.000.000-239.063.255.255 Reserved
  - 239.064.000.000-239.127.255.255 Reserved
  - 239.128.000.000-239.191.255.255 Reserved
  - 239.192.000.000-239.251.255.255 Organization-Local Scope
  - 239.252.0.0/16 Site-Local Scope (reserved)
  - 239.253.0.0/16 Site-Local Scope (reserved)
  - 239.254.0.0/16 Site-Local Scope (reserved)
  - 239.255.0.0/16 Site-Local Scope
  - 239.255.002.002 rasadv

Converting Multicast Group to Ethernet Address

Could have been a simple mapping of the 28 bits of multicast group to 28 bits of Ethernet multicast space (which is 2^{27} in size), but this would have meant that IEEE would have to allocate multiple blocks of MAC addresses to this purpose, but:

- they didn't want to allocate multiple blocks to one organization
- a block of 2^{24} addresses costs $1,000 ==> $16K for 2^{27} addresses

Mapping Multicast (Class D) address to Ethernet MAC Address

Solution IANA has one block of ethernet addresses 00:00:5e as the high 24 bits

- they decided to give 1/2 this address space to multicast -- thus multicast has the address range: 00:00:5e:00:00:00 to 00:00:5e:7f:ff:ff
- since the first bit of an ethernet multicast has a low order 1 bit (which is the first bit transmitted in link layer order), the addresses are 01:00:5e:00:00:00 to 01:00:5e:7f:ff:ff
- thus there are 23 bits available for use by the 28 bits of the multicast group ID; we just use the bottom 23 bits
  - therefore 32 different multicast group addresses map to the same ethernet address
  - the IP layer will have to sort these 32 out
  - thus although the filtering is not complete, it is very significant

The multicast datagrams are delivered to all processes that belong to the same multicast group.

To extend beyond a single subnet we use IGMP.

Problems

Unfortunately many links do not support link layer multicasts at all!

For example:

- ATM
- Frame relay
- many cellular wireless standards
- …

IGMP: Internet Group Management Protocol

IGMP: Internet Group Management Protocol (RFC 1112) [70]:

- Used by hosts and routers to know which hosts currently belong to which multicast groups.
- multicast routers have to know which interface to forward datagrams to
- IGMP like ICMP is part of the IP layer and is transmitted using IP datagrams (protocol = 2) I

IP header	IGMP message
20 bytes	8 bytes

Figure 95: Encapsulation of IGMP message in IP datagram (see Stevens, Vol. 1, figure 13.1, pg. 179)

4 bit IGMP version (1)	4-bit IGMP type (1-2)	Unused	16 bit checksum
32 bit group address (class D IP address)			

- type =1 ⇒ query sent by a router, type =2 ⇒ response sent by a host

How does IGMP fit into the protocol stack

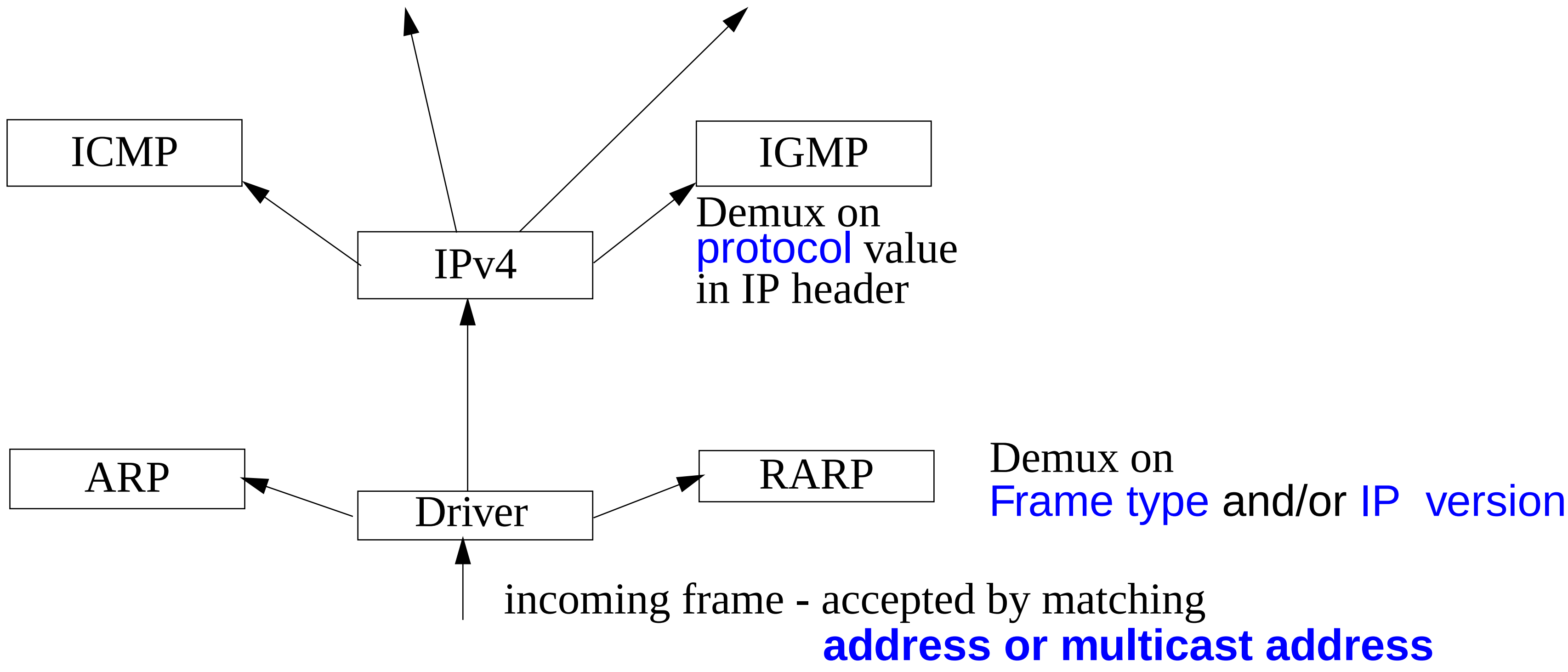


Figure 96: IGMP - adapted from earlier figure (See “Demultiplexing” on page 35.)

So it used IP packets with a protocol value of 2.

Joining a Multicast Group

- a process joins a multicast group on a given interface
- host keeps a table of all groups which have a reference count ≥ 1

IGMP Reports and Queries

- Hosts sends a report when first process joins a given group
- Nothing is sent when processes leave (not even when the last leaves), but the host will no longer send a report for this group
- IGMP router sends queries (to address 224.0.0.1) periodically (one out each interface), the group address in the query is 0.0.0.0

In response to a query, a host sends a IGMP report for every group with at least one process

Routers

- Note that routers have to listen to all 2^{23} link layer multicast addresses!
- Hence they listen promiscuously to all LAN multicast traffic

IGMP Implementation Details

In order to improve its efficiency there are several clever features:

- Since initial reports could be lost, they are resent after a random time [0, 10 sec]
- Response to queries are also delayed randomly - but if a node hears someone else report membership in a group it is interested in, its response is cancelled

Note: multicast routers don't care which host is a member of which group; only that *someone* attached to the subnet on a given interface is!

Time to Live

- TTL generally set to 1, but you can perform an expanding ring search for a server by increasing the value
- Addresses in the special range 224.0.0.0 through 224.0.0.255 - should never be forwarded by routers - regardless of the TTL value

All-Hosts Group

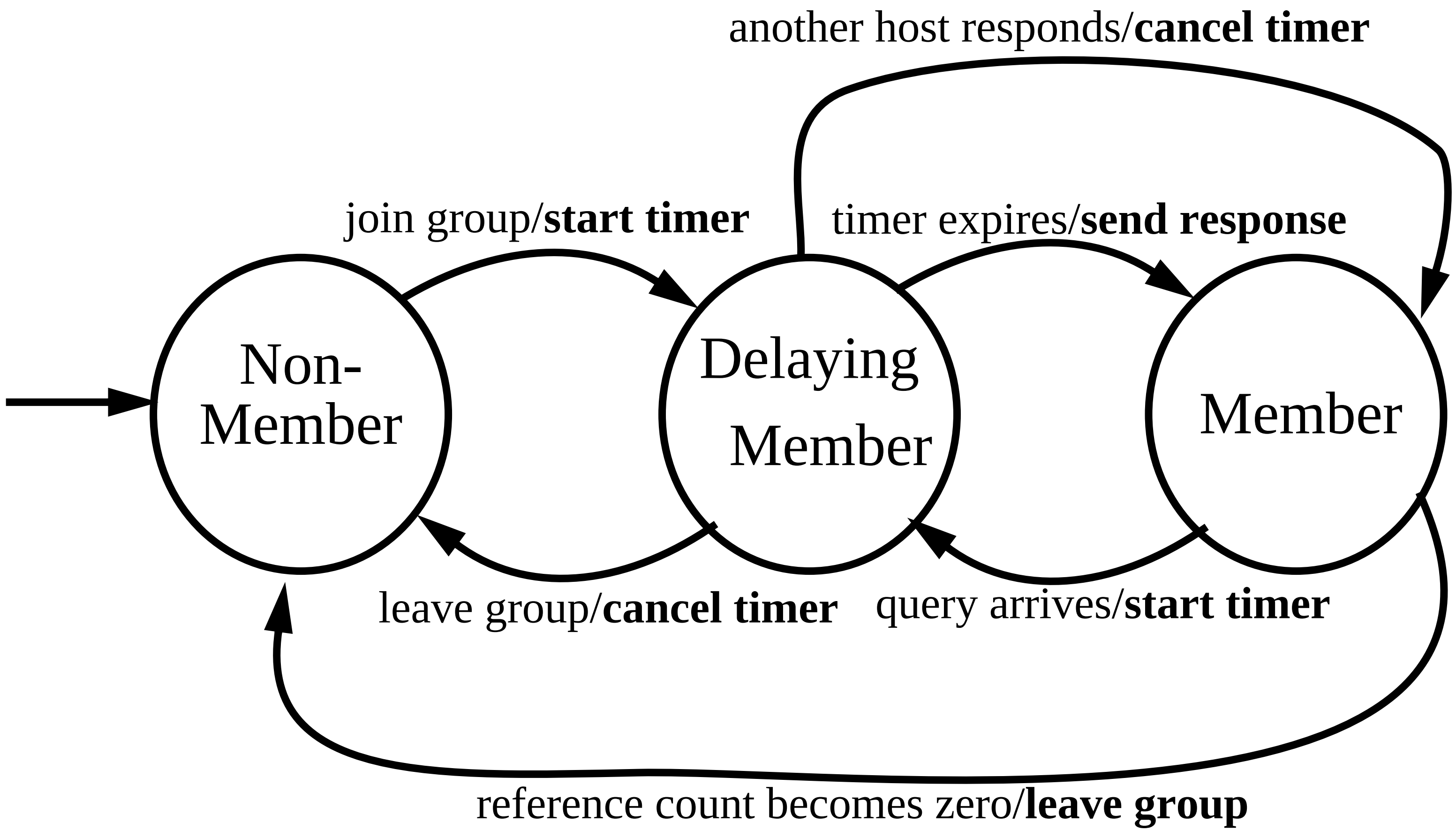
- all-hosts group address 224.0.0.1 - consists of all multicast capable hosts and routers on a given physical network; membership is *never* reported (sometimes this is called the “all-systems multicast address”)

All-Routers Group

- all-routers group address 224.0.0.2

Group membership State Transitions

adapted from Comer figure 17.4 pg. 330



IGMP Version 2 [71]

Allows a host to send a message (to address 224.0.0.2) when they want to explicitly leave a group -- after this message the router sends a *group-specific* query to ask if there is anyone still interested in listening to this group.

- however, the router may have to ask multiple times because this query could be lost
- hence the leave is not immediate -- even if there had been only one member (since the router can't know this)

IGMP Version 3 [72]

- Joining a multicast group, but with a specified set of sender(s) -- so that a client can limit the set of senders which it is interested in hearing from (i.e., source filtering)
- all IGMP replies are now set to a single layer 2 multicast address (e.g., 224.0.0.22) which all IGMPv3-capable multicast routers listen to:
  - because most LANs are now *switched* rather than shared media -- it uses less bandwidth to **not** forward all IGMP replies to all ports
  - most switches now support IGMP snooping -- i.e., the switch is IGMP aware and knows which ports are part of which multicast group (this requires the switch to know which ports other switches and routers are on -- so it can forward IGMP replies to them)
    - switches can listen to this specific layer 2 multicast address - rather than having to listen to all multicast addresses
  - it is thought that rather than have end nodes figure out if all the multicast senders which it is interested in have been replied to - simply make the switch do this work.

IGMP - ethereal

No.	Time	Source	Destination	Protocol	Info
1	0.000000	130.237.15.194	224.0.0.1	IGMP	V2 Membership Query
2	0.632486	130.237.15.225	239.255.255.250	IGMP	V2 Membership Report
3	0.727178	130.237.15.218	239.255.255.250	IGMP	V2 Membership Report
4	1.910951	130.237.15.227	224.0.0.252	IGMP	V2 Membership Report
5	6.953857	130.237.15.229	224.0.1.60	IGMP	V1 Membership Report
6	60.000053	130.237.15.194	224.0.0.1	IGMP	V2 Membership Query
7	61.998827	130.237.15.227	224.0.0.252	IGMP	V2 Membership Report
8	66.711434	130.237.15.225	239.255.255.250	IGMP	V2 Membership Report
9	66.953288	130.237.15.229	224.0.1.60	IGMP	V1 Membership Report
10	120.004228	130.237.15.194	224.0.0.1	IGMP	V2 Membership Query
11	120.872195	130.237.15.218	239.255.255.250	IGMP	V2 Membership Report
12	126.952839	130.237.15.229	224.0.1.60	IGMP	V1 Membership Report
13	129.597716	130.237.15.227	224.0.0.252	IGMP	V2 Membership Report
14	154.655463	211.105.145.186	224.0.0.2	IGMP	V2 Leave Group
15	154.656338	211.105.145.186	224.0.0.2	IGMP	V2 Leave Group
16	180.004408	130.237.15.194	224.0.0.1	IGMP	V2 Membership Query
17	180.943331	130.237.15.217	239.255.255.250	IGMP	V2 Membership Report

⊞ Frame 1 (60 bytes on wire, 60 bytes captured)

Figure 97: IGMP packets as seen with Ethereal

Frame 1: IGMP Membership Query

```
Ethernet II, Src: 00:02:4b:de:ea:d8, Dst: 01:00:5e:00:00:01
    Destination: 01:00:5e:00:00:01 (01:00:5e:00:00:01)
    Source: 00:02:4b:de:ea:d8 (Cisco_de:ea:d8)
    Type: IP (0x0800)
Internet Protocol, Src Addr: 130.237.15.194 (130.237.15.194), Dst
Addr: 224.0.0.1 (224.0.0.1)
   Version: 4
    Header length: 20 bytes
    Differentiated Services Field: 0xc0 (DSCP 0x30: Class Selector
6; ECN: 0x00)
    Total Length: 28
    Identification: 0x6fa3 (28579)
    Flags: 0x00
      Fragment offset: 0
    Time to live: 1
    Protocol: IGMP (0x02)
    Header checksum: 0xd6cc (correct)
    Source: 130.237.15.194 (130.237.15.194)
    Destination: 224.0.0.1 (224.0.0.1)
Internet Group Management Protocol
    IGMP Version: 2
    Type: Membership Query (0x11)
    Max Response Time: 10.0 sec (0x64)
    Header checksum: 0xee9b (correct)
    Multicast Address: 0.0.0.0 (0.0.0.0)
```

Frame 2: IGMP v2 Membership Report

```
    Ethernet II, Src: 00:06:1b:d0:98:c6, Dst: 01:00:5e:7f:ff:fa
    Destination: 01:00:5e:7f:ff:fa (01:00:5e:7f:ff:fa)
    Source: 00:06:1b:d0:98:c6 (Portable_d0:98:c6)
    Type: IP (0x0800)
Internet Protocol, Src Addr: 130.237.15.225 (130.237.15.225), Dst
Addr: 239.255.255.250 (239.255.255.250)
    Version: 4
    Header length: 24 bytes
    Differentiated Services Field: 0x00 (DSCP 0x00: Default; ECN: 0x00)
    Total Length: 32
    Identification: 0x1f8b (8075)
    Flags: 0x00
    Time to live: 1
    Protocol: IGMP (0x02)
    Header checksum: 0x8284 (correct)
    Source: 130.237.15.225 (130.237.15.225)
    Destination: 239.255.255.250 (239.255.255.250)
    Options: (4 bytes)
        Router Alert: Every router examines packet
Internet Group Management Protocol
    IGMP Version: 2
    Type: Membership Report (0x16)
    Max Response Time: 0.0 sec (0x00)
    Header checksum: 0xfa04 (correct)
    Multicast Address: 239.255.255.250 (239.255.255.250)
```

Frame 12: IGMP v1 Membership Report

```
Ethernet II, Src: 00:01:e6:a7:d3:b9, Dst: 01:00:5e:00:01:3c
    Destination: 01:00:5e:00:01:3c (01:00:5e:00:01:3c)
    Source: 00:01:e6:a7:d3:b9 (Hewlett-_a7:d3:b9)
    Type: IP (0x0800)
Internet Protocol, Src Addr: 130.237.15.229 (130.237.15.229), Dst
Addr: 224.0.1.60 (224.0.1.60)
    Version: 4
    Header length: 20 bytes
    Differentiated Services Field: 0x00 (DSCP 0x00: Default; ECN:
0x00)
    Total Length: 28
    Identification: 0x01f6 (502)
    Flags: 0x00
    Fragment offset: 0
    Time to live: 1
    Protocol: IGMP (0x02)
    Header checksum: 0x43dc (correct)
    Source: 130.237.15.229 (130.237.15.229)
    Destination: 224.0.1.60 (224.0.1.60)
Internet Group Management Protocol
    IGMP Version: 1
    Type: Membership Report (0x12)
    Header checksum: 0x0cc3 (correct)
    Multicast Address: 224.0.1.60 (224.0.1.60)
```

Frame 15: IGMP v2 Leave Group

```
Ethernet II, Src: 00:02:8a:78:91:8f, Dst: 01:00:5e:00:00:02
    Destination: 01:00:5e:00:00:02 (01:00:5e:00:00:02)
    Source: 00:02:8a:78:91:8f (AmbitMic_78:91:8f)
    Type: IP (0x0800)
Internet Protocol, Src Addr: 211.105.145.186 (211.105.145.186), Dst
Addr: 224.0.0.2 (224.0.0.2)
    Version: 4
    Header length: 24 bytes
    Differentiated Services Field: 0x00 (DSCP 0x00: Default; ECN: 0x00)
    Total Length: 32
    Identification: 0x9391 (37777)
    Flags: 0x00
    Fragment offset: 0
    Time to live: 1
    Protocol: IGMP (0x02)
    Header checksum: 0x4c20 (correct)
    Source: 211.105.145.186 (211.105.145.186)
    Destination: 224.0.0.2 (224.0.0.2)
    Options: (4 bytes)
        Router Alert: Every router examines packet
Internet Group Management Protocol
    IGMP Version: 2
    Type: Leave Group (0x17)
    Max Response Time: 0.0 sec (0x00)
    Header checksum: 0xff71 (correct)
    Multicast Address: 239.192.249.204 (239.192.249.204)
```

Multicast routing

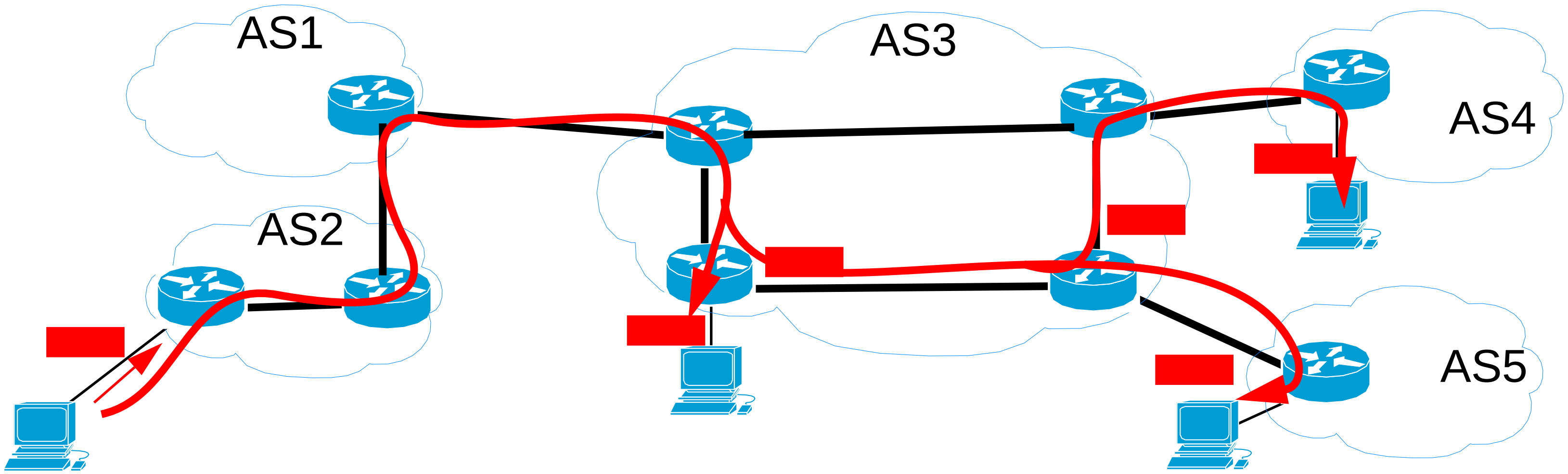


Figure 98: Multicast routing: packet replicated by the routers -- not the hosts

- packet forwarded one or more interfaces
  - router replicates the packet as necessary
- need to build a delivery tree - to decide on which paths to forward

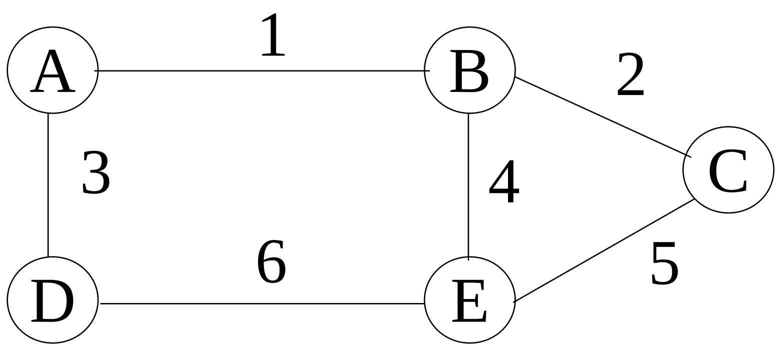
Therefore a Multicast Router

- Listens to all multicast traffic and forwards *if* necessary
  - Listens promiscuously to all LAN multicast traffic
- Listens to all multicast addresses
  - For an ethernet this means all 2^{23} link layer multicast addresses
- Communicates with:
  - directly connected hosts via IGMP
  - other multicast routers with multicast routing protocols

Multicasting

Example: Transmitting a file from C to A, B, and D.

✘ Using point-to-point transfer, some links will be used more than once to send the same file



✔ Using Multicast

Point-to-point						
Link	A	B	E	D	Total	Multicast
1	1				1	1
2	1	1			2	1
5			1	1	2	1
6				1	1	1
	2	1	1	2		4

Multicast Routing - Flooding

- maintaining a list of recently seen packets (last 2 minutes), if it has been seen before, then delete it, otherwise copy to a cache/database and send a copy on all (except the incoming) interface.

  ✘ Disadvantages:

    - Maintaining a list of “last-seen” packets. This list can be fairly long in high speed networks
    - The “last-seen” lists guarantee that a router will not forward the same packet twice, but it certainly does not guarantee that the router will receive a packet only once.

  ✔ Advantages

    - Robustness
    - It does not depend on any routing tables.

Delivery Trees: different methods

- Source-based Trees
  - Notation: (S, G) ⇒ only specific sender(s) [S= source, G=Group]
  - Uses memory proportional to O(S*G), can find optimal paths ⇒ minimizes delay
- Group Shared Trees
  - Notation: (*, G) ⇒ All senders
  - Uses less memory (O(G)), but uses suboptimal paths ⇒ higher delay
- Data-driven
  - Build only when data packets are sent
- Demand-driven
  - Build the tree as members join

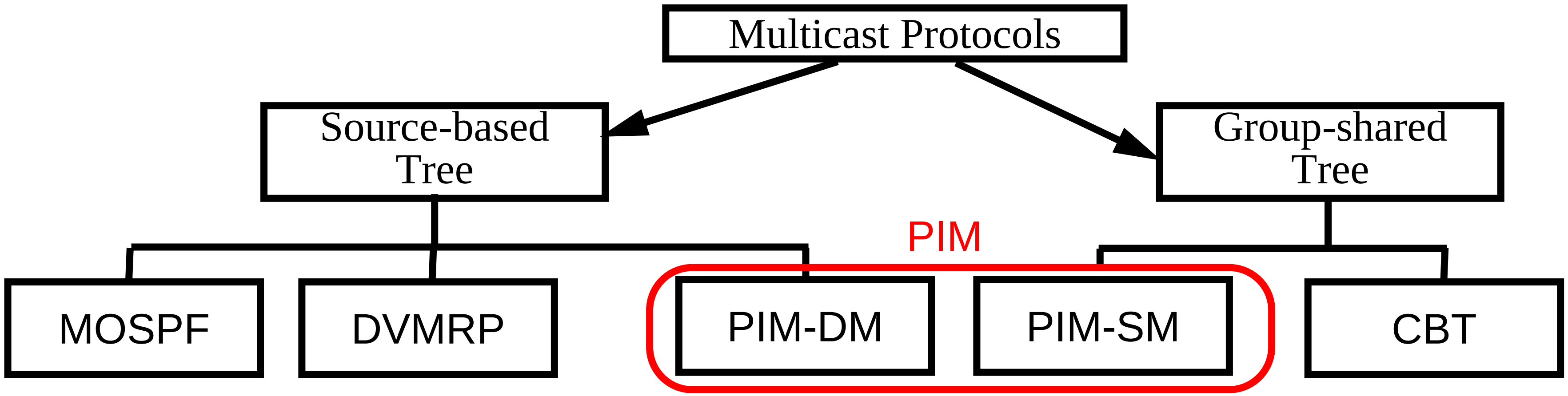
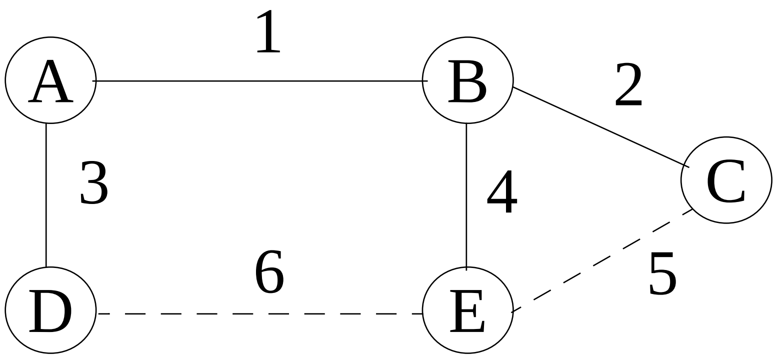


Figure 99: Taxonomy of Multicast Routing Protocols (see Forouzan figure 15.7 pg. 444)

Multicast Routing - Spanning Trees

The "spanning tree" technique is used by "media-access-control (MAC) bridges".

- Simply build up an "overlay" network by marking some links as "part of the tree" and other links as "unused" (produces a loopless graph).



Drawbacks

✘ It does not take into account group membership

✘ It concentrates all traffic into a small subset of the network links.

Link-State Multicast: MOSPF [73]

Just add multicast to a link-state routing protocol thus OSPF ⇒ MOSPF

- Use the multiprotocol facility in OSPF to carry multicast information
- Extended with a group-membership LSA
  - This LSA lists only members of a given group
- Use the resulting link-state database to build delivery trees
  - Compute least-cost source-based trees considering metrics using Dijkstra's algorithm
  - A tree is computed for each (S,G) pair with a given source (S), this is done for all S
  - Remember that as a link-state routing protocol that every router will know the topology of the complete network
- However, it is expensive to keep store all this information (and most is unnecessary)
  - Cache only the active (S,G) pairs
  - Use a data-driven approach, i.e., only computes a new tree when a multicast datagram arrives for this group

Reverse -Path Forwarding (RPF)

RPF algorithm takes advantage of a routing table to "orientate" the network and to compute an implicit tree per network source.

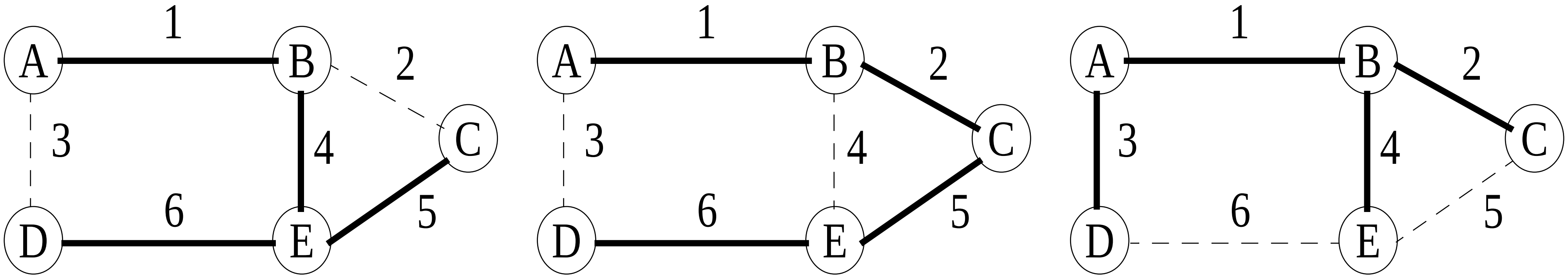
Procedure

1.When a multicast packet is received, note source (S) and interface (I)

2.If I belongs to the shortest path toward S, forward to all interfaces except I.

- Compute shortest path **from** the **source** to the node rather than from the node to the source.
- Check whether the local router is on the shortest path between a neighbor and the source before forwarding a packet to that neighbor. If this is not the case, then there is no point in forwarding a packet that will be immediately dropped by the next router.

- RPF results in a different spanning tree for each source.



RPF tree from E RPF tree from C RPF tree from A

These trees have two interesting properties:

- They guarantee the fastest possible delivery, as multicasting follows the shortest path from source to destination
- Better network utilization, since the packets are spread over multiple links.

Drawback

✘Group membership is **not** taken into account when building the tree
⇒ a network can receive two or more copies of a multicast packet

Reverse Path Broadcast (RPB)

- We define a parent router for each network
- For each source, a router will forward a multicast packet **only** if it is the designated parent

$\Rightarrow$ each network gets only one copy of each multicast packet

RPB + Prunes ⇒ Reverse Path Multicast (RPM)

When source S starts a multicast transmission the first packet is propagated to all the network nodes (i.e., flooding). Therefore all leaf nodes receive the first multicast packet. However, if there is a leaf node that does **not** want to receive further packets, it will send back a “prune” message to the router that sent it this packet - saying effectively “don’t send further packets from source S to group G on this interface I.”

There are two obvious drawback in the flood and prune algorithm:

- The first packet is flooded to the whole network
- The routers must keep states per group and source

When a listener joins at a leaf that was pruned, we add this leaf back by grafting.

Flood and prune was acceptable in the experimental MBONE which had only a few tens of thousands of nodes, but for the Internet where both the number of sources and the number of groups becomes very large, there is a risk of exhausting the memory resources in network routers.

Distance-Vector Multicast Routing Protocol (DVMRP) [74]

- Start with a unicast distance-vector routing protocol (e.g., RIP), then extend (Destination, Cost, Nexthop) $\Rightarrow$ (Group, Cost, Nexthops)
  - Routers only know their next hop (i.e., which neighbor)
- Reverse Path Multicasting (RPM)
- DVMRP is data-driven and uses source-based trees

Multicast Routing - Steiner Tree's

Assume source C and the recipients are A and D.

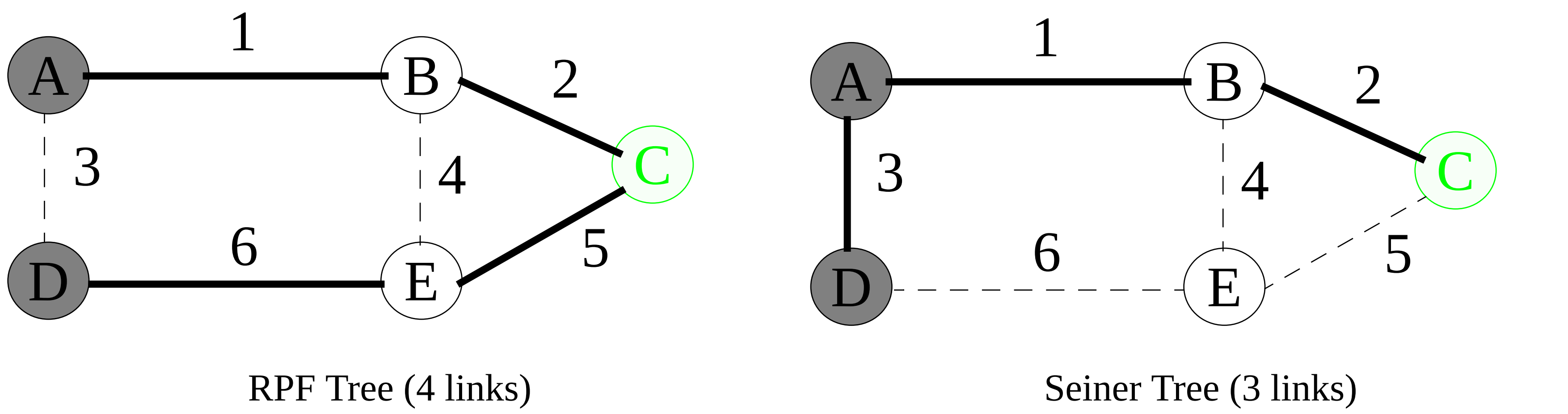


Figure 100: RPF vs. Steiner Tree

- Steiner tree uses less resources (links), but are *very hard* to compute (N-P complete)
- In Steiner trees the routing changes widely if a new member joins the group, this leads to instability. Thus the Steiner tree is more a mathematical construct that a practical tool.

Core-Based Trees (CBT)

A fixed point in the network chosen to be the center of the multicast group, i.e., “core”. Nodes desiring to be recipients send “join” commands toward this core. These commands will be processed by all intermediate routers, which will mark the interface on which they received the command as belonging to the group’s tree. The routers need to keep one piece of state information per group, listing all the interface that belong to the tree. If the router that receives a join command is already a member of the tree, it will mark only one more interface as belong to the group. If this is the first join command that the router receives, it will forward the command one step further toward the core.

Advantages

- CBT limits the expansion of multicast transmissions to precisely the set of all recipients (so it is demand-driven). This is in contrast with RPF where the first packet is sent to the whole network.
- The amount of state is less; it depends only on the number of the groups, not the number of pairs of sources and groups ⇒ Group-shared multicast trees (*, G)
- Routing is based on a spanning tree, thus CBT does **not** depend on multicast or unicast routing tables

Disadvantages

- The path between some sources and some receivers may be suboptimal.
- Senders sends multicast datagrams to the core router encapsulated in unicast datagrams

Protocol-Independent Multicast (PIM)

Two modes:

- PIM-dense mode (PIM-DM) [76]
  - Dense mode is an implementation of RPF and prune/graft strategy
  - Relies on unicast routing tables providing an optimal path
  - However, it is independent of the underlying unicast protocol
- PIM-sparse mode (PIM-SM) [75]
  - Sparse mode is an implementation of CBT where join points are called “rendezvous points”
  - A given router may know of more than one rendezvous point
  - Simpler than CBT as there is no need for acknowledgement of a join message
  - Can switch from group-shared tree to source-based tree if there is a dense cluster far from the nearest rendezvous point

The adjectives “dense” and “sparse: refer to the density of group members in the Internet. Where a group is send to be **dense** if the probability is high that the area contains at least one group member. It is send to be **sparse** if that probability is low.

Multiprotocol BGP (MBGP) [78]

Extends BGP to enable multicast routing policy, thus it connects multicast topologies within and between BGP autonomous systems

Add two new (optional and non-transitive) attributes:

- Multiprotocol Reachable NLRI (MP_REACH_NLRI)
- Multiprotocol Unreachable NLRI (MP_UNREACH_NLRI)

As these are optional and non-transitive attributes - routers which do not support these attributes ignore then and don’t pass them on.

Thus MBGP allows the exchange of multicast routing information, but one must still use PIM to build the distribution tree to actually forward the traffic!

Multicast backbone (MBONE) [70]

Why can you do when all router's and networks don't support multicasting: Tunnel!

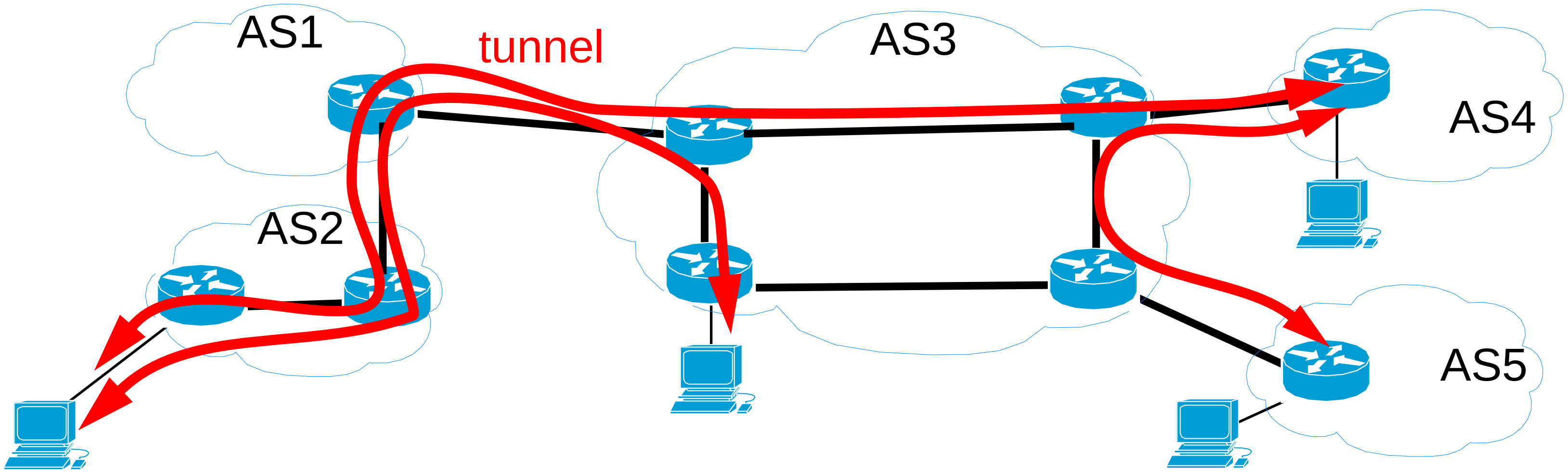


Figure 101: Multicast routing via tunnels - the basis of MBONE (see Forouzan figure 15.14 pg. 453)

See the IETF MBONE Deployment Working Group (MBONED)

http://antc.uoregon.edu/MBONED/ and their charter *http://www.ietf.org/html.charters/mboned-charter.html*

Telesys class was multicast over MBONE

Already in Period 2, 1994/1995 "Telesys, gk" was multicast over the internet and to several sites in and near Stockholm.

Established ports for each of the data streams:

- electronic whiteboard
- video stream
- audio stream

The technology works - but it is very important to get the audio packets delivered with modest delay and loss rate. Poor audio quality is perceived a major problem.

NASA and several other organizations regularly multicast their audio and video “programs”.

Benefits for Conferencing

- IP Multicast is efficient, simple, robust
- Users can join a conference without enumerating (or even knowing) other participants
- User can join and leave at any time
- Dynamic membership

MBONE Chronology

Nov. 1988	Small group proposes testbed net to DARPA. This becomes DARTNET
Nov. 1990	Routers and T1 lines start to work
Feb. 1991	First packet audio conference (using ISI's vt)
Apr. 1991	First multicast audio conference
Sept. 1991	First audio+video conference (hardware codec)
Mar. 1992	Deering & Casner broadcast San Diego IETF to 32 sites in 4 countries
Dec. 1992	Washington DC IETF - four channels of audio and video to 195 watchers in 12 countries
Jan. 1993	MBONE events go from one every 4 months to several a day
1994/1995	Telesys gk -- multicast from KTH/IT in Stockholm
July 1995	KTH/IT uses MBONE to multicast two parallel sessions from IETF meeting in Stockholm
...	
today	lots of users and "multicasters"

IETF meetings are *now* regularily multicast - so the number of participants that can attend is not limited by physical space or travel budgets.

MBONE growth

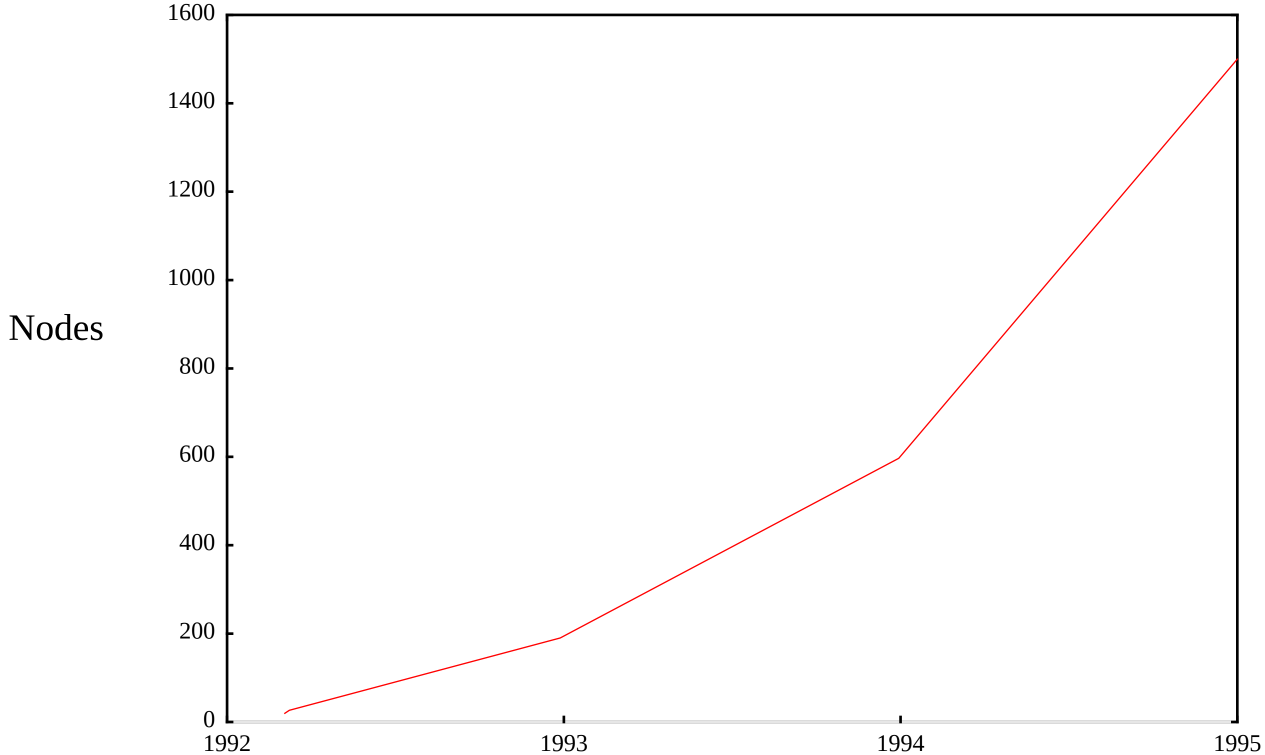


Figure 102: MBONE Growth - Doubling time ~8 months

For the some statistics see: *http://www.caida.org/tools/measurement/mantra/*

Multicast	2003 02/06/2003,15:25:38 PST	2002 01/21/2002,11:30 PST (Pacific Standard Time)	Old state 2000
Entity	Value		
#Groups	4473	1002	330
#Participants	6059	average 4	
#Unique Participants	1446		
#ASes	137		
#RPs	197		

But we are still waiting for multicast to “take off”.

MBONE connections

MBONE is an “overlay” on the Internet

- multicast routers were distinct from normal, unicast routers - but increasingly routers support multicasting
- it is not trivial to get hooked up
- requires cooperation from local and regional people

MBONE is changing:

- Most router vendors now support IP multicast
- MBONE will go away as a distinct entity once ubiquitous multicast is supported throughout the Internet.
- Anyone hooked up to the Internet can participate in conferences

mrouted

mrouted UNIX deamon

tunneling to other MBONE routers

See: “Linux-Mrouted-MiniHOWTO: How to set up Linux for multicast routing” by Bart Trojanowski <bart@jukie.net>, v0.1, 30 October 1999
http://jukie.net/~bart/multicast/Linux-Mrouted-MiniHOWTO.html

and *http://www.linuxdoc.org/HOWTO/Multicast-HOWTO-5.html*

Multicast Source Discovery Protocol (MSDP)[81]

As the routing protocols deployed in the multicast networks operating in sparse mode do not support flooding information, a mechanism was needed to propagate information about sources (i.e., hosts sourcing data to a multicast group) and the associated multicast groups to all the multicast networks.

Sends Source Active (SA) messages containing (S,G,RP):

- Source Address,
- Group Address,
- and RP Address

these are propagated by Rendezvous Points over TCP

MSDP connects multiple PIM-SM domains together. Each domain uses its own **independent** Rendezvous Point (RP) and does not depend on RPs in other domains.

GLOP addressing

Traditionally multicast address allocation has been dynamic and done with the help of applications like SDR that use Session Announcement Protocol (SAP).

GLOP is an example of a policy for allocating multicast addresses (it is still experimental in nature). It allocated the 233/8 range of multicast addresses amongst different ASes such that each AS is statically allocated a /24 block of multicast addresses. See [77]

0 … 7	8 … 23	… 31
233	16 bits AS	local bits

Single Source Multicast (SSM) [83]

- A single source multicast-address space was allocated to 232/8
- Each AS is allocated a unique 232/24 address block that it can use for multicasting.

Other multicast efforts

PGM: Pragmatic General Multicast Protocol [82]

Administratively Scoped IP Multicast [84]

…

Tools for managing multicast

"Managing IP Multicast Traffic" A White Paper from the IP Multicast Initiative (IPMI) and Stardust Forums for the benefit of attendees of the 3rd Annual IP Multicast Summit, February 7-9, 1999

http://techsup.vcon.com/whtpprs/Managing%20IP%20Multicast%20Traffic.pdf

Mrinfo	shows the multicast tunnels and routes for a router/mrouted.
Mtrace	traces the multicast path between two hosts.
RTPmon	displays receiver loss collected from RTCP messages.
Mhealth	monitors tree topology and loss statistics.
Multimon	monitors multicast traffic on a local area network.
Mlisten	captures multicast group membership information.
Dr. Watson	collects information about protocol operation.

Mantra (Monitor and Analysis of Traffic in Multicast Routers)

http://www.caida.org/tools/measurement/mantra/

SNMP-based tools and multicast related MIBs

Management Information Bases (MIBs) for multicast:

RTP MIB	designed to be used by either host running RTP applications or intermediate systems acting as RTP monitors; has tables for each type of user; collect statistical data about RTP sessions.
Basic Multicast Routing MIB	includes only general data about multicast routing. such as multicast group and source pairs; next hop routing state, forwarding state for each of a router's interfaces, and information about multicast routing boundaries.

Protocol-Specific Multicast Routing MIBs

Provide information specific to a particular routing protocol

PIM MIB	list of PIM interfaces that are configured; the router's PIM neighbors; the set of rendezvous points and an association for the multicast address prefixes; the list of groups for which this particular router should advertise itself as the candidate rendezvous point; the reverse path table for active multicast groups; and component table with an entry per domain that the router is connected to.
CBT MIB:	configuration of the router including interface configuration; router statistics for multicast groups; state about the set of group cores, either generated by automatic bootstrapping or by static mappings; and configuration information for border routers.
DVMRP MIB	interface configuration and statistics; peer router configuration states and statistics; the state of the DVMRP (Distance-Vector Multicast Routing Protocol) routing table; and information about key management for DVMRP routes.
Tunnel MIB	lists tunnels that might be supported by a router or host. The table supports tunnel types including Generic Routing Encapsulation (GRE) tunnels, IP-in-IP tunnels, minimal encapsulation tunnels, layer two tunnels (LTTP), and point-to-point tunnels (PPTP).
IGMP MIB	only deals with determining if packets should be forwarded over a particular leaf router interface; contains information about the set of router interfaces that are listening for IGMP messages, and a table with information about which interfaces currently have members listening to particular multicast groups.

SNMP tools for working with multicast MIBs

Merit SNMP-Based Management Project has release two freeware tools which work with multicast MIBs:

Mstat	queries a router or SNMP-capable mrouted to generate various tables of information including routing tables, interface configurations, cache contents, etc.
Mview	"application for visualizing and managing the MBone",allows user to display and interact with the topology, collect and monitor performance statistics on routers and links

HP Laboratories researchers investigating IP multicast network management are building a prototype integrated with HP OpenView -- intended for use by the network operators who are not experts in IP multicast; provides discovery, monitoring and fault detection capabilities.

QoS & Scheduling algorithms

Predictable delay is thought to be required for interactive real-time applications: Alternatives:

1. use a network which guarantees fixed delays
2. use a packet scheduling algorithm
3. retime traffic at destination

Since queueing at routers, hosts, etc. has traditionally been simply FIFO; which does not provide guaranteed end-to-end delay both the 2nd and 3rd method use alternative algorithms to maintain a predictable delay.

Algorithms such as: Weighted Fair Queueing (WFQ)

These algorithms normally emulate a fluid flow model.

As it is very hard to provide fixed delays in a network, hence we will examine the 2nd and 3rd methods.

RSVP: Resource Reservation Setup Protocol [87]

- RSVP is a network control protocol that will deal with resource reservations for certain Internet applications.
- RSVP is a component of “Integrated services” Internet, and can provide both best-effort and QoS.
  - Applications request a specific quality of service for a data stream
- RSVP delivers QoS requests to each router along the path.
  - Maintains router and host state along the data stream during the requested service.
  - Hosts and routers deliver these request along the path(s) of the data stream
  - At each node along the path RSVP passes a new resource reservation request to an admission control routine

RSVP is a signalling protocol carrying no application data

- First a host sends IGMP messages to join a group
- Second a host invokes RSVP to reserve QoS

Functionality

- RSVP is receiver oriented protocol.
  The receiver is responsible for requesting reservations.
- RSVP handles heterogeneous receivers.
  Hosts in the same multicast tree may have different capabilities and hence need different QoS.
- RSVP adapts to changing group membership and changing routes.
  RSVP maintains “Soft state” in routers. The only permanent state is in the end systems. Each end system sends their RSVP control messages to refresh the router state.
  In the absence of refresh message, RSVP state in the routers will time-out and be deleted.
- RSVP is **not** a routing protocol.
  A host sends IGMP messages to join a multicast group, but it uses RSVP to reserve resources along the delivery path(s) from that group.

Resource Reservation

- Interarrival variance reduction / jitter
- Capacity assignment / admission control
- Resource allocation (who gets the bandwidth?)

Jitter Control

- if network has enough capacity
  average departure rate = receiver arrival rate
- Then jitter is caused by queue waits due to competing traffic
- Queue waits should be at most the amount of competing traffic in transit, total amount of in transit data should be at most round trip propagation time
  (100 ms for transcontinental path)
  (64 kbit/sec => buffer = 8 kb/s*0.1 sec = 800 bytes)

See: Jonathan Rosenberg, Lili Qiu, and Henning Schulzrinne, "Integrating Packet FEC into Adaptive Voice Playout Buffer Algorithms on the Internet",INFOCOM, (3), 2000, pp. 1705-1714.

See also *http://citeseer.nj.nec.com/rosenberg00integrating.html*

Capacity Assignment

- end-nodes ask network for bandwidth.
- Can get “yes” or “no” (busy signal)
- Used to control available transmission capacity

RSVP Protocol Mechanism

- Sender sends RSVP PATH message which records path
- Receiver sends RSVP RESV message backwards along the path indicating desired QoS
- In case of failure a RSVP error message is returned

Figure 103:

RSVP Soft State

- “soft state” in hosts and routers
- create by PATH and RESV messages
- refreshed by PATH and RESV messages
- Time-outs clean up reservations
- Removed by explicit “tear-down” messages

RSVP operation

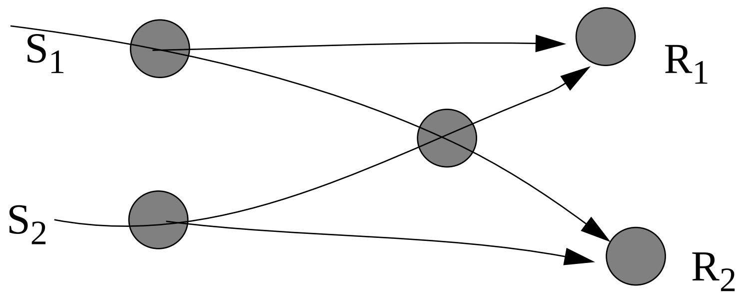


Figure 104:

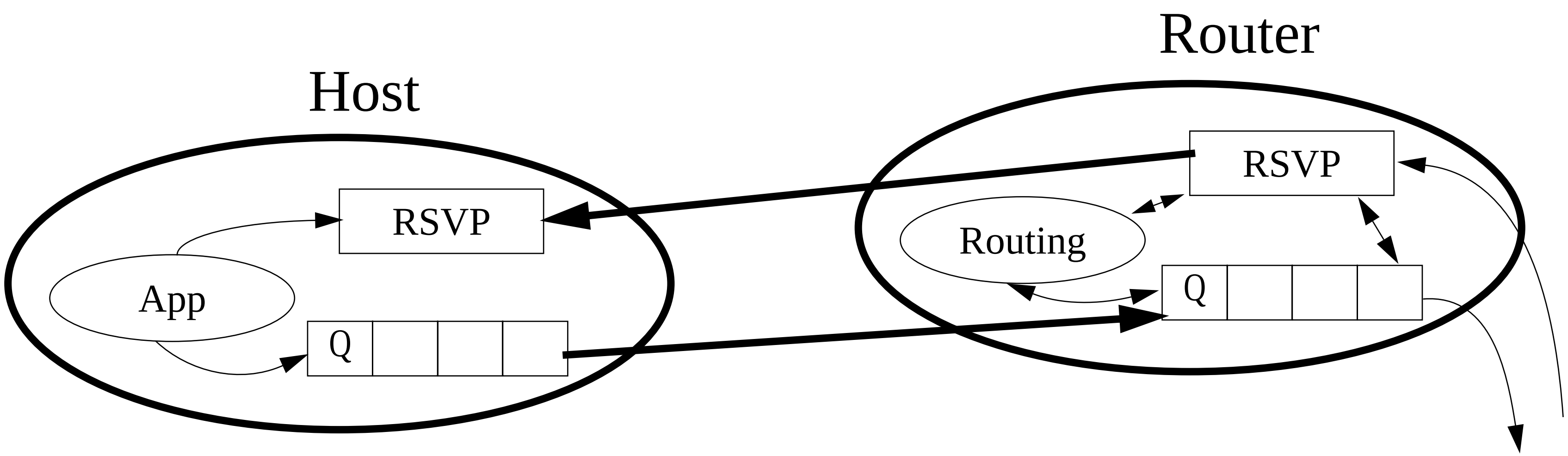


Figure 105:

RSVP operations (continued)

- At each node, RSVP applies a local decision procedure “admission control” to the QoS request. If the admission control succeeds, it set the parameters to the classifies and the packet schedule to obtain the desired QoS. If admission control fails at any node, RSVP returns an error indication to the application.
- Each router in the path capable of resource reservation will pass incoming data packets to a packet classifier and then queue these packet in the packet scheduler. The packet classifier determines the route and the QoS class for each packet. The schedule allocates a particular outgoing link for packet transmission.
- The packet schedule is responsible for negotiation with the link layer to obtain the QoS requested by RSVP. The scheduler may also negotiate a “CPU time”.

RSVP Summary

- RSVP supports multicast and unicast data delivery
- RSVP adapts to changing group membership and routes
- RSVP reserves resources for simplex data streams
- RSVP is receiver oriented, i.e., the receiver is responsible for the initiation and maintenance of a flow
- RSVP maintains a “soft-state” in routers, enabling them to support gracefully dynamic memberships and automatically adapt to routing changes
- RSVP provides several reservation models
- RSVP is transparent for routers that do not provide it

Argument against Reservation

Given, the US has 126 million phones:

- Each conversation uses 64 kbit/sec per phone
- Therefore the total demand is: 8×10^{12} b/s (1 Tbyte/s)

One optical fiber has a bandwidth of ~25×10^{12} b /s

There are well over 1000 transcontinental fibers!

Why should bandwidth be a problem?

Further reading

IETF *Routing Area,* especially:

- Inter-Domain Multicast Routing (*idmr*)
- Multicast Extensions to OSPF (*mospf*)

IETF *Transport Area* especially:

- Differentiated Services (*diffserv*)
- RSVP Admission Policy (*rap*)
- Multicast-Address Allocation (*malloc*)

With lots of traditional broadcasters and others discovering multicast -- it is going to be an exciting area for the next few years.

Summary

This lecture we have discussed:

- Multicast, IGMP, RSVP

References

[69] Joe Abley, f.root-servers.net, NZNOG 2005, February 2005, Hamilton, NZ
http://www.isc.org/pubs/pres/NZNOG/2005/F%20Root%20Server.pdf

[70] S. Deering, “Host Extensions for IP Multicasting”, IETF RFC 1112, August 1989 *http://www.ietf.org/rfc/rfc1112.txt*

[71] W. Fenner, “Internet Group Management Protocol, Version 2”, IETF RFC 2236 , November 1997 *http://www.ietf.org/rfc/rfc2236.txt*

[72] B. Cain, S. Deering, I. Kouvelas, B. Fenner, and A. Thyagarajan, “Internet Group Management Protocol, Version 3”, IETF RFC 3376, October 2002
http://www.ietf.org/rfc/rfc3376.txt

[73] J. Moy, “Multicast Extensions to OSPF”, IETF RFC 1584, March 1994 http://www.ietf.org/rfc/rfc1584.txt

[74] D. Waitzman, C. Partridge, and S. Deering, “Distance Vector Multicast Routing Protocol”, IETF RFC 1075 , November 1988

http://www.ietf.org/rfc/rfc1075.txt

[75] D. Estrin, D. Farinacci, A. Helmy, D. Thaler, S. Deering, M. Handley, V. Jacobson, C. Liu, P. Sharma, and L. Wei, “Protocol Independent Multicast-Sparse Mode (PIM-SM): Protocol Specification”, IETF RFC 2362, June 1998 http://www.ietf.org/rfc/rfc2362.txt

[76] A. Adams, J. Nicholas, and W. Siadak, “Protocol Independent Multicast - Dense Mode (PIM-DM): Protocol Specification (Revised)”, IETF RFC 3973, January 2005 http://www.ietf.org/rfc/rfc3973.txt

[77] D. Meyer and P. Lothberg, “GLOP Addressing in 233/8”, IETF RFC 3180 September 2001 http://www.ietf.org/rfc/rfc3180.txt

[78] T. Bates, Y. Rekhter, R. Chandra, and D. Katz, “Multiprotocol Extensions for BGP-4”, IETF RFC 2858, June 2000 http://www.ietf.org/rfc/rfc2858.txt

[79] Beau Williamson, *Developing IP Multicast Networks*, Cisco Press, 2000

[80] Internet Protocol Multicast, Cisco, Wed Feb 20 21:50:09 PST 2002

http://www.cisco.com/univercd/cc/td/doc/cisintwk/ito_doc/ipmulti.htm

[81] B. Fenner and D. Meyer (Editors), "'Multicast Source Discovery Protocol (MSDP)", IETF RFC 3618, October 2003 http://www.ietf.org/rfc/rfc3618.txt

[82] T. Speakman, J. Crowcroft, J. Gemmell, D. Farinacci, S. Lin, D. Leshchiner, M. Luby, T. Montgomery, L. Rizzo, A. Tweedly, N. Bhaskar, R. Edmonstone, R. Sumanasekera and L. Vicisano, "PGM Reliable Transport Protocol Specification", IETF RFC 3208 , December 2001

[83] S. Bhattacharyya (Ed.), "An Overview of Source-Specific Multicast (SSM)", IETF RFC 3569, July 2003 http://www.ietf.org/rfc/rfc3569.txt

[84] D. Meyer, "Administratively Scoped IP Multicast", IETF RFC 2365, July 1998 http://www.ietf.org/rfc/rfc2365.txt

[85] B. Quinn and K. Almeroth, "IP Multicast Applications: Challenges and Solutions", IETF RFC 3170,September 2001 http://www.ietf.org/rfc/rfc3170.txt

[86] R. Braden (Ed.), L. Zhang, S. Berson, S. Herzog, and S. Jamin, "Resource

ReSerVation Protocol (RSVP) -- Version 1 Functional Specification”, IETF RFC 2205, September 1997 *http://www.ietf.org/rfc/rfc2205.txt*

[87] Y. Snir, Y. Ramberg, J. Strassner, R. Cohen, and B. Moore, “Policy Quality of Service (QoS) Information Model”, IETF RFC 3644, November 2003

http://www.ietf.org/rfc/rfc3644.txt