2011

Introduction to IP multicast in production networks

Ganesh Vidyadharan Girija

ROCHESTER INSTITUTE OF TECHNOLOGY
COLLEGE OF APPLIED SCIENCE AND TECHNOLOGY
DEPARTMENT OF ELECTRICAL, COMPUTER &
TELECOMMUNICATIONS ENGINEERING TECHNOLOGY

Introduction to IP Multicast in production networks

By
Ganesh Vidyadharan Girija
9/29/2011

Project submitted is in partial fulfillment of the requirements for the degree of Master of Science in Telecommunications Engineering Technology

Approval

Ganesh Vidyadharan Girija Project approved by:

Professor Ronald G Fulle

____

Professor Mark J. Indelicato

____

Professor William P. Johnson

____

Date: September 29th, 2011

Table of Contents

Table of Figures

Abstract

The objective of this paper is to introduce the reader to the world of IP multicasting. I intend to achieve this goal by providing an introduction that bridges the gap between the existing unicast networks and the developing multicast network. The basics of multicast that is covered in the earlier chapter includes the multicast addressing scheme, different protocols used for multicast transmission, various distribution trees that are formed by these protocols and various aspects of multicast forwarding. We take a look at IGMP which is the protocol that runs between the host devices and their first hop multicast routers, enabling the host to join/leave a multicast group. The protocols used for running IP multicast over networks are discussed in detail with additional emphasis on PIM-SM which is the most common among the available selection. The paper concludes with a general overlook on the avenues where multicasting could play a major role benefitting the Internet Service Providers and even large corporate networks, and a glance on the pros and cons of multicasting.

1 Introduction

We are living in an age that is witnessing the fastest pace in the growth of Information Technology and this growth is facilitated by the networking infrastructure that binds the communication world. While the information carrying capacity of transport layer has been increased manifold by the advancements in fiber optic communication, the efficient utilization of the available network infrastructure is being honed by development of various layer 2/3 protocols. Most of the developmental studies are aimed at achieving the goal of optimal utilization of the available bandwidth, which comes at a cost.

One such topic that is gaining popularity is the development of Multicast Networks. Multicasting is based on a specific scheme of IP addressing and making use of it to optimize the bandwidth utilization. The role of multicasting is to minimize the number of packets that are sent as duplicates when communication is happening between groups of select devices in a network. Examples of these could be a training session for employees in a particular department, sharing trading information to a group of members, paging a message to a specific group, IPTV applications, etc. This paper will cover the basics of IP Multicasting including the IP addressing scheme, the basic protocols used in Multicast routing, the mechanism used to communicate between the hosts in a network and the first hop routers about its intent to join/leave a multicast group and a quick look on the applications that use IP multicasting and the advantages that it offers when compared to using unicast as the method of choice from an application perspective. But, before going into the details of Multicast Communication, here is a briefing on the common methods of group communication between devices in a network, which are Unicast Communication and Broadcast Communication.

Unicast

Unicast is the method used for sending packets to a single device in the network i.e. one to one communication. A Unicast transmission can be achieved using both TCP and UDP which are IP delivery methods which are session-based protocols. The bandwidth utilization in unicast communication increases linearly with the number of users. If there is a source device sending 100Kbps packets to a bunch of devices, say 10, the total bandwidth utilization will be 1000Kbps even though the same information is being sent in all the 10 packets flowing across the same network infrastructure. In the following pages of this paper, we will go over the advancements in communication that will help to keep the bandwidth usage to a minimum irrespective of the number of recipients.

Broadcast

This method is used for sending packets to all devices in the network i.e. one to all communication. Broadcast packets are not forwarded by routers, so as to avoid flooding of packets in large networks.

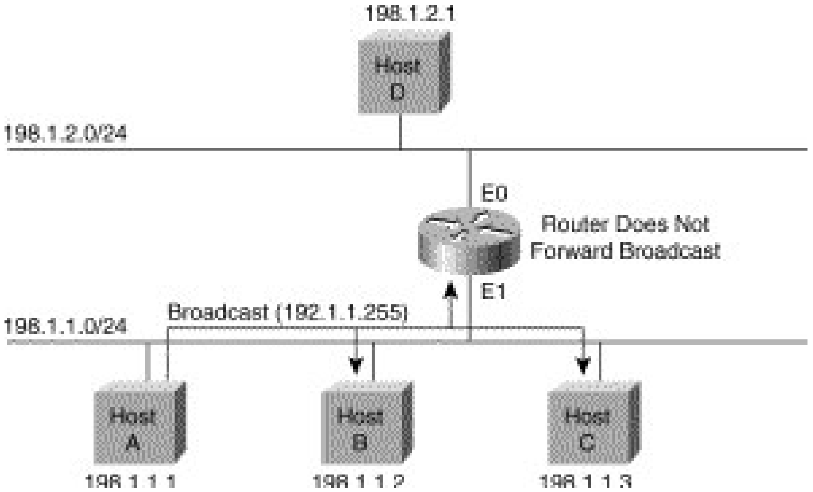


Figure 1: IP Broadcast being blocked by a Router [2]

Broadcast can be done over UDP only. Broadcast packets use all 1's in destination IP field. It is also possible to do directed broadcast for specific networks. For example, if a device with host IP of 192.168.15.3 wants to send a broadcast to all devices in the 192.168.15.0/24 network, the broadcast packets destination IP will be 192.168.15.255, which means that all the devices in the network will receive that packet.

Figure 1 shows a directed broadcast within the 198.1.1.0/24 network which reaches all the devices in that network Host A, B and C but not Host D as it is in 198.1.2.0/24 network. This mechanism is helpful in avoiding broadcast storms in networks.

Why Multicast?

Optimal use of the available network bandwidth is one of the main goals of a network design. There are lots of applications that involve communication between select groups of devices which are part of a larger network. Some of the fast growing examples are networked gaming applications and IPTV along with other traditional information sharing fields like stock trading firms, medical transcription, online education systems, etc that can take advantage of developments in selective broadcasting. Using unicast for these applications means that the number of packets containing the same information has to be replicated a number of times depending on the number of recipients, while using broadcast would mean that there would be a bunch of devices that receive some junk packets or in some cases open up security concerns depending on the criticality of the information being shared.

Multicast routing requires the presence of Multicast aware routing devices in the network. This is the reason why the earlier deployment of multicast networks would be seen in the private networks of large corporations, than on the internet where there would be a whole lot of old school devices that are not multicast enabled. The multicast enabled routers in the network will have the responsibilities of duplicating the packets received from the source on its outgoing interfaces depending on the presence of recipients on its outgoing interface. This ensures that there is only one source packet irrespective of the number of recipients in the network. “Multicast in corporate environments where all routers are multicast-enabled can save quite a bit of bandwidth. “ [1]

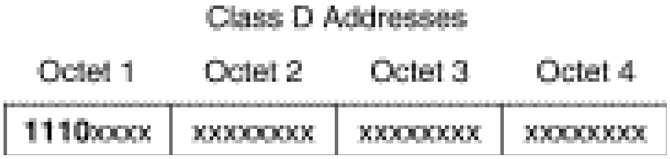
With reference to Figure 1, if a source device or say a server wants to send directed broadcast to select devices in other subnets separated by routers, we will need IP multicasting. This would necessitate the use of a “special form of IP address called an IP multicast group address, in the place of destination address. This would also make it necessary that the routers in the network are multicast aware, so as to forward the incoming multicast packets out all the interfaces that connect to members of the multicast group. The multicast group address mentioned above will be specified in the IP destination address field of the packet.” [2]

2 Multicast Basics

The idea of this part is to introduce some of the common terminologies related to IP Multicasting, like IP multicast address itself, sending and receiving of IP multicast packets, and IGMP v1 (Internet Group Management Protocol) along with multicast distribution trees and multicast forwarding.

2.1 Multicast Addresses

"Multicast addressing is different from the regular unicast addressing. While the devices participating in unicast are identifiable with unique IP's, the devices participating in multicast are identified by an arbitrary group of IP hosts specified by the multicast IP addresses, for the hosts that have joined a multicast group or wishing to receive traffic sent to the group." [2] The multicast addresses belong to the IANA assigned Class D address space, which ranges from 224.0.0.0 to 239.255.255.255. These IP's are identified by 1110 in the first four places of the first octet, in binary format.



Class D Addresses

Octet 1	Octet 2	Octet 3	Octet 4
1110xxxx	xxxxxxxx	xxxxxxxx	xxxxxxxx

Figure 2: Multicast Address Format [2]

2.2 Assigned Multicast Addresses

One of the major concerns with regard to multicast IPs is its scarcity, just like in the case of Class A, B and C IP's which are already used up for internet and regular data network that we had to move on to IPv6. Just to make things easier with IP addressing, the idea of classful IP addressing scheme has been eliminated when it comes to Multicast addresses. This means that the boundary between network portion and host portion when it comes to a Multicast address is no longer there. IANA controls the assignment of these addresses. "They does not assign individual IP multicast addresses without a really good technical justification. They tend to assign individual IP multicast addresses for use by specific network protocols." [2]

This leaves the rest of the world to share the remaining IP’s on a lease basis like we do with DHCP IP allocation.

2.2.1 Link-Local Multicast Addresses

“The IANA has reserved the range of 224.0.0.0 through 224.0.0.255 for use by network protocols on a local network segment. Packets with an address in this range are local in scope, are note forwarded by IP routers and therefore go no further than the local network.

The table 2-1 is a partial list of reserved multicast addresses taken directly from the IANA database. The table lists the reserved link-local address, the network protocol function to which they have been assigned, and the person that requested the address or the RFC associated with the protocol.” [2]

Table 2-1: Link-Local Multicast Addresses

Address	**Usage**	**Reference**
224.0.0.1	All Hosts	[RFC 1112, JBP]
224.0.0.2	All Multicast Routers	[JBP]
224.0.0.3	Unassigned	[JBP]
224.0.0.4	DVMRP Routers	[RFC 1075, JBP]
224.0.0.5	OSPF Routers	[RFC 1583, JXM1]
224.0.0.6	OSPF Designated Routers	[RFC 1583, JXM1]
224.0.0.7	ST Routers	[RFC 1190, KS14]
224.0.0.8	ST Hosts	[RFC 1190, KS14]
224.0.0.9	RIP2 Routers	[RFC 1723, SM11]
224.0.0.10	IGRP Routers	[Farinacci]
224.0.0.11	Mobile-Agents	[Bill Simpson]
224.0.0.12	DHCP Server/Relay Agent	[RFC 1884]
224.0.0.13	All PIM Routers	[Farinacci]
224.0.0.14	RSVP-Encapsulation	[Braden]
224.0.0.15	All CBT Routers	[Ballardie]
224.0.0.16	Designated-SBM	[Baker]
224.0.0.17	All SBMS	[Baker]
224.0.0.18	VRRP	[Hinden]
224.0.0.19 to 224.0.0.255	Unassigned	[JBP]

2.2.2 Other Reserved Addresses

"The IANA typically assigns single multicast address requests for network protocols or network applications out of the 224.0.1.xxx address range. Multicast routers will forward these multicast addresses, unlike multicast address in the 224.0.0.xxx address range, which are local in scope and are never forwarded by routers.

The table 2-2 is a partial list of these single multicast address assignments." [2]

Table 2-2: Other Reserved Multicast Addresses

Address	Usage	Reference
224.0.1.0	VMTP Managers Group	[RFC 1045, DRC3]
224.0.1.1	NTP-Network Time Protocol	[RFC 1119, DLM1]
224.0.1.2	SGI-Dogfight	[AXC]
224.0.1.3	Rwhod	[SXD]
224.0.1.6	NSS-Name Service Server	[BXS2]
224.0.1.8	SUN NIS+ Information Service	[CXM3]
224.0.1.20	Any Private Experiment	[JBP]
224.0.1.21	DVMRP on MOSPF	[John Moy]
224.0.1.32	Mtrace	[Casner]
224.0.1.33	RSVP-encap-1	[Braden]
224.0.1.34	RSVP-encap-2	[Braden]
224.0.1.39	Cisco-RP-Announce	[Farinacci]
224.0.1.40	Cisco-RP-Discovery	[Farinacci]
224.0.1.52	Mbone-VCR-Directory	[Holfelder]
224.0.1.78	Tibco Multicast1	[Shum]
224.0.1.79	Tibco Multicast2	[Shum]
224.0.1.80 to 224.0.1.255	Unassigned	[JBP]

2.2.3 Administratively Scoped Multicast Addresses

In addition to the above mentioned ranges, IANA has reserved the IP range from 239.0.0.0 to 239.255.255.255 as administratively scoped addresses for use in private multicast domains. This range of IP's is similar to the private IP ranges in Class A, B and C. Hence they are available to be used without any restrictions in private networks.

2.3 Multicast Distribution Trees

To understand the flow of multicast traffic in a network it is important to know the models or modes of multicast transmission that is designed to control the traffic flow. Unlike unicast transmission where it is host to host traffic over a network link, multicast works on a tree model better known as multicast distribution trees. Here the traffic is send from a source often referred to as the root of the distribution tree, to a bunch of arbitrary group of hosts, represented by the multicast group address which can be seen as the branches of the tree.

The distribution trees define the path taken by the IP multicast traffic in the network ensuring that the packets are delivered to all receivers. "Two basic types of multicast distribution trees are source trees and shared trees." [2]

2.3.1 Source Trees

"The simplest form of multicast distribution tree is a source tree whose root is the source of the multicast traffic and whose branches form a spanning tree through the network to the receivers. Because this tree uses the shortest path through the network, it also is referred to frequently as a shortest path tree (SPT).

Figure 3 shows an example of an SPT for group 224.1.1.1 rooted at the source, Host A, and connecting two receivers, Hosts B and C." [2]

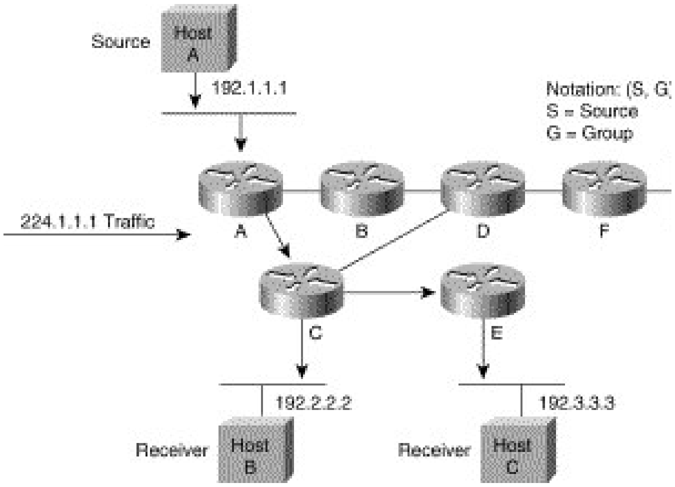


Figure 3: Host A Shortest Path Tree [2]

"The special notation of (S, G) pronounced "S comma G", enumerates an SPT where S is the IP address of the source and G is the multicast group address." [2]. Hence, the multicast group in the above example can be notated as (192.1.1.1, 224.1.1.1).

This notation also emphasizes the point that there is no direct one to one communication with respect to multicast communication. For example, if Host B is sending traffic back to the group 224.1.1.1 which has Host A and C as recipients, there will be another SPT noted as (192.2.2.2, 224.1.1.1), shown in Figure 4. There will be a separate tree for every source in this set up.

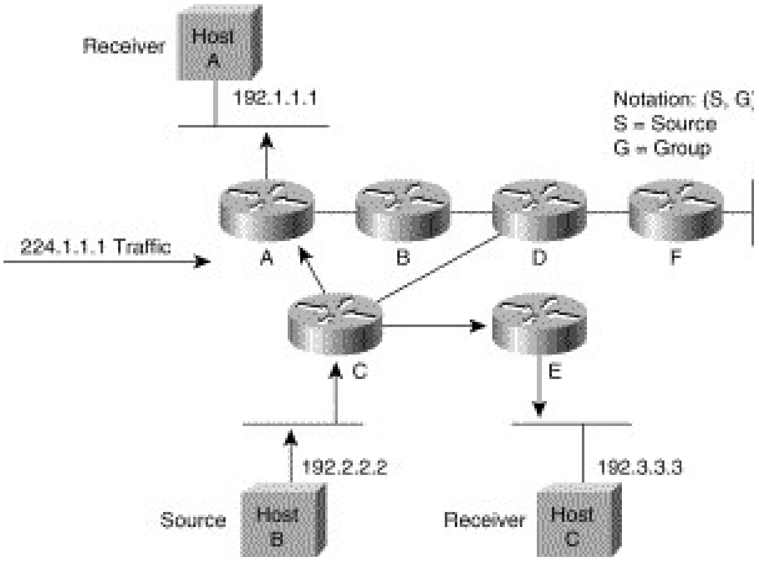


Figure 4: Host B Shortest Path Tree [2]

2.3.2 Shared Trees

"Unlike source trees that have their roots at the source, shared trees use a single common root placed at some chosen point in the network. Depending on the multicast routing protocol, this root is often called a rendezvous point (RP) or core, which lends itself to shared trees' other common names: RP Trees (RPT) or core-based trees (CBT).

The following figure shows a shared tree for multicast address group 224.2.2.2, with the RP located at Router D. In case of a shared tree, the source should send their traffic to the RP in order to reach all the recipients." [2]

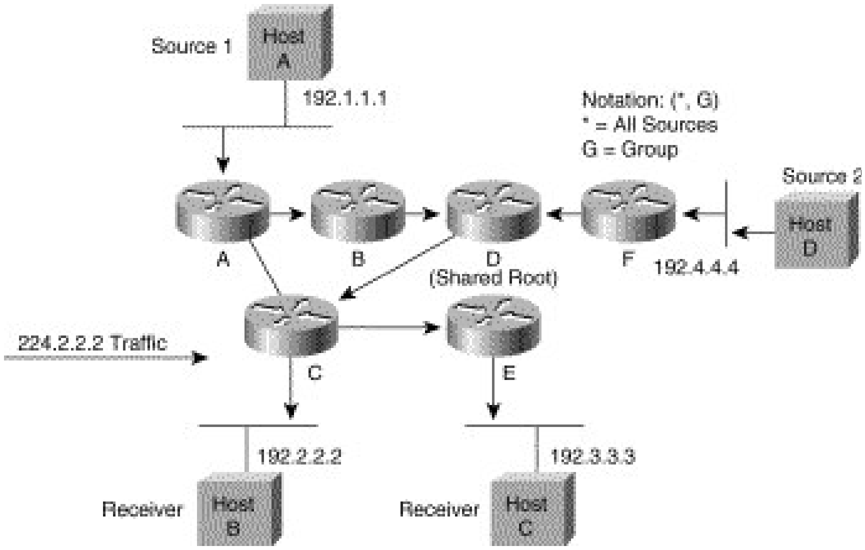


Figure 5: Shared Distribution Tree [2]

This particular example has two sources, Host A and Host D sending multicast group traffic to the shared root as shown in the diagram, which is Router D. This traffic then flows down the tree to the two receivers Host B and Host C. In Source tree, each source had its own tree with the source as the root, but in this case since any source in the group will share the same root, the notation for this would be a wild card notation represented as (*,G), pronounced as “star comma G”. This notation represents *, any source and G is the multicast group address. The tree in the example can be named (*, 224.2.2.2).

2.3.2.1 Bidirectional Shared Trees

“Shared trees can be subdivided into two types: unidirectional and bidirectional. In the bidirectional case, multicast traffic may flow up and down the shared tree to reach all receivers. Figure 6 shows an example of a bidirectional shared tree.

Notice that multicast traffic from Host B is being forwarded by its first hop router *up the tree* towards the root of the shared tree as well as down the tree toward the other receivers (in this case, Host A)” [2]. The link between Router B and Router D has the source traffic towards the root going in one direction (up) and the recipient traffic to Host A flowing in the other direction (down).

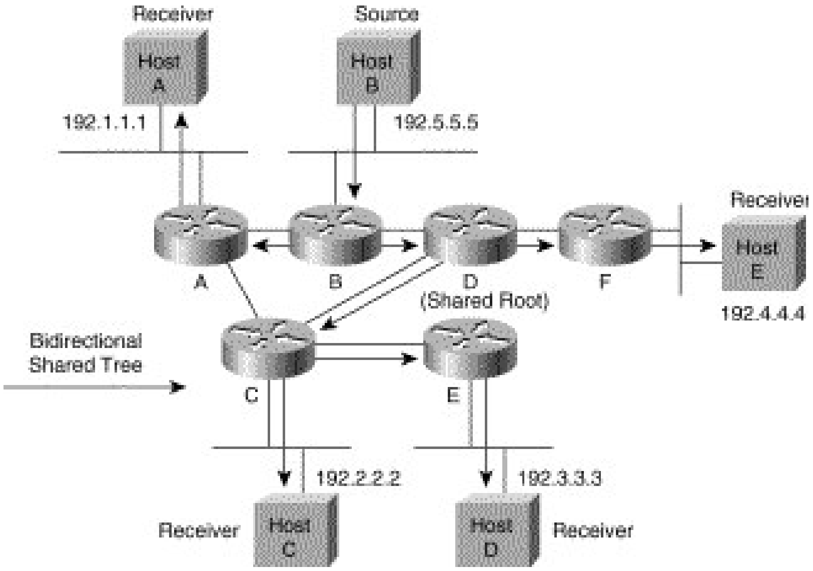


Figure 6: Bidirectional Shared Tree [2]

2.3.2.2 Unidirectional Shared Trees

Unidirectional shared trees allow multicast traffic to flow down the shared tree from root to receivers. So, with reference to the example in figure 6, the traffic from Source B will not be able to use the link between Router B and Router D as part of the multicast group to send the source traffic to the shared root. "The source of multicast traffic must use some other means to first get the traffic to the root so that it can be forwarded down the shared tree". [2]

"One method that can be used is to have the root join an SPT rooted at the source to pull the traffic to the root for forwarding down the shared tree. Figure 7 shows a unidirectional shared tree where the root has joined the SPT to source Host B to pull Host B's multicast traffic to the root. When the root receives the traffic, it is forwarded down the shared tree to the other receivers. Protocol Independent Multicast (PIM) uses this method to get source multicast traffic to the root or RP.

Another method that gets source multicast traffic to the root is for the first-hop router (Router B) to unicast the traffic directly to the root. The CBT multicast routing protocol uses this method when a source-only host wants to send multicast traffic to the group, as depicted in Figure 8. Host A is a source-only host that has not joined the multicast group and is therefore not on a branch of the bidirectional shared tree." [2]

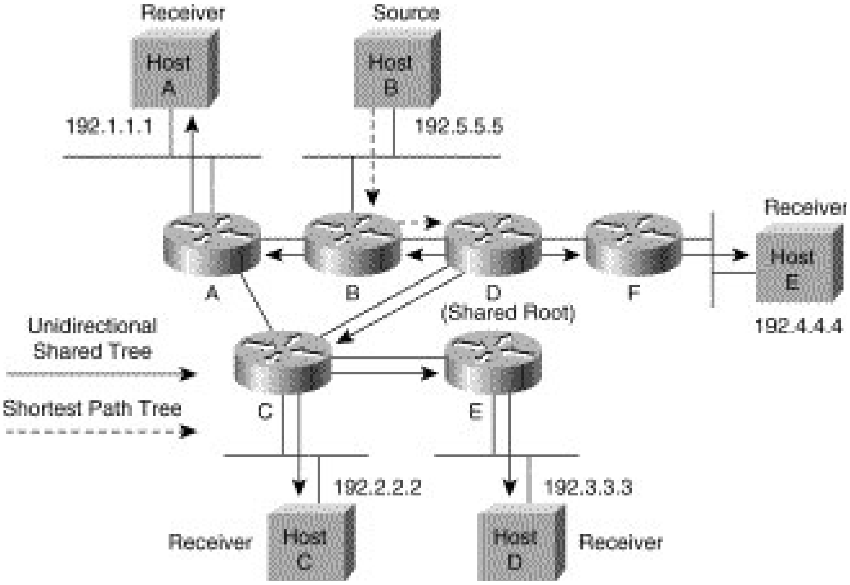


Figure 7: Unidirectional Shared Tree Using SPTs to Get Traffic to the Root [2]

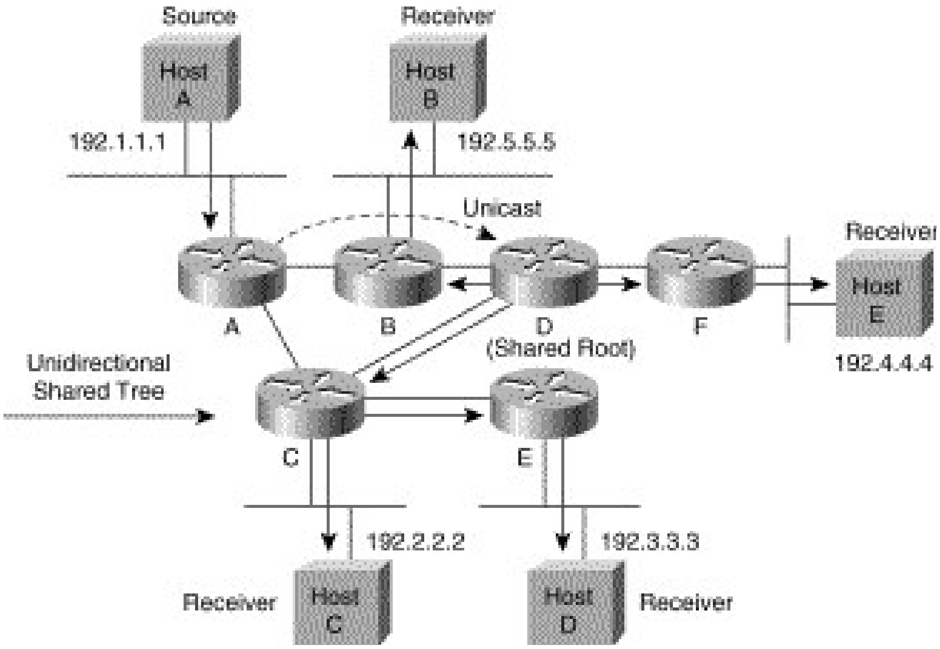


Figure 8: CBT Bidirectional Shared Tree Using Unicast to Get Traffic to the Root [2]

“In this example, Host A is the source, and Host B is now a receiver. Router B is encapsulating the multicast traffic received from Host A and unicasting it directly to the root via an IP-IP tunnel. The root de-encapsulates the packet and sends it down the shared tree.” [2]

2.4 Multicast Forwarding

The basic concept of packet forwarding in Multicast is a little bit different from the usual unicast model. In unicast model, the traffic flows between the source and destination in a single path, where in the intermediate routers forward the traffic to the next hop interface, based on the combined information from the destination address field in the IP packet and the routing table in the router. "In the multicast model, the source is supposed to send traffic to an arbitrary group of hosts represented by the multicast group address in the destination address field of the IP packet."[2] Unlike unicast model, the forwarding decision is not based on the destination address in the IP packet, but the router may end up forwarding the traffic to more than one out going interfaces in order to ensure the delivery of the packets to all participants, hence making it a complex process.

The basic multicast forwarding process used in most of the multicast routing protocols is called Reverse Path Forwarding (RPF), is explained in this section along with information on multicast forwarding caches, TTL thresholds, and administratively scoped boundaries.

2.4.1 Reverse Path Forwarding

"Virtually all IP multicast routing protocols make use of some form of RPF *or incoming interface check* as the primary mechanism to determine whether to forward or drop an incoming multicast packet. When a multicast packet arrives at a router, the router performs an RPF check on the packet. If the RPF check is successful, the packet is forwarded; otherwise, it is dropped.

For traffic flowing down a source tree, the RPF check mechanism works as follows:

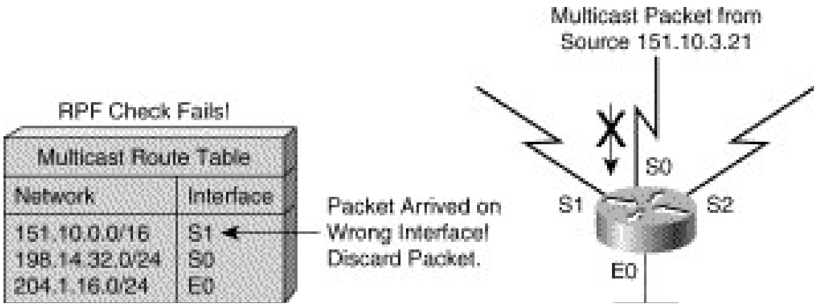
1. The router examines the source address of the arriving multicast packet to determine whether the packet arrived *via an interface that is on the reverse path back to the source.*

2. If the packet arrives on the interface leading back to the source, the RPF check is successful and the packet is forwarded.

3. If the RPF check fails, the packet is discarded." [2]

The method used to determine if the packets was received from an interface traceable back to the source could vary, depending on the routing protocol in use. Some multicast routing protocols uses its own routing table to make this decision, while some other use the already available unicast routing table. An example for the former is the Distance Vector Multicast Routing Protocol (DVMRP), whereas “PIM and CBT are examples of multicast protocols that typically use the unicast routing table to perform the RPF check. PIM and CBT are not limited to using just the unicast routing table for the RPF check, however. They also can use reachability information from a DVMRP router table or a Multicast Border Gateway Protocol (MBGP) route table, or they can statically configure RPF information.

Figure 9 illustrates the RPF check process. This example uses a separate multicast routing table, although the concept is the same if the unicast routing table or some other reachability table is used.

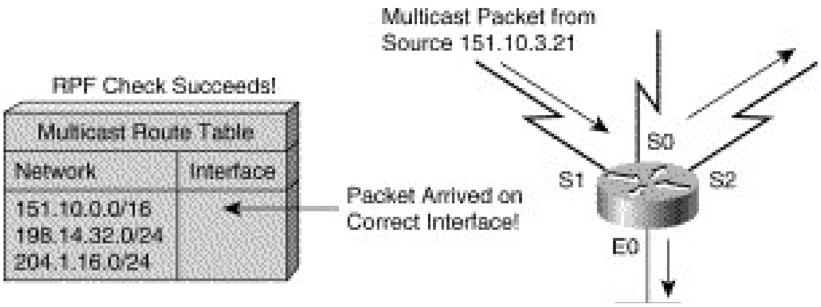


Network	Interface
151.10.0.0/16	S1
198.14.32.0/24	S0
204.1.16.0/24	E0

Figure 9: RPF Check Fails [2]

A multicast packet from source 151.10.3.21 is received on interface S0. A check of the multicast routing table shows that the interface on the reverse path back to the source is S1, not S0. Therefore, the RPF check fails, and the packet is discarded.

Figure 10 is another example of a multicast packet from source 151.10.3.21 arriving at the router, this time via interface S1.



Network	Interface
151.10.0.0/16	
198.14.32.0/24	
204.1.16.0/24	

Figure 10: RPF Check Succeeds [2]

In this case, the RPF check succeeds as interface S1 is on the reverse path back to the source, and therefore the packet is forwarded to all interfaces on the outgoing interface list. (Notice that the outgoing interfaces don't have to necessarily include all interfaces on the router.)"[2]

2.4.2 Multicast Forwarding Cache

As explained in the previous section, determining the authenticity of the source address for an incoming multicast packet is important in the whole process of multicast routing. This is determined with the help of checks performed on various routing tables associated to the multicast routing protocol implemented. This check on incoming packets has a substantial impact on the performance of the router. The concept of Multicast Forwarding Cache is developed to ease this effort.

"From the router's point of view, each source or shared tree can be represented in a *multicast forwarding cache* entry as an incoming interface, associated with zero or more outgoing interfaces. This *multicast forwarding cache* entry is sometimes referred to as a *multicast route table* entry. Note that bidirectional shared trees modify this process slightly, as they don't make a distinction between incoming and outgoing interfaces because traffic can flow up and down the tree."[2]

Creation of the multicast forwarding cache entry helps the multicast router to determine the RPF interface which is now mapped as the incoming interface of the multicast forwarding cache entry. "If a change occurs in the routing table used by the RPF check process, the RPF interface must be recomputed and the multicast forwarding cache entry updated to reflect this information. Note that outgoing interfaces are determined in various ways depending on the multicast routing protocol in use."[2]

Following example shows a Cisco multicast routing table entry:

```
151.10.3.21/32, 224.2.127.254), 00:04:15/00:01:10, flags: T
Incoming interface: Serial1, RPF nbr 171.68.0.91
Outgoing interface list:
Serial2, Forward/Sparse, 00:04:15/00:02:17
Ethernet0, Forward/Sparse, 00:04:15/00:02:13
```

This *(S, G)* entry describes the (151.10.3.21/32, 224.2.127.254) SPT as seen by the router in Figure 9 and Figure 10. From this information you can see that the entry has an incoming interface, **Serial1**, and two outgoing interfaces, **Serial2** and **Ethernet0**.

2.4.3 TTL Thresholds

The TTL values which is part of the IP multicast packet, is a value that is decremented by one each time the packet is forwarded by a router. Once this value reaches zero, the packet is dropped by the router. This value can be made use of, to determine network boundaries, by defining TTL thresholds on the outgoing interfaces of a router. An interface configured with a TTL threshold value will not forward packets with a TTL less than the threshold value. “For example, Figure 11 shows a multicast router with various TTL thresholds applied to its interfaces.

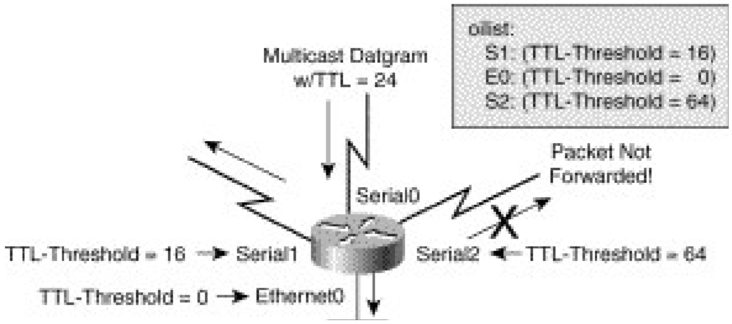


Figure 11: TTL Thresholds [2]

In this example, a multicast packet arrives via **Serial0** with its current TTL value at 24. Assuming that the RPF check succeeds and that interfaces **Serial1**, **Serial2,** and **Ethernet0** are all in the outgoing interface list, the packet therefore normally would be forwarded out these interfaces. Because some TTL thresholds have been applied to these interfaces, however, the router must make sure that the packet's TTL value, which is now down to 23, is greater than or equal to the interface's TTL threshold before forwarding the packet out the interface.

As you can see in this example, the packet is forwarded out interfaces **Serial1** and **Ethernet0**. Note that a TTL threshold of zero means that there is no TTL threshold on this interface. The packet's TTL value of 23 was below the TTL threshold value on interface **Serial2,** however, and therefore the packet was not forwarded out this interface.

TTL thresholds provide a simple method to prevent the forwarding of multicast traffic beyond the boundary of a site or region based on the TTL field in a multicast packet. This technique is referred to as *TTL scoping*. Multicast applications that must keep their traffic inside of a site or region transmit their multicast traffic with an initial TTL value so as not to cross the TTL threshold boundaries.

Figure 12 shows an example of TTL threshold boundaries being used to limit the forwarding of multicast traffic. Company ABC has set a TTL threshold of 128 on all router interfaces at the perimeter of its network.

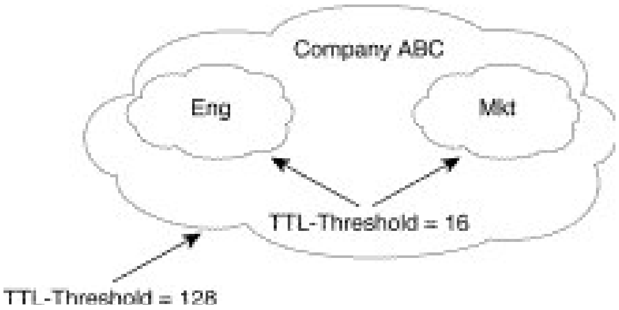


Figure 12: TTL Threshold Boundaries [2]

Multicast applications that want to constrain their traffic to within Company ABC's network need to transmit multicast packets with an initial TTL value set to 127. Furthermore, the engineering and marketing departments have set a TTL threshold of 16 at the perimeter of their networks. Therefore, multicast applications running inside of these networks can prevent their multicast transmissions from leaving their respective networks.

Table shows the typical initial TTL values and router interface TTL thresholds for various TTL boundaries." [2]

Table 2-3: Typical TTL Boundary Values

TTL Scope	**Initial TTL Value**	**TTL Threshold**
Local net	1	N/A
Site	15	16
Region	63	64
World	127	128

2.4.4 Administratively Scoped Boundaries

"Like TTL thresholds, administratively scoped boundaries may also be used to limit the forwarding of multicast traffic outside of a domain or sub domain. This approach uses a special range of multicast addresses, called *administratively scoped* addresses, as the boundary mechanism. If we configure an administratively scoped boundary on a router's interface, multicast traffic whose multicast group addresses fall in this range will not be allowed to enter or exit this interface, thereby providing a firewall for multicast traffic in this address range.

Figure 13 depicts the administratively scoped boundary mechanism at work. Here an administratively scoped boundary is set for the multicast address range 239.0.0.0 through 239.255.255.255 on interface Serial0. This mechanism effectively sets up a firewall that multicast packets in this range cannot cross.

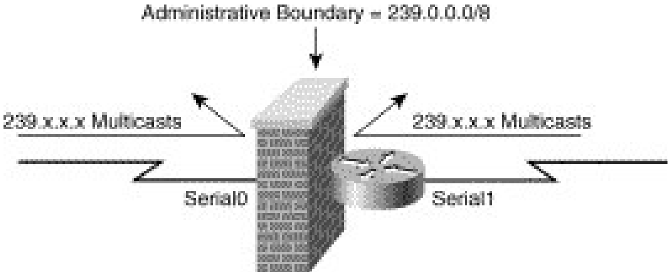


Figure 13: Administrative Boundary Mechanism [2]

As mentioned earlier in the paper, the administratively scoped multicast addresses fall into the range of 239.0.0.0 through 239.255.255.255 and are considered to be locally assigned i.e. they are not used in public networks. The administratively scoped boundary mechanism allows for the enforcement of this convention,"[2] by allowing the usage of the same network address on two sides of the network if separated by an interface that is acting as an administrative boundary, thereby preventing the multicast traffic that falls in this range from entering or leaving the network.

"Figure 14 is an example of the use of administratively scoped boundaries. Here, Company ABC has used different ranges of administrative addresses to prevent the forwarding of multicast traffic outside of specific boundaries.

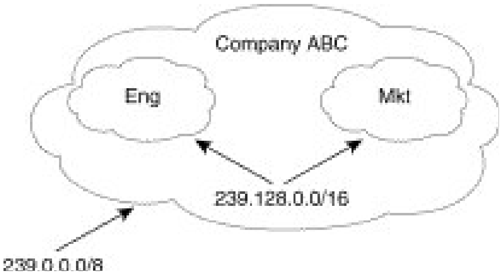


Figure 14: Administratively Scoped Boundaries [2]

In this example, Company ABC has configured an administratively scoped boundary of 239.0.0.0/8 on all router interfaces at the perimeter of its network. This boundary prevents any multicast traffic in the range of 239.0.0.0 through 239.255.255.255 from entering or leaving the network. Similarly, the engineering and marketing departments have configured an administratively scoped boundary of 239.128.0.0/16 around the perimeter of their networks. This boundary prevents multicast traffic in the range of 239.128.0.0 through 239.128.255.255 from entering or leaving their respective networks. It also means that the multicast address range 239.128.0.0 through 239.128.255.255 is being used independently by both the engineering and marketing departments. This reuse of multicast address space permits multicast addresses to be used more efficiently inside of Company ABC." [2]

2.5 Multicast Routing Protocol Categories

"The current multicast protocols can be subdivided into three basic categories:

1) Dense mode protocols (DVMRP and PIM-DM)
2) Sparse mode protocols (PIM-SM and CBT)
3) Link-state protocols (MOSPF)

Some protocols, such as PIM, are capable of operating in either dense or sparse mode depending on how the router is configured. It is also possible to configure Cisco PIM routers to make the sparse/dense decision dynamically on a multicast group basis." [2]

2.5.1 Dense Mode Protocols

Dense mode protocols such as DVMRP and PIM DM make use of shortest path tress in order to deliver the multicast traffic using a *push* principle. This principle makes an assumption that all subnets in the network have a recipient for the multicast traffic being sent and hence floods the traffic to all points in the network. This can be compared to the FM radio transmission where the signal is available at all locations and the interested users tuning in to the right frequency.

Flood and Prune Behavior

One major disadvantage of this kind of flooding is that, there could be multiple nodes and links that have unwanted traffic flowing through them there by consuming few of the most precious commodities in networking like bandwidth, CPU processing speed and performance. "Hence, to avoid the unnecessary consumption of valuable network resources, routers send Prune messages back up the source distribution tree to shut off unwanted multicast traffic. The result is that branches without receivers are *pruned* off the distribution tree, leaving only branches that contain receivers.

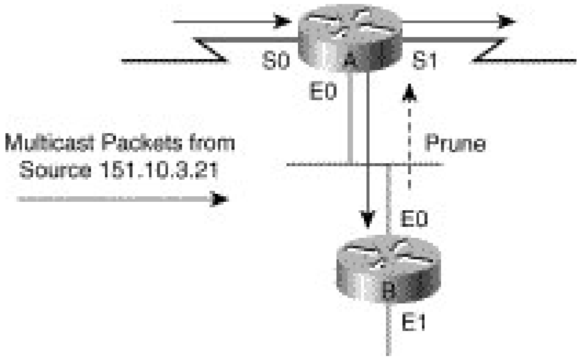


Figure 15: Pruning a Dense Mode Flow [2]

In Figure 15 Router B is responding to unwanted multicast traffic with a Prune message. When Router A receives the Prune message for the (S, G) multicast traffic stream on an outgoing interface (in this example, Ethernet0), the router places that interface into Pruned state and stops forwarding the (S, G) traffic out the interface.

These prune message have a timeout value associated with them such that, when they time out, they cause the router to put the interface back into forward state and to start flooding multicast traffic out this interface again.

Following example shows the Cisco multicast route table entry for Router A in Figure 15.

```
(151.10.3.21/32, 224.2.127.254), 00:04:15/00:01:10, flags: T
Incoming interface: Serial0, RPF nbr 171.68.0.91
Outgoing interface list:
Serial1, Forward/Dense, 00:04:15/00:00:00
Ethernet0, Prune/Dense, 00:00:25/00:02:35
```

Notice that interface Ethernet0 is in prune state (denoted by the **Prune/Dense** indicator) and that no group 224.2.127.254 traffic (from source 151.10.3.21) is being forwarded out this interface. The example also shows that the Prune will timeout in 2 minutes and 35 seconds (as indicated by the last timer value on the line). When the prune times out, the state of this interface will return to **Forward/ Dense** and traffic will once again begin flowing out this interface. Assuming that the downstream router (Router B in this case) still has no need to receive the multicast traffic, it again sends a Prune to shut off the unwanted traffic.

The timeout values used for Prunes depend on the multicast routing protocol in use but typically range from 2 to 3 minutes. This periodic flood and Prune behavior is characteristic of dense mode protocols, such as DVMRP and PIM-DM.” [2]

Grafting

Grafting is a mechanism that ensures that a new recipient joining the multicast group on a network portion that was part of a previously pruned branch does not have to wait for the Prune timer to expire before it can start participating. When the new receiver joins the multicast group, “the router detects the new receiver and immediately sends a Graft message up the distribution tree toward the source. When the upstream router receives the Graft message, the router immediately puts the interface on which the Graft was received into forward state so that the multicast traffic begins flowing down to the receiver.

Figure 16 shows the Graft process. In this example, the source, Host E, is transmitting multicast traffic down the SPT (denoted by the solid arrows) to receivers Host A, B, and C.

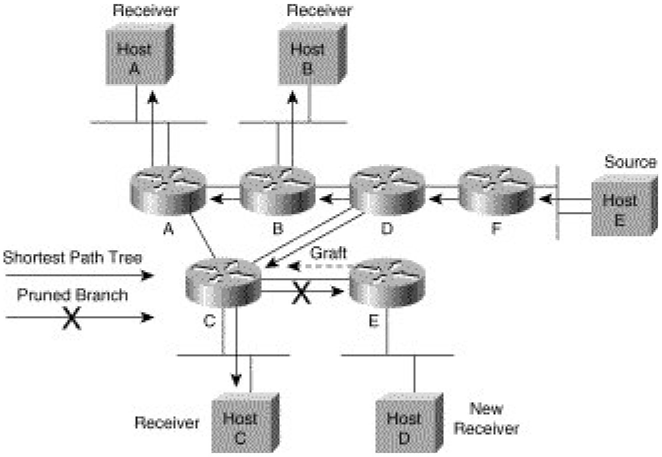


Figure 16: Dense Mode Grafting [2]

Router E previously has pruned its link to Router C, as it initially had no directly connected receivers. At this point in the example, Host D joins the multicast group as a new receiver. This action prompts Router E to send a Graft message up the SPT to Router C to immediately restart the flow of multicast traffic. By using this Graft process, Router E can avoid having to wait for the previous Prune to time out, thereby reducing the join latency as seen by Host D.” [2]

2.5.2 Sparse Mode Protocols

The name sparse mode itself denotes that the amount of traffic is kept comparatively low with respect to dense mode protocol which uses flooding of IP packets in the network as a method for transmitting multicast traffic. Sparse mode protocols uses pull mechanism in place of push, so that the traffic is pulled from the receivers on demand, rather than being available on all nodes to be tuned in. This request from the receiver is sent using an explicit

Join mechanism. This ensures that there is not unwanted multicast traffic moving around in the network looking for recipients. When most of the dense mode protocols uses SPT's for passing traffic across the multicast network, sparse mode protocols mostly make use of shared trees and occasionally, as in the case of PIM-SM, SPTs to distribute multicast traffic to multicast receivers in the network.

Shared Tree Join Messages

"To pull the multicast traffic down to a receiver in a sparse mode network, a shared tree branch must be constructed from the root node (the RP in PIM-SM or the core in CBT) to the receiver. To construct this shared tree branch, a router sends a shared tree Join message toward the root of the shared tree. This Join message travels router by router toward the root, constructing a branch of the shared tree as it goes.

Figure 17 shows Joins being sent up the shared tree to the root. In this example, Router E has a locally connected receiver and therefore sends a Join message (represented by the dashed arrow) toward the root via Router C. The message travels hop by hop until it reaches the root and builds a branch of the shared tree (as shown by the solid arrows).

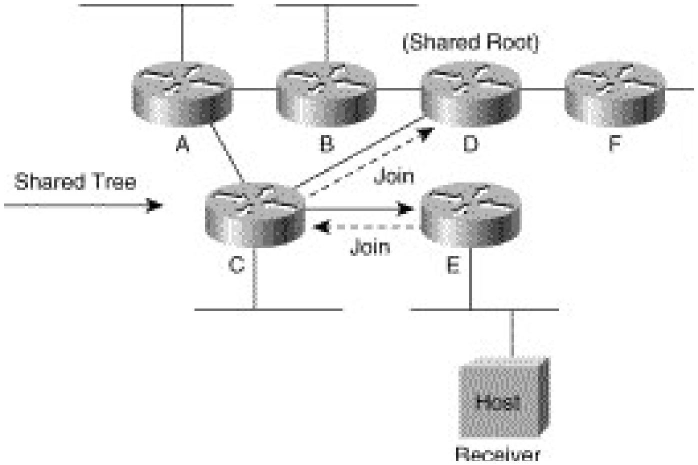


Figure 17: Shared Tree Join Message [2]

SPT Join Messages

In some cases (PIM-SM, for example), SPT Join messages may also be sent in the direction of the source to construct an SPT from an individual multicast source to receivers in the network. SPTs allow routers that have directly connected receivers to cut through the network and bypass the root node so that multicast traffic from a source can be received via a more direct path.

Figure 18 depicts an SPT being built using Join messages sent toward a specific multicast source. In this example, Router E sends an SPT Join message (shown by dashed arrows) toward the source via Router C. The SPT Join travels hop by hop until it reaches Router A, building the SPT (shown by solid arrows) as it goes.

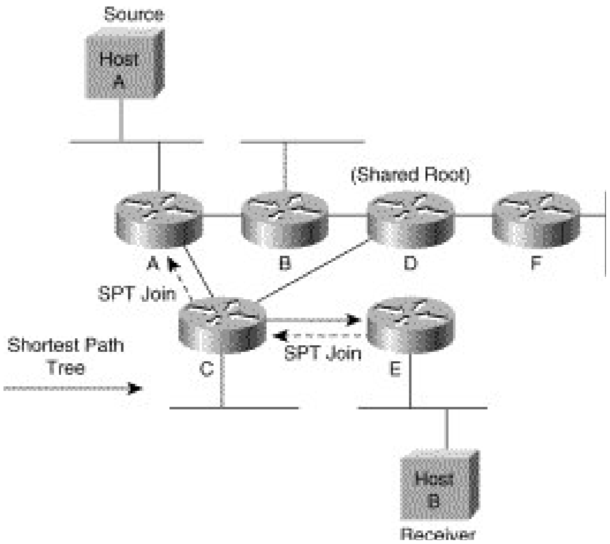


Figure 18: SPT Join Messages [2]

It is important to note that if the branches of distribution trees in a sparse mode network (either shared trees or SPTs) are not refreshed, they will time out and be deleted, thereby stopping traffic flow down the branch of the shared tree. To avoid this problem, the branches of sparse mode distribution trees are maintained by some form of periodic ***Join refresh*** mechanism that the routers send along the branch. Some protocols (PIM-SM, for example) handle the refresh by resending the Join message up the tree to refresh the branch periodically.

Prune Messages

In sparse mode, Prune messages are sent up the distribution tree when multicast group traffic is no long desired. This action permits branches of the shared tree or SPT that were created via explicit Joins messages to be torn down when they are no longer needed. For example, if a leaf router detects that it no longer has any directly connected hosts (or downstream multicast routers) for a particular multicast group, the router sends a Prune message up the distribution tree to shut off the flow of unwanted multicast group traffic. Sending Prune messages, instead of waiting for the branch of the sparse mode distribution tree to time out, greatly improves leave latency in the network.

Figure 19 shows this process in action. Host A has just left the multicast group; therefore, Router A no longer needs the traffic flowing down the shared tree (indicated by the solid arrows) and sends a Prune message up the shared tree toward the RP. This message prunes the link between Router A and Router B from the shared tree and stops the now unnecessary multicast traffic flow to Router A.

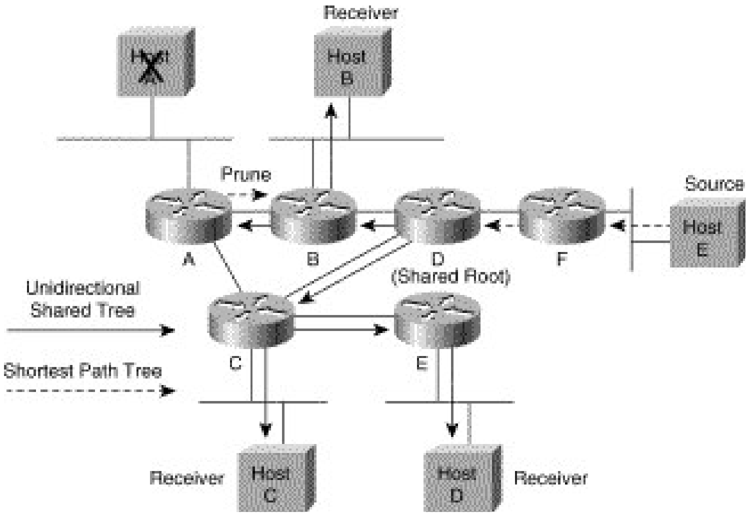


Figure 19: Sparse Mode Prune [2]

2.5.3 Link-State Protocols

Link-state protocols such as MOSPF function much like dense mode protocols in that they both use SPTs to distribute multicast traffic to the receivers in the network. Link-state protocols, however, don't use the flood and Prune mechanism that is used in DVMRP or PIM-DM. Instead, they flood special multicast, link-state information that identifies the whereabouts of group members (that is, receivers) in the network. All routers in the network use this group membership information to build shortest path trees from each source to all receivers in the group." [2]

2.6 Summary

This section gives a good introduction to the most important terminologies of IP multicast like different types of distribution trees, and multicast forwarding. The next section covers another basic IP multicast building block, IGMP (Internet Group Management Protocol), in detail. IGMP provides the necessary group membership signaling between hosts and routers and mostly stays within the local network.

3 Internet Group Management Protocol

“IGMP messages are used *primarily* by multicast hosts to signal their local multicast router when they wish to join a specific multicast group and begin receiving group traffic. Hosts may also (with some extensions defined in IGMPv2) signal to the local multicast router that they wish to leave an IP multicast group and, therefore, are no longer interested in receiving the multicast group traffic.” [2]

On top of this primary function, IGMP is a protocol used for a bunch of other purposes. As mentioned above, the primary function of IGMP is to form the Host Membership model. But “several multicast routing protocols, such as Distance Vector Multicast Routing Protocol (DVMRP) and Protocol Independent Multicast (PIM) version 1, make use of special IGMP message types to transmit their ***routing control information***. Other multicast control and diagnostic functions (such as mtrace), which are totally unrelated to the Host Membership model, also use a special IGMP message type to accomplish their task, and others are being proposed still.” [2]

Since IGMP contains information generated by host machines and is sent to the local multicast router, the routers can make use of this information to develop a list of members per interface who are participating in multicast communication. This means that even if at least one host on a particular outgoing interface of a router has signaled its desire, via IGMP, to receive multicast group traffic, the group membership for that interface is maintained as active.

The following section gives further details on IGMP v1 and 2 so as to understand the operational aspects and also a quick peek at the features that may be offered in IGMP v3.

3.1 IGMP Version 1

The reason why IGMPv1 is still important even after having IGPMv2 in place and IGMPv3 on the way is to have the understanding of the protocol since most IP stacks in today’s hosts still use IGMPv1. “The pervasive Microsoft Windows 95 operating system includes built-in support for IP multicast, but unless you download an upgraded version of Microsoft's Winsock DLL, you will be running IGMPv1. On the other hand, if you have upgraded to Windows 98, it contains full support for IGMPv2. The same is true for many UNIX implementations. Unless you install a patch or are running the very latest version of the UNIX operating system, you very possibly will be using IGMPv1. Because you are likely to be dealing with older versions of these platforms that support only IGMPv1, you need to understand how it works and know its limitations.

“This section focuses on the details of IGMPv1, including:

1) IGMPv1 Message Format
2) The IGMPv1 Query Process
3) IGMPv1 Report Suppression Mechanism
4) IGMPv1 Query Router
5) The IGMPv1 Join Process
6) The IGMPv1 Leave Process

3.1.1 IGMPv1 Message Format

IGMP messages are transmitted inside IP datagram’s and denoted by an IP protocol number of 2. IGMP messages are transmitted with the IP TTL field set to 1 and, therefore, are local in scope and not forwarded by routers. Figure 20 shows the format of an IGMPv1 message.

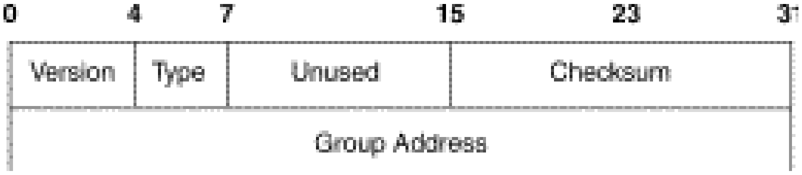


Figure 20: IGMPv1 Message Format [2]

The following sections define the fields, as depicted in Figure 20, which make up an IGMPv1 message.

Version Field

The Version field contains the IGMP version identification and is therefore set to 1. This field has been eliminated in version 2.

Type Field

In version 1 of IGMP, the following two message types are used between hosts and routers:

a) Membership Query
b) Membership Report

Checksum Field

The Checksum field is a 16-bit, one's complement of the one's complement sum of the IGMP message. The Checksum field is zeroed when making the checksum computation.

Group Address Field

The Group Address field contains the multicast group address when a Membership Report is being sent. This field is zero when used in the Membership Query and should be ignored by hosts.

3.1.2 The IGMPv1 Query-Response Process

IGMP primarily uses a Query-Response model that allows the multicast router to determine which multicast groups are active (that is, have one or more hosts interested in a multicast group) on the local subnet. Figure 21 shows the Query-Response process in operation.” [2]

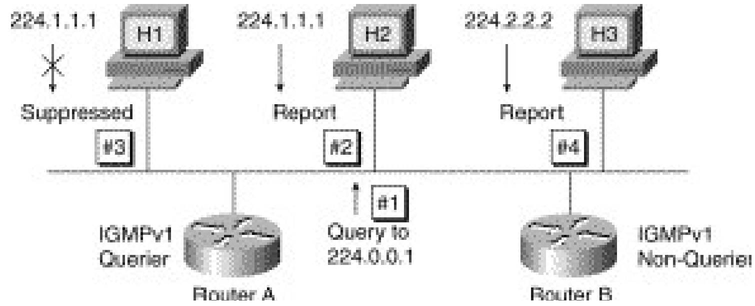


Figure 21: IGMPv1 Query-Response Process [2]

In the above example, Host H1 and H2 wants to participate in the multicast group 224.1.1.1 and Host H3 wants to be part of the multicast group 224.2.2.2. Out of the two routers in the network Router A is the IGMP Querier and Router B is non-querier. The election process for querier router is briefed later in this section. The Querier router will be responsible for performing the queries and the non-querier just listens and records the hosts' responses.

“The IGMPv1 Query-Response mechanism for this example works as follows:

1. Router A (the IGMP Querier) periodically (the default is every 60 seconds) multicasts an IGMPv1 *Membership Query* to the All-Hosts multicast group (224.0.0.1) on the local subnet. All hosts must listen to this group as long as they have enabled multicast so that these queries can be received.

2. All hosts receive the IGMPv1 Membership Query, and one host (in this example it's H2) responds first by multicasting an IGMPv1 *Membership Report* to the multicast group, 224.1.1.1, of which the host is a member. This report informs the routers on the subnet that a host is interested in receiving multicast traffic for group 224.1.1.1.

3. Because Host H1 is listening to multicast group 224.1.1.1, it hears the IGMPv1 Membership Report that was multicast by Host H2. Host H1, therefore, suppresses the

sending of its report for group 224.1.1.1 because H2 already has informed the routers on the subnet that there is at least one host interested in receiving multicast traffic for group 224.1.1.1. This *Report Suppression* mechanism helps reduce the amount of traffic on the local network.

4. Host H3 has also received the IGMPv1 Membership Query, and it responds by multicasting an IGMPv1 Membership Report to the multicast group 224.2.2.2, of which it is a member. This report informs the routers on the subnet that a host is interested in receiving multicast traffic for group 224.2.2.2.

As a result of this Query-Response exchange, Router A now knows that there are active receivers for multicast groups 224.1.1.1 and 224.2.2.2 on the local subnet. In addition, Router B has been eavesdropping on the whole process and also knows the same information.” [2]

3.1.3 Report Suppression Mechanism

As a result of the suppression of report to be sent by Host 1, as in this example, or if there are more than two hosts in an actual networking scenario, where there are n number of hosts participating in a particular multicast group, the directly connected routers will not be able to keep track of the number of hosts receiving the multicast traffic for a group. They will only have information about the multicast groups that has active recipients in the subnet. Not being able to keep track of the number of hosts, helps IGMP to scale networks with large number of hosts and at the same time it is not important for the routers to have that information since the packets are not send with individual hosts IP’s as the destination, but the multicast group address.

“The IGMP Report Suppression Mechanism helps to reduce the amount of IGMP traffic on a subnet to the minimum necessary to maintain a multicast group state. The following describes this mechanism in more detail:

1. When a host receives an IGMP Membership Query, the host starts a countdown report-timer for each multicast group it has joined. Each report-timer is initialized to a random value between zero and the maximum response interval. The default is 10 seconds.

2. If a report-timer expires, the host multicasts an IGMP Membership report for the active multicast group associated with the report-timer.

3. If the host hears another host send an IGMP Membership Report, it cancels its report-timer associated with the received Membership Report, thereby suppressing the sending of a Membership Report for the group." [2]

3.1.4 IGMPv1 Querier

If there is more than one router on a subnet, we would be able to save some bandwidth by ensuring that only one of the routers sends IGMPv1 queries to the hosts in the subnet. Here we call it the IGMPv1 Querier. RFC 1112 does not specify the Querier election process but mentions that the router elected as a Designated Router for other specific functions, will be chosen as the IGMPv1 Querier too. These additional function of DR as the querier was separated in IGMPv2.

3.1.5 The IGMPv1 Join Process

Typically, in case a host in a subnet is joining a multicast group and that host happens to be the first in that network to join this group, it may have to wait for the next Membership Query from the Querier before it can start participating in the group. The latency to join caused by waiting for the Query can be eliminated by sending one or more unsolicited Membership Reports for the multicast group it desires to join. This is an actual Membership Report, though it may sometimes be referred to as a Join message, but there is no such thing as an IGMP Join packet.

"Figure 22 illustrates this unsolicited join process. Here, Host H3 wants to receive traffic for multicast group 224.3.3.3.

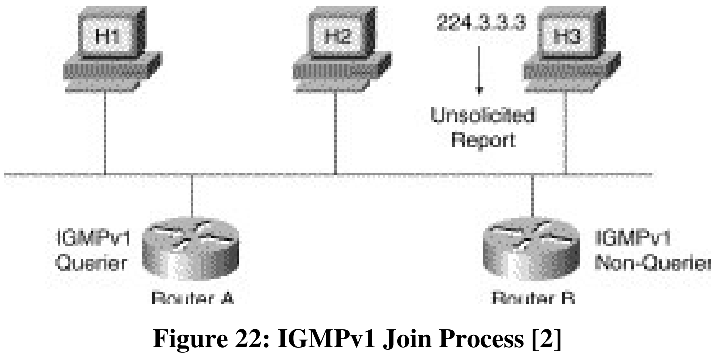


Figure 22: IGMPv1 Join Process [2]

Instead of waiting for the next Membership Query from Router A, it immediately multicasts an unsolicited IGMPv1 Membership Report to group 224.3.3.3 to inform the routers on the subnet of its desire to join this group.

***Note:** It is very important to understand that all the discussion we are having here regarding IGMP is for the hosts to inform the local router its intent to start or stop receiving IP multicast traffic. This has no significance if a host just has intent to* **send or initiate or be the source of** *multicast traffic for the group. "In this case, the host does not have to join the group to send traffic to the group. These kinds of hosts, which have send-only multicast application, just have to start sending traffic addressed to the multicast group in order to prompt the local multicast router to start forwarding the traffic to the receivers elsewhere in the network."[2]*

3.1.6 The IGMPv1 Leave Process

IGMPv1 has no specific way of leaving the multicast group. They just stop processing the packets when they no longer want to be part of the group. "There is no Leave Group message in IGMPv1 to notify the routers on the subnet that a host no longer wants to receive the multicast traffic from a specific group. They just stop responding to IGMP Queries from the Querier router. [2]

Once the last host in a subnet leaves the multicast group, the only way the local multicast router would get to know that there are no active receivers is when they stop getting Membership Reports. "To facilitate this process, IGMPv1 routers associate a countdown timer with an IGMP group on a subnet. When a Membership Report is received for the group on the subnet, the timer is reset. For IGMPv1 routers, this timeout interval is typically three times the Query Interval, or 3 minutes. This timeout interval means that the router may continue to forward multicast traffic onto the subnet for up to 3 minutes after all hosts have left the multicast group. This worst-case timing scenario is shown in Figure 23.

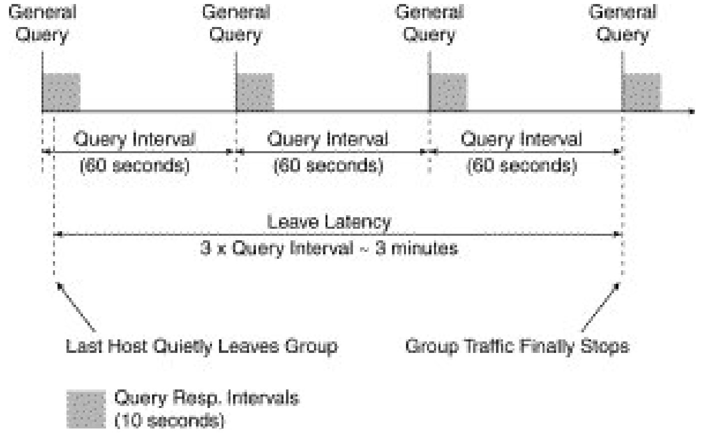

General Query
General Query
General Query
General Query
Query Interval (60 seconds)
Query Interval (60 seconds)
Query Interval (60 seconds)
Leave Latency
3 x Query Interval ~ 3 minutes
Last Host Quietly Leaves Group
Group Traffic Finally Stops
Query Resp. Intervals (10 seconds)

Figure 23: IGMPv1 Leave Group Timing [2]

This 3-minute leave latency can cause problems sometimes. Assume for the moment that company ABC is sourcing five different channels of high-rate training videos from a central, video-training server via five different multicast groups. Assume also that the bandwidth available to each remote field office in ABC's network has been sized to support two active training-video streams at a time.

If a user at one of these remote sites is unsure which "channel" (multicast group) contains which training video, the user may simply "channel surf" to locate the desired training video. This action would result in the user's workstation joining and leaving each multicast group as the user surfs the multicast groups for the desired video. In the worst-case scenario, the user may have joined five of the multicast groups and left four of them in rapid succession while surfing. As far as the user is concerned, he or she is now receiving the correct training video (albeit with lots of errors due to some serious congestion problems). Because the IGMPv1 leave latency can be as high as 3 minutes, however, the router will be forwarding all five high-rate video streams onto the subnet for up to 3 minutes.

If some method had been available for the user's workstation to signal the router that it had left the other four groups, this leave latency could be shortened substantially and the probability of this problem occurring reduced. This was one of the primary reasons for developing IGMP version 2." [2]

3.2 IGMP Version 2

"IGMPv2 was developed primarily to address some of the shortcomings of IGMPv1 that were discovered over time and through practical experience.

The Query and Membership Report messages in IGMPv2 are identical to the IGMPv1 messages with two exceptions. The first difference is that IGMPv2 Query messages are broken into two categories: General Queries, which perform the same function as the old IGMPv1 Queries, and Group-Specific Queries, which are queries directed to a single group. The second difference is that IGMPv1 Membership Reports and IGMPv2 Membership Reports have different IGMP Type codes. The (General) Query-Response process in IGMPv2, however, is the same as IGMPv1.

IGMPv2 includes several new features that are discussed in the sections that follow. Here is a quick summary of the key features added to IGMPv2:

1) **Querier election process**---Provides the capability for IGMPv2 routers to elect the Query Router without having to rely on the multicast routing protocol to perform this process.
2) **Maximum Response Time field**---A new field in Query messages permits the Query Router to specify the maximum Query-Response time. This field permits the tuning of the Query-Response process to control response burstiness and to fine-tune leave latencies.
3) **Group-Specific Query messages**---Permits the Query Router to perform the query operation on a specific group instead of all groups.
4) **Leave Group messages**---Provides hosts with a method of notifying routers on the network that they wish to leave the group.

The last two features enable hosts and routers to reduce the leave latency, which was such a problem in IGMPv1, from minutes down to a few seconds." [2]

3.2.1 IGMPv2 Message Format

The format of IGMPv2 messages is shown in Figure 24. The Type field has been merged with IGMPv1's Version field and now occupies a full octet. The values assigned to the various message types have been chosen carefully to provide backward compatibility with IGMPv1.

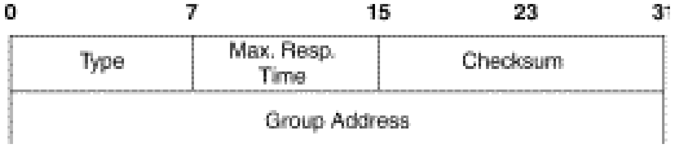


Figure 24: IGMPv2 Message Format [2]

The following sections define the fields, as depicted in Figure 24 that make up an IGMPv2 message.

Type Field

In version 2 of IGMP, the following four message types are used between hosts and routers:

1) Membership Query (Type code = 0x11)

There are two subtypes of Membership Query messages:

General Query---Used to determine which multicast groups are active in the same fashion as IGMPv1 does. A General Query is denoted by an all-zeros Group Address field.

Group-Specific Query---Used to determine whether a specific multicast group has any remaining members. A Group-Specific Query contains the address of the group being queried.

2) Version 1 Membership Report (Type code = 0x12)

This message type is provided solely for backward compatibility with IGMPv1.

3) Version 2 Membership Report (Type code = 0x16)

4) Leave Group (Type code = 0x17)

Maximum Response Time Field

The Maximum Response Time field is used only in Membership Query messages and specifies the maximum time in units of 1/10 of a second that a host may wait to respond to a Query message. The default value for this field is 100 (10 seconds).

Hosts use the Maximum Response Time value in this field as the upper limit for the random setting of their group report-timers, which are used by the Report Suppression mechanism. The value in this field may be tuned to control either the burstiness of membership responses or leave latency.

Checksum Field

The Checksum field is a 16-bit, one's complement of the one's complement sum of the IGMP message.

The Checksum field is zeroed when making the checksum computation.

Group Address Field

When a General Query is sent, the Group Address field is set to zero to differentiate it from a Group- Specific Query, which contains the multicast group of the group being queried.

When a Membership Report or Leave Group message is sent, this field is set to the target multicast group address." [2]

3.2.2 Query-Response Tuning

Query response tuning is a mechanism introduced as part of IGMPv2 with the idea of decreasing the burstiness in response from hosts where there are a number of active multicast groups. The current implementation has a general query interval of 60 seconds, query response interval of 10 seconds and a leave latency of 180 seconds. The hosts in the network pick a random value within the 10 second query response interval to respond to the Multicast query as explained earlier. This may cause issue with bursty responses from hosts if there are a number of hosts and a number of active multicast group. The Maximum Response Time field allows to expand the response time over a longer span than the usual 10 second window so that the burstiness that may be caused due to smaller window within which multiple hosts need to respond to multiple multicast groups can be avoided. This field can be configured on the IGMP Querier so as to inform the hosts what the upper limit has been set to.

The example below in Figure 25 shows "timing diagram with General Queries and Responses for IGMPv2 default timer settings for a subnet with 18 active groups spread across 18 different hosts." [2]

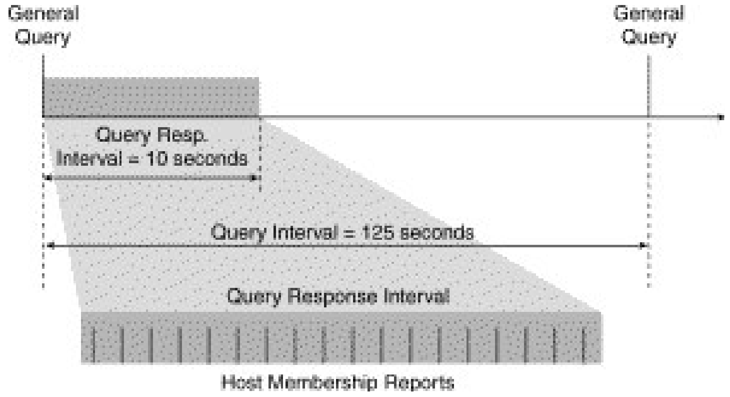


Figure 25: IGMPv2 Query-Response Tuning [2]

This short interval for that many responses can cause a bursty response since they need to be contained in the Query Response Interval. "By increasing the Maximum Response Time value, as shown in Figure 26, the period over which hosts may spread their responses to the General Query increases, thereby decreases the burstiness of the responses.

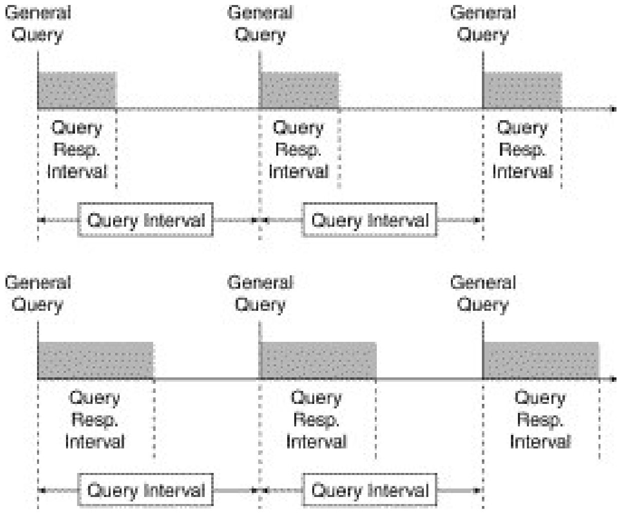


Figure 26: Decreasing Response Burstiness [2]

Reducing this burstiness comes with some penalties. Increasing the Query Response Interval by using a larger Maximum Response Time value also increases the leave latency because the Query Router must now wait longer to make sure that there are no more hosts for the group on the subnet. Therefore, your network design must strike a balance between burstiness and leave latency." [2]

3.2.3 IGMPv2 Leave Group Messages

"IGMPv2 defines a new Leave Group message type that hosts should send when they leave the group. The RFC says, "When a host leaves a multicast group, if it was the last host to respond to a query with a Membership Report for that group, it should send a Leave Group message to the all-routers multicast group (224.0.0.2)." That statement is more specific to the last host to respond, where as with respect to any host, RFC states that "a host may always send a Leave Group message when it leaves a group." Even though the wording is *may* and not *must*, most IGMPv2 implementations find it easier to implement the Leave Group processing by always sending a Leave Group message when the host leaves the group. Checking whether the host was the last to respond to a query for the group takes considerably more code.

3.2.4 IGMPv2 Group-Specific Query Messages

Another message type in IGMPv2 is the Group-Specific Query, which is sent by the IGMP Query Router, and whose purpose is to query a single group instead of all groups. In a Group-Specific Query, the Group Address field contains the target group being queried. IGMPv2 hosts that receive this message respond in the same manner as they do to a General Query. Another difference between Group-Specific Queries and General Queries is that Group-Specific Queries further reduce Leave Group latency by using a much smaller value of Maximum Response Time. The default is 1 second." [2]

3.3 IGMPv2 Leave Process

We have seen earlier how the lack of an efficient Leave process in IGMPv1 could cause bandwidth issues, since the router would continue to forward multicast traffic onto the local subnet for the extent of the Leave latency period. The addition of the Leave Group and Group-Specific IGMPv2 messages, coupled with the Maximum Response Time field, permit

IGMPv2 to reduce the leave latency to only a few seconds, which is a significant improvement over the default IGMPv1 value of 3 minutes.

The following example in Figure 27 shows IGMPv2's Leave process. The two hosts H2 and H3 are members of multicast group 224.1.1.1 and Host H2 wants to leave the group.

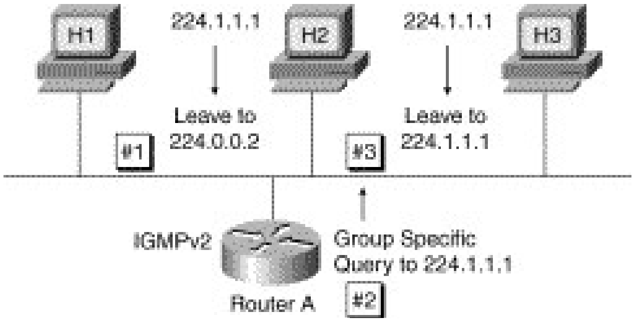


Figure 27: IGMPv2 Leave Process---Host H2 Leaves [2]

"The sequence of events for Host H2 to leave the group is as follows:

1. Host H2 multicasts an IGMPv2 Leave Group message to the All-Routers (224.0.0.2) multicast group to inform all routers on the subnet that it is leaving the group.

2. Router A (assumed to be the IGMP Query Router in this example) hears the Leave Group message from Host H2. However, because routers keep a list only of the group memberships that are active on a subnet---not individual hosts that are members---Router A sends a Group-Specific Query to determine whether any hosts remain for group 224.1.1.1. Note that the Group-Specific Query is multicast to the target group (that is, 224.1.1.1). Therefore, only hosts that are members of this group will respond.

3. Host H3 is still a member of group 224.1.1.1 and, therefore, hears the Group-Specific Query and responds to the query with an IGMPv2 Membership Report to inform the routers on the subnet that a member of this group is still present. Notice that the Report Suppression mechanism is used here, just as in the General Query case, to avoid an implosion of responses when multiple members of the group are on the subnet.

Host H3 is now the last remaining member of group 224.1.1.1. Now, assume that Host H3 also wants to leave the group, as shown in Figure 28.

The following sequence of events occurs when Host H3 leaves the group:

1. Host H3 multicasts an IGMPv2 Leave Group message to the All-Routers (224.0.0.2) multicast group to inform all routers on the subnet that it is leaving the group.

2. Again, Router A hears the Leave Group message (this time from Host H3) and sends a Group- Specific Query to determine whether any hosts remain for group 224.1.1.1.

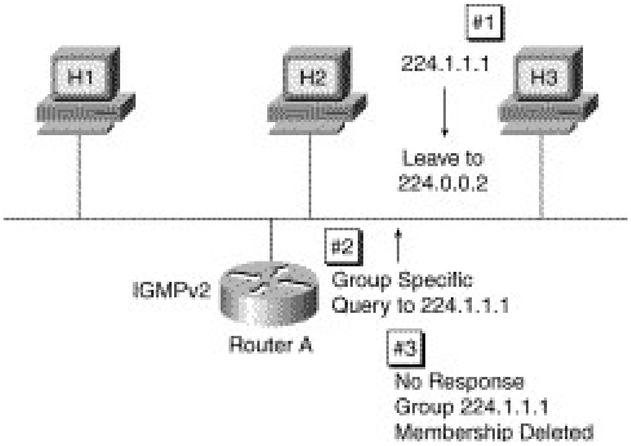


Figure 28: IGMPv2 Leave Process---H3 Leaves [2]

3. There are now no remaining members of group 224.1.1.1 on the subnet; therefore, no hosts respond to the Group-Specific Query. Getting no response, Router A waits a Last Member Query Interval (the default is 1 second) and sends another Group-Specific Query to which there is still no response. (The default number of tries is two.) At this point, Router A times out the group and stops forwarding its traffic onto the subnet.

Figure 29 depicts how the new mechanisms in IGMPv2 have reduced the leave latency.

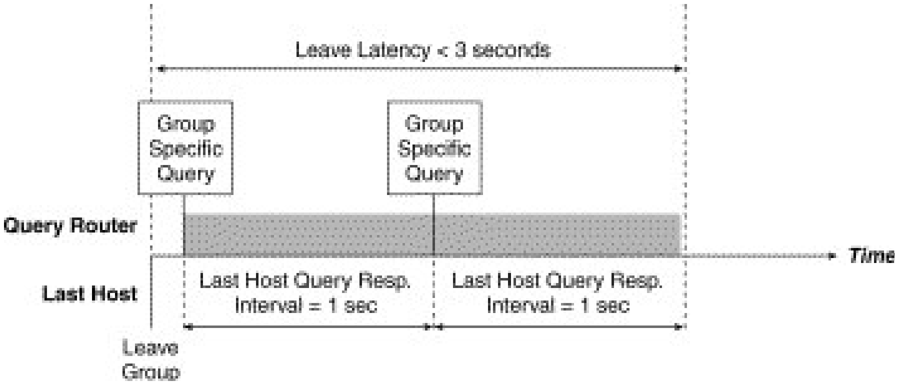

Leave Latency < 3 seconds
Group Specific Query
Group Specific Query
Query Router
Time
Last Host
Last Host Query Resp. Interval = 1 sec
Last Host Query Resp. Interval = 1 sec
Leave Group

Figure 29: IGMPv2 Leave Group Timing [2]

Instead of a total leave latency of roughly two complete query intervals, or 3 minutes, the leave latency is now less than 3 seconds.

3.4 Querier Election Process

Another important feature added in version 2 of IGMP is the Querier Election process. Instead of depending on the upper-layer multicast routing protocol, IGMPv2 uses the IP addresses in General Query messages to elect the IGMP Query Router via the following procedure:

1. When IGMPv2 routers start, they each multicast an IGMPv2 General Query message to the All-Multicast-Systems group (224.0.0.1) with their interface address in the Source IP Address field of the message.

2. When an IGMPv2 router receives a General Query message, the router compares the source IP address in the message with its own interface address. The router with the lowest IP address on the subnet is elected the IGMP Querier.

3. All non-querier routers start a querier timer that is reset whenever a General Query message is received from the IGMP Querier. The default duration of this timer is two times the Query Interval, or 250 seconds. If the querier timer expires, it is assumed that the IGMP Querier has gone down, and the election process is run again to elect a new IGMP Querier." [2]

3.5 Summary

In this section, we have gone over how IGMP is used by hosts to inform the routers in the subnet of its intention to join a multicast group. IGMPv2 was introduced to overcome some of the limitations that of version 1 that were exposed over time like leave messages and response time tuning. "This extension to the protocol significantly has reduced the leave latency that, in turn, allows routers and switches to respond quickly and shut off the flow of unnecessary multicast traffic to parts of the networks where it is no longer needed.

Finally, it is important to remember that IGMP is the only mechanism a host can use to signal routers of its desire to receive multicast traffic for a specific group. Hosts are neither aware of, nor should they be concerned with, which routing protocol is in use by the routers in the network. Instead, the routers in the network are responsible for knowing and understanding the multicast routing protocol in use and for making sure that the multicast traffic is delivered to the members of the group throughout the network." [2]

4 Distance Vector Multicast Routing Protocol

"Distance Vector Multicast Routing Protocol (DVMRP) was the first true multicast routing protocol to see widespread use. DVMRP is similar in many ways to Routing Information Protocol (RIP) with some minor variations added to support multicast.

Some key characteristics of DVMRP are

1) Distance vector based (similar to RIP)
2) Periodic route updates (every 60 seconds)
3) Infinity = 32 hops (versus 16 for RIP)
4) Poison Reverse has special meaning
5) Classless (that is, route updates include masks)

Currently, not all router vendors implement the same multicast routing protocols. However, most vendors support DVMRP to some degree and hence can be used between virtually all routers. This chapter explores some of the DVMRP related topics like: DVMRP Neighbor Discovery, route tables, exchanging DVMRP Route Reports, source distribution trees, multicast forwarding, pruning, grafting, and scalability." [2] Some of the basic features have already been discussed in the earlier section that covers the multicast basics and will not be elaborated here.

4.1 DVMRP Neighbor Discovery

"DVMRP Neighbor Discovery is important because DVMRP routers must maintain a database of DVMRP adjacencies with other DVMRP routers, especially when DVMRP is operating over multi access networks, such as Ethernet, FDDI, and so forth, because the network can have many DVMRP routers. The normal operation of a DVMRP router requires it to know its DVMRP neighbors on each interface.

This is accomplished by periodically multicasting DVMRP Probe messages to the All DVMRP Router group address (224.0.0.4). Figure 30 "shows the DVMRP Neighbor Discovery mechanism in action between two DVMRP routers connected to a common Ethernet network.

Here's an explanation of the DVMRP Neighbor Discovery mechanism depicted in Figure 30:

1. Router 1 sends a Probe packet first. Because Router 1 has not yet heard any other Probes from other routers, the Neighbor List in the Probe packet is empty.

2. Router 2 hears the Probe sent by Router 1 and adds the IP address of Router 1 to its internal list of DVMRP neighbors on this interface.

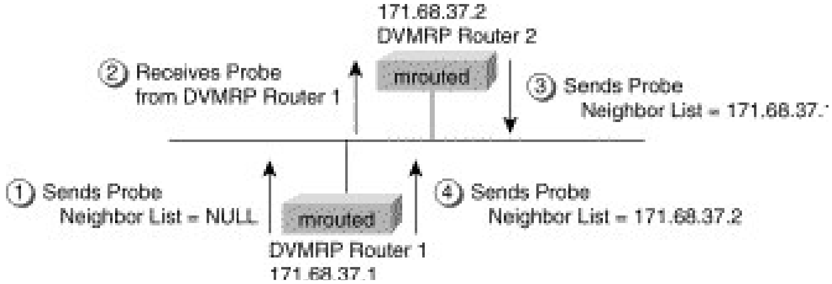


Figure 30: DVMRP Neighbor Discovery [2]

3. Router 2 then sends a Probe of its own with the IP address of Router 1 in the Neighbor List.

4. Router 1 hears the Probe sent by Router 2 and adds the IP address of Router 2 to its internal list of DVMRP neighbors on this interface. At the next Probe interval, Router 1 sends a Probe with the IP address of Router 2 in the Neighbor List.

When a DVMRP router receives a Probe with its own IP address listed in the Neighbor List, the router knows that a two-way adjacency has been successfully formed between itself and the neighbor that sent the Probe.” [2]

4.2 DVMRP Route Table

“In DVMRP, source network routing information is exchanged in the same basic manner as it is in RIP. That is, periodic (every 60 seconds) Route Report messages are sent between DVMRP neighbors. These Route Reports contain entries that advertise a source network (with a mask) and a hop-count that is used as the routing metric.

The routing information stored in the DVMRP routing table is separate from the unicast routing table and is used to

a) Build source distribution trees

b) Perform multicast forwarding (that is, Reverse Path Forwarding [RPF] checks)

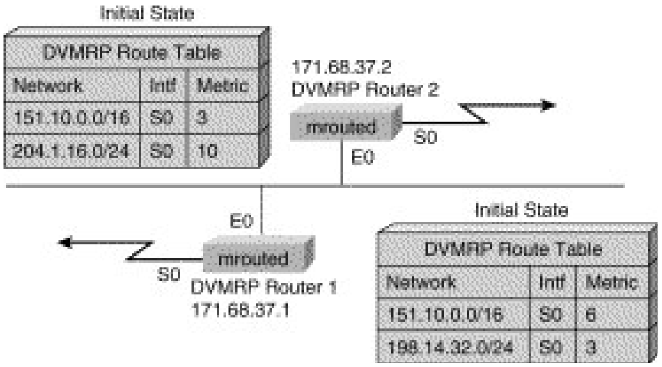
Following example shows a DVMRP route table in a Cisco router.

```
DVMRP Routing Table - 2 entries
130.1.0.0/16 [0/3] uptime 00:19:03, expires 00:02:13
via 135.1.22.98, Tunnel0, [version mrouted 3.8] [flags: GPM]
135.1.0.0/16 [0/3] uptime 00:19:03, expires 00:02:13
via 135.1.22.98, Tunnel0, [version mrouted 3.8] [flags: GPM]
```

4.3 Exchanging DVMRP Route Reports

DVMRP Route Reports are periodically exchanged in a manner similar to the way RIP unicast routing protocol information is exchanged. The key difference is that DVMRP routes are advertised with a subnet mask, making the protocol effectively a classless protocol. This section takes a closer look at the DVMRP Route Report exchange mechanism.

Figure 31 shows a portion of a multicast network consisting of two DVMRP routers connected via a common Ethernet." [2]



DVMRP Route Table

Network	Intf	Metric
151.10.0.0/16	S0	3
204.1.16.0/24	S0	10

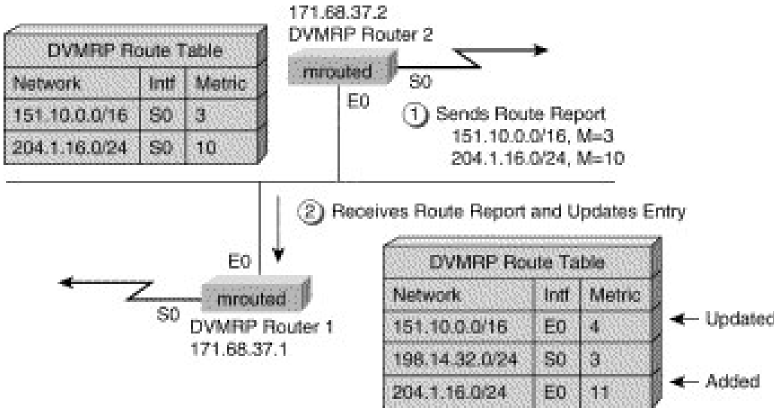
DVMRP Route Table

Network	Intf	Metric
151.10.0.0/16	S0	6
198.14.32.0/24	S0	3

Figure 31: DVMRP Route Exchange---Initial State [2]

The contents in the DVMRP route tables shows the routes that were learned from the Serial links and the state before any Route Reports are exchanged on the Ethernet link. It can be noticed that the network 151.10.0.0/16 is learned by both routers via their respective serial links already.

Let us assume that the first Route Report is sent by Router 2 as shown by Step 1 in Figure 32.



DVMRP Route Table

Network	Intf	Metric
151.10.0.0/16	S0	3
204.1.16.0/24	S0	10

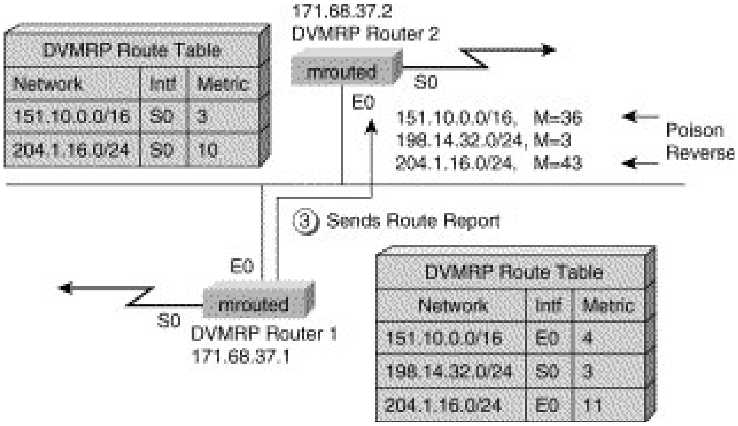
DVMRP Route Table

Network	Intf	Metric
151.10.0.0/16	E0	4
198.14.32.0/24	S0	3
204.1.16.0/24	E0	11

Figure 32: DVMRP Route Exchange---Steps 1 and 2 [2]

This report contains two route advertisements and is received by Router 1 noted in the figure as Step 2. The Router 1, upon receiving the routes from Router 2, responds by adding the newly found network to its own routing table and increments the metric by 1. It also notes that the route was received via E0 interface. It also notices that the second route received for network 151.10.0.0 from Router 2 has a better metric (4) than the one already in its routing table (6). So it updates its routing table with the new metric and the next hop interface is changed to E0. The behavior of incrementing the metric of received routes by one is similar to the function in RIP.

In the next step shown in Figure 33, we analyze the response Route Report sent by Router 1.



DVMRP Route Table

Network	Intf	Metric
151.10.0.0/16	S0	3
204.1.16.0/24	S0	10

DVMRP Route Table

Network	Intf	Metric
151.10.0.0/16	E0	4
198.14.32.0/24	S0	3
204.1.16.0/24	E0	11

Figure 33: DVMRP Route Exchange---Step 3 [2]

It is seen that in the Route report sent by Router 1, it has added 32, which is the maximum hop count for DVMRP, to the routes that is has received via E0. This process is called Poison Reverse and is used to inform "Router 2 that Router 1 is a child (that is, Router 1 is downstream of Router 2) and that Router 1 expects to receive multicast traffic from these source networks from Router 2." [2]

Note Most of the unicast DV routing protocols use Poison Reverse to advertise the unreachability of a particular network, where as DVMRP uses it for a different function. In DVMRP, a route that has been Poison Reversed indicates the upstream neighbor that this router with the Poison Reversed route is downstream in the multicast distribution tree.

Router 2 which initially had only two entries in its route table, upon receiving the Route report from Router 1 adds the network 198.14.32.0/24 to its route table (Step 4 in Figure 34).

In Step 5 (shown in Figure 34), Router 2 sends one more route report, after performing Poison Reverses on the 198.14.32.0/24 network that was earlier received from Router 1 by adding infinity (32) to the current metric, there by informing Router 1 that Router 2 will be acting as a downstream router for traffic from source network 198.14.32.0/24 via Router 1.

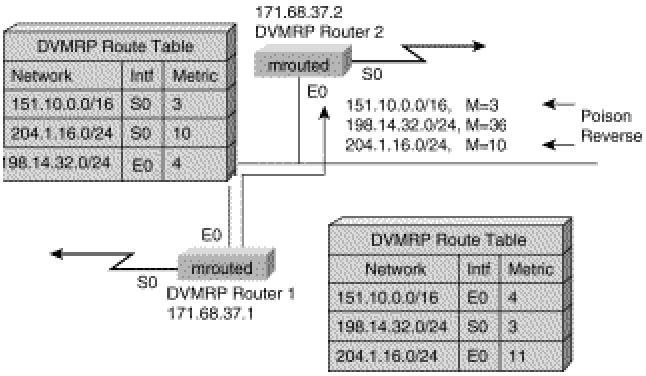


Figure 34: DVMRP Route Exchange---Steps 4 and 5 [2]

4.4 DVMRP Truncated Broadcast Trees

DVMRP is a dense mode protocol that uses *source distribution trees*, also known as *shortest path trees (SPTs)*, to forward multicast traffic. The basic building blocks for these DVMRP source distribution trees are the *truncated broadcast trees* built by the DVMRP routers using the metrics in the routers' DVMRP route tables. “To build a truncated broadcast tree, routers signal their upstream router (the neighbor advertising the best metric to a source network) that they are downstream by advertising a special Poison Reverse route metric for the source network back to the upstream router.

When sending a Route Report to the upstream DVMRP router for source network X, Poison Reverse the route by adding infinity (32) to the current metric for Network X and advertising it back to the upstream neighbor.

These Poison Reverse route advertisements tell the upstream (parent) router to forward any multicast traffic from the source network out this interface so that the downstream child router can receive it. Basically, the downstream router is telling the upstream router to "put me on the truncated broadcast tree for this source network." [2]

The following two Figures 35 and 36 shows how a sample DVMRP truncated broadcast tree is being built for Network S. Figure 35 is the status before the process of creating the truncated tree has begun and Figure 36 shows how the resulting truncated broadcast tree would look like.

“In this example, both Routers A and B advertise a route to Network S (shown by the solid arrows) with a metric (hop-count) of 1 to Routers C and D. Because Router D is downstream of Router B for Network S, Router D adds infinity (32) to the received metric to Poison Reverse the route advertisement (shown by the dashed arrows) to Network S and returns the route advertisement to Router B. This informs Router B that Router D is a child router and that the link to D should be placed in the outgoing interface list (that is, on the truncated broadcast tree) for Network S.

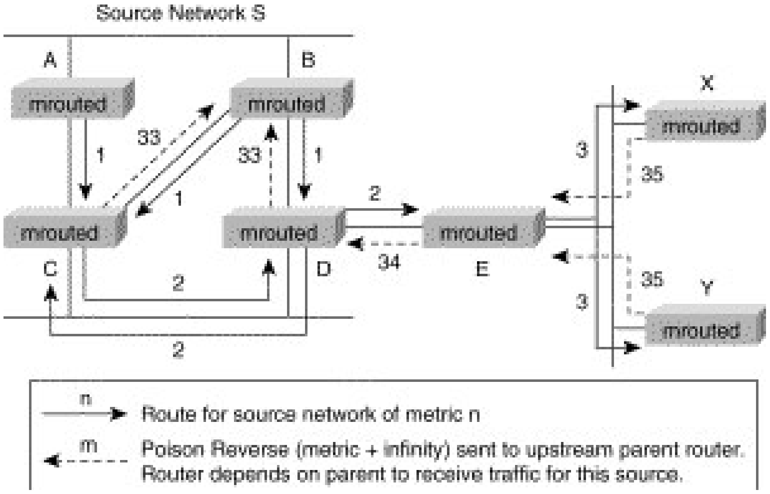

Source Network S
A
B
mrouted
33
1
1
33
1
2
C
D
E
2
34
2
X
Y
3
35
35
3
n
Route for source network of metric n
m
Poison Reverse (metric + infinity) sent to upstream parent router.
Router depends on parent to receive traffic for this source.

Figure 35: DVMRP Truncated Broadcast Tree [2]

Likewise, Router C receives advertisements with a metric of 1 from both Routers A and B. Using the lowest IP address as the tiebreaker, Router C selects Router B as its parent (that is, upstream) router toward Network S and sends a Poison Reverse (shown by the dashed arrow) to Router B. As a result, Router B places the link to Router C in the outgoing interface list that describes the truncated broadcast tree for Network S.

Routers C and D now both advertise a route to Network S out their common Ethernet. To avoid the delivery of duplicate packets, the router with the best metric to Network S is elected as the *designated forwarder* that is responsible for forwarding multicast packets from Network S to hosts on the Ethernet. In this case, the metrics are equal, so again, the lowest IP address is used as the tiebreaker, which results in Router D being the designated forwarder.

Route advertisements (solid arrows) continue to propagate away from Network S through Routers D and E on down to Routers X and Y as shown in Figure 35. At each point, the upstream router is sent a Poison Reverse advertisement (dashed arrows) to tell it to put the interface in the outgoing interface list for Network S. In the end, a truncated broadcast tree for Source Network S has been built as depicted by the solid arrows in Figure 36.

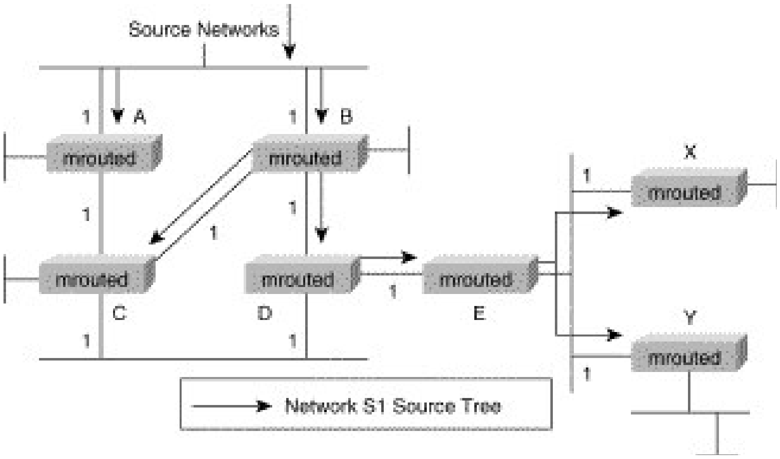


Figure 36: Resulting Truncated Broadcast Tree for Network S [2]

Truncated broadcast trees (such as the one shown in Figure 36) describe the distribution tree that is used to deliver multicast traffic from a specific source network to all other locations in the network *regardless of whether there are any group members in the network*. When a source begins to transmit, the multicast data is flooded down the truncated broadcast tree to all points in the network. DVMRP routers then prune this flow where the traffic is unwanted. The preceding example kept things simple by showing only the advertisements and the resulting truncated broadcast tree for Network S. In fact, each source network is associated with a truncated broadcast tree. Figure 37 shows the truncated broadcast tree that is built for Source Network S1.

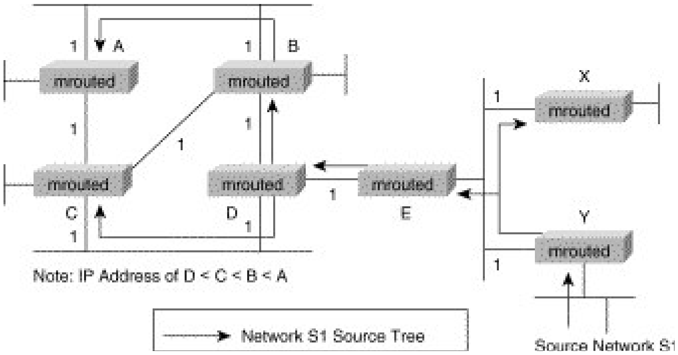


Figure 37: Truncated Broadcast Tree for Network S1 [2]

Because each source network has its own truncated broadcast tree, the example in Figure 37 shows a completely different tree that is rooted at Source Network S1 than the tree shown in Figure 38.

The key point here is that every subnet has a unique truncated broadcast tree that is defined by the DVMRP route metrics and the Poison Reverse mechanism. Multicast data is initially flooded over these truncated broadcast trees to reach all routers in the network.

Though the tree is formed in this fashion, it is not populated with traffic till some source starts sending traffic. But once a source, say Si starts sending traffic, an (Si, G) entry will be created in the multicast forwarding table. From this point, “the outgoing interface list in the (Si, G) entry is populated based on the Poison Reverse information for Network S in the DVMRP route table, thereby creating the distribution tree. This on-demand state creation mechanism saves considerable memory in the routers.” [2]

4.5 DVMRP Scalability

Even though DVMRP is used in MBone and other intra-domain multicast networks, there are some major scalability issues that prevent it from being used in large-scale multicast environments, the primary one being the max hop-count of 32. This limits the span of the network to a maximum of 31 hops beyond which the packets will be dropped before reaching the routers. This limitation will eliminate DVMRP being considered as the protocol in any application that interacts with internet since we cannot limit the traffic within 31 hops, due to obvious reasons.

Other than the max hop count limitation, DVMRP comes with the other limitations of a DV protocol, like slow convergence and a periodic route update mechanism that can pose a huge obstacle while having to handle nearly 50,000 prefixes active in the Internet. This would mean that if we have to deploy IP multicast over the internet, we need some significant developments over DVMRP to be used in the backbone.

4.6 Summary

Though DVMRP was the earliest of the multicast protocols developed, when it comes to deployment in large multicast networks, network engineers will have to treat it the same way as RIP is currently deployed in unicast networks. Just like RIP has give way to more efficient unicast routing protocols like OSPF, EIGRP, ISIS and BGP, unless there is a strong reason like the need to provide interface with existing DVMRP infrastructures, most network engineers wouldn't even consider using DVMRP in new network designs and that too, only until the network can be migrated to some other more efficient multicast protocol.

5 PIM Dense Mode

Protocol Independent Multicast (PIM) is one of the popular routing protocols in the multicast world. Just as the name suggests, it functions *independent of the IP routing protocol* deployed in the network. “That is, regardless of which unicast routing protocol(s) is (are) used to populate the unicast routing table (including static routes), PIM uses this information to perform multicast forwarding. Thus, even though we refer to PIM as a *multicast routing protocol*, it actually uses the existing unicast routing table to perform the RPF check function instead of maintaining a separate multicast route table.” [2] This property itself places a huge advantage with respect to resource utilization. The fact that PIM doesn't have to maintain its own routing table ensures that it doesn't have to send and/or receive multicast route updates like other protocols, such as MOSPF or DVMRP, thereby reducing PIM's overhead in a significant way in comparison to other multicast protocols.

“Cisco Systems' PIM implementation also permits the RPF check function to use other sources of routing information, such as a DVMRP route table, static multicast routes called *mroutes,* and most recently, special multicast Border Gateway Protocol (BGP) routes.

Some key characteristics of PIM dense mode (PIM-DM) are

1) Protocol independent (uses unicast route table for RPF check)
2) No separate multicast routing protocol (à la DVMRP)
3) Flood-and-prune behavior (3-minute cycle)
4) Classless (as long as classless unicast routing is in use)

Although PIM can be configured to operate in either sparse or dense mode, this chapter provides a brief overview of the dense mode operation including information on neighbor discovery, source trees, asserts, and scalability. Next chapter has a brief overview of PIM sparse mode.” [2]

5.1 PIM Neighbor Discovery

“Like DVMRP, PIM uses a Neighbor Discovery mechanism to establish PIM neighbor adjacencies. To establish these adjacencies, every Hello-Period (default: 30 seconds) a PIM multicast router multicasts a PIM Hello message to the *All-PIM-Routers* (224.0.0.13) multicast address on each of its multicast enabled interfaces. In PIMv1, these PIM Hello messages were called *PIM Query messages*. Like all PIMv1 packets, these packets are multicast to the 224.0.0.2 *All Routers* multicast group address and ride inside of Internet

Group Membership Protocol (IGMP) packets that have special type codes. On the other hand, PIMv2 has its own assigned protocol number (103) and, therefore, does not ride inside of IGMP packets.

5.1.1 PIM Hello Messages

PIM Hello messages contain a Holdtime value, which tells the receiver *when to expire the neighbor adjacency* associated with the sender if no further PIM Hello messages are received. The value that is sent in the Holdtime field is typically three times the sender's PIM Hello-Period, or 90 seconds if the default interval of 30 seconds is used. Keeping track of adjacent PIM-DM routers is very important to building and maintaining PIM-DM source distribution trees." [2]

5.1.2 PIM-DM Designated Router

In addition to establishing PIM Neighbor adjacencies, PIM Hello messages are also used to elect the Designated Router (DR) for a multi-access network. PIM routers makes use of the PIM Hello messages to determine the router with the highest IP address and this router is chosen as the DR for the network. "The DR is primarily used in sparse mode networks and has little meaning in dense mode networks, except when IGMPv1 is in use on an interface. In this case, the PIM-DR also functions as the IGMP Query Router since IGMPv1 does not have a Query Router election mechanism. An enhancement to the PIM protocol in PIMv2 is the use of new *DR-Priority option* which gives the network engineers an option to set the DR priority of each router in the LAN. The router with the highest priority is chosen as the DA and the default value is 1. So the network engineer can select any router he wants in the network, set the priority value greater than one and make it the DR and if he need to change it to a different one, he just have to assign a bigger value to the newly elected router. If all routers have the same DR priority value and/or have it set to the default value, the router with the highest IP will be chosen to break the tie.

"On Cisco routers, PIM neighbor adjacencies as well as the elected DRs can be displayed using the **show ip pim neighbor** IOS command. Following example 6-1 shows some sample output of a **show ip pim neighbor** command on a Cisco router.

```
Wan-gw8>show ip pim neighbor
PIM Neighbor Table
Neighbor Address Interface Uptime Expires Mode
153.68.0.70 FastEthernet0 2w1d 00:01:24 Dense
153.68.0.91 FastEthernet0 2w6d 00:01:01 Dense (DR)
153.68.0.82 FastEthernet0 7w0d 00:01:14 Dense
153.68.0.86 FastEthernet0 7w0d 00:01:13 Dense
153.68.0.80 FastEthernet0 7w0d 00:01:02 Dense
153.68.28.70 Serial2.31 22:47:11 00:01:16 Dense
153.68.28.50 Serial2.33 22:47:22 00:01:08 Dense
```

In this example, router **wan-gw8** has several neighbors on interface **FastEthernet0**. Notice that neighbor 153.68.0.91 is the DR on this Fast Ethernet segment and has been up for 2 weeks and 6 days (as indicated by **2w6d** in the **Uptime** column). Furthermore, the adjacency with this neighbor expires in 1 minute and 1 second if router **wan-gw8** doesn't receive any further PIM Hellos from this neighbor (for example, if the neighbor router went down or connectivity to it was lost). In this case, a new DR would be elected for the Fast Ethernet segment." [2]

5.2 PIM-DM Source Distribution Trees

"Sind PIM-DM is a dense-mode protocol, it uses the source distribution or shortest path trees (SPTs) as the sole means of distributing multicast traffic to receivers in the network. These trees are built on the fly, using the flood-and-prune mechanism as soon as a multicast source begins transmitting.

Unlike DVMRP, which uses its own multicast routing table and a special Poison Reverse mechanism to initially construct a minimal spanning distribution tree, PIM-DM uses its PIM neighbor information to construct a similar source distribution tree. In PIM-DM, neighbors are initially considered to be on the SPT, with the incoming interface being the interface in the direction of the source (based on the unicast routing table) and all other PIM-DM neighbors being downstream for this source. This initial form of SPT is referred to as a *Broadcast Tree* because a router sends the multicast traffic to all neighbors in a broadcast-like fashion. (In contrast, DVMRP routers use Truncated Broadcast Trees to initially flood multicast traffic to all downstream routers.)

This is a slight modification of a technique called *Reverse Path Flooding* where incoming traffic that passes the RPF check is flooded out all other interfaces. The difference here is that the flooding occurs only out interfaces where at least one PIM-DM neighbor has been detected or that have a directly connected receiver(s).

Figure 38 shows an example of the initial flooding of multicast traffic in a PIM-DM network down a Broadcast Tree.

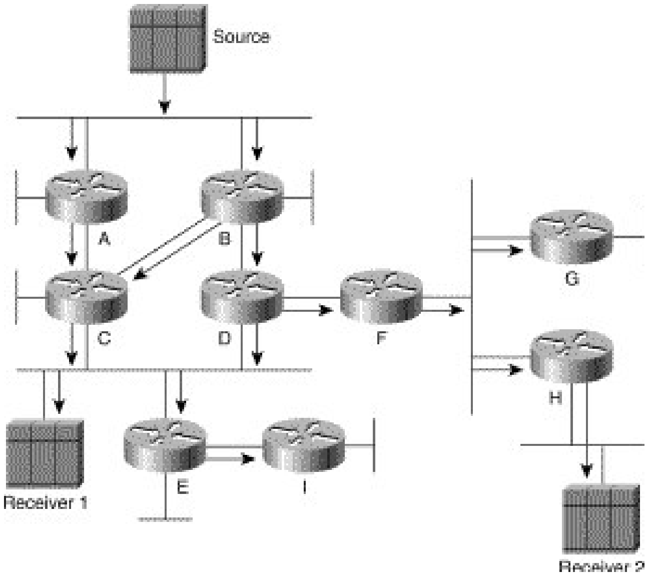


Figure 38: PIM-DM Distribution Tree (Initial Flooding) [2]

Here, you see a multicast source transmitting data that is picked up by Routers A and B and flooded to their downstream PIM-DM neighbors Routers C and D. Keep in mind that you are looking at the *initial* flow of traffic before any pruning takes place because of redundant paths (such as the two incoming paths to Router C) or duplicate forwarders (such as Routers C and D on their common Ethernet).

This tree is trimmed back to a minimal spanning tree of all PIM-DM routers after all pruning has occurred.

As a rule a router can have only ONE incoming interface for any entry in its multicast routing table.

In PIM (either dense mode or sparse mode), when multiple entries exist in the unicast routing table, the entry with the *highest* next-hop IP address is used for the RPF check and hence the incoming interface. For example, consider Router R4 in the network shown in Figure 39.

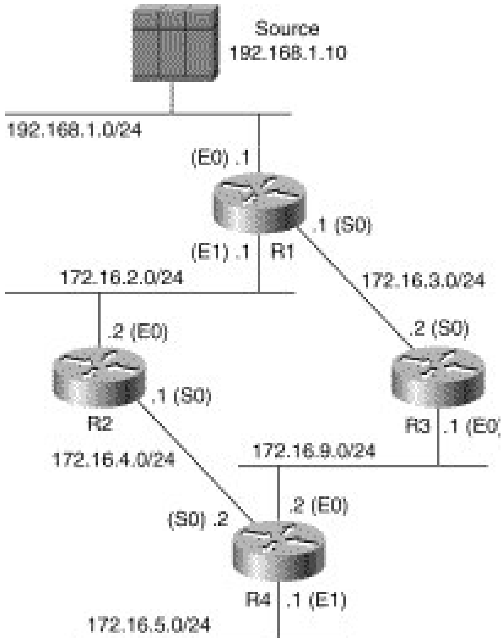


Figure 39: Sample Network [2]

The routing information in Router R4 for the 192.168.1.0 source network is shown in the routing table in the following example.

```
R4>sh ip route
. . .
Gateway of last resort is not set
172.16.0.0/24 is subnetted, 10 subnets
D 172.16.8.0 [90/2195456] via 172.16.4.1, 20:02:31, Serial1
C 172.16.9.0 is directly connected, Ethernet0
D 172.16.10.0 [90/2195456] via 172.16.9.1, 20:02:31, Ethernet0
C 172.16.4.0 is directly connected, Serial1
C 172.16.5.0 is directly connected, Ethernet1
D 172.16.6.0 [90/2707456] via 172.16.4.1, 20:02:31, Serial1
D 172.16.7.0 [90/2195456] via 172.16.4.1, 20:02:31, Serial1
D 172.16.1.0 [90/2707456] via 172.16.9.1, 00:00:13, Ethernet0
D 172.16.2.0 [90/2221056] via 172.16.9.1, 00:00:13, Ethernet0
D 172.16.3.0 [90/2195456] via 172.16.9.1, 00:00:18, Ethernet0
D   192.168.1.0/24   [90/2733056]   via   172.16.4.1,   00:00:14,   Serial1
[90/2733056] via 172.16.9.1, 00:00:14, Ethernet0
D 192.168.100.0/24 [90/2835456] via 172.16.4.1, 00:00:14,
Serial1 [90/2835456] via 172.16.9.1, 00:00:14, Ethernet0
```

Notice that there are two equal cost paths in the unicast routing table for source network 192.168.1.0. However, using the rule regarding only having one incoming interface for multicast, the path **via 172.16.9.1, 00:00:14, Ethernet0** is used as the RPF or incoming interface. You can confirm this outcome by using the **show ip rpf** command, which produces the output shown.

```
R4>sh ip rpf 192.168.1.10
RPF information for? (192.168.1.10)
RPF interface: Ethernet0
RPF neighbor: R3 (172.16.9.1)
RPF route/mask:
192.168.1.0/255.255.255.0
RPF type: unicast
```

Here, the RPF information for multicast source 192.168.1.10 is via interface Ethernet0 through the RPF neighbor R3 (172.16.9.1)." [2]

5.3 PIM-DM Asserts

One of the issues that need to be addressed with regard to attaining the optimal network with traffic flowing only to the necessary segments of the network (between the initial flow and the final stage after the segments with no active receivers are pruned) is the duplicate traffic being sent out to the same Ethernet segment by multiple routers connected to the same segment. "To address this issue and shut off all but one flow of multicast to a network, PIM uses an Assert mechanism to elect a "forwarder" for a particular multicast source. The Assert mechanism is triggered by the following rule.

If a router receives a multicast packet via an interface in the outgoing interface list associated with a multicast source, send a PIM Assert message out the interface to resolve which router will continue forwarding this traffic.

When the Assert mechanism is triggered on an interface, a PIM router sends a PIM Assert message containing its metric to the source. All PIM routers on the network examine the metric in the PIM Assert message to determine which router has the best metric back to the multicast source. The router with the best metric continues to forward traffic from this source onto the network while all other PIM routers prune their interface. If there is a tie in the metrics, the router addresses are used as the tiebreaker and the highest IP address wins." [2]

Figure 40 shows the PIM Assert mechanism in action.

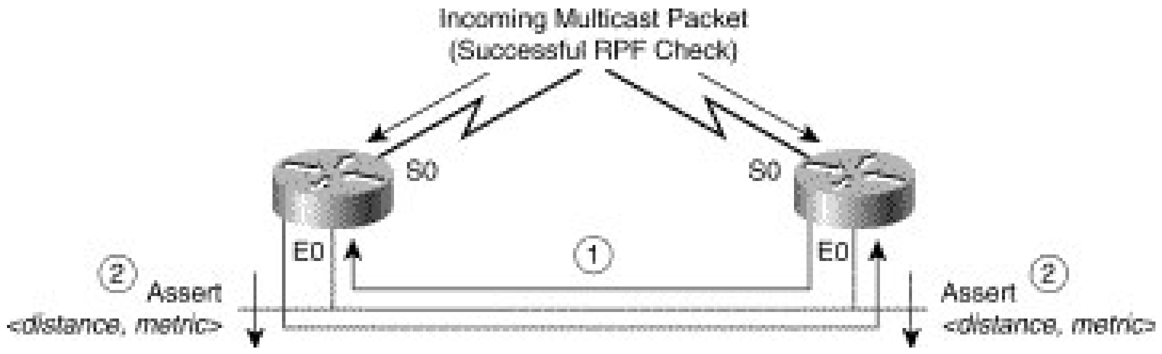


Figure 40: PIM Assert Mechanism [2]

The PIM Assert mechanism shown in Figure 40 is described in the following two steps:

Step 1 Both routers start to receive traffic from same source on their respective S0 interface and starts forwarding it out through the E0 onto the common Ethernet network. These packets are received by the other routers on the Ethernet segment via an interface (in this case, Ethernet0) that is on the routers outgoing interface list. Once it is found that a packet has been received on an interface which is supposed to be an outgoing interface and not an incoming interface, it triggers both routers to send PIM Assert messages to stop this from happening and to determine who should be the forwarder.

"Step 2 The routers send and receive PIM Assert messages that contain the administrative distance (used as the high-order portion of the compared value) and the routing protocol metric for the source (used as the low-order portion of the compared value). The values in the PIM Assert messages are compared and the lowest value (that is, has the best metric to the source when both administrative distance and route metric are considered) wins the Assert battle. The loser(s) stop sending the source traffic onto the network by pruning their interface(s) for this source traffic." [2]

So, going back to the sample network in Figure 41(a), Routers C and D will be sending PIM Assert messages out on the Ethernet interface since they have received a multicast packet from the source, on an interface that is in the outgoing interface list for its multicast traffic. If we assume that both routers C and D have the same metrics and C has a higher IP address, C wins the Assert battle due to the higher IP address and continues to forward traffic onto the Ethernet while Router D prunes its interface, as shown in Figure 41(b)

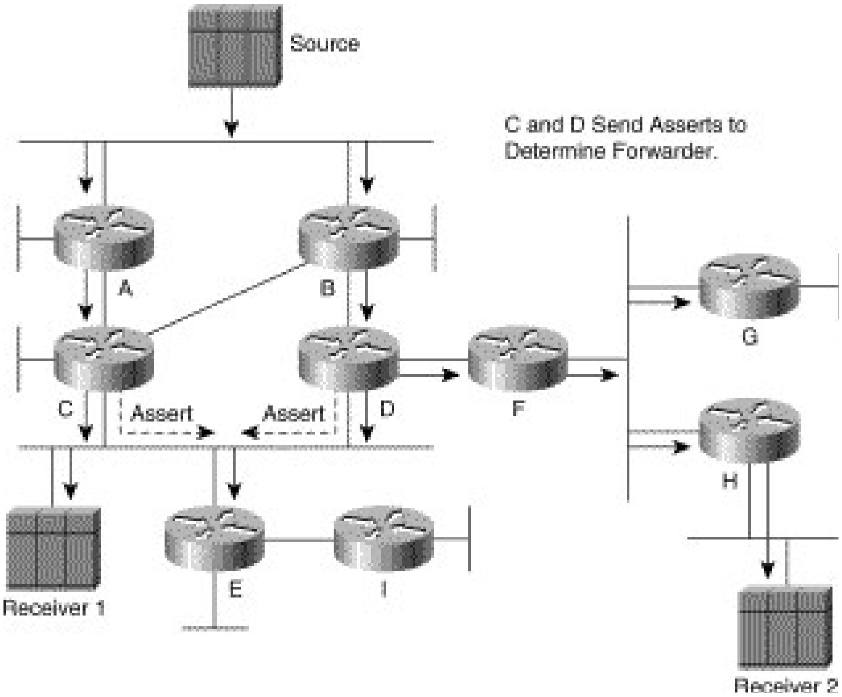


Figure 41a: PIM Assert Example [2]

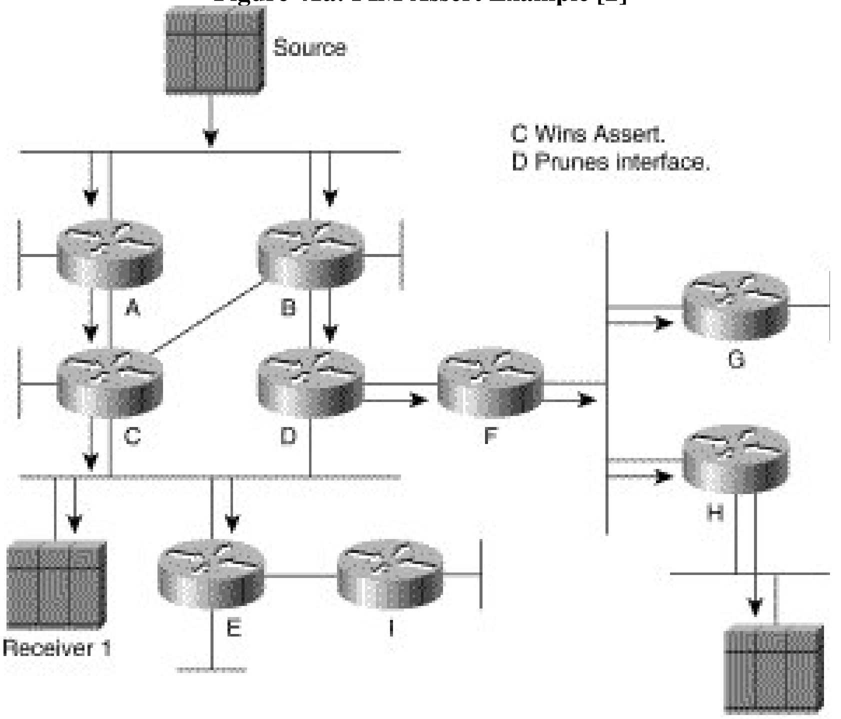


Figure 41b: PIM Network---After Assert [2]

5.4 PIM-DM Scalability

PIM-DM is dependent on the routing information available from the underlying unicast network. So a well designed unicast network can make PIM-DM capable of scaling much better than DVMRP networks. One of the parameters that would make the network designers hesitant about using PIM-DM would be its flood-and-prune behavior that could cause unwanted traffic being sent across the network domain. A proposal is being made to do a *State-Refresh* extension to the PIM specification to try preventing the prune state from timing out and hence avoiding the periodic flooding of unwanted traffic.

5.5 Summary

As mentioned above, the biggest impediment for the network designers to embrace PIM-DM as a chosen protocol for a multicast network is the periodic flooding of the network with the flood-prune traffic that could be periodic in the network based on the expiry of the hold timer expiry. This obstacle could be considered insignificant in high-speed networks where the bandwidth consumed by periodic flooding can be neglected. "Using PIM-DM for general-purpose multicast on networks that employ medium- to low-speed WAN links is generally not desirable. Even when the State-Refresh extension becomes a reality, PIM-DM will still create (S, G) state in every router in the network. In networks that have large numbers of active sources and groups, the amount of multicast routing state that must be maintained in the routers in the network can become an issue. Furthermore, networks that have large numbers of sources, such as financial networks, or in which RTP-based multimedia applications are used, tend to suffer from bursty source problems.

Most of the sources in these networks tend to send a single multicast packet every few minutes. The State-Refresh enhancement will not help in these scenarios because the (S, G) state in the first-hop routers will time out if the time interval between packets sent by the source is greater than 3 minutes. Because of these factors, most PIM multicast network engineers are opting for PIM sparse mode, which requires less state in the routers and does not suffer from a flood-and-prune behavior." [2]

6 PIM Sparse Mode

Just like in PIM dense mode, Protocol Independent Multicast sparse mode (PIM-SM) uses the contents from the unicast routing table, irrespective of how it is populated, weather it uses dynamic routing or static, to perform the Reverse Path Forwarding (RPF) check function; hence, it too is protocol independent.

"Some key characteristics of PIM-SM are

1) Protocol independent (uses unicast route table for RPF check)
2) No separate multicast routing protocol
3) Explicit Join behavior
4) Classless (as long as classless unicast routing is in use)

This chapter provides an overview of the basic mechanisms used by PIM-SM, which include Explicit Join model, Shared trees, Shortest path trees (SPT), Source registration, Designated Router (DR), SPT switchover, State-Refresh, Rendezvous point (RP) discovery." [2]

In addition, some of the mechanisms like PIM Neighbor Discovery and PIM Asserts that are used in PIM-DM are also used by PIM-SM. These are already explained in the earlier section and hence not repeated here.

6.1 Explicit Join Model

The very idea of Explicit Join is what gives PIM-SM the edge of maintaining a sparse mode nature. It means that multicast traffic is only sent to locations of the network that has active recipients. This is made possible by the use of PIM Joins, "which are sent hop by hop toward the root node of tree. The root node of a tree in PIM-SM is the RP in the case of a shared tree or the first-hop router that is directly connected to the multicast source in the case of a SPT. As this Join travels up the tree, routers along the path set up multicast forwarding state so that the requested multicast traffic will be forwarded back down the tree. Likewise, when multicast traffic is no longer needed, a router sends a PIM Prune up the tree toward the root node to prune off the unnecessary traffic. As this PIM Prune travels hop by hop up the tree, each router updates its forwarding state appropriately. This update often results in the deletion of the forwarding state associated with a multicast group or source.

The key point here is that in the Explicit Join model, forwarding state in the routers is set up as a result of these Joins. This is a substantial departure from flood-and-prune protocols such as PIM-DM where router forwarding state is set up by the arrival of multicast data." [2]

6.2 PIM-SM Shared Trees

PIM-SM operates around a single, *unidirectional* shared tree whose root node is called the rendezvous point (RP), hence sometimes known as RP trees. Shared trees or RP trees are frequently known as RPTs to avoid confusion with source trees also known as shortest path trees having the acronym SPTs.

The routers that have a directly connected receiver for the multicast group (also known as Last-hop routers) that need to receive the traffic from a specific multicast group will join this shared tree, while the last-hop router that no longer needs the traffic of a specific multicast group router prunes itself from the shared tree. PIM-SM uses a unidirectional shared tree which means that once the tree is formed with the root at RP, there cannot be traffic flowing from the source (which could be a host connected to a router in the network and hence a downstream device with respect to the RP) to the RP. This would mean that the sources will have to get the multicast traffic by some other means for the RP to send the traffic down the tree. This is done by registering the source with the RP and the "registration process actually triggers an SPT Join by the RP toward the Source when there are active receivers for the group in the network." [2]

6.2.1 Shared Tree Joins

"Figure 42 shows the first step of a Shared Tree Join in a sample PIM-SM. In this step, a single host (Receiver 1) has just joined multicast Group G via an Internet Group Membership Protocol (IGMP) Membership Report.

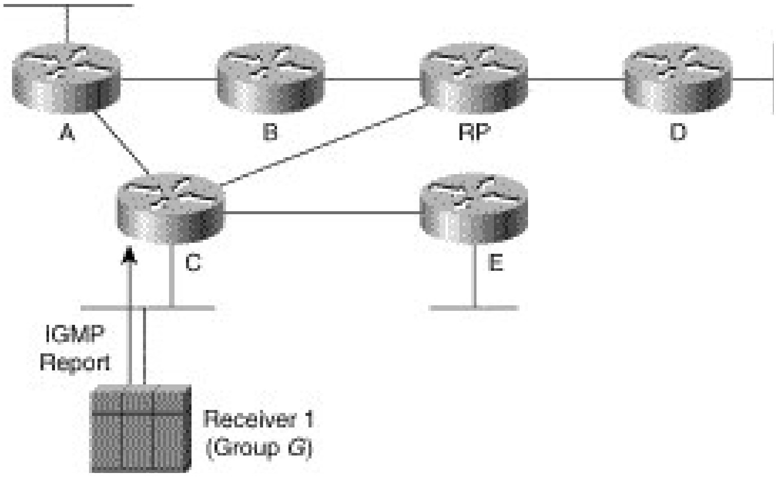


Figure 42: PIM Shared Tree Joins---Step 1 [2]

Because Receiver 1 is the first host to join the multicast group in the example, Router C had to create a (*, G) state entry in its multicast routing table for this multicast group. Router C then places the Ethernet interface in the outgoing interface list of the (*, G) entry as shown by the solid arrow in Figure 43.

Because Router C had to create a new (*, G) state entry, it must also send a PIM (*, G) Join (indicated by the dashed arrow in Figure 43) toward the RP to join the shared tree. (Router C uses its unicast routing table to determine the interface toward the RP.)

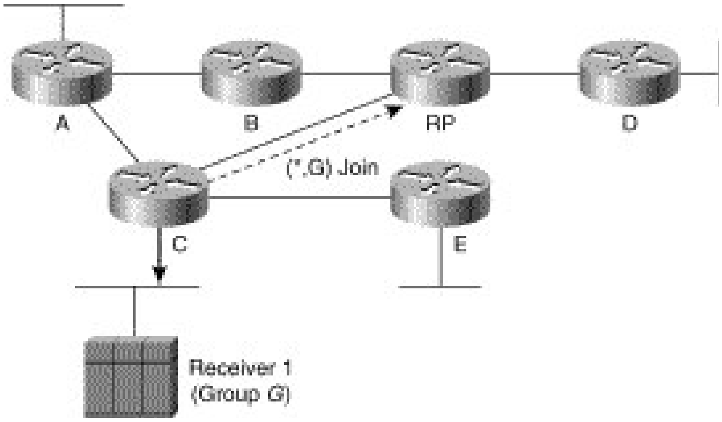


Figure 43: PIM Shared Tree Join---Step 2 [2]

The RP receives the (*, G) Join, and because it too had no previous state for multicast Group G, creates a (*, G) state entry in its multicast routing table and adds the link to Router C to the outgoing interface list. At this point, a shared tree for multicast Group G has been constructed from the RP to Router C and Receiver 1, as shown by the solid arrows in Figure 44. Now, any traffic for multicast Group G that reaches the RP can flow down the shared tree to Receiver 1.

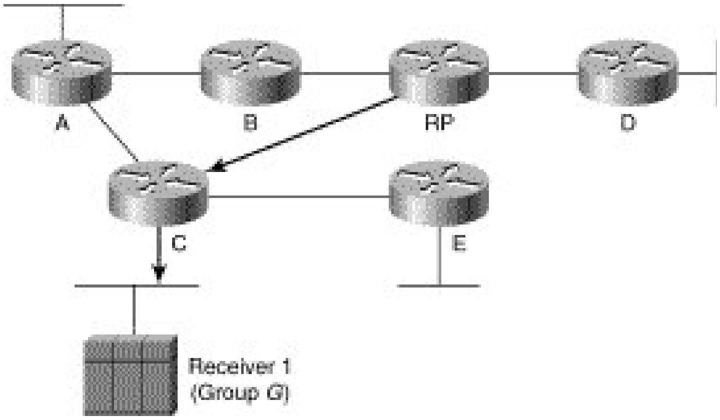


Figure 44: Shared Tree Joins---Step 3 [2]

Let's continue the example and assume that another host (Receiver 2) joins the multicast group as shown in Figure 45. Again, the host has signaled its desire to join multicast Group G by using an IGMP Membership Report that was received by Router E.

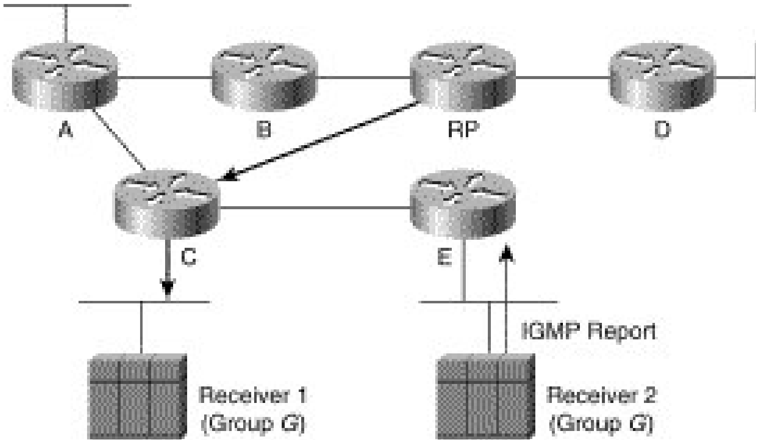


Figure 45: Shared Tree Joins---Step 4 [2]

Because Router E didn't have any state for multicast Group G, it creates a (*, G) state entry in its multicast routing table and adds the Ethernet interface to its outgoing interface list (shown by the solid arrow in Figure 46). Because Router E had to create a new (*, G) entry, it sends a (*, G) Join (indicated by the dashed arrow in Figure 46) toward the RP to join the shared tree for Group G.

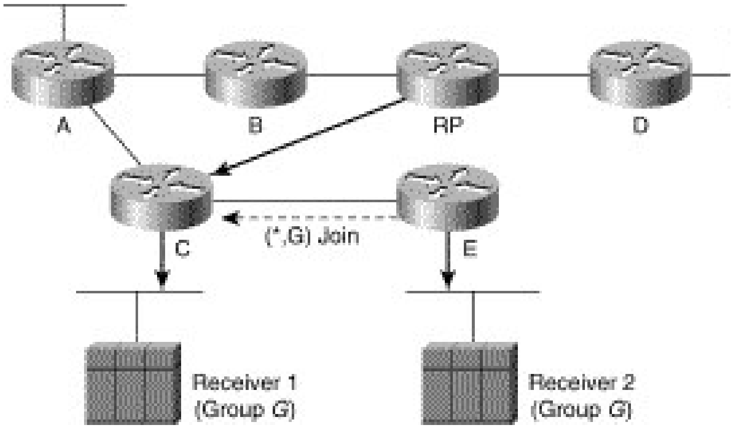


Figure 46: Shared Tree Joins---Step 5 [2]

When Router C receives the (*, G) Join from Router E, it finds that it already has (*, G) state for Group G (that is, it's already on the shared tree for this group). As a result, Router C simply adds the link to Router E to the outgoing interface list in it's (*, G) entry.

Figure 47 shows the resulting shared tree (indicated by the solid arrows) that includes Routers C and E along with their directly connected hosts (Receiver 1 and Receiver 2)." [2]

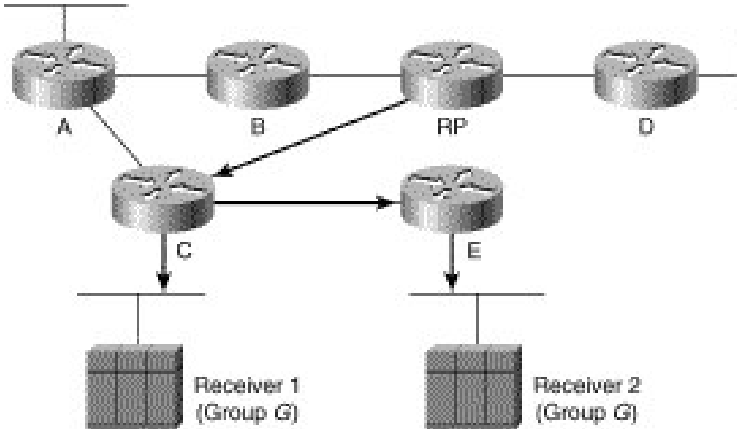


Figure 47: Shared Tree Joins---Step 6 [2]

6.2.2 Shared Tree Prunes

Just like PIM-SM uses the Explicit Join model to build distribution trees as and when needed, it also uses Prunes to tear down the trees when they are no longer needed, rather than just allowing the branches to time out due to the absence of Join messages which would not be the most efficient use of network resources.

"For example, assume that Receiver 2 leaves multicast Group G by sending an IGMP Leave message (shown in Figure 48).

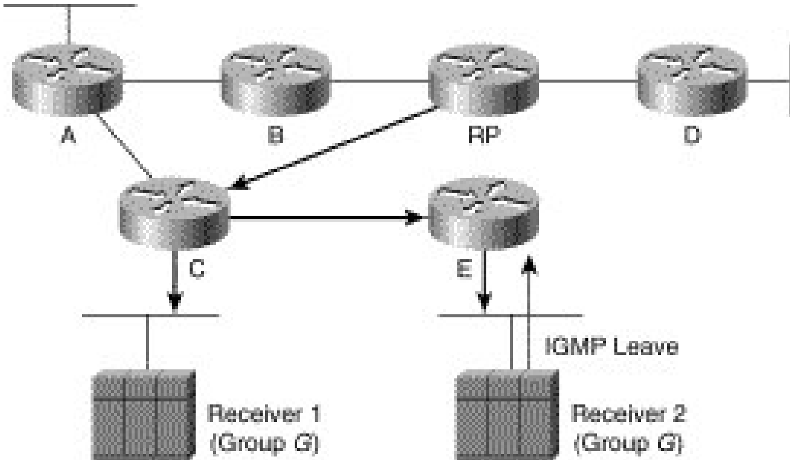


Figure 48: Shared Tree Prunes---Step 1 [2]

Because Receiver 2 was the only host joined to the group on Router E's Ethernet interface, this interface is removed from the outgoing interface list in it's (*, G) entry. (This is represented by the removal of the arrow on Router E's Ethernet in Figure 49.) When the interface is removed, the outgoing interface list for this (*, G) entry is null (empty), which indicates that Router E no longer needs the traffic for this group. Router E responds to the fact that its outgoing interface list is now null by sending a *(*, G) Prune* (shown in Figure 49) toward the RP to prune itself off the shared tree.

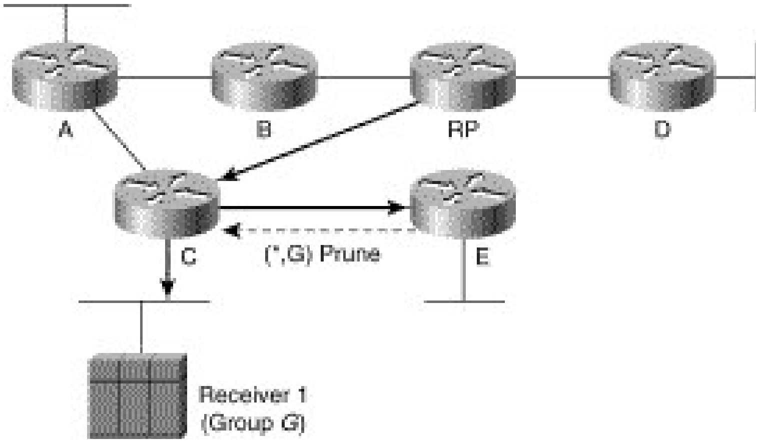


Figure 49: Shared Tree Prunes---Step 2 [2]

When Router C receives this Prune, it removes the link to Router E from the outgoing interface list of the (*, G) entry (as indicated by the removal of the arrow between Router C and Router E in Figure 50).

However, because Router C still has a directly connected host for the group (Receiver 1), the outgoing interface list for the (*, G) entry is not null (empty). Therefore, Router C must remain on the shared tree, and a Prune is not sent up the shared tree toward the RP.

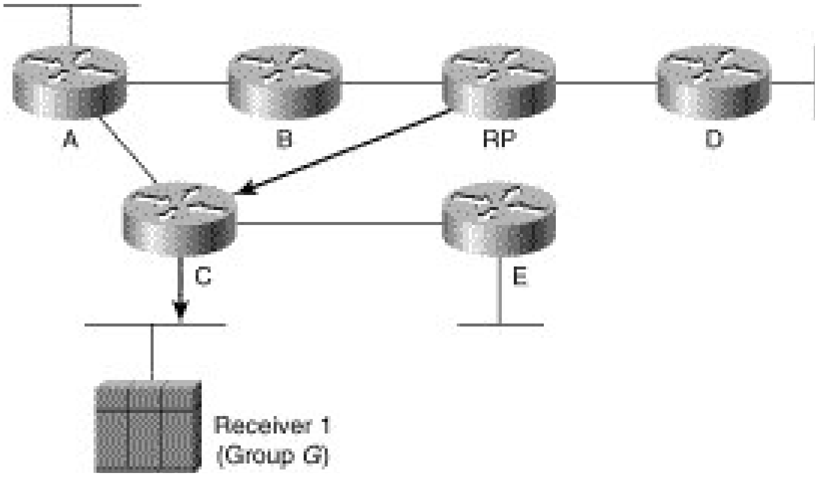


Figure 50: Shared Tree Prunes---Step 3 [2]

The prune examples in this section did not cover the situation in which a (*, G) prune is being sent on a multi-access network with several other PIM-SM routers still joined to the same shared tree. However, a Prune Override mechanism is used in PIM-SM by the other routers on the network to prevent the shared tree from being pruned prematurely.” [2]

6.3 PIM-SM Shortest Path Trees

An advantage that comes with PIM-SM is that “unlike other sparse mode protocols (such as core based trees), it doesn't limit us to receiving multicast traffic only via the shared tree. Just like using the Explicit Join mechanism to join the shared tree with the root at the RP, this mechanism can be used to join the SPT with the root as a particular source. The advantage is that by doing this the multicast traffic can now be routed directly to the receivers without having to go through the RP, thereby reducing network latency and possible congestion at the RP. On the flip side, the disadvantage is that routers must create and maintain (S, G) state entries in their multicast routing tables along the (S, G) SPT which could be consuming more router resources.

Even with all that additional states to be maintained, the overall amount of (S, G) information maintained by the routers in a PIM-SM network that uses SPTs is generally much less than is necessary for dense mode protocols.” [2] This is because of the fact that the Flood-and-Prune mechanism will require all the routers in the network having to maintain (S, G) state entries in their routing tables for all the active sources in the network, even if there are no active listeners for the groups to which the sources are transmitting the multicast traffic. “By joining SPTs in PIM-SM, we gain the advantage of an optimal distribution tree without suffering from the overhead and inefficiencies associated with other dense mode protocols such as PIM-DM, DVMRP, and MOSPF. These are some of the reasons why PIM-SM is gaining popularity over the dense mode protocols.

The reason why we cannot have the routers join the SPT directly before having to join the RPT is that, without having received a few packets from shared tree with the root at RP, the routers will not have a method of knowing that there is an active source or even more importantly the wear about of the source router.

The basic mechanism of joining the SPT is explained in the following section. Later on we discuss the why and when the switch happens.

6.3.1 Shortest Path Tree Joins

Usually in PIM-SM operation, the routers send a (*, G) Join to the RP to join the shared tree to receive traffic from Group G. But, by sending a (S, G) Join towards an active source S the router can become part of an SPT for the source and receive traffic sent by S to the specific group G.

“Figure 51 shows an example of an (S, G) Join being sent toward an active source to join the SPT. (The solid arrows in the drawing indicate the path of the SPT down which traffic from Source S1 flows.) In this example, Receiver 1 has already joined Group G (indicated by the solid arrow from Router E).

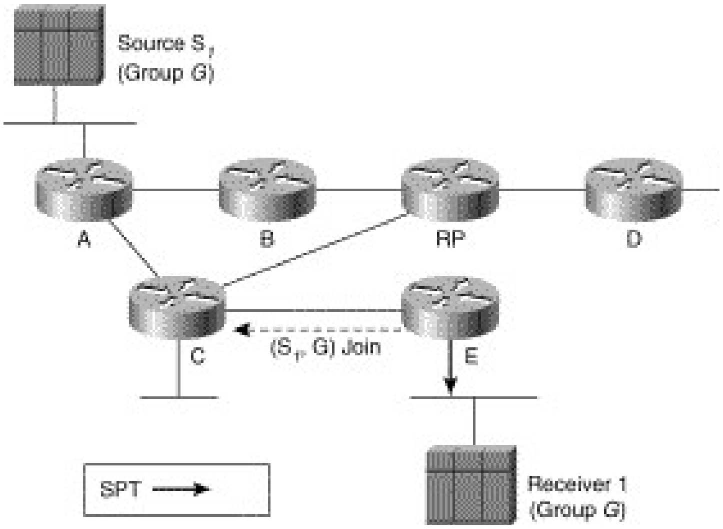


Figure 51: SPT Join---Step 1 [2]

Router E would have learned that Source S1 is active because it would have received a packet from the source via the shared tree. Because Router E wants to join the SPT for Source S1, it sends an (S1, G) Join toward the source. Router E determines the correct interface to send this Join out by calculating the RPF interface toward Source S1 using its unicast routing table. This will show that Router C is the next-hop router to Source S1.

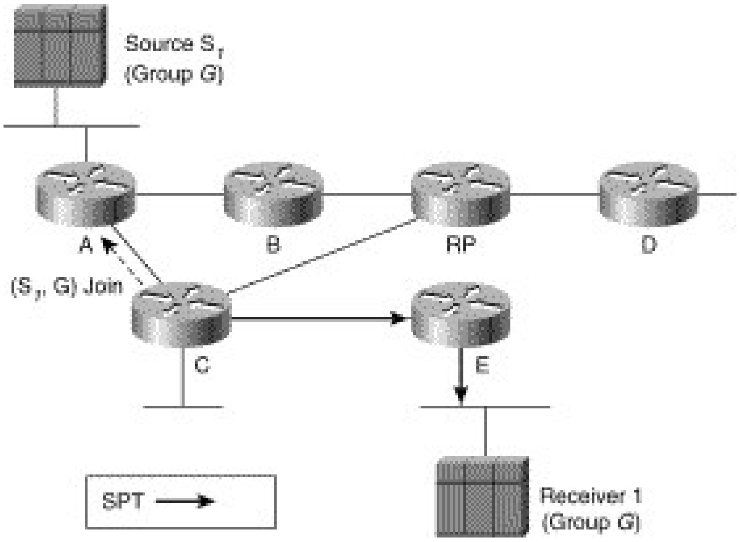


Figure 52: SPT Join---Step 2 [2]

When Router C receives the (S1, G) Join, it creates an (S1, G) entry in its multicast forwarding table and adds the interface on which the Join was received to the entry's outgoing interface list (indicated by the solid arrow from Router C to Router E in Figure 52). Because Router C had to create state for (S1, G), it also sends an (S1, G) Join (as shown by the dashed arrow in Figure 52) toward the source.

Finally, when Router A receives the (S1, G) Join, it adds the link to Router C to the outgoing interface list of its existing (S1, G) entry as shown by the solid arrow in Figure 53. Router A is referred to as the first hop router for Source S1 and would have already created an (S1, G) entry as soon as it received the first packet from the source." [2]

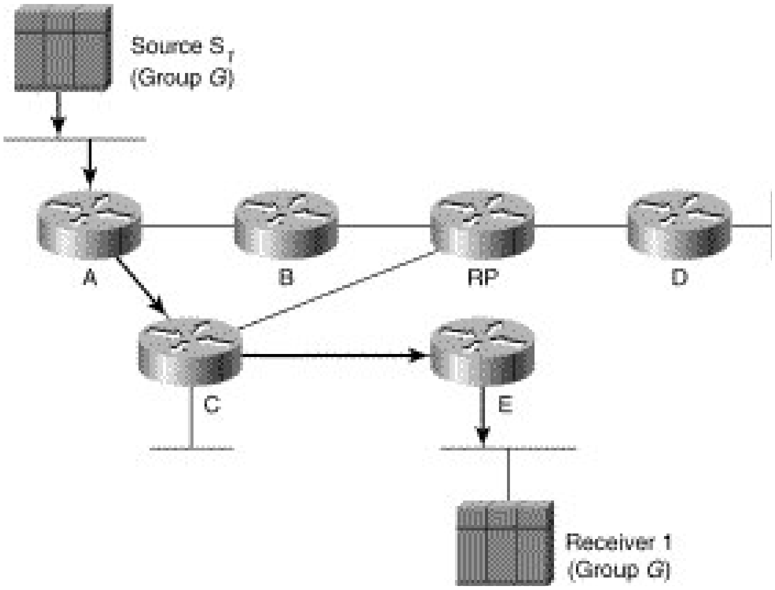


Figure 53: SPT Join---Step 3 [2]

6.3.2 Shortest Path Tree Prunes

SPTs can be pruned by using (S1, G) Prunes in the same manner that shared trees were pruned by using (*, G) Prunes. Still, there could be situations where an (S, G) prune is being sent on a multi-access network (such as an Ethernet segment) with several other PIM-SM routers still joined to the same SPT. The Prune Override mechanism is used in PIM-SM by the other routers on the network to prevent the SPT from being pruned prematurely.

The following three figures shows the same network example used for PIM Joins and shows the steps followed in pruning the states from the network once there are no more active receivers in that segment of the network. Since the Receiver stopped receiving the multicast traffic from Router E, there is no more solid line between Router E and receiver. Further, the

Router E no longer needs a (S1, G) state since there are no more entries in its outgoing interface list. This triggers Router E to send a (S1, G) Prune toward Source S1 (shown by the dashed arrow in Figure 54).

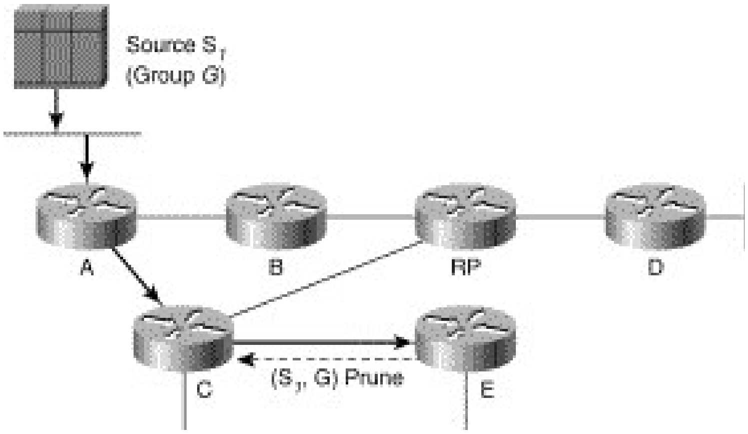


Figure 54: SPT Prunes---Step 1 [2]

Similar steps are taken by Router C once it receives the Prune for (S1, G) from Router E. Router C removes the interface on which the message was received from the outgoing interface list of its (S1, G) entry (indicated by the absence of the solid arrow between Routers C and E in Figure 55). This will lead to Router C having no entries in its outgoing interface list for (S1, G) and hence sending an (S1, G) Prune toward the source S1 (as shown by the dashed arrow in Figure 55).

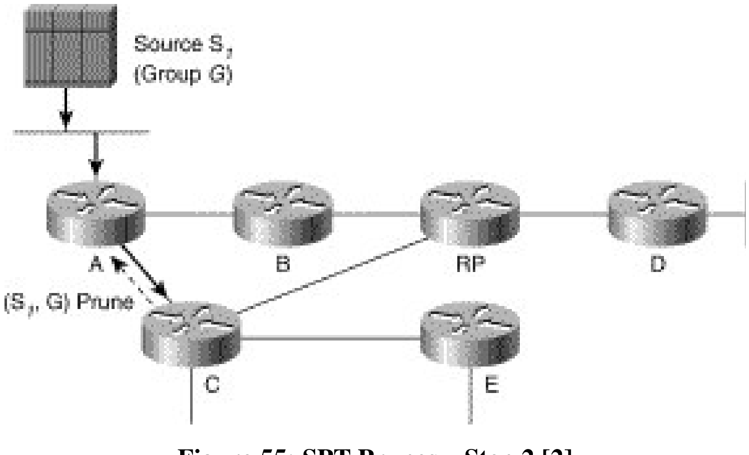


Figure 55: SPT Prunes---Step 2 [2]

Router A on receiving the (S1, G) Prune from Router C takes the same steps as removing the interface on which the Prune was received from the outgoing interface list for (S1, G) (No solid line between Router A and Router C in Figure 56). “However, because Router A is the first-hop router for Source S1 (in other words, it is directly connected to the source), no further action is taken. Router A continues to drop any packets from Source S1 because the outgoing interface list in Router A's (S1, G) entry is empty." [2]

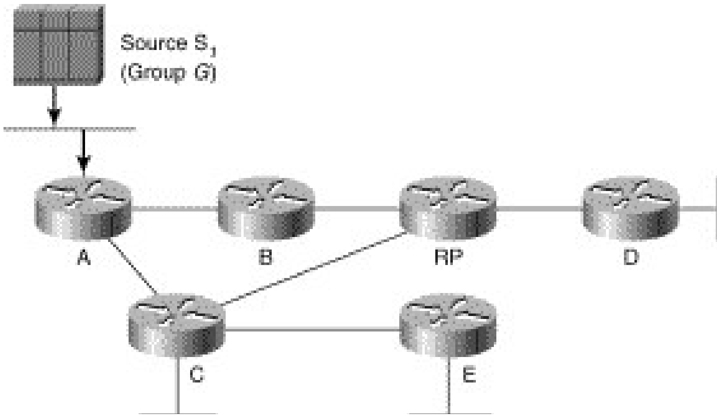


Figure 56: SPT Prunes---Step 3 [2]

The key point here is that the PIM-SM explicit Join/Prune mechanism can be used to Join/Prune SPTs as well as RPTs/shared trees. This capability becomes important in later sections on source registering and SPT switchover.

6.4 PIM Join/Prune Messages

Even though in the examples above, for the sake of explanation, we were referring to PIM Prunes and Joins in different contexts, there is only a single PIM Join/Prune message type. Each PIM Join/Prune message contains both a Join list and a Prune list, either one of which may be empty, depending on the information being conveyed up the distribution tree. This facilitates the chances of improving the efficiency of the periodic refresh mechanism, by providing the ability to include multiple entries in Join and/or Prune lists there by making it possible for the router to allow multiple sources and/or groups to Join and/or Prune with a single message.

"The entries in the Join and Prune lists of PIM Join/Prune messages share a common format, containing (among other things) the following information:

- **Multicast source address**---IP address of the multicast source to Join/Prune. (If the Wildcard flag is set, this is the address of the RP.)
- **Multicast group address**---Class D multicast group address to Join/Prune.
- **WC bit (Wildcard flag)** ---This entry is a shared tree (*, G) Join/Prune message.
- **RP bit (RP Tree flag)** ---This Join/Prune information is applicable to and should be forwarded up the shared tree." [2]

The above mentioned information in the Join/Prune lists can be varied to send different requests to the upstream routers.

“For example, a PIM Join/Prune message with an entry in the *Join list* of

Source address = 192.16.10.1

Group address = 224.1.1.1

Flags = WC, RP indicates that this item is a (*, G) (denoted by the WC and RP flags being set) Join for Group 224.1.1.1 whose RP is 192.16.10.1.

A PIM Join/Prune message with an entry in the *Prune list* of

Source address = 191.1.2.1

Group address = 239.255.1.1

Flags = none indicates that this is an (S, G) (denoted by the WC and RP flags being clear) Prune for source 191.1.2.1, Group 239.255.1.1.” [2]

These parameters have a huge significance in the situation where the traffic has to switch from a shared tree to an SPT as and when required as part of traffic flow optimization, which will be discussed in the later sections of the chapter.

6.5 PIM-SM State-Refresh

In order to prevent the situation of a PIM-SM forwarding state being stuck on a router, even when there are no active receivers downstream or due to the upstream router missing a Prune message due to congestion in the network, they are given a life time of 3 minutes. This lifetime is established by associating an expiration timer with each (*, G) and (S, G) state entry in the multicast routing table. If a new PIM Join/Prune message, a.k.a State-Refresh message in this context is not received before the life time of 3 minutes expires, the state entry is deleted from the router. To avoid this eventuality, routers send PIM Join/Prune messages to the appropriate upstream neighbor once a minute and once it is received; the upstream router resets the expiration timer and refreshes its existing multicast forwarding state. Shared tree refreshes (which are (*, G) Joins) are periodically send to the upstream neighbor in the direction of the RP and SPT refreshes (which are (S, G) Joins) are send to the upstream neighbor in the direction of the source as long as they have a nonempty outgoing interface list in their associated (*, G) and (S, G) entries (or a directly connected host for multicast Group G).

6.6 Source Registration

We learned that routers use (*, G) Joins to join the shared tree for a multicast group G. But, since PIM-SM is a unidirectional tree, multicast traffic can only flow down the root of the tree which is the RP. So the flow of traffic from the Source to RP cannot be part of the shared tree traffic. Hence, the multicast sources should somehow get their traffic to the RP so that the traffic can flow down the shared tree. In PIM-SM this is achieved by having the RP form an SPT back to the source so it can receive the source's traffic. Before this SPT can be formed, "the RP need to be notified that the source exists. PIM-SM makes use of PIM Register and Register-Stop messages to implement a source registration process to accomplish this task. A common misconception is that a source must register before any receivers can join the shared tree. However, receivers can join the shared tree, even though there are currently no active sources. But when a source does become active, the RP then joins the SPT to the source and begins forwarding this traffic down the shared tree. By the same token, sources can register first even if there are no active receivers in the network. Later, when a receiver does join the group, the RP Joins the SPT toward all sources in the group and begins forwarding the group traffic down the shared tree.

The following sections discuss the mechanics of the source registration process that makes use of *PIM Register* and *PIM Register-Stop* messages. This process notifies an RP of an active source in the network and delivers the initial multicast packets to the RP to be forwarded down the shared tree.

6.6.1 PIM Register Messages

PIM Register messages are sent by first-hop DRs (that is, a DR directly connected to a multicast source) to the RP. The purpose of the PIM Register message is twofold:

1) Notify the RP that Source S1 is actively sending to Group G.

2) Deliver the initial multicast packet(s) sent by Source S1 (each encapsulated inside a single PIM Register message) to the RP for delivery down the shared tree."

To explain the process we are referring back to Figure 47 which has a shared tree formed with an RP and two receivers. "When a multicast source begins to transmit, (Figure 57, as shown by the solid line from the Source to the Router A) the DR receives the multicast packets sent by the source and creates an (S, G) state entry in its multicast routing table.

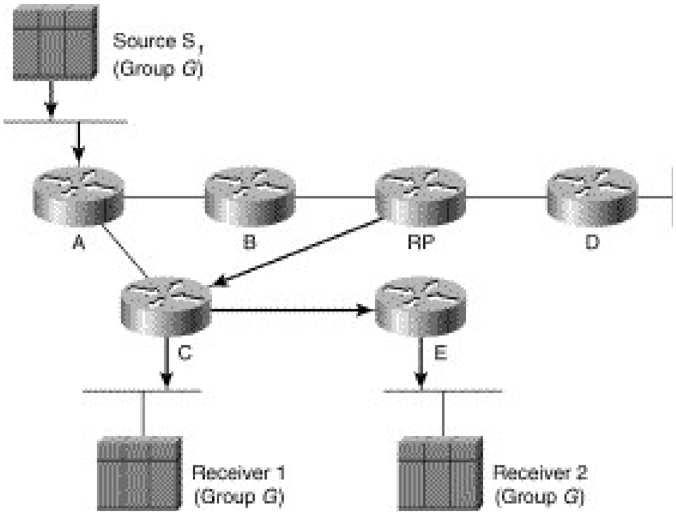


Figure 57: Source Registration---Step 1 [2]

In addition, because the source is directly connected (to the DR), the DR encapsulates each multicast packet in a separate PIM Register message and unicast it to the RP (Figure 58). How the DR learns the IP address of the RP is discussed in the "RP Discovery" section.

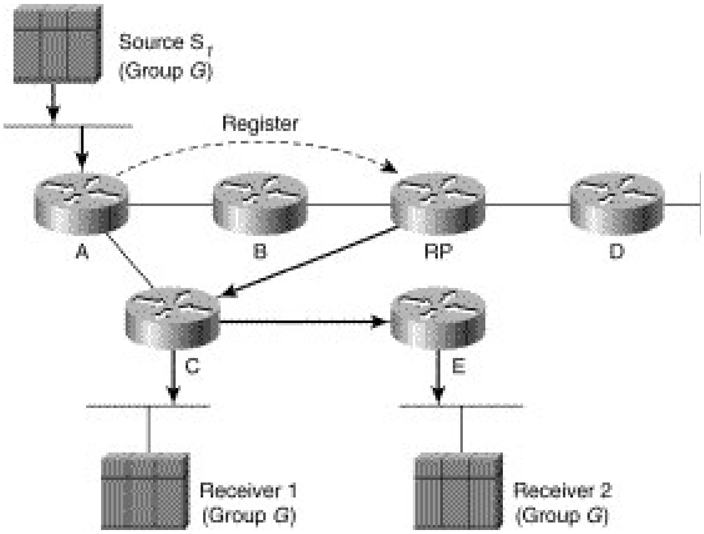


Figure 58: Source Registration---Step 2 [2]

Unlike the other PIM messages that are multicast on a local segment and travel hop by hop through the network, PIM Register messages and PIM Register-Stop messages are unicast between the first-hop router and the RP.

When an RP receives a PIM Register message, it first de-encapsulates the message so it can examine the multicast packet inside. If the packet is for an active multicast group (that is, shared tree Joins for the group have been received), the RP forwards the packet down the shared tree. The RP then joins the SPT for Source S1 so that it can receive (S1, G) traffic natively instead of it being sent encapsulated inside of PIM Register messages. If, on the other hand, there is no active shared tree for the group, the RP simply discards the multicast packets and does not send a Join toward the source. (Figure 59)

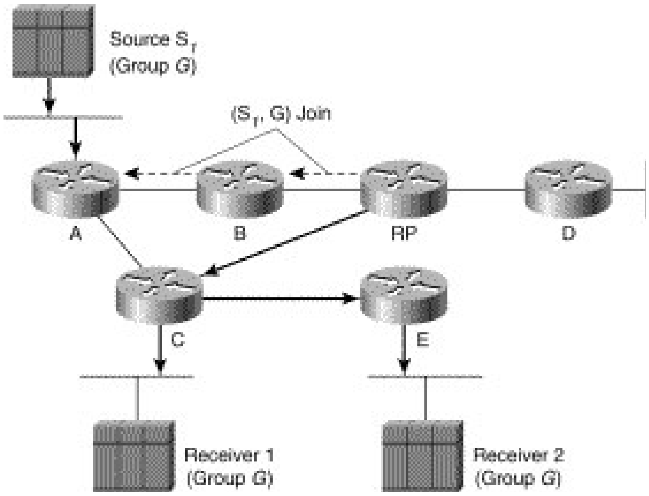


Figure 59: Source Registration---Step 3 [2]

6.6.2 PIM Register-Stop Messages

The RP unicasts PIM Register-Stop messages to the first-hop DR, instructing it to stop sending (S1, G) Register messages under any of the following conditions:

1) When the RP begins receiving multicast traffic from Source S1 via the (S1, G) SPT between the source and the RP.
2) If the RP has no need for the traffic because there is no active shared tree for the group.

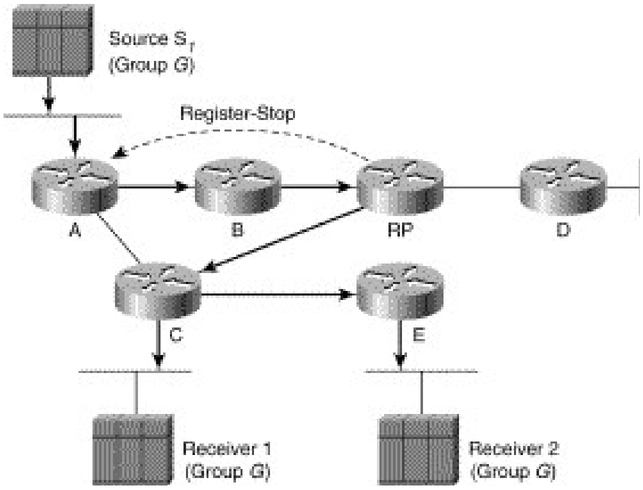


Figure 60: Source Registration---Step 4 [2]

When a first-hop DR receives a Register-Stop message, the router knows that the RP has received the Register message and one of the two conditions above has been met. In either case, the first-hop DR terminates the Register process and stops encapsulating (S1, G) packets in PIM Register messages." [2]

Let us extend the same example by introducing another multicast source, S2, to the network with Router D as the first-hop DR sending multicast traffic to the same group G. The PIM registration and register-stop process will happen in the same manner as explained above, causing the RP to join an (S2, G) SPT to get the traffic for (S2, G) till the RP so as to be forwarded down the shared tree for Group G. "At this point, the RP has joined both the (S1, G) and (S2, G) SPTs, (as seen in Figure 61) for the two active sources (S1 and S2) for Group G. This traffic is being forwarded down the (*, G) shared tree to Receivers 1 and 2. Now, the paths are complete between the sources and the receivers, and multicast traffic is flowing properly." [2]

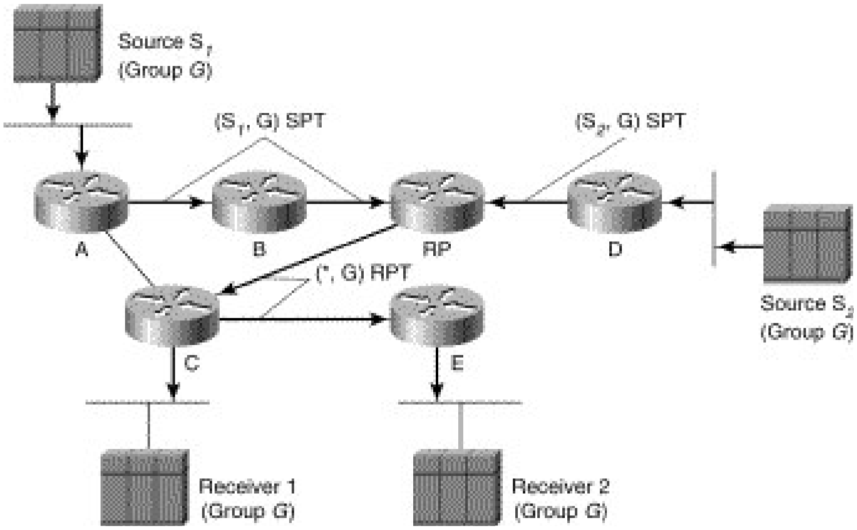


Figure 61: Source Registration---Step 5 [2]

6.7 Shortest Path Tree Switchover

PIM-SM gives the ability "to a last hop DR (a DR with directly connected hosts that have joined a multicast group) to switch from the shared tree to the SPT for a specific source. This step is usually accomplished by specifying an ***SPT-Threshold*** in terms of bandwidth. If this threshold is exceeded, the last-hop DR joins the SPT. This threshold is set to zero by default on Cisco routers, which means that the SPT is joined as soon the first multicast packet from a source has been received via the shared tree." [2]

Let us go back to the example in Figure 61, where there are two receivers and two active sources. "Since Router C is a last-hop DR, it has the option of switching to the SPT's for

Source S1 and Source S2. Let us see how the switch happens for the traffic from Source S1. To accomplish this, Router C would send an (S1, G) Join toward Source S1, as shown by the dashed arrow in Figure 62.

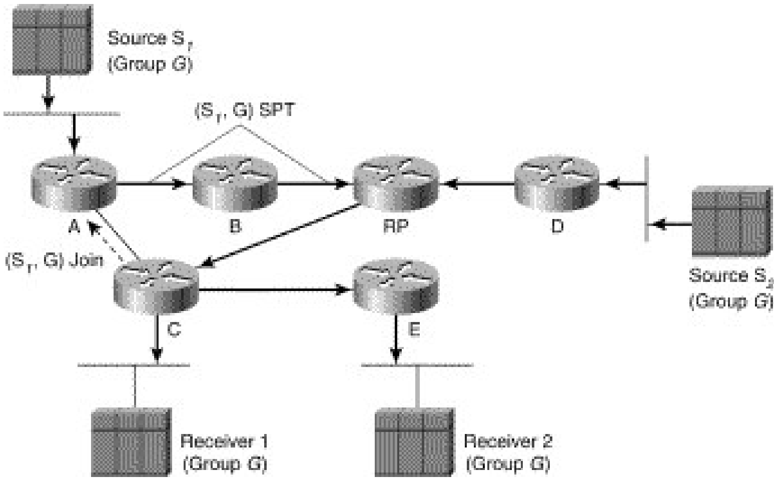


Figure 62: SPT Switchover---Step 1 [2]

When Router A receives this Join, it adds the interface over which the Join was received to the outgoing interface list of the (S1, G) entry in its multicast forwarding table. This effectively adds the link from Router A to Router C to the (S1, G) SPT, as indicated in Figure 63. At this point, (S1, G) multicast traffic can flow directly to Router C via the (S1, G) SPT.

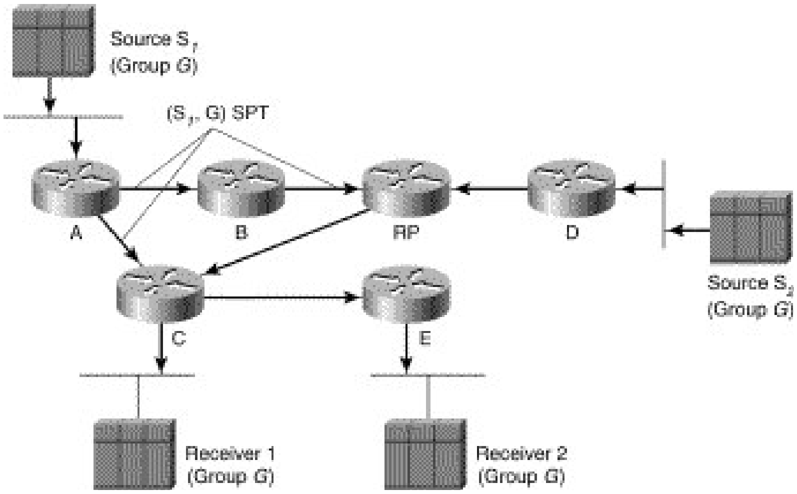


Figure 63: SPT Switchover---Step 2 [2]

Normally, Group SPT-Thresholds are configured consistently on all routers in the network, in which case Router E would also initiate a switch to the SPT by sending an (S, G) Join to its upstream router, Router C. Finally, remember that the routers, not the receivers, initiate the switch over to the SPT." [2]

At this point, we have a shared tree (*, G) from the RP down to the receiver 1, a (S1, G) SPT from RP to S1 and a (S1, G) from Router C to Router A. This leaves us with two different paths for Router C to receive multicast traffic from S1: over the shared tree and the SPT. "This would result in duplicate packets being delivered to Router C and is a waste of network bandwidth. So, we need to tell the RP to prune the (S1, G) multicast traffic from the shared tree, which is the topic of our next section.

6.7.1 Pruning Sources from the Shared Tree

When encountering the situation shown in Figure 63, in which source traffic is flowing down the shared tree that is also receiving via the SPT, a special type of Prune, referred to as an ***(S, G) RP-bit Prune,*** is used to tell the RP to prune this source's traffic from the shared tree. This special prune has the RP flag set (indicating that this message is applicable to the shared tree) in the Prune list entry. Setting this flag/bit in an (S1, G) Prune and sending it up the shared tree tells the routers along the shared tree to prune Source S1 multicast traffic from the shared tree." [2] (Figure 64)

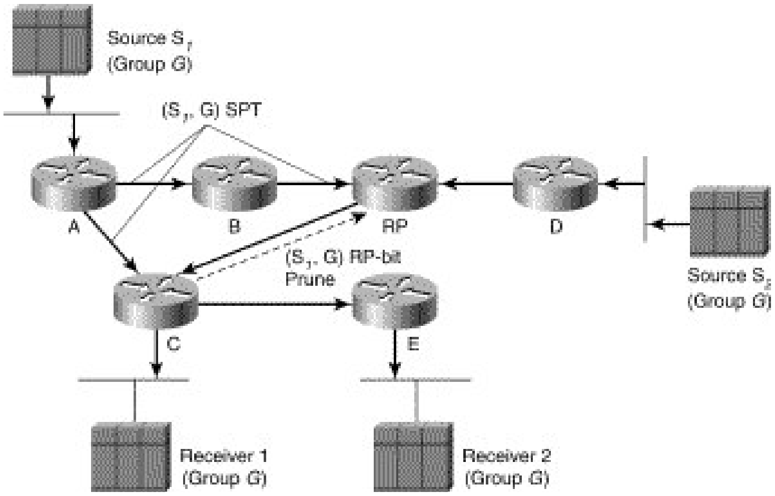


Figure 64: Pruning Sources from the Shared Tree---Step 1 [2]

Once the RP receives this special prune, it removes Router C from the outgoing interface list for (S1, G) state. Now since the RP has an empty outgoing interface list for this state, it means that RP no long need any traffic for the (S1, G) group and ends up sending an (S1, G) prune through Router B and Router A to Source S1 in a hop by hop manner. (Figure 65)

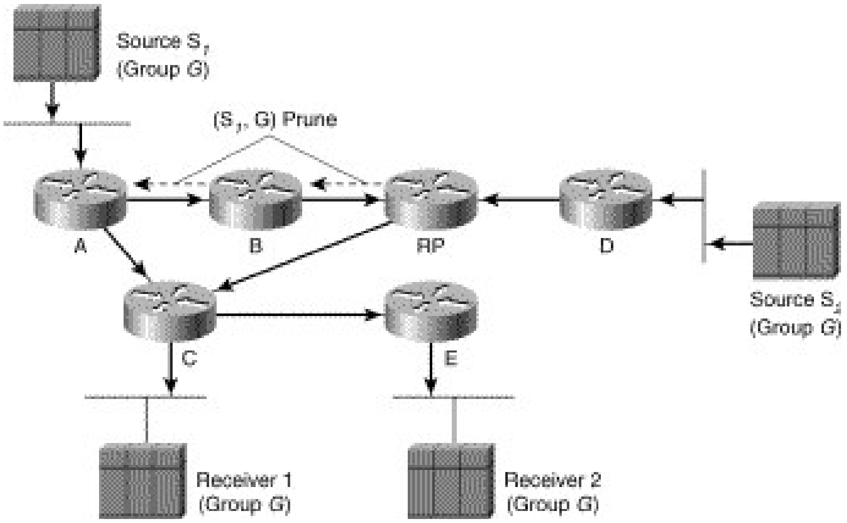


Figure 65: Pruning Sources from the Shared Tree---Step 2 [2]

“Figure 66 shows the result. Now, the (S1, G) SPT has been pruned, leaving only the link between Router A and Router C. Note that Router E is still receiving (S1, G) traffic from Router C (solid arrow from C to E), unaware that its upstream neighbor (Router C) has switched over to the SPT for Source S1.

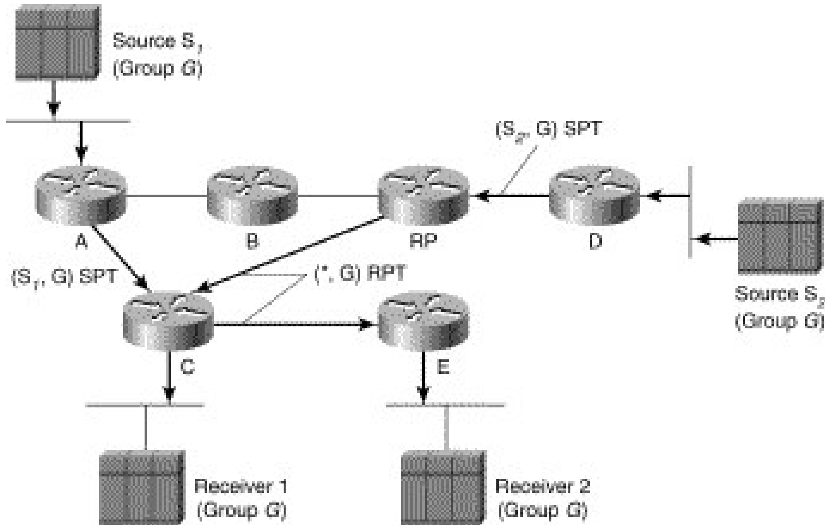


Figure 66: Pruning Sources from the Shared Tree---Step 3 [2]

Also note in Figure 66 that (S2, G) traffic is still flowing to the RP and down the shared tree to reach Receiver 1 and Receiver 2.

6.8 PIM-SM Designated Router

PIM elects a DR on each multi-access network (for example an Ethernet segment) using PIM Hello messages. In PIM-SM, DR has a very important role unlike in PIM-DM, where a DR was significant only in case of IGMPv1 where it plays the role of IGMP Querier, since IGMPv1 did not have a mechanism to select an IGMP Querier.

6.8.1 The Role of the Designated Router

Consider the network example shown in Figure 67, in which two PIM-SM routers are connected to a common multi-access network with an active receiver for Group G. Because the Explicit Join model is used, only the DR (in this case, Router A) should send Joins to the RP to construct the shared tree for Group G. If both routers are permitted to send (*, G) Joins to the RP, parallel paths would be created and Host A would receive duplicate multicast traffic.

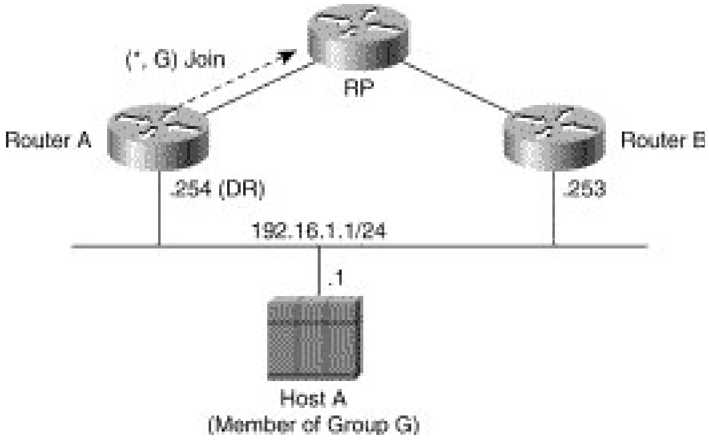


Figure 67: PIM-SM Designated Router [2]

Similarly, if Host A begins to source multicast traffic to the group, the DR (Router A) is the router responsible for sending Register messages to the RP. Again, if both routers were permitted to send Register messages, the RP would receive duplicate multicast packets.

6.8.2 Designated Router Failover

When more than one router is connected to a LAN segment, PIM-SM provides not only a method to elect the DR but also a way to detect the failure of the current DR. If the current DR (Router A) shown in Figure 67 were to fail, Router B would detect this situation when its neighbor adjacency with Router A times out. Now, a new DR election takes place, and Router B becomes the active DR for the network.

In this case, Router B is already aware that an active receiver (Host A) exists on the network because it has been hearing IGMP Membership Reports from the host. As a result, Router B already has IGMP state for Group G on this interface, which would cause B to send a Join to the RP as soon as it was elected the new DR. This step re-establishes traffic flow down a new branch of the shared tree via Router B. Additionally, if Host A were sourcing traffic, Router B would initiate a new Register process immediately after receiving the next multicast packet from Host A. This action would trigger the RP to join the SPT to Host A via a new branch through Router B.” [2]

6.9 RP Discovery

For PIM-SM to work properly, it is necessary for all the routers in the PIM-SM domain to know the address of the RP. Though it might be feasible to manually specify the IP address of the RP in the configuration of each router, in case of a small network that use a single RP for all multicast groups, as the size of the network grows and/or if the RP changes frequently, manual configuration of every router will soon become an administration nightmare. This could go even worse if different multicast groups get configured with different routers in the domain as RPs, in order to either optimize the shared tree or spread the RP workload across multiple routers.

“PIMv2 defines a *Bootstrap* mechanism that permits all PIM-SM routers within a domain to dynamically learn all *Group-to-RP* mappings and avoid the manual RP configuration problem. In addition, Cisco's PIM implementation has another mechanism, *Auto-RP*, which can accomplish the same thing. Cisco's Auto-RP was developed before the PIMv2 specification was written so that existing Cisco PIM-SM networks could dynamically learn *Group-to-RP* mappings.

6.10 PIM-SM Suitability/Scalability

Because PIM-SM uses the Explicit Join model, multicast traffic is better constrained to only those portions of the network where it is actually desired. Therefore, PIM-SM does not suffer from the inefficiencies found in flood-and-prune protocols. As a result, PIM-SM is better suited to multicast networks that have potential members at the end of WAN links.

PIM-SM also enables network engineers to use SPTs to reduce the network latency commonly associated with the use of shared trees. The decision to use or not use SPTs can be made on a group-by-group basis. For example, a low-rate, many-to-many multicast application (such as SDR) running over a star-topology network may not warrant the use of SPTs. In this case, the use of Infinity as the group's SPT-Threshold could force all group traffic to remain on the shared tree. This ability to control the use of SPTs gives network engineers greater control over the amount of state created in the routers in the network. Ultimately, the amount of state in the routers in the network is one of the primary factors that affect the scalability of any multicast routing protocol.

PIM-SM is arguably the best choice for an intra-domain multicast routing protocol for most general-purpose multicast networks. The possible exceptions are dedicated, special purpose networks that are designed to run very specific network applications under the complete control of the network administrators. (In these cases, PIM-SM may still be the best choice, although other protocols could possibly be made to work adequately given the tight controls that the network administrators have over the network and its applications.)" [2]

6.11 Summary

The chapter covers the fundamentals of PIM-SM beginning with the Explicit Join model that ensures that the multicast traffic in the domain is available only at the segments that need the traffic, rather than all the multicast enabled routers as in the case of protocols using the Flood-Prune behavior. It explains with example how PIM-SM Join/Prunes are used to build shared trees and SPTs. In addition, it explains the need for finding a method to get the Source traffic to the RP since the traffic on the shared tree is unidirectional with the root at RP. The method used for this is the RP forming an SPT toward the source. Also, the RP is made aware of the existence of a source by the use of PIM Register and PIM Register-Stop messages which carry the first few multicast traffic embedded in it in an IP-to-IP manner. Also briefly explains the possibility for the receivers to switch from a shared tree to an SPT directly with the Source in order to optimize the traffic flow in the network and methods to prune an interface when there is a chance for duplicate packets.

7 Core-Based Trees

The Core-Based Tree (CBT) multicast routing protocol is truly a work-in-progress and has been so for several years. The original version, CBTv1, was superseded by CBTv2 and now a draft specification for CBTv3 is available. Though these versions are not backward compatible, it is not expected to be a major issue since the previous versions have not been implemented on a wide spread. This chapter provides a brief discussion of the concepts and mechanisms used by version 2 of the CBT multicast routing protocol. The coverage includes a brief overview of CBT, followed by a very short discussion of some of the proposals in version 3 of the CBT protocol and concludes with an examination of the scalability and suitability to the use of CBT in today's multicast network.

7.1 CBT Overview

One of the major design parameters of CBT was to make "the protocol scalable as the number of active groups in the network grows and to work towards the goal of reducing multicast state in the routers in the network to *O(G)* (pronounced "orders G"). To accomplish this goal, CBT was designed as a sparse mode protocol (similar in many ways to PIM-SM) that uses only *bidirectional* shared trees to deliver multicast group traffic to portions of the network that have specifically joined the group. These bidirectional shared trees are rooted at a *Core* router, (as the name Core-Based Trees suggests) and permit multicast traffic to flow in both directions, up or down the tree. The bidirectional nature of these trees means that routers already on the shared tree do not have to perform any special tasks to forward locally sourced multicast traffic to the Core. Instead, the first-hop CBT router can simply send the traffic up the tree. Each router on the tree simply forwards the traffic out all interfaces on the tree other than the interface on which the packet was received.

Figure 68 is an example of a CBT that has several member hosts, M1 to M7, joined to the tree. In this example, member M3 is also a source of multicast traffic to the group. Because it is both a member and a source, M3 is called a *member source*.

Notice that the traffic sent by M3 flows both up and down the shared tree as shown by the arrows in Figure 68. The bidirectional nature of CBT means that no special processing is necessary to get the traffic to the Core so that it can flow down the other branches of the CBT to other members. The key benefit here is that there is also no need for additional forwarding state in the routers for member senders.

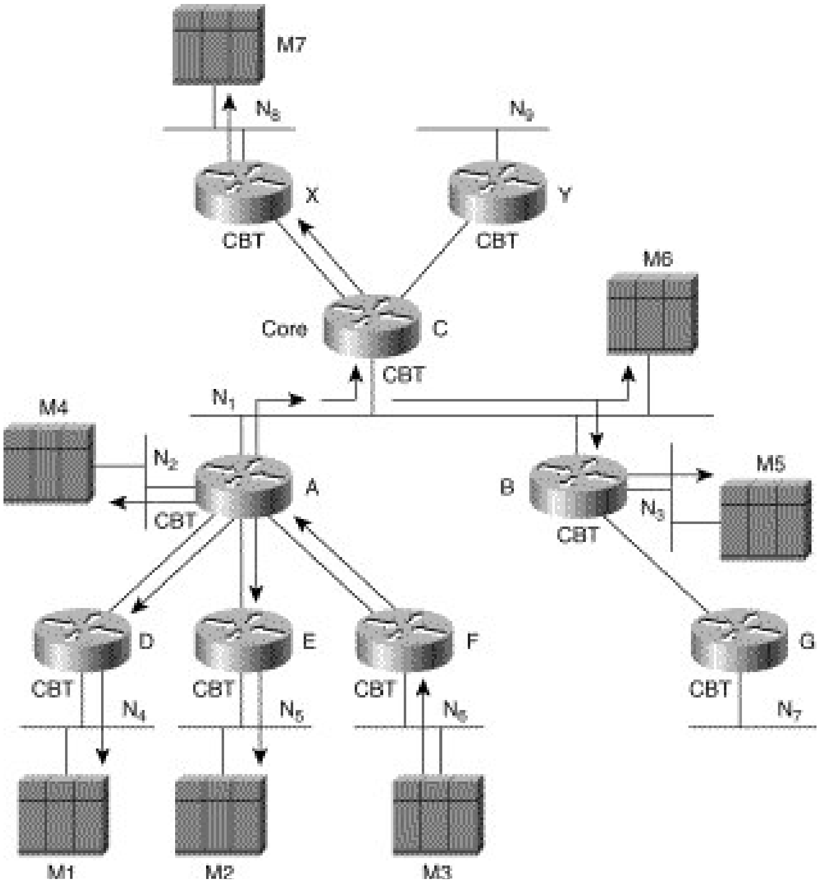


Figure 68: Traffic Flow on CBTs [2]

CBT differs from PIM-SM in that source traffic cannot flow *up* the shared tree because it is *unidirectional*. Instead, PIM-SM uses shortest path trees (SPTs) between the rendezvous point (RP) and the sources to get the traffic to the RP so it can be forwarded down the shared tree." b One challenge for CBT is how to allow sources that are not members of the CBT to send traffic to the core so that it can be distributed down the tree. This traffic will have to be sent over an IP-in-IP tunnel from the non member to the Core, since CBT has no way to encapsulate this traffic from the non member like PIM-SM that used Source Registration to encapsulate the multicast traffic from the source to the root, even when it is not part of the shared tree.

As mentioned earlier, CBT was designed with the aim of minimizing the states in the router and hence does not support SPTs. So there is no flexibility for the paths to switch between shared trees and SPTs depending on which is the optimal path, leading to the chances of increased latency due to the selection of non optimal paths depending on the placement of the Core (as shown in Figure 69), even though it tags along with the design aspect of CBT by reducing the amount of multicast state in the routers to only (*, G) state entries in the multicast routing table.

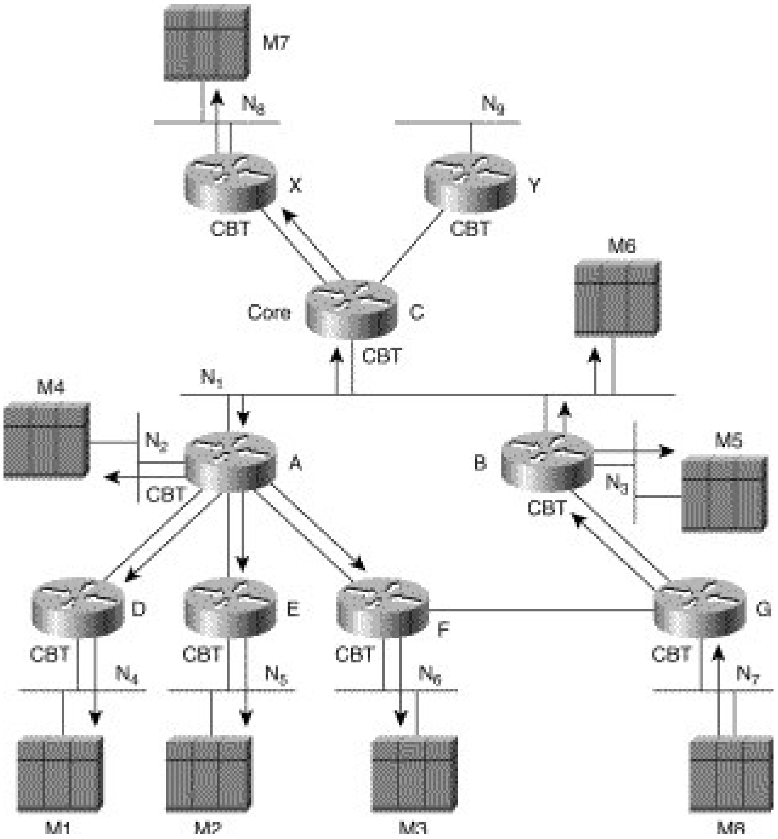


Figure 69: Suboptimal Traffic Flow [2]

The member M8 in Figure 69 has joined the CBT to connect to the Core Router C via Router B. When M8 starts sourcing multicast traffic to network N7 it will be send up the shared tree to the core. Consider the Member M3 receiving this traffic. It will be receiving the traffic from the Core via the segments N1- Router A – Router F – segment N6. So the total path from the source for Member M3 is via Routers G-B-A-F, even though there is a direct path between G-F. In case of PIM-SM, the traffic would have switched over to the STP there by reducing latency.

7.2 CBT Version 3

“CBTv3 is primarily concerned with extensions that permit it to better handle inter-domain multicast by the use of CBT border routers (BRs). Compared to its predecessors, version 3 requires significantly more state information in the routers so that this BR function can be implemented efficiently.

For example, CBTv2 needed to maintain (*, G) state information only in the form of its forwarding cache entries. Minimizing router state to O(G) is one of the stated advantages of CBT. However, CBTv3 defines new (*, Core) and (S, G) states in order to support pruning of multicast traffic flowing into, out of, or across the domain through the BRs. This, in turn, has resulted in a fairly substantial change and extension to the CBT packet formats that are not backward compatible with CBTv2." [2]

7.3 CBT Suitability/Scalability

The biggest advantage of CBT over protocols that support SPT is its efficiency in terms of the amount of multicast state it creates in the router, limiting it to O(G) in networks with large number of sources and groups. The only downside being that this comes at the compromise of optimal paths, thereby adding to latency, in certain situations due to the lack of support of SPT.

"Furthermore, the simple (*, G) state model that is maintained in CBTv2 does not handle nonmember senders or inter-domain border routers particularly well. Attempts are being made to overcome this in CBTv3. CBTv3 is a substantially more complex protocol that departs from the simple bidirectional (*, G)-state-only model and introduces new (*, Core) and (S, G) states as well as the concept of *unidirectional* branches of the CBT. These extensions can potentially cause the amount of state in a CBT domain to approach the amount of state in a similar PIM-SM domain." [2]

7.4 Summary

The chapter covers a very brief overview of the mechanisms of version 2 of the CBT multicast routing protocol. "CBTv2 is based on the use of a bidirectional, shared tree rooted at a Core router in the network. Because CBTv2 supports only traffic forwarding along this shared tree and does not support the notion of SPTs, the amount of state that must be maintained is considerably reduced." b The idea of keeping the multicast states in a network is quiet appealing when viewed at the perspective of large networks with a number of active sources and members, this is still a work-in-progress with little or no actual network implementations, and hence the not-too-much details in the chapter.

8 Multicast Open Shortest Path First

This chapter is only an introduction to the link state protocol MOSPF which is developed as an extension to OSPF. The details on this protocol is beyond the scope of this paper since it is intended to cover the protocol that is most common in production networks, which is PIM-SM, beyond doubt. It is good to start the discussion with a briefing on OSPF.

"OSPFv2 is a link-state unicast routing protocol that uses a two-tier, network hierarchy. At the top of the hierarchy is Area 0, aka the backbone area, to which all other areas in the second tier must connect, either physically or through a virtual link. Open Shortest Path First (OSPF) is hierarchical because all inter area traffic must flow through the backbone area, Area 0. Inside each OSPF area, routers flood link-state information that describes the network topology within the area, and each router maintains a copy of this information in its area database. Each time the topology change, new link-state information is flooded throughout the area so that all routers get an updated picture of the topology. Using the topology information in the area databases, each router (using itself as the root) constructs a lowest cost, spanning tree of the networks within the area via a special Dijkstra algorithm. This lowest cost, spanning tree is then used to build a unicast forwarding table. Special area border routers (ABRs) connect second-tier areas to Area 0 and maintain separate area databases for each area it connects (including Area 0) so that it can forward traffic across the area boundaries.

Multicast Open Shortest Path First (MOSPF), defined in RFC 1584, "Multicast Extensions to OSPF" is an extension to the OSPFv2 unicast routing protocol. MOSPF provides additional OSPF data format definitions and operating specifications. These extensions to the OSPF protocol permit multicast traffic to be forwarded within an OSPF unicast network using shortest path trees (SPTs) by either a partial or full set of MOSPF routers. An MOSPF router is one that supports the multicast extensions to OSPF and has been configured to function as an MOSPF router. All routers in the OSPF network need not be running MOSPF extensions for multicast traffic to be forwarded. However, routers that are configured to function as MOSPF routers perform multicast routing in addition to performing normal OSPF routing.

8.1 MOSPF Suitability/Scalability

The biggest advantage to MOSPF is that it shares OSPF's capability to respond rapidly to changes in network topology as it uses link-state routing methods to compute multicast distribution trees. This capability, however, comes at the huge expense of rapidly increasing router CPU resources as the number of (S, G) pairs in the network increase, thereby increasing the number of Dijkstra computations in each router. Highly dynamic networks (where group membership and/or the network topology changes frequently because of unstable WAN links) also suffer from an increased number of Dijkstra computations required to reconstruct the (S, G) SPTs in the area." [2]

MOSPF can be considered best suited for special-purpose multicast networks where the network administrators have absolute rigid control over very crucial factors that affect the performance and scalability of MOSPF, such as Location of sources, Number of sources, Number of groups, Group membership. Because most of these factors are often ***not*** under absolute rigid control of today's corporate IS networks, the long-term suitability and scalability of MOSPF in these networks should be carefully questioned. In addition to the factors listed above, the multicast applications that run on host machines could make the things even worse for the network administrators. For example, the net admin will have less to no control over users installing freeware MBone multimedia conferencing tools on their workstations. Therefore, "MOSPF is probably truly suited only for a small percentage of all general-purpose IP networks in use today." [2]

8.2 Summary

The chapter provides a quick glance of the link state protocol that has the advantages of quick learning of network changes since it learns network changes from link state advertisements. But this computation algorithm could be taxing on the CPU resources as the number of states in the network grows or if there are rapid changes in the network due to other reasons. The basic concept employed by MOSPF for intra-area multicast routing is the flooding of group membership information throughout the area, using a special group membership LSA. When the MOSPF routers know where all members are in the area, SPTs can be constructed for each source-subnet, using the same Dijkstra algorithm that is used for OSPF unicast routing. This concept can be extended to inter area multicast by introducing M Area Border Routers that inject the additional information to the Area 0.

"Finally, due to the debatable topic of scalability of MOSPF in larger networks, even though MOSPF is currently deployed in some production networks, it is still too early to tell if it will stand the test of times." [2]

9 Multicast Applications

One of the most common applications of multicasting is video conferencing that people often think that all video conferencing application works on multicasting or even fail to differentiate between one and the other. There are multiple other applications that make use of multicasting and not all video conferencing applications works on multicasting technology and it has to be agreed that video conferencing was one of the earliest applications of multicasting. Some of the initial experiments with video conferencing on multicast network proved that the additional bandwidth consumed for the application could not be justified when compared with the value added to the conference. "This section looks at some other IP multicast applications that have the potential for improving productivity, including multimedia conferencing, data replication, real-time data multicasts, and gaming and simulation applications." [2]

9.1 Multimedia Conferencing

"Some excellent IP multicast, multimedia conferencing tools were developed for the UNIX environment for use over the MBone" b (which is discussed in detail in the coming sections). Even though these tools were initially developed for UNIX environment, most of them have been ported to Windows 95 and NT platforms. "These tools permit a many-to-many audio only or audio/video conference to take place via IP multicast. In addition to the audio and video tools, a UNIX-based Whiteboard tool was developed that permits users to share a common, electronic whiteboard. Besides these MBone freeware tools for multimedia conferencing over IP multicast networks, other companies are now beginning to offer commercial forms of these tools with other value added features." [2]

Though people start using video conferencing with audio, since it is a novel way of having a group conversation, once the freshness of the application fades away and the users start looking into the aspects like resource utilization (mainly bandwidth, considering that each participant can be a source of multicast traffic), it is quite possible that regular audio-only conferences added with the shared Whiteboard application where people can share screens with charts, diagrams, notes etc becomes the norm. This could result in an "extremely powerful form of multimedia conferencing that does not consume much bandwidth." [2]

9.2 Data Distribution

"Data replication is another IP multicast application area that is rapidly becoming very popular. By using IP multicasting, IS departments are adopting a push model of file and database updates. Applications such as Starburst's (http://www.starburstcom.com) MFTP product permit the reliable delivery of files and data to groups of nodes in the network. As the name MFTP implies, this product is like a multicast form of FTP. One or more files may be sent simultaneously with FTP to a group of nodes in the network by using IP multicasting. This sort of technology permits companies to push new information such as price and product information to their remote stores every night so that the stores have up-to-date information the next business day." [2]

9.3 Real-Time Data Multicasts

There are various industrial sectors that can make use of the delivery of real-time data to their internal users and their clients, for example the stock ticker information to the trading floor. IP multicasting can be used to deliver this information to the traders in real time. Initial beneficiaries of the early applications using real time delivery of stock ticker using multicasting was the trading floor but "more and more financial and investment firms are now investigating the use of IP multicasting to deliver information to their customers as another revenue-generating financial and trading service. By assigning different financial categories (bonds, transportations, pharmaceuticals, and so forth) to different multicast groups, traders can use their workstations to receive only the real-time financial data for which they are interested." [2]

9.4 Gaming and Simulations

One of the biggest benefactors of the multicast applications is going to be the online gaming sector which involves networked gaming and simulation applications. Even though most of these gaming and simulations allow groups of people to interactively participate in the action, they are connected to each other via unicast, point-to-point connections. These connections are maintained either as individual connection to multiple user which could be growing in the Order N2 for N users, or through a central gaming or simulation server to which each user gets connected to via an unicast connection. Either of the two methods places additional load on the individual host or the server as the case may be and restrict the number of concurrent users to 5 to 10 participants, thereby limiting the fun of the application.

“IP multicasting can be used to extend gaming and simulations to extremely large numbers of participants. Participating PCs or workstations simply join the IP multicast group and begin sending and receiving gaming and simulation data. Dividing the simulation data into more than one stream and then communicating this information via separate IP multicast groups can further extend this concept. This division of data permits the PCs or workstations to limit the amount of simulation data that they are sending and receiving (and, hence, the number of IP multicast groups they need to join) to what they currently need to participate in a game or simulation situation.

For example, each room in a fantasy battle game could be assigned a separate IP multicast group. Only those PCs or workstations whose participants are in this room need to join this multicast group to send and receive simulation data about what is happening there. When players leave the room and go into another room, they leave the IP multicast group associated with the first room and join the IP multicast group associated with the new room.

As more IP networks become multicast enabled, more game and simulation application developers are expected to make use of IP multicasting for large-scale simulations. It's not unthinkable that sometime in the near future, thousands of gamers will be simultaneously battling it out over the Internet in the ultimate Doom game.” [2]

9.5 MBone---The Internet’s Multicast Backbone

The Internet's *Multicast Backbone (MBone)* is the small subset of Internet routers and hosts that are interconnected and capable of forwarding IP multicast traffic. The usage of the word subnet of Internet is due to the reason that all parts of the internet are not yet capable of handling multicast traffic. It is quite possible for the new users to assume that if their workstation is installed with a multicast application and if they are connected to an internet gateway that has multicasting enabled they would be able to access multicast traffic. This need not be the case all the time, since there are many other variable involved in ensuring a successful multicast exchange which will be briefed in the following pages with reference to MBone.

The next few sections describe various MBone session examples, a history of the MBone, and the MBone architecture of today and tomorrow.

9.5.1 MBone Sessions

“One of the most popular sessions on the MBone is the audio/video multicast of NASA's shuttle missions. Some individuals, mostly engineers who work on the technology, have used MBone for some interesting and rather bizarre purposes, for example, like setting up pet-cams/cat-cam to broadcast video of their pets to monitor his cat's recovery from its recent surgery. Some other occasions include the live CNN feed of the O. J. Simpson verdict which had over 350 members tuned in at one point. Other major media events that have been multicast over the MBone include the 1994 Rolling Stones concert which was multicast over the MBone from the DEC Systems Research Center. Interestingly, the rock group Severe Tire Damage (several members of which were Internet engineers) began transmitting audio and video of their band performing live music half-hour before the Rolling Stones concert began, thereby "opening" for the Rolling Stones via the MBone.” [2]

Apart from the popular NASA shuttle missions and the other media events or individual efforts for using the MBone architecture, the biggest users of this technology is going to be the various conventions and seminars by various technical organizations like IETF, IEEE and even large businesses. All that an individual may need to connect to one of these sessions may be a Cisco 1600 router and ISDN line that connect the user Cisco's corporate network. Participation in these kinds of events by interested parties and the display of effectiveness of the application would keep the demand for MBone connectivity to new levels and thereby promoting more research and development in the field helping its growth to larger parts of the Internet. This will ensure that commercial and private multicasting over the MBone will soon become part of the new Internet experience.

9.5.2 History of the MBone

In the early 1990s, as internet was becoming more and more popular among public and was no longer exclusively available for the researches for experimentation with networking technologies, the Defense Advanced Research Projects Agency (DARPA - the governing body of the Internet at that time) developed the DARPA Test bed Network (DARTNet) to give the researchers a playground network on which they could test and evaluate new tools and technologies without affecting the production Internet.

“This DARTNet was initially composed of T1 lines connecting various sites including Xerox PARC, Lawrence Berkley Labs, SRI, ISI, BBN, MIT, and the University of Delaware. The sites used Sun SPARCstations running *routed* as the unicast routing daemon as well as *mrouted* as the DVMRP multicast routing daemon, thereby providing native IP multicast support between all sites. With this exclusive DARTNet platform available for researchers the normal research and development continued as normal with weekly audio conferences between various DARTNet sites around the United States, till early 1992 when IETF made plans to hold their next meeting in March in San Diego, California.

This created a situation, as one of the researches was displeased as she was not going to be able to make the trip to San Diego though she still wanted to participate. Several DARTNet researchers, including Steve Deering and Steve Casner, decided to audio multicast the IETF proceedings, which gave them a chance to let the displeased researcher to be part of the session via the DARTNet network and also allowed the researchers to further test the concepts of IP multicasting over the Internet.

Steve Deering and Steve Casner made possible the feeding of audio into a borrowed Sun SPARCstation at the San Diego IETF and in order to get the multicast audio back into DARTNet, a DVMRP tunnel was configured between the SPARCstation at the IETF and the DARTNet backbone. With this set up available, invitations to participate in this IETF audio multicast were also sent out ahead of time to various Internet research organizations in the United States, Australia, Sweden, and England, along with information on how to configure a Sun SPARCstation with a DVMRP tunnel through the Internet back to the DARTNet backbone. Several sites responded to the invitation and setup DVMRP tunnels to the DARTNet backbone. The result was the first audio multicast of the IETF to several locations on the Internet around the world.

During one of the plenary sessions at the IETF, Steve Deering and Steve Casner had made arrangements to transmit the audio of the meeting on the public address system in the room. “The attendees of the session were informed that the session was being audio multicast over the Internet to several locations throughout the United States, Australia, Sweden, and England. During the presentation, the plenary speaker posed a question to the multicast audience. Immediately, the voice of one of the multicast participants in Australia came through as clear as a bell over the public address system, and the participant proceeded to answer the speaker's question! Multimedia conferencing by way of IP multicast over the Internet had come of age.” [2]

Even though the DVMRP tunnels were torn down after the March 1992 meeting, considering the success of audio multicasting "plans were made to multicast both audio and video from the next IETF convention. Invitations were again sent out to even more sites on the Internet, and DVMRP tunnels were again built from these sites back to the DARTNet backbone. Like the March IETF multicast from San Diego, the Washington, D.C., IETF held that summer was also successfully audio and video multicast to participants all over the world.

People were, by now, beginning to see the power of IP multicasting. The network administrators at DARTNet and the other participating sites decided to leave the DVMRP tunnels in place for on-going multimedia conferencing over the Internet. These initial tunnel sites, coupled with DARTNet serving as the initial multicast core network, were soon dubbed the MBone." [2]

9.5.3 Today's MBone Architecture

The MBone network that started in March 1992 with a few locations around the world for attending the IETF meetings over the DVMRP tunnels and Sun SPARCstations running mrouted has steadily grown over the years.

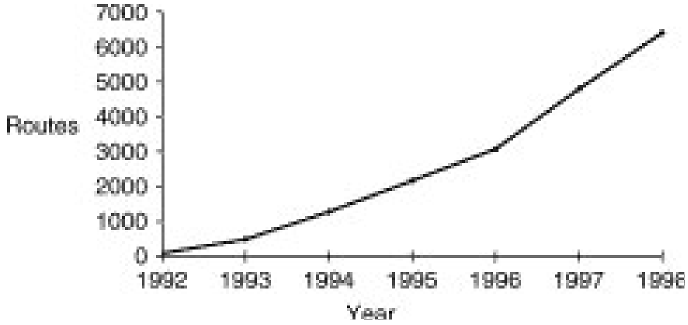


Figure 70: MBone Growth [2]

Even though the growth in the number of DVMRP routes advertised in MBone has shown an increase from a few hundred users in 1992 to a few thousands it is still a long way from being advertised in numbers similar to the total number of unicast routing prefixes which is over 50,000. It means that there is still a long way to go before the entire Internet supports IP multicasting.

One of the points that need mention with the MBone architecture is that even though there has been a significant increase in the number of routes, "the basic architecture of today's MBone has not changed substantially since it was built in 1992. With just a few exceptions, DVMRP routes are still being exchanged between MBone routers over a network of DVMRP tunnels. The downside of this is that, since DVMRP was not designed to be an inter-domain multicast routing protocol or to scale to the size of the Internet, we clearly need a new protocol, a new MBone architecture, or both to make Internet multicast traffic as ubiquitous as Internet unicast traffic.

9.5.4 Tomorrow's MBone Architecture

As mentioned above, most of the recent works involve attempts to design new protocols and architectures to permit IP multicasting to be extended to all points on the Internet. There is the need to develop inter-domain multicast routing protocols and forwarding algorithms to give ISPs the control over multicast peering and traffic management for them to offer a reliable multicast service without significantly impacting the existing unicast service. We will also need dynamic multicast address to support new multicast architectures and to carefully manage the limited IP multicast address space.

In addition to solving the issues with inter-domain multicast routing, the ISPs that make up the lion's share of today's Internet must develop the financial and billing models to offer IP multicast as a service to their customers. This would involve making the decision on whether it is the sender or the receiver who should be paying for making use of the multicast technology over the service provider's network. So, defining the financial and business models is considered to be nearly as complex a process as solving the routing issues and will have to be addressed.

9.6 Summary

Although IP multicasting has been around since the early 1990s, its power is only now beginning to be realized. Corporations are trying to exploit the benefits in bandwidth utilization and the ability deliver content to large numbers of receivers simultaneously, while ISP's are seeing benefits in offering IP multicast as a service to their customers. The MBone which has seen rapid growth in the last few years is undergoing significant research and development in order to keep the trend going by making itself able to scale to larger networks. Most importantly since we are witnessing the development of a new technology, the networks must be carefully designed, using some new design philosophies, to support IP multicasting without posing problems to the existing unicast architecture.

10 Conclusions

Over the span of the paper so far, we have gone over the basics of the Multicast protocol with over view on the various protocols like DVMRP, PIM-DM, PIM-SM, CBT and MOSPF, that are used in multicast networks. We have gone over in detail about IGMP which is used between the host devices and the next hop multicast routers to address its interest in joining/leaving a multicast group. There is a short overview of the sectors in which multicast can play a significant role in improving the performance of many applications are currently in use on the unicast network at the expense of huge bandwidth consumption. The paper is being concluded by going over the Pros and Cons of IP Multicasting.

10.1 The Pros of IP Multicast

With the growth of number of connected users in the Internet and even in company intranets there is a very good chance that many of the users will be trying to access the same contents simultaneously. “The use of IP multicast techniques to distribute this information can often substantially reduce the overall bandwidth demands on the network. A good example of this approach is the rapidly growing area of audio and video Web content.

For the following discussion we are going to use the example of ABC Company which uses a suite of audio servers to transmit popular radio talk show content, such as the Rush Limbaugh and Howard Stern shows, in real time to connected subscribers over the Internet. Some of the pros of IP multicasting discussed here include: bandwidth, server load, and network loading.

10.1.1 Bandwidth

To get a measure of how multicasting can help in conserving bandwidth for corporate, let us consider for example that “ABC Company is transmitting real-time feeds of the Rush Limbaugh talk show via an audio compression technique that requires an 8-kbps data stream to deliver. As shown in Figure 71, as the number of *unicast* subscribers increases the amount of bandwidth consumed also increases linearly as shown by the dotted lines, while “if you are *multicasting* the same program (represented by the solid line), a single 8-kbps multicast data stream can deliver the program to all the subscribers. It should be pretty straight forward to assume that ABC's revenues would be based on the number of subscribers, the prime goal of the marketing department to have thousands of clients. This would demand a huge available bandwidth in the range of 100 Mbps to support this scenario if unicast is being used.

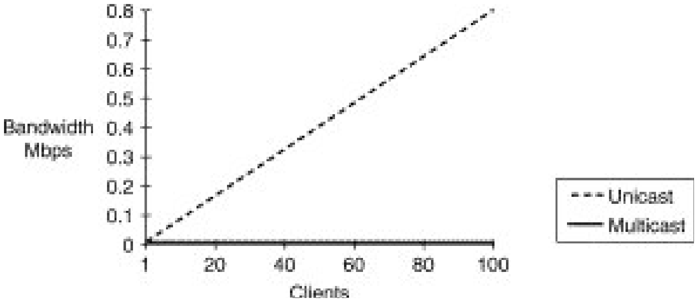


Figure 71: Unicast versus Multicast Bandwidth for Audio [2]

To stretch the application even further, suppose that ABC" has been very successful with this product and wants to extend its service offering to include highly compressed, low-rate, 120-kbps video streams to go along with the 8-kbps audio programs. Figure 72 shows that if the unicast model continues to be used as the delivery method, the bandwidth requirements are driven even higher.

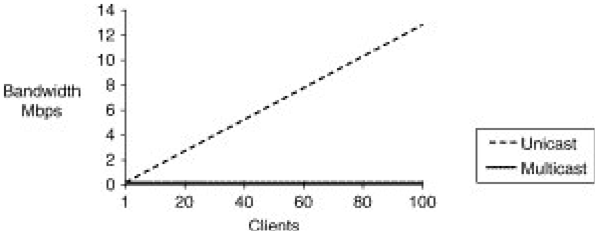


Figure 72: Unicast versus Multicast Bandwidth for Video [2]

The bandwidth requirements can be consider too taxing as more and more users on the internet tune in to watch ABC's content and it would even get multiplied if there are a few other competitors for ABC. The availability of multicasting as an option for delivering this kind of content will have a huge impact on the bandwidth requirements.

10.1.2 Server Load

Going back to the example of the ABC Company's delivery of real-time audio to connected subscribers via the Internet, if they keep using unicast delivery mechanism, it will have to continue to increase the power and number of its real-time audio servers to keep up with the increasing number of connected subscribers. Figure 73 shows an example of the difference between the number of flows that a real-time audio server must source to deliver Rush Limbaugh's talk show to three clients using unicast technology (shown at the top of Figure 73) and how many flows would need to be sourced if IP multicast is used (shown at the bottom of Figure 73. It is clear that for each client tuned in to receive the content, there should be a separate flow sourced from the server in case of unicast, which could place a significant load on the server with increasing number of users. As more number of users equate to more revenue, the excess load on the server would mean either congestion on the network leading to poor quality or deployment of additional servers to share the load.

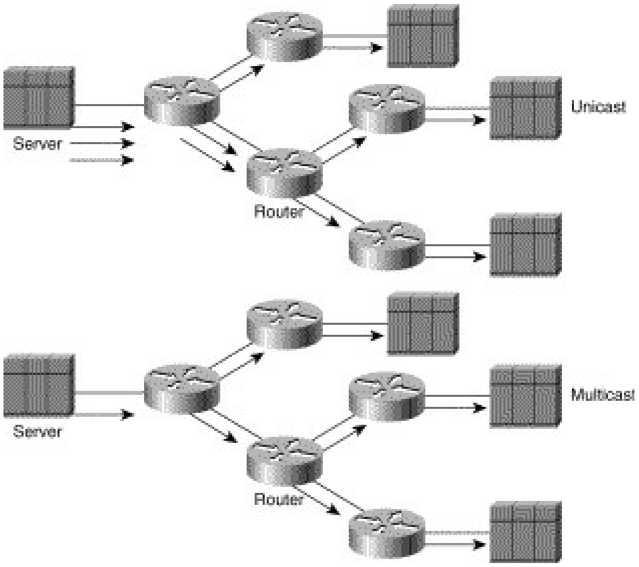


Figure 73: Server Load [2]

Switching the services to IP Multicasting can save the compromise on quality and at the same time, ensure that we don't need to invest on additional servers. This is due to the fact that, "(as shown at the bottom of Figure 73), only a single real-time data stream needs to be sourced to deliver the program to all of the connected clients." [2]

10.1.3 Network Loading

Even though the uses of IP multicasting can reduce the load on the server, it does not guarantee that it would directly imply that it reduces the load place on the routers in the network. In general, this assumption is true, but it's important to note that, in some cases, the router workload in terms of CPU and memory resources can increase at certain points in the network. This could be attributed to the fact that since the server is sending only one stream to the first hop router, it is now the routers responsibility to duplicate or multiply the packet, if there are multiple outgoing interfaces on the router that needs this traffic. "If a router does not have an efficient replication mechanism, the router load can increase significantly when the number of outgoing interfaces is high. The newer versions of forwarding code take care of this load by using a pointer to the data in the original packet to each outgoing interface.

10.2 The Cons of IP Multicast

Although there are a number of good reasons for wanting to use IP multicasting in networks, we need to keep in mind that there are limitations and downsides to this technology. This needs to be clearly understood, particularly if you are developing new applications that plan to use IP multicasting.

Some of the main drawbacks associated with the implementation of an IP multicast system include unreliable packet delivery, packet duplication, and network congestion." [2]

10.2.1 Unreliable Packet Delivery

Just like IP unicast, the unicast makes the router inherently unreliable. It is only through the use of TCP at Layer 4 (or some other higher layer protocol) that IP unicast data streams can be made reliable. "IP multicast packets typically use the User Datagram Protocol (UDP), which is best-effort in nature. Therefore, an application that uses IP multicasting must expect occasional packet loss and be prepared to either accept the loss or to somehow handle this at the application layer or via a reliable multicast protocol layered on top of UDP. Studies by Dr. Deering state that during periods when paths are being changed immediately due to a topology change, multicast packets that are in flight has a lower probability of reaching their destinations than the unicast packets. This is due to the fact that even if erroneous unicast forwarding information exists at some routers in the network during a topology change, the network may eventually succeed in forwarding the packet to the destination due to the availability of destination IP address in the packet, even while the network topology is in transition, though the actual path may be somewhat circuitous. The forwarding mechanisms of IP multicast, on the other hand, are based on the source IP address, and to prevent loops, the packet is discarded if it does not arrive on the interface that would lead back to the source.

10.2.2 Packet Duplication

Duplicate packets are, just as in the UDP unicast world, a fact of life. However, a key difference between unicast and multicast routing is that routers intentionally send copies of a multicast packet out multiple interfaces. This increases the probability that multiple copies of the multicast packet may arrive at a receiver. For example, in certain redundant network topologies in which multiple paths exist to the receiver, duplicate packets can occur until the multicast routing protocol converges and eliminates the redundant path. Typically, this means that only an occasional packet is duplicated within the network, although under some transient network-error conditions, a number of duplicates may arrive at the receiver. Again, well-designed IP multicast applications should be prepared to detect and handle the arrival of the occasional duplicate packet.

10.2.3 Network Congestion

In the TCP unicast case, the standard TCP *backoff* and *slow-start* window mechanisms automatically adjust the speed of the data transfer and therefore provide a degree of congestion avoidance within the network. Because IP multicasting cannot use TCP (due to its connectionless, one-to-many nature), there is no built-in congestion avoidance mechanism to prevent a multicast stream from exhausting link bandwidth or other critical router resources. Having said that, it is important for you to note that UDP unicast data streams suffer the same congestion avoidance problems! Furthermore, the recent growth in popularity of multimedia audio and video applications both on the Internet and within private intranets is increasing the amount of UDP unicast traffic. Going forward, you will find that there is no provision to prevent you from joining a multicast group that is sourcing data at a rate that exceeds the total available bandwidth in your portion of the network.

Figure 74 shows two IP multicast servers sourcing the same video content. One server sources the program at 500 kbps, intended for use only in the local corporate headquarters network environment, while the other server sources the program at 128 kbps, intended for use by the remote sales offices.

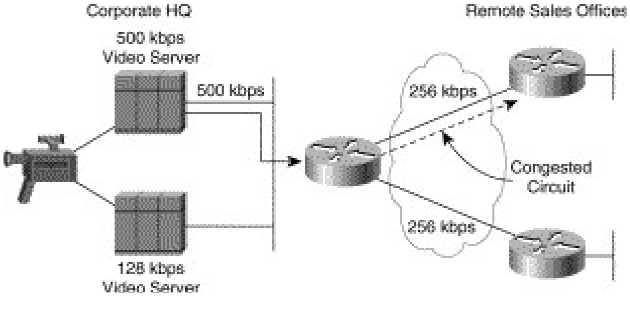


Figure 74: Exceeding Network Bandwidth with Multicast Traffic [2]

If a user at a remote sales office joins the 500-kbps multicast group by mistake, the result will be that the 256-kbps circuit to the remote sales office will be completely consumed by the 500-kbps video multicast traffic. There are methods to configure the 256-kbps circuit to limit the amount of bandwidth that the multicast traffic can consume which will need to be applied here to prevent the situation. Another alternative is to use administratively scoped boundaries to prevent users in the remote office from joining the 500 kbps group.” [2]

With this note on going over the pros and cons of IP multicast networks, I would like to conclude this paper. As it stands today, PIM-SM is still the protocol of choice for network engineers, though further developments in other protocols could possibly provide a better contented for the top spot depending on its scalability and suitability. Also, the world is looking forward to the developments on the MBone architecture so as to make it available to the larger portion of the Internet world.

11 References

[1] http://support.microsoft.com/kb/291786

[2] Developing IP Multicast Networks by Beau Williamson

[3] http://www.tml.tkk.fi/Opinnot/T-111.350/2005/Slides/Multicast_6.pdf

[4] http://www.cs.clemson.edu/~jzwang/0808360/cpsc36015.pdf

[5] http://www.ic.uff.br/~michael/EngRedes/TKK/13TKK_multicast.pdf

[6] http://www.youtube.com/watch?v=TApIo_BiX6U