Network Working Group
Internet-Draft
Category: Informational
Expire in six months

T. Maufer
C. Semeria
3Com Corporation
July 1997

Introduction to IP Multicast Routing

<draft-ietf-mboned-intro-multicast-03.txt>

Status of this Memo

This document is an Internet Draft. Internet Drafts are working documents of the Internet Engineering Task Force (IETF), its Areas, and its Working Groups. Note that other groups may also distribute working documents as Internet Drafts.

Internet Drafts are draft documents valid for a maximum of six months. Internet Drafts may be updated, replaced, or obsoleted by other documents at any time. It is not appropriate to use Internet Drafts as reference material or to cite them other than as a "working draft" or "work in progress."

To learn the current status of any Internet-Draft, please check the "1id-abstracts.txt" listing contained in the internet-drafts Shadow Directories on:

```
    ftp.is.co.za               (Africa)
    nic.nordu.net              (Europe)
    ds.internic.net      (US East Coast)
    ftp.isi.edu          (US West Coast)
    munnari.oz.au          (Pacific Rim)
```

ABSTRACT

The first part of this paper describes the benefits of multicasting, the MBone, Class D addressing, and the operation of the Internet Group Management Protocol (IGMP). The second section explores a number of different techniques that may potentially be employed by multicast routing protocols:

- Flooding
- Spanning Trees
- Reverse Path Broadcasting (RPB)
- Truncated Reverse Path Broadcasting (TRPB)
- Reverse Path Multicasting (RPM)
- ''Shared-Tree'' Techniques

The third part contains the main body of the paper. It describes how the previous techniques are implemented in multicast routing protocols available today (or under development).

- Distance Vector Multicast Routing Protocol (DVMRP)
- Multicast Extensions to OSPF (MOSPF)
- Protocol-Independent Multicast - Dense Mode (PIM-DM)
- Protocol-Independent Multicast - Sparse Mode (PIM-SM)
- Core-Based Trees (CBT)

FOREWORD

This document is introductory in nature. We have not attempted to describe every detail of each protocol, rather to give a concise overview in all cases, with enough specifics to allow a reader to grasp the essential details and operation of protocols related to multicast IP. Every effort has been made to ensure the accurate representation of any cited works, especially any works-in-progress. For the complete details, we refer you to the relevant specification(s).

If internet-drafts were cited in this document, it is only because they were the only sources of certain technical information at the time of this writing. We expect that many of the internet-drafts which we have

cited will eventually become RFCs. See the IETF's internet-drafts shadow directories for the status of any of these drafts, their follow-on drafts, or possibly the resulting RFCs.

TABLE OF CONTENTS

Section

[This space was intentionally left blank.]

1. INTRODUCTION

There are three fundamental types of IPv4 addresses: unicast, broadcast, and multicast. A unicast address is used to transmit a packet to a single destination. A broadcast address is used to send a datagram to an entire subnetwork. A multicast address is designed to enable the delivery of datagrams to a set of hosts that have been configured as members of a multicast group across various subnetworks.

Multicasting is not connection-oriented. A multicast datagram is delivered to destination group members with the same "best-effort" reliability as a standard unicast IP datagram. This means that multicast datagrams are not guaranteed to reach all members of a group, nor to arrive in the same order in which they were transmitted.

The only difference between a multicast IP packet and a unicast IP packet is the presence of a 'group address' in the Destination Address field of the IP header. Instead of a Class A, B, or C IP destination address, multicasting employs a Class D address format, which ranges from 224.0.0.0 to 239.255.255.255.

1.1 Multicast Groups

Individual hosts are free to join or leave a multicast group at any time. There are no restrictions on the physical location or the number of members in a multicast group. A host may be a member of more than one multicast group at any given time and does not have to belong to a group to send packets to members of a group.

1.2 Group Membership Protocol

A group membership protocol is employed by routers to learn about the presence of group members on their directly attached subnetworks. When a host joins a multicast group, it transmits a group membership protocol message for the group(s) that it wishes to receive, and sets its IP process and network interface card to receive frames addressed to the multicast group.

1.3 Multicast Routing Protocols

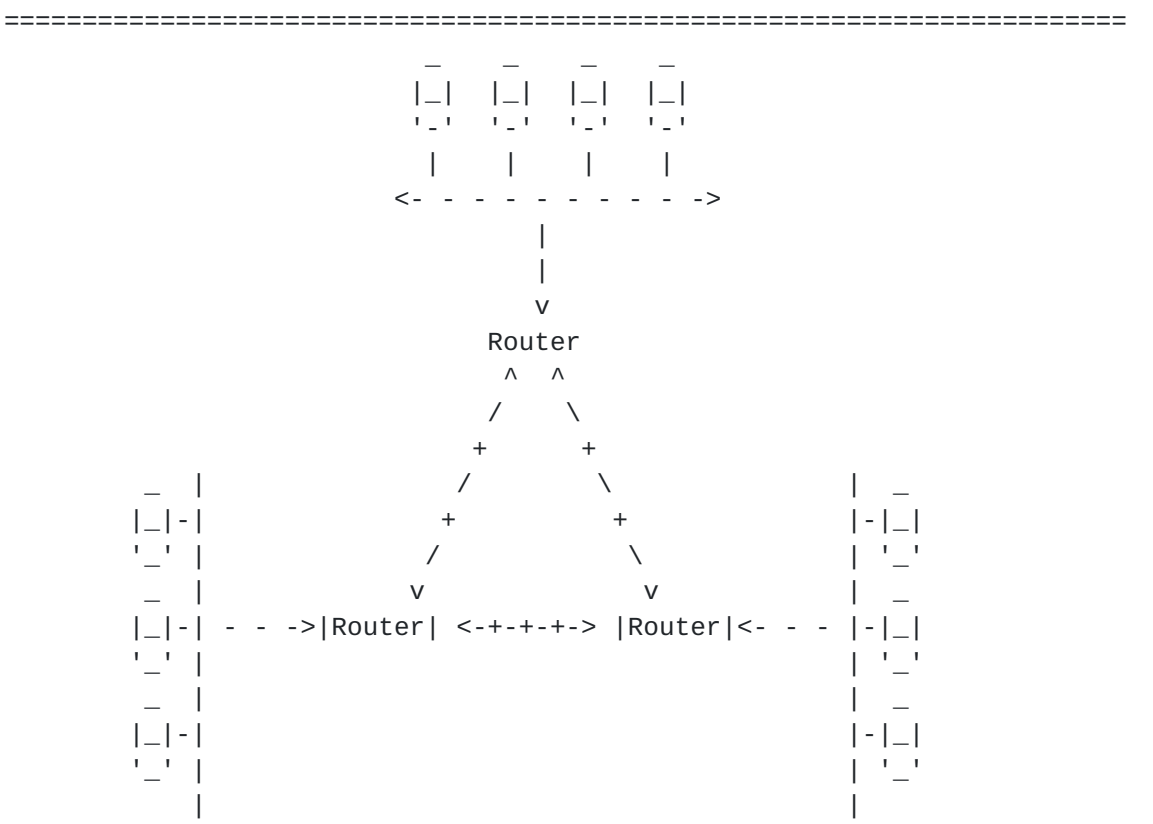
Multicast routers execute a multicast routing protocol to define delivery paths that enable the forwarding of multicast datagrams across an internetwork.

1.3.1 Multicast Routing vs. Multicast Forwarding

Multicast routing protocols establish or help establish the distribution tree for a given group, which enables multicast forwarding of packets

addressed to the group. In the case of unicast, routing protocols are also used to build a forwarding table (commonly called a routing table).

Unicast destinations are entered in the routing table, and associated with a metric and a next-hop router toward the destination. The key difference between unicast forwarding and multicast forwarding is that



LEGEND

<- - - -> Group Membership Protocol
<-+-+-+-> Multicast Routing Protocol

Figure 1: Multicast IP Delivery Service

multicast packets must be forwarded away from their source. If a packet is ever forwarded back toward its source, a forwarding loop could have formed, possibly leading to a multicast "storm."

Each routing protocol constructs a forwarding table in its own way. The forwarding table may tell each router that for a certain source, or for a given source sending to a certain group (a (source, group) pair), or for any source sending to some group, how to forward such packets. This may take the form of rules expecting certain packets to arrive on some "inbound" or "upstream"interface, and the (set of) "downstream" or "outbound" interface(s) required to reach all known subnetworks with

group members. Not all multicast routing protocols use the same style forwarding state, and some use techniques not mentioned here.

2. MULTICAST SUPPORT FOR EMERGING INTERNET APPLICATIONS

Today, the majority of Internet applications rely on point-to-point transmission. The utilization of point-to-multipoint transmission has traditionally been limited to local area network applications. Over the past few years the Internet has seen a rise in the number of new applications that rely on multicast transmission. Multicast IP conserves bandwidth by forcing the network to do packet replication only when necessary, and offers an attractive alternative to unicast transmission for the delivery of network ticker tapes, live stock quotes, multiparty videoconferencing, and shared whiteboard applications (among others). It is important to note that the applications for IP Multicast are not solely limited to the Internet. Multicast IP can also play an important role in large commercial internetworks.

2.1 Reducing Network Load

Assume that a stock ticker application is required to transmit packets to 100 stations within an organization's network. Unicast transmission to this set of stations will require the periodic transmission of 100 packets where many packets may in fact be traversing the same link(s). Multicast transmission is the ideal solution for this type of application since it requires only a single packet stream to be transmitted by the source which is replicated at forks in the multicast delivery tree.

Broadcast transmission is not an effective solution for this type of application since it affects the CPU performance of each and every station that sees the packet. Besides, it wastes bandwidth.

2.2 Resource Discovery

Some applications utilize multicast instead of broadcast transmission to transmit packets to group members residing on the same subnetwork. However, there is no reason to limit the extent of a multicast transmission to a single LAN. The time-to-live (TTL) field in the IP header can be used to limit the range (or "scope") of a multicast transmission. "Expanding ring searches" are one often-cited generic multicast application, and they will be discussed in detail once the mechanisms used by each multicast routing protocol have been described.

2.3 Support for Datacasting Applications

Since 1992, the IETF has conducted a series of "audiocast" experiments in which live audio and video were multicast from the IETF meeting site to destinations around the world. In this case, "datacasting" takes compressed audio and video signals from the source station and transmits them as a sequence of UDP packets to a group address. Multicast delivery today is not limited to audio and video. Stock quote systems

are one example of a (connectionless) data-oriented multicast

application. Someday reliable multicast transport protocols may facilitate efficient inter-computer communication. Reliable multicast transport protocols are currently an active area of research and development.

3. THE INTERNET'S MULTICAST BACKBONE (MBone)

The Internet Multicast Backbone (MBone) is an interconnected set of subnetworks and routers that support the delivery of IP multicast traffic. The goal of the MBone is to construct a semipermanent IP multicast testbed to enable the deployment of multicast applications without waiting for the ubiquitous deployment of multicast-capable routers in the Internet.

The MBone has grown from 40 subnets in four different countries in 1992, to more than 4300 subnets worldwide by July 1997. With new multicast applications and multicast-based services appearing, it seems likely that the use of multicast technology in the Internet will keep growing at an ever-increasing rate.

The MBone is a virtual network that is layered on top of sections of the physical Internet. It is composed of islands of multicast routing capability connected to other islands, or "regions," by virtual point-to-point links called "tunnels." The tunnels allow multicast traffic to pass through the non-multicast-capable parts of the Internet. Tunneled IP multicast packets are encapsulated as IP-over-IP (i.e., the protocol number is set to 4) so they are seen as regular unicast packets to the intervening routers. The encapsulating IP header is added on entry to a tunnel and stripped off on exit. This set of multicast routers, their directly-connected subnetworks, and the interconnecting tunnels comprise the MBone.

Since the MBone and the Internet have different topologies, multicast routers execute a separate routing protocol to decide how to forward multicast packets. In some cases, this means that they include their own internal unicast routing protocol, but in other cases the multicast routing protocol relies on the routing table provided by the underlying unicast routing protocols.

The majority of the MBone regions are currently interconnected by the Distance Vector Multicast Routing Protocol (DVMRP). Internally, the regions may execute any routing protocol they choose, i.e., Multicast extensions to OSPF (MOSPF), or the Protocol-Independent Multicast (PIM) routing protocol(s), or the DVMRP.

As multicast routing software features become more widely available on the routers of the Internet, providers may gradually decide to use "native" multicast as an alternative to using lots of tunnels. "Native"

multicast happens when a region of routers (or set of regions) operates

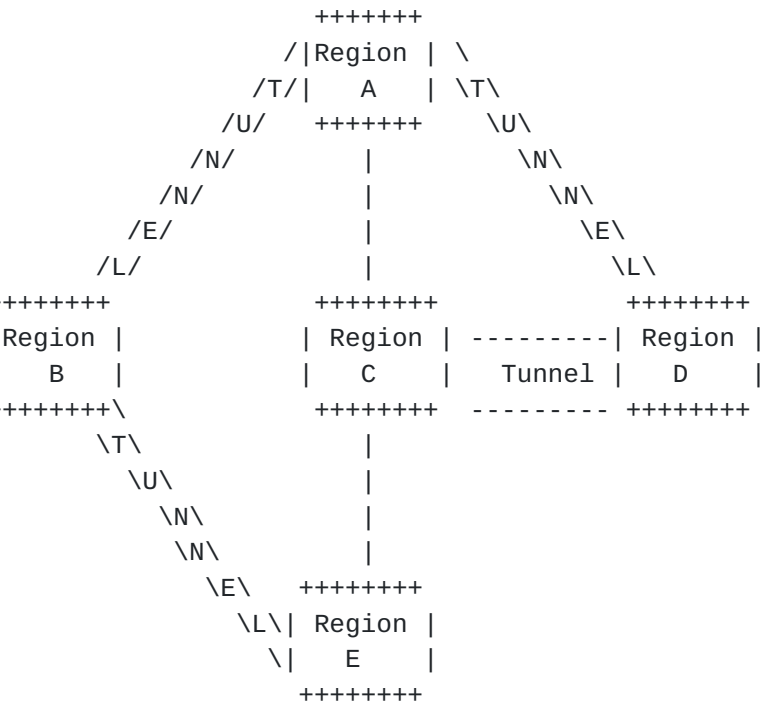


Figure 2: Internet Multicast Backbone (MBone)

without tunnels. All subnetworks are connected by at least one router which is capable of forwarding their multicast packets as required. In this case, tunnels are not required: Multicast packets just flow where they need to.

The MBone carries audio and video multicasts of Internet Engineering Task Force (IETF) meetings, NASA Space Shuttle Missions, US House and Senate sessions, and other content. There are public and private sessions on the MBone. Sessions that are meant for public consumption are announced via the session directory (SDR) tool. Users of this tool see a list of current and future public sessions, provided they are within the sender's "scope."

4. MULTICAST ADDRESSING

A multicast address is assigned to a set of Internet hosts comprising a multicast group. Senders use the multicast address as the destination IP address of a packet that is to be transmitted to all group members.

4.1 Class D Addresses

An IP multicast group is identified by a Class D address. Class D addresses have their high-order four bits set to "1110" followed by a 28-bit multicast group ID. Expressed in standard "dotted-decimal" notation, multicast group addresses range from 224.0.0.0 to 239.255.255.255 **(shorthand:** 224.0.0.0/4).

Figure 3 shows the format of a 32-bit Class D address:

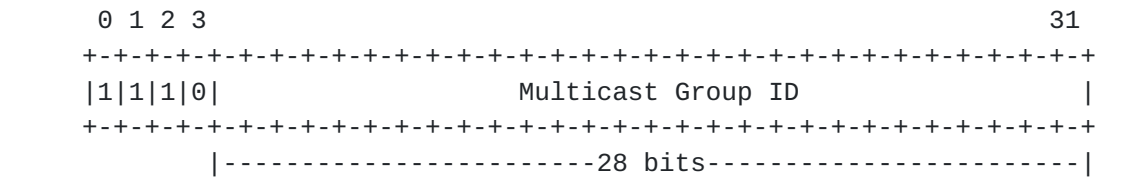


Figure 3: Class D Multicast Address Format

The Internet Assigned Numbers Authority (IANA) maintains a list of registered IP multicast groups. The base address 224.0.0.0 is reserved and cannot be assigned to any group. The block of multicast addresses ranging from 224.0.0.1 to 224.0.0.255 is reserved for permanent assignment to various uses, including routing protocols and other protocols that require a well-known permanent address. Multicast routers should not forward any multicast datagram with destination addresses in this range, (regardless of the packet's TTL).

Some of the well-known groups include:

"all systems on this subnet"	224.0.0.1
"all routers on this subnet"	224.0.0.2
"all DVMRP routers"	224.0.0.4
"all OSPF routers"	224.0.0.5
"all OSPF designated routers"	224.0.0.6
"all RIP2 routers"	224.0.0.9
"all PIM routers"	224.0.0.13
"all CBT routers"	224.0.0.15

The remaining groups ranging from 224.0.1.0 to 239.255.255.255 are either permanently assigned to various multicast applications or are available for dynamic assignment (via SDR or other methods). From this range, the addresses from 239.0.0.0 to 239.255.255.255 are reserved for various "administratively scoped" applications, within private networks,

not necessarily Internet-wide applications.

The complete list may be found in the Assigned Numbers RFC (RFC 1700 or its successor) or at the assignments page of the IANA Web Site:

<URL:http://www.iana.org/iana/assignments.html>

4.2 Mapping a Class D Address to an IEEE-802 MAC Address

The IANA has been allocated a reserved portion of the IEEE-802 MAC-layer multicast address space. All of the addresses in IANA's reserved block begin with 01-00-5E (hex); to be clear, the range from 01-00-5E-00-00-00 to 01-00-5E-FF-FF-FF is used for IP multicast groups.

A simple procedure was developed to map Class D addresses to this reserved MAC-layer multicast address block. This allows IP multicasting to easily take advantage of the hardware-level multicasting supported by network interface cards.

The mapping between a Class D IP address and an IEEE-802 (e.g., FDDI, Ethernet) MAC-layer multicast address is obtained by placing the low-order 23 bits of the Class D address into the low-order 23 bits of IANA's reserved MAC-layer multicast address block. This simple procedure removes the need for an explicit protocol for multicast address resolution on LANs akin to ARP for unicast. All LAN stations know this simple transformation, and can easily send any IP multicast over any IEEE-802-based LAN.

Figure 4 illustrates how the multicast group address 234.138.8.5 (or EA-8A-08-05 expressed in hex) is mapped into an IEEE-802 multicast address. Note that the high-order nine bits of the IP address are not mapped into the MAC-layer multicast address.

The mapping in Figure 4 places the low-order 23 bits of the IP multicast group ID into the low order 23 bits of the IEEE-802 multicast address. Because the class D space has more groups (2^28) than the IETF's OUI may contain at the MAC layer (2^23), multiple group addresses map to each IEEE-802 address. To be precise, 32 class D addresses map to each MAC-layer multicast addresses. If two or more groups on a LAN have, by some astronomical coincidence, chosen class D addresses which map to the same MAC-layer multicast address, then the member hosts will receive traffic for all those groups. It will then be up to their IP-layer software to interpret which packets are for the group(s) to which they really belong. The class D addresses 224.10.8.5 (E0-0A-08-05) and 225.138.8.5 (E1-8A-08-05) are shown to map to the same IEEE-802 MAC-layer multicast address (01-00-5E-0A-08-05).

4.3 Transmission and Delivery of Multicast Datagrams

When the sender and receivers are members of the same (LAN) subnetwork, the transmission and reception of multicast frames is a straightforward

process. The source station simply addresses the IP packet to the multicast group, the network interface card maps the Class D address to

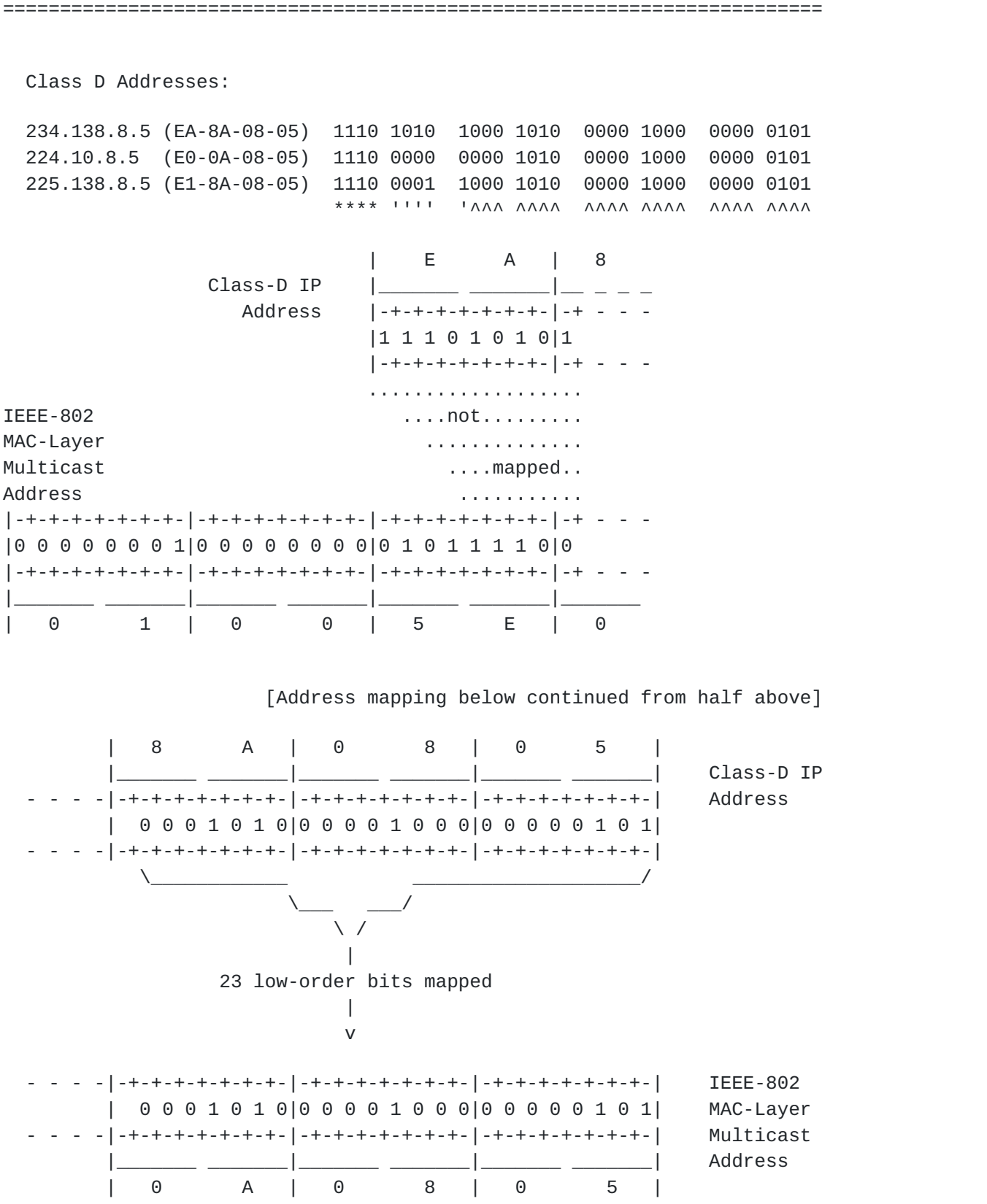


Figure 4: Mapping between Class D and IEEE-802 Multicast Addresses

the corresponding IEEE-802 multicast address, and the frame is sent. Internet hosts that need to receive selected multicast frames, whether because a user has executed a multicast application, or because the host's IP stack is required to receive certain groups (e.g., 224.0.0.1, the "all-hosts" group), notify their driver software of which group addresses to filter.

Things become somewhat more complex when the sender is attached to one subnetwork and receivers reside on different subnetworks. In this case, the routers must implement a multicast routing protocol that permits the construction of multicast delivery trees and supports multicast packet forwarding. In addition, each router needs to implement a group membership protocol that allows it to learn about the existence of group members on its directly attached subnetworks.

5. INTERNET GROUP MANAGEMENT PROTOCOL (IGMP)

The Internet Group Management Protocol (IGMP) runs between hosts and their immediately-neighboring multicast routers. The mechanisms of the protocol allow a host to inform its local router that it wishes to receive transmissions addressed to a specific multicast group. Also, routers periodically query the LAN to determine if any group members are still active. If there is more than one IP multicast router on the LAN, one of the routers is elected "querier" and assumes the responsibility of querying the LAN for the presence of any group members.

Based on the group membership information learned from the IGMP, a router is able to determine which (if any) multicast traffic needs to be forwarded to each of its "leaf" subnetworks. "Leaf" subnetworks are those that have no further downstream routers; they either contain receivers for some set of groups, or they do not. Multicast routers use the information derived from IGMP, along with a multicast routing protocol, to support IP multicasting across the MBone.

5.1 IGMP Version 1

IGMP Version 1 was specified in RFC-1112. According to the specification, multicast routers periodically transmit Host Membership Query messages to determine which host groups have members on their directly-attached networks. IGMP Query messages are addressed to the all-hosts group (224.0.0.1) and have an IP TTL = 1. This means that Query messages sourced from a router are transmitted onto the directly-attached subnetwork, but are not forwarded by any other multicast routers.

When a host receives an IGMP Query message, it responds with a Host Membership Report for each group to which it belongs, sent to each group to which it belongs. (This is an important point: IGMP Queries are

sent to the "all hosts on this subnet" class D address (224.0.0.1), but IGMP Reports are sent to the group(s) to which the host(s) belong. IGMP Reports, like Queries, are sent with the IP TTL = 1, and thus are not forwarded beyond the local subnetwork.)

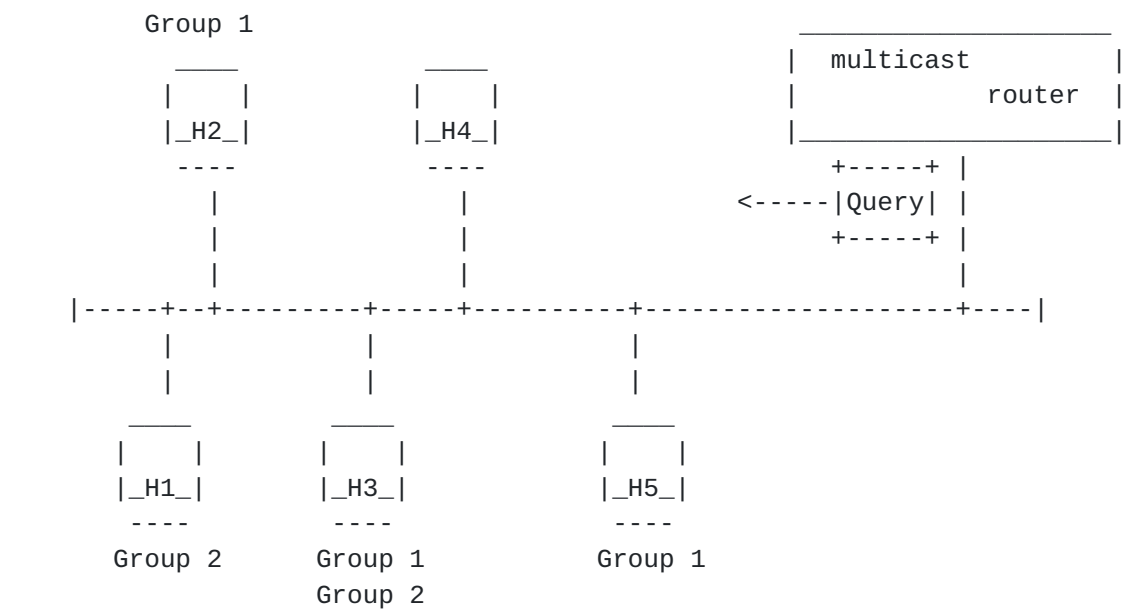


Figure 5: Internet Group Management Protocol - Query Message

In order to avoid a flurry of Reports, each host starts a randomly-chosen Report delay timer for each of its group memberships. If, during the delay period, another Report is heard for the same group, every other host in that group must reset its timer to a new random value. This procedure spreads Reports out over a period of time and thus minimizes Report traffic for each group that has at least one member on a given subnetwork.

It should be noted that multicast routers do not need to be directly addressed since their interfaces are required to promiscuously receive all multicast IP traffic. Also, a router does not need to maintain a detailed list of which hosts belong to each multicast group; the router only needs to know that at least one group member is present on a given network interface.

Multicast routers periodically transmit IGMP Queries to update their knowledge of the group members present on each network interface. If the router does not receive a Report from any members of a particular

group after a number of Queries, the router assumes that group members

are no longer present on an interface. Assuming this is a leaf subnet (i.e., a subnet with group members but no multicast routers connecting to additional group members further downstream), this interface is removed from the delivery tree(s) for this group. Multicasts will continue to be sent on this interface only if the router can tell (via multicast routing protocols) that there are additional group members further downstream reachable via this interface.

When a host first joins a group, it immediately transmits an IGMP Report for the group rather than waiting for a router's IGMP Query. This reduces the "join latency" for the first host to join a given group on a particular subnetwork. "Join latency" is measured from the time when a host's first IGMP Report is sent, until the first packet for that group arrives on that host's subnetwork. Of course, if the group is already active, the join latency is negligible.

5.2 IGMP Version 2

IGMP version 2 was distributed as part of the Distance Vector Multicast Routing Protocol (DVMRP) implementation ("mrouted") source code, from version 3.3 through 3.8. Initially, there was no detailed specification for IGMP version 2 other than this source code. However, the complete specification has recently been published in <draft-ietf-idmr-igmp-v2-06.txt>, a work-in-progress which will update the specification contained in the first appendix of RFC-1112. IGMP version 2 extends IGMP version 1 while maintaining backward compatibility with version 1 hosts.

IGMP version 2 defines a procedure for the election of the multicast querier for each LAN. In IGMP version 2, the multicast router with the lowest IP address on the LAN is elected the multicast querier. In IGMP version 1, the querier election was determined by the multicast routing protocol.

IGMP version 2 defines a new type of Query message: the Group-Specific Query. Group-Specific Query messages allow a router to transmit a Query to a specific multicast group rather than all groups residing on a directly attached subnetwork.

Finally, IGMP version 2 defines a Leave Group message to lower IGMP's "leave latency." When a host wishes to leave that specific group, the host transmits an IGMPv2 "Leave Group" message to the all-routers group (224.0.0.2), with the group field set to the group being left. After receiving a Leave Group message, the IGMPv2-elected querier must find out if this was the last member of this group on this subnetwork. To do this, the router begins transmitting Group-Specific Query messages on the interface that received the Leave Group message. If it hears no

Reports in response to the Group-Specific Query messages, then (if this is a leaf subnet) this interface is removed from the delivery tree(s) for this group (as was the case of IGMP version 1). Even if there are

no group members on this subnetwork, multicasts must continue to be sent onto it if the router can tell that additional group members are further away from the source (i.e., "downstream") that are reachable via other routers attached to this subnetwork.

"Leave latency" is measured from a router's perspective. In version 1 of IGMP, leave latency was the time from a router's hearing the last Report for a given group, until the router aged out that interface from the delivery tree for that group (assuming this is a leaf subnet, of course). Note that the only way for the router to tell that this was the LAST group member is that no reports are heard in some multiple of the Query Interval (this is on the order of minutes). IGMP version 2, with the addition of the Leave Group message, allows a group member to more quickly inform the router that it is done receiving traffic for a group. The router then must determine if this host was the last member of this group on this subnetwork. To do this, the router quickly queries the subnetwork for other group members via the Group-Specific Query message. If no members send reports after several of these Group-Specific Queries, the router can infer that the last member of that group has, indeed, left the subnetwork.

The benefit of lowering the leave latency is that the router can quickly use its multicast routing protocol to inform its "upstream" neighbors ("upstream" is the direction that a router considers to be toward a source), allowing the delivery tree for this group to quickly adapt to the new shape of the group (without this subnetwork and the branch that lead to it). The alternative was having to wait for several rounds of unanswered Queries to pass (on the order of minutes). If a group was experiencing high traffic levels, it can be very beneficial to stop transmitting data for this group as soon as possible.

5.3 IGMP Version 3 (Future)

IGMP version 3 is a preliminary work-in-progress published in <draft-cain-igmp-00.txt>. IGMP version 3 as it is currently defined will introduce support for Group-Source Report messages so that a host may elect to receive traffic only from specific multicast within a group. An Inclusion Group-Source Report message will allow a host to specify the IP address(es) of the specific sources it wants to receive, and an Exclusion Group-Source Report message will allow a host to explicitly ask that traffic from some (list of) sources be blocked. With IGMP versions 1 and 2, if a host wants to receive any traffic for a group, then traffic from all the group's sources will be forwarded onto the host's subnetwork.

IGMP version 3 will help conserve bandwidth by allowing a host to select only the specific sources from which it wants to receive traffic. Also, multicast routing protocols will be able to use this information to help

conserve bandwidth when constructing the branches of their multicast delivery trees.

Finally, support for Leave Group messages, first introduced in IGMPv2, has been enhanced to support Group-Source Leave messages. This feature will allow a host to leave an entire group, or to specify the specific IP address(es) of the (source, group) pair(s) that it wishes to leave. Note that at this time, not all existing multicast routing protocols can support such requests. This is one issue that needs to be addressed during the development of IGMP version 3.

6. MULTICAST FORWARDING TECHNIQUES

IGMP provides the final step in a multicast packet delivery service since it is only concerned with the forwarding of multicast traffic from a router to group members on its directly-attached subnetworks. IGMP is not concerned with the delivery of multicast packets between neighboring routers or across an internetwork.

To provide an internetwork delivery service, it is necessary to define multicast routing protocols. A multicast routing protocol is responsible for the construction of multicast delivery trees and enabling multicast packet forwarding. This section explores a number of different techniques that may potentially be employed by multicast routing protocols:

- o "Simpleminded" Techniques
  - Flooding
  - Multicast Extensions to MAC-layer Spanning Trees
- o Source-Based Tree (SBT) Techniques
  - Reverse Path Broadcasting (RPB)
  - Truncated Reverse Path Broadcasting (TRPB)
  - Reverse Path Multicasting (RPM)
- o "Shared-Tree" Techniques

Later sections will describe how these algorithms are implemented in the most prevalent multicast routing protocols in the Internet today (e.g., Distance Vector Multicast Routing Protocol (DVMRP), Multicast extensions to OSPF (MOSPF), Protocol-Independent Multicast (PIM), and Core-Based Trees (CBT).

6.1 "Simpleminded" Techniques

Flooding and Multicast Extensions to MAC-layer Spanning Trees are two algorithms that could be used to build primitive multicast routing protocols. The techniques are primitive because they tend to waste bandwidth or require a large amount of computational resources within the participating multicast routers. Also, protocols built on these techniques may work for small networks with few senders, groups, and

routers, but do not scale well to larger numbers of senders, groups, or

routers. Finally, the ability to handle arbitrary topologies may be absent, or may only be present in limited ways.

6.1.1 Flooding

The simplest technique for delivering multicast datagrams to all routers in an internetwork is to implement a flooding algorithm. The flooding procedure begins when a router receives a packet that is addressed to a multicast group. The router employs a protocol mechanism to determine whether or not it has seen this particular packet before. If it is the first reception of the packet, the packet is forwarded on all interfaces (except the one on which it arrived) guaranteeing that the multicast packet reaches all routers in the internetwork. If the router has seen the packet before, then the packet is discarded.

A flooding algorithm is very simple to implement since a router does not have to maintain a routing table and only needs to keep track of the most recently seen packets. However, flooding does not scale for Internet-wide applications since it generates a large number of duplicate packets and uses all available paths across the internetwork instead of just a limited number. Also, the flooding algorithm makes inefficient use of router memory resources since each router is required to maintain a distinct table entry for each recently seen packet.

6.1.2 Multicast Extensions to MAC-layer Spanning Trees

A more effective solution than flooding would be to select a subset of the internetwork topology which forms a spanning tree. The spanning tree defines a structure in which only one active path connects any two routers of the internetwork. Figure 6 shows an internetwork and a spanning tree rooted at router RR.

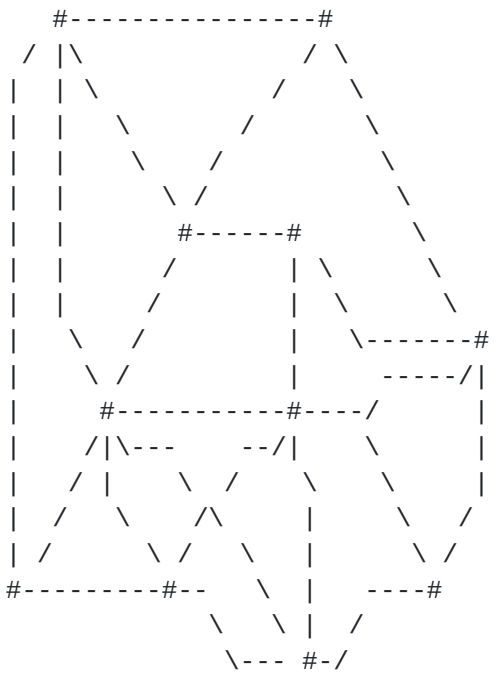
Once the spanning tree has been built, a multicast router simply forwards each multicast packet to all interfaces that are part of the spanning tree except the one on which the packet originally arrived. Forwarding along the branches of a spanning tree guarantees that the multicast packet will not loop and that it will eventually reach all routers in the internetwork.

A spanning tree solution is powerful and would be relatively easy to implement since there is a great deal of experience with spanning tree protocols in the Internet community. However, a spanning tree solution can centralize traffic on a small number of links, and may not provide the most efficient path between the source subnetwork and group members. Also, it is computationally difficult to compute a spanning tree in large, complex topologies.

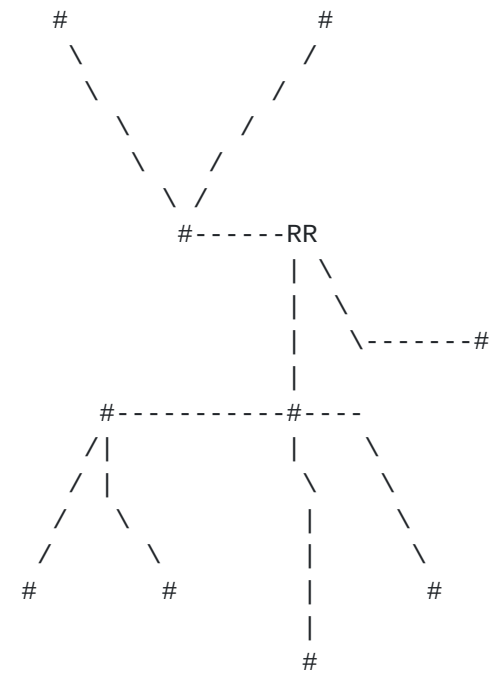
[This space was intentionally left blank.]

==========

A Sample Internetwork:



One Possible Spanning Tree for this Sample Internetwork:



LEGEND

Router
RR Root Router

Figure 6: Spanning Tree

6.2 Source-Based Tree Techniques

The following techniques all generate a source-based tree by various means. The techniques differ in the efficiency of the tree building process, and the bandwidth and router resources (i.e., state tables) used to build a source-based tree.

6.2.1 Reverse Path Broadcasting (RPB)

A more efficient solution than building a single spanning tree for the entire internetwork would be to build a spanning tree for each potential source [subnetwork]. These spanning trees would result in source-based delivery trees emanating from the subnetworks directly connected to the source stations. Since there are many potential sources for a group, a different delivery tree is constructed rooted at each active source.

6.2.1.1 Reverse Path Broadcasting: Operation

The fundamental algorithm to construct these source-based trees is referred to as Reverse Path Broadcasting (RPB). The RPB algorithm is actually quite simple. For each source, if a packet arrives on a link that the local router believes to be on the shortest path back toward the packet's source, then the router forwards the packet on all interfaces except the incoming interface. If the packet does not arrive on the interface that is on the shortest path back toward the source, then the packet is discarded. The interface over which the router expects to receive multicast packets from a particular source is referred to as the "parent" link. The outbound links over which the router forwards the multicast packet are called "child" links for this source.

This basic algorithm can be enhanced to reduce unnecessary packet duplication. If the local router making the forwarding decision can determine whether a neighboring router on a child link is "downstream," then the packet is multicast toward the neighbor. (A "downstream" neighbor is a neighboring router which considers the local router to be on the shortest path back toward a given source.) Otherwise, the packet is not forwarded on the potential child link since the local router knows that the neighboring router will just discard the packet (since it will arrive on a non-parent link for the source, relative to that downstream router).

The information to make this "downstream" decision is relatively easy to derive from a link-state routing protocol since each router maintains a topological database for the entire routing domain. If a distance-vector routing protocol is employed, a neighbor can either advertise its previous hop for the source as part of its routing update messages or "poison reverse" the route toward a source if it is not on the delivery

tree for that source. Either of these techniques allows an upstream router to determine if a downstream neighboring router is on an active branch of the distribution tree for a certain source.

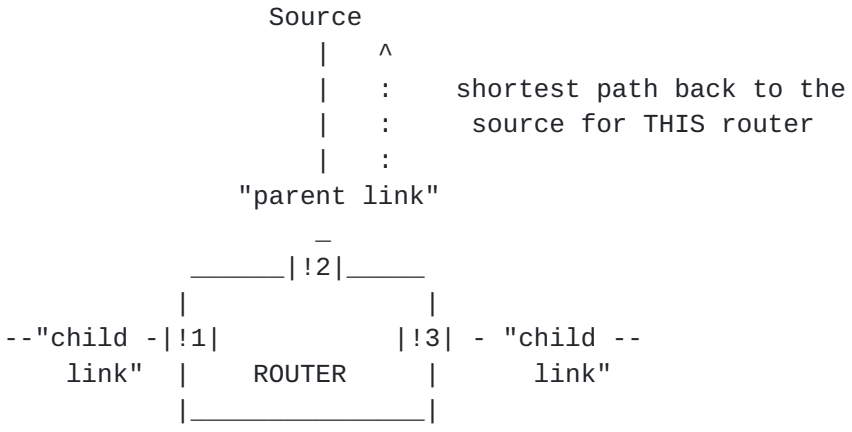


Figure 7: Reverse Path Broadcasting - Forwarding Algorithm

Note that the source station (S) is attached to a leaf subnetwork directly connected to Router A. For this example, we will look at the RPB algorithm from Router B's perspective. Router B receives the multicast packet from Router A on link 1. Since Router B considers link 1 **to be the parent link for the (source, group) pair, it forwards the** packet on link 4, link 5, and the local leaf subnetworks if they contain group members. Router B does not forward the packet on link 3 because it knows from routing protocol exchanges that Router C considers link 2 as its parent link for the source. Router B knows that if it were to forward the packet on link 3, it would be discarded by Router C since the packet would not be arriving on Router C's parent link for this source.

Figure 8 illustrates the preceding discussion of the enhanced RPB algorithm's basic operation.

6.2.1.2 RPB: Benefits and Limitations

The key benefit to reverse path broadcasting is that it is reasonably efficient and easy to implement. It does not require that the router know about the entire spanning tree, nor does it require a special mechanism to stop the forwarding process (as flooding does). In addition, it guarantees efficient delivery since multicast packets always follow the "shortest" path from the source station to the destination group. Finally, the packets are distributed over multiple links, resulting in better network utilization since a different tree is computed for each source.

One of the major limitations of the RPB algorithm is that it does not

take into account multicast group membership when building the delivery

tree for a source. As a result, extraneous datagrams may be forwarded onto subnetworks that have no group members.

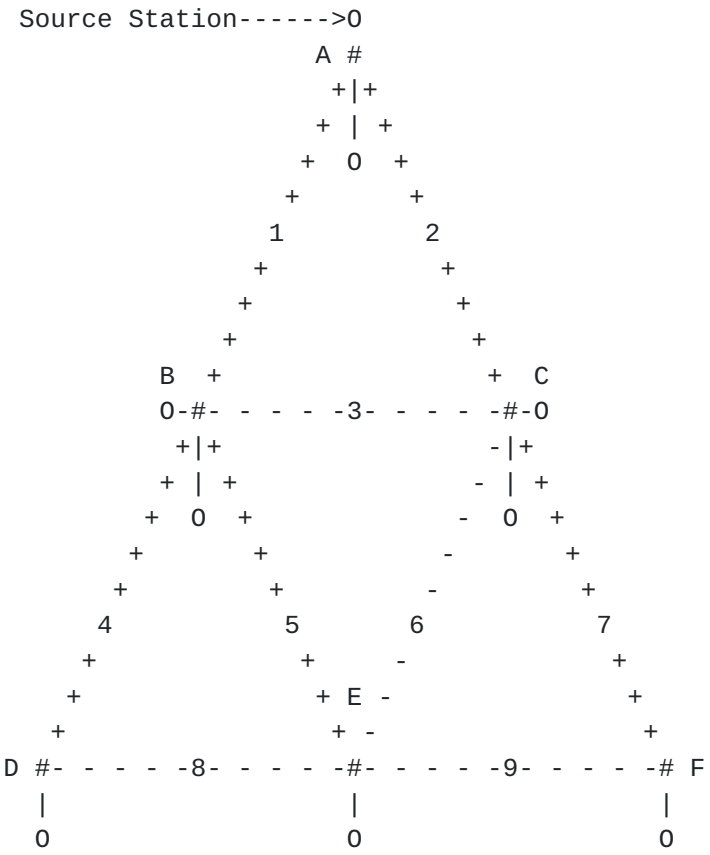


Figure 8: Reverse Path Broadcasting - Example

6.2.2 Truncated Reverse Path Broadcasting (TRPB)

Truncated Reverse Path Broadcasting (TRPB) was developed to overcome the limitations of Reverse Path Broadcasting. With information provided by IGMP, multicast routers determine the group memberships on each leaf subnetwork and avoid forwarding datagrams onto a leaf subnetwork if it does not contain at least one member of a given destination group. Thus, the delivery tree is "truncated" by the router if a leaf subnetwork has

no group members.

Figure 9 illustrates the operation of TRPB algorithm. In this example the router receives a multicast packet on its parent link for the Source. The router forwards the datagram on interface 1 since that interface has at least one member of G1. The router does not forward the datagram to interface 3 since this interface has no members in the destination group. The datagram is forwarded on interface 4 if and only if a downstream router considers this subnetwork to be part of its "parent link" for the Source.

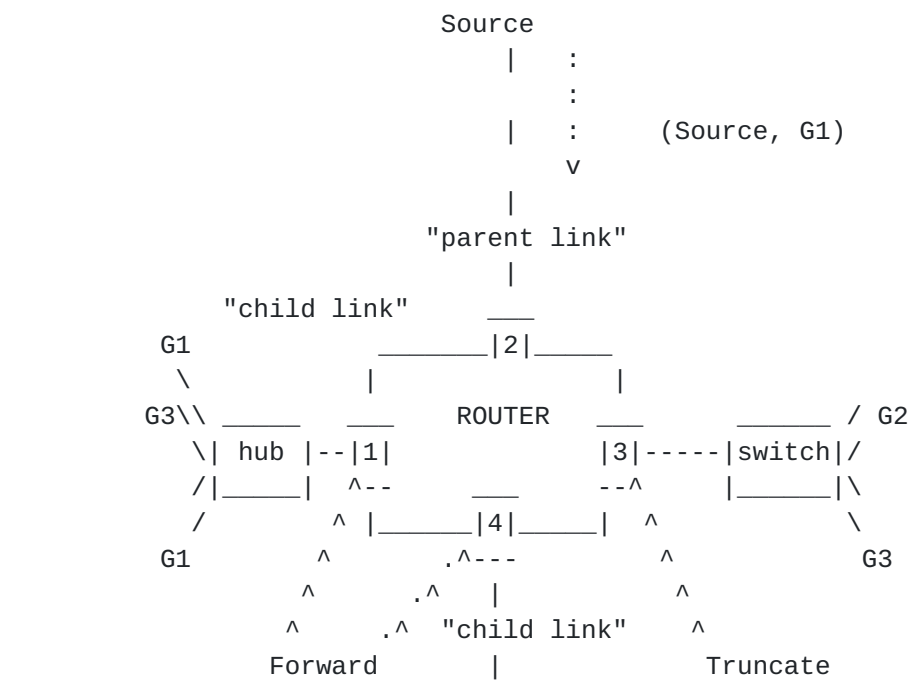


Figure 9: Truncated Reverse Path Broadcasting - (TRPB)

TRPB removes some limitations of RPB but it solves only part of the problem. It eliminates extraneous traffic on leaf subnetworks, but it does not consider group memberships when building the branches of the delivery tree.

6.2.3 Reverse Path Multicasting (RPM)

Reverse Path Multicasting (RPM) is an enhancement to Reverse Path Broadcasting and Truncated Reverse Path Broadcasting.

RPM creates a delivery tree that spans only 1) subnetworks with group members, and 2) routers and subnetworks along the shortest path to those subnetworks. RPM allows the source-based "shortest-path" tree to be

"pruned" so that datagrams are only forwarded along branches that lead to active members of the destination group.

6.2.3.1 Operation

When a multicast router receives a packet for a (source, group) pair, the first packet is forwarded following the TRPB algorithm across all routers in the internetwork. Routers on the edge of the network (which have only leaf subnetworks) are called leaf routers. The TRPB algorithm guarantees that each leaf router will receive at least the first multicast packet. If there is a group member on one of its leaf subnetworks, a leaf router forwards the packet based on this group membership information.

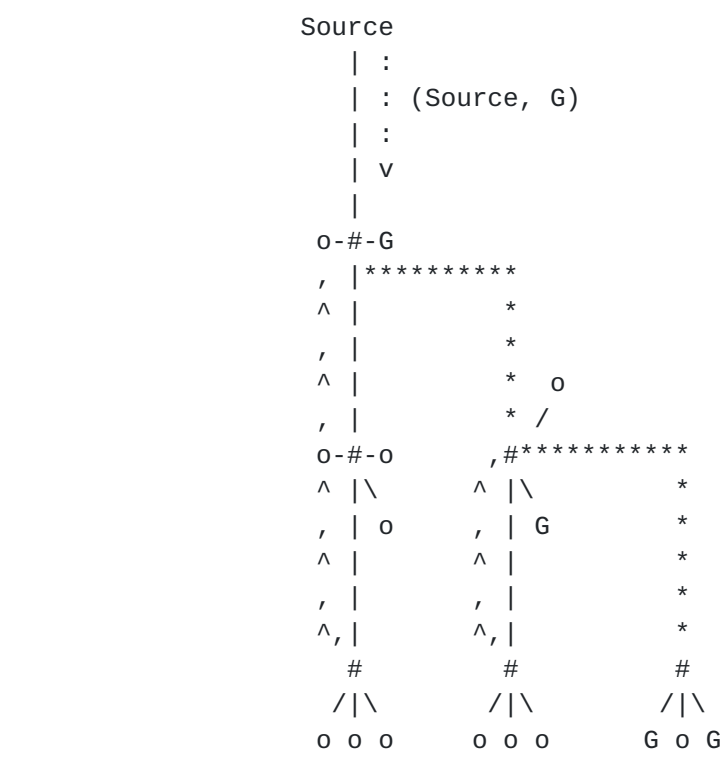


Figure 10: Reverse Path Multicasting (RPM)

If none of the subnetworks connected to the leaf router contain group

members, the leaf router may transmit a "prune" message on its parent link, informing the upstream router that it should not forward packets

for this particular (source, group) pair on the child interface on which it received the prune message. Prune messages are sent just one hop back toward the source.

An upstream router receiving a prune message is required to store the prune information in memory. If the upstream router has no recipients on local leaf subnetworks and has received prune messages from each downstream neighbor on each of the child interfaces for this (source, group) pair, then the upstream router does not need to receive any more packets for this (source, group) pair. Therefore, the upstream router can also generate a prune message of its own, one hop further back toward the source. This cascade of prune messages results in an active multicast delivery tree, consisting exclusively of "live" branches (i.e., branches that lead to active receivers).

Since both the group membership and internetwork topology can change dynamically, the pruned state of the multicast delivery tree must be refreshed periodically. At regular intervals, the prune information expires from the memory of all routers and the next packet for the (source, group) pair is forwarded toward all downstream routers. This allows "stale state" (prune state for groups that are no longer active) to be reclaimed by the multicast routers.

6.2.3.2 Limitations

Despite the improvements offered by the RPM algorithm, there are still several scaling issues that need to be addressed when attempting to develop an Internet-wide delivery service. The first limitation is that multicast packets must be periodically broadcast across every router in the internetwork, onto every leaf subnetwork. This "broadcasting" is wasteful of bandwidth (until the updated prune state is constructed).

This "broadcast and prune" paradigm is very powerful, but it wastes bandwidth and does not scale well, especially if there are receivers at the edge of the delivery tree which are connected via low-speed technologies (e.g., ISDN or modem). Also, note that every router participating in the RPM algorithm must either have a forwarding table entry for a (source, group) pair, or have prune state information for that (source, group) pair.

It is clearly wasteful (especially as the number of active sources and groups increase) to place such a burden on routers that are not on every (or perhaps any) active delivery tree. Shared tree techniques are an attempt to address these scaling issues, which become quite acute when most groups' senders and receivers are sparsely distributed across the internetwork.

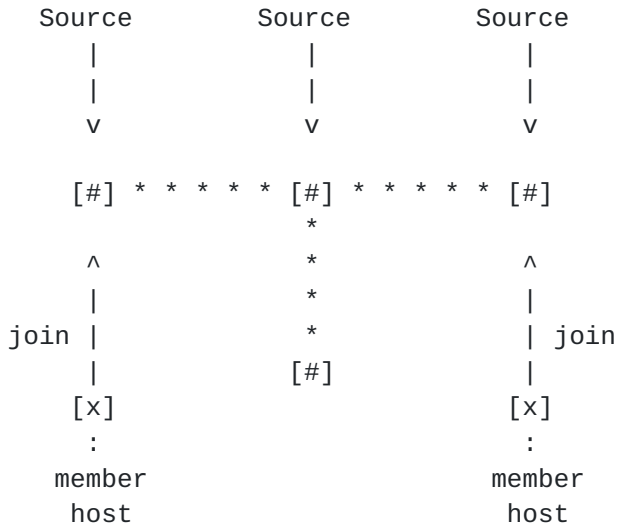
6.3 Shared Tree Techniques

The most recent additions to the set of multicast forwarding techniques are based on a shared delivery tree. Unlike shortest-path tree

algorithms which build a source-based tree for each source, or each (source, group) pair, shared tree algorithms construct a single delivery tree that is shared by all members of a group. The shared tree approach is quite similar to the spanning tree algorithm except it allows the definition of a different shared tree for each group. Stations wishing to receive traffic for a multicast group must explicitly join the shared delivery tree. Multicast traffic for each group is sent and received over the same delivery tree, regardless of the source.

6.3.1 Operation

A shared tree may involve a single router, or set of routers, which comprise(s) the "core" of a multicast delivery tree. Figure 11 illustrates how a single multicast delivery tree is shared by all sources and receivers for a multicast group.



LEGEND

[#] Shared Tree "Core" Routers
* * Shared Tree Backbone
[x] Member-hosts' directly-attached routers

Figure 11: Shared Multicast Delivery Tree

Similar to other multicast forwarding algorithms, shared tree algorithms do not require that the source of a multicast packet be a member of a

destination group in order to send to a group.

6.3.2 Benefits

In terms of scalability, shared tree techniques have several advantages over source-based trees. Shared tree algorithms make efficient use of router resources since they only require a router to maintain state information for each group, not for each source, or for each (source, group) pair. (Remember that source-based tree techniques required all routers in an internetwork to either a) be on the delivery tree for a given source or (source, group) pair, or b) to have prune state for that source or (source, group) pair: So the entire internetwork must participate in the source-based tree protocol.) This improves the scalability of applications with many active senders since the number of source stations is no longer a scaling issue. Also, shared tree algorithms conserve network bandwidth since they do not require that multicast packets be periodically flooded across all multicast routers in the internetwork onto every leaf subnetwork. This can offer significant bandwidth savings, especially across low-bandwidth WAN links, and when receivers sparsely populate the domain of operation. Finally, since receivers are required to explicitly join the shared delivery tree, data only ever flows over those links that lead to active receivers.

6.3.3 Limitations

Despite these benefits, there are still several limitations to protocols that are based on a shared tree algorithm. Shared trees may result in traffic concentration and bottlenecks near core routers since traffic from all sources traverses the same set of links as it approaches the core. In addition, a single shared delivery tree may create suboptimal routes (a shortest path between the source and the shared tree, a suboptimal path across the shared tree, a shortest path between the group's core router and the receiver's directly-attached router), resulting in increased delay which may be a critical issue for some multimedia applications. (Simulations indicate that latency over a shared tree may be approximately 10% larger than source-based trees in many cases, but by the same token, this may be negligible for many applications.)

Finally, expanding-ring searches will probably not work as expected inside shared-tree domains. The searching host's increasing TTL will cause the packets to work their way up the shared tree, and while a desired resource may still be found, it may not be as topologically close as one would expect.

7. "DENSE MODE" ROUTING PROTOCOLS

Certain multicast routing protocols are designed to work well in

environments that have plentiful bandwidth and where it is reasonable to assume that receivers are rather densely distributed. In such scenarios, it is very reasonable to use periodic flooding, or other

bandwidth-intensive techniques that would not necessarily be very scalable over a wide-area network. In section 8, we will examine different protocols that are specifically geared toward efficient WAN operation, especially for groups that have widely dispersed (i.e., sparse) membership.

So-called "dense"-mode routing protocols include:

o the Distance Vector Multicast Routing Protocol (DVMRP),

o Multicast Extensions to Open Shortest Path First (MOSPF), and

o Protocol Independent Multicast - Dense Mode (PIM-DM).

These protocols' underlying designs assume that the amount of protocol overhead (in terms of the amount of state that must be maintained by each router, the number of router CPU cycles required, and the amount of bandwidth consumed by protocol operation) is appropriate since receivers densely populate the area of operation.

7.1. Distance Vector Multicast Routing Protocol (DVMRP)

The Distance Vector Multicast Routing Protocol (DVMRP) is a distance-vector routing protocol designed to support the forwarding of multicast datagrams through an internetwork. DVMRP constructs source-based multicast delivery trees using the Reverse Path Multicasting (RPM) algorithm. Originally, the entire MBone ran only DVMRP. Today, over half of the MBone routers still run some version of DVMRP.

DVMRP was first defined in RFC-1075. The original specification was derived from the Routing Information Protocol (RIP) and employed the Truncated Reverse Path Broadcasting (TRPB) technique. The major difference between RIP and DVMRP is that RIP calculates the next-hop toward a destination, while DVMRP computes the previous-hop back toward a source. Since mrouted 3.0, DVMRP has employed the Reverse Path Multicasting (RPM) algorithm. Thus, the latest implementations of DVMRP are quite different from the original RFC specification in many regards. There is an active effort within the IETF Inter-Domain Multicast Routing (IDMR) working group to specify DVMRP version 3 in a well-documented form.

The current DVMRP v3 Internet-Draft, representing a snapshot of a work-in-progress, is:

<draft-ietf-idmr-dvmrp-v3-04.txt>, or <draft-ietf-idmr-dvmrp-v3-04.ps>

7.1.1 Physical and Tunnel Interfaces

The interfaces of a DVMRP router may be either a physical interface to

a directly-attached subnetwork or a tunnel interface (a.k.a virtual interface) to another multicast-capable region. All interfaces are

configured with a metric specifying their individual costs, and a TTL threshold which any multicast packets must exceed in order to cross this interface. In addition, each tunnel interface must be explicitly configured with two additional parameters: The IP address of the local router's tunnel interface and the IP address of the remote router's interface address.

TTL Threshold	Scope
0	Restricted to the same host
1	Restricted to the same subnetwork
15	Restricted to the same site
63	Restricted to the same region
127	Worldwide
191	Worldwide; limited bandwidth
255	Unrestricted in scope

Table 1: TTL Scope Control Values

A multicast router will only forward a multicast datagram across an interface if the TTL field in the IP header is greater than the TTL threshold assigned to the interface. Table 1 lists the conventional TTL values that are used to restrict the scope of an IP multicast. For example, a multicast datagram with a TTL of less than 16 is restricted to the same site and should not be forwarded across an interface to other sites in the same region.

TTL-based scoping is not always sufficient for all applications. Conflicts arise when trying to simultaneously enforce limits on topology, geography, and bandwidth. In particular, TTL-based scoping cannot handle overlapping regions, which is a necessary characteristic of administrative regions. In light of these issues, "administrative" scoping was created in 1994, to provide a way to do scoping based on multicast address. Certain addresses would be usable within a given administrative scope (e.g., a corporate internetwork) but would not be forwarded onto the global MBone. This allows for privacy, and address reuse within the class D address space. The range from 239.0.0.0 to **239.255.255.255 has been reserved for administrative scoping.** While administrative scoping has been in limited use since 1994 or so, it has yet to be widely deployed. The IETF MBoneD working group is working on the deployment of administrative scoping. For additional information, please see <draft-ietf-mboned-admin-ip-space-03.txt> or the successor to this work-in-progress, entitled "Administratively Scoped IP Multicast."

7.1.2 Basic Operation

DVMRP implements the Reverse Path Multicasting (RPM) algorithm. According to RPM, the first datagram for any (source, group) pair is forwarded across the entire internetwork (providing the packet's TTL and router interface thresholds permit this). Upon receiving this traffic, leaf routers may transmit prune messages back toward the source if there are no group members on their directly-attached leaf subnetworks. The prune messages remove all branches that do not lead to group members from the tree, leaving a source-based shortest path tree.

After a period of time, the prune state for each (source, group) pair expires to reclaim router memory that is being used to store prune state pertaining to groups that are no longer active. If those groups happen to still in use, a subsequent datagram for the (source, group) pair will be broadcast across all downstream routers. This will result in a new set of prune messages, serving to regenerate the (source, group) pair's source-based shortest-path tree. Current implementations of RPM (notably DVMRP) do not transmit prune messages reliably, so the prune lifetime must be kept short to compensate for potential lost prune messages. (The internet-draft of DVMRP 3.0 incorporates a mechanism to make prunes reliable.)

DVMRP also implements a mechanism to quickly "graft" back a previously pruned branch of a group's delivery tree. If a router that had sent a prune message for a (source, group) pair discovers new group members on a leaf network, it sends a graft message to the previous-hop router for this source. When an upstream router receives a graft message, it cancels out the previously-received prune message. Graft messages cascade (reliably) hop-by-hop back toward the source until they reach the nearest "live" branch point on the delivery tree. In this way, previously-pruned branches are quickly restored to a given delivery tree.

7.1.3 DVMRP Router Functions

In Figure 12, Router C is downstream and may potentially receive datagrams from the source subnetwork from Router A or Router B. If Router A's metric to the source subnetwork is less than Router B's metric, then Router A is dominant over Router B for this source.

This means that Router A will forward any traffic from the source subnetwork and Router B will discard traffic received from that source. However, if Router A's metric is equal to Router B's metric, then the router with the lower IP address on its downstream interface (child link) becomes the Dominant Router for this source. Note that on a subnetwork with multiple routers forwarding to groups with multiple sources, different routers may be dominant for each source.

7.1.4 DVMRP Routing Table

The DVMRP process periodically exchanges routing table updates with its DVMRP neighbors. These updates are independent of those generated by any unicast Interior Gateway Protocol.

Since the DVMRP was developed to route multicast and not unicast traffic, a router will probably run multiple routing processes in practice: One to support the forwarding of unicast traffic and another to support the forwarding of multicast traffic. (This can be convenient: A router can be configured to only route multicast IP, with no unicast

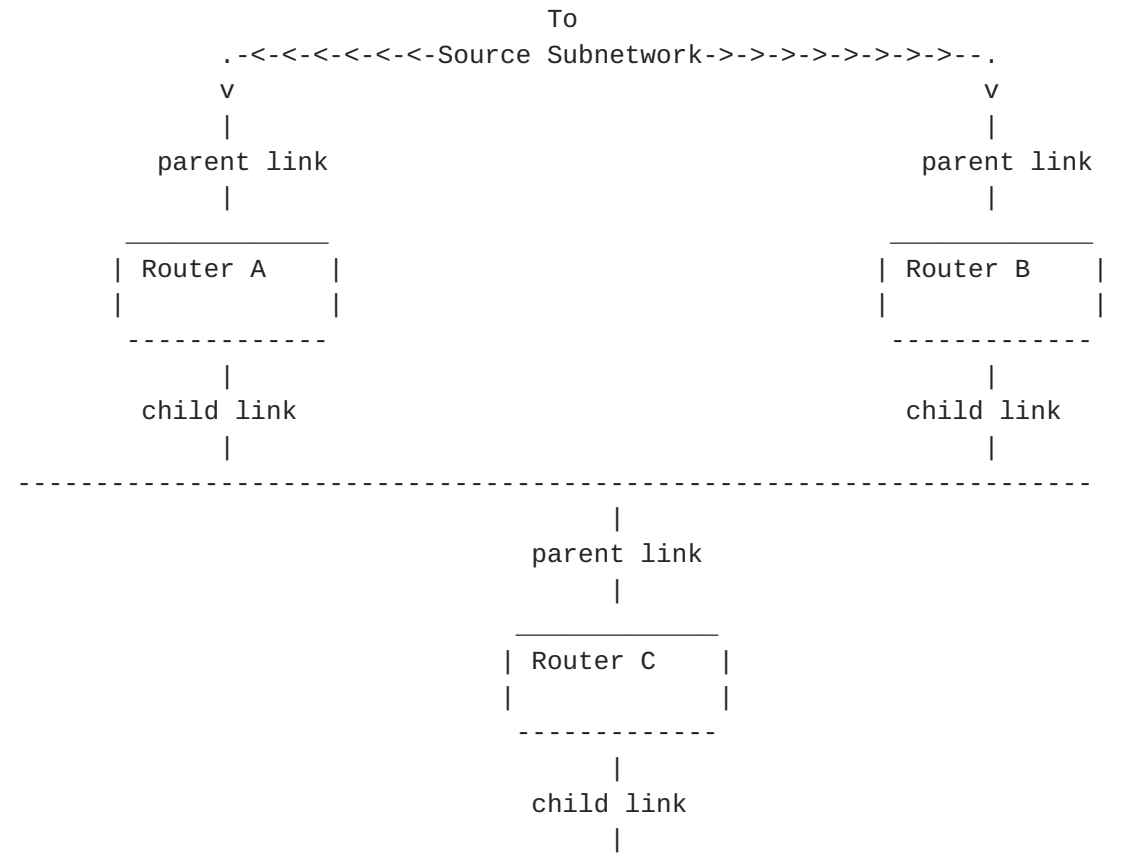


Figure 12. DVMRP Dominant Router in a Redundant Topology

IP routing. This may be a useful capability in firewalled environments.)

Again, consider Figure 12: There are two types of routers in figure 12: Dominant and Subordinate. Assume in this example that Router B is dominant, Router A is subordinate, and Router C is part of the

downstream distribution tree. In general, which routers are dominant

or subordinate may be different for each source! A subordinate router is one that is NOT on the shortest path tree back toward a source. The dominant router can tell this because the subordinate router will 'poison-reverse' the route for this source in its routing updates which are sent on the common LAN (i.e., Router A sets the metric for this source to 'infinity'). The dominant router keeps track of subordinate routers on a per-source basis...it never needs or expects to receive a prune message from a subordinate router. Only routers that are truly on the downstream distribution tree will ever need to send prunes to the dominant router. If a dominant router on a LAN has received either a poison-reversed route for a source, or prunes for all groups emanating from that source subnetwork, then it may itself send a prune upstream toward the source (assuming also that IGMP has told it that there are no local receivers for any group from this source).

Source Prefix	Subnet Mask	From Gateway	Metric	Status	TTL
128.1.0.0	255.255.0.0	128.7.5.2	3	Up	200
128.2.0.0	255.255.0.0	128.7.5.2	5	Up	150
128.3.0.0	255.255.0.0	128.6.3.1	2	Up	150
128.3.0.0	255.255.0.0	128.6.3.1	4	Up	200

Figure 13: DVMRP Routing Table

A sample routing table for a DVMRP router is shown in Figure 13. Unlike the table that would be created by a unicast routing protocol such as the RIP, OSPF, or the BGP, the DVMRP routing table contains Source Prefixes and From-Gateways instead of Destination Prefixes and Next-Hop Gateways.

The routing table represents the shortest path (source-based) spanning tree to every possible source prefix in the internetwork--the Reverse Path Broadcasting (RPB) tree. The DVMRP routing table does not represent group membership or received prune messages.

The key elements in DVMRP routing table include the following items:

Source Prefix — A subnetwork which is a potential or actual source of multicast datagrams.

Subnet Mask — The subnet mask associated with the Source Prefix. Note that the DVMRP provides the subnet mask for each source subnetwork (in other words, the DVMRP is classless).

From-Gateway	The previous-hop router leading back toward a particular Source Prefix.
TTL	The time-to-live is used for table management and indicates the number of seconds before an entry is removed from the routing table. This TTL has nothing at all to do with the TTL used in TTL-based scoping.

7.1.5 DVMRP Forwarding Table

Since the DVMRP routing table is not aware of group membership, the DVMRP process builds a forwarding table based on a combination of the information contained in the multicast routing table, known groups, and received prune messages. The forwarding table represents the local router's understanding of the shortest path source-based delivery tree for each (source, group) pair--the Reverse Path Multicasting (RPM) tree.

Source Prefix	Multicast Group	TTL	InIntf	OutIntf(s)
128.1.0.0	224.1.1.1	200	1 Pr	2p3p
	224.2.2.2	100	1	2p3
	224.3.3.3	250	1	2
128.2.0.0	224.1.1.1	150	2	2p3

Figure 14: DVMRP Forwarding Table

The forwarding table for a sample DVMRP router is shown in Figure 14. The elements in this display include the following items:

Source Prefix	The subnetwork sending multicast datagrams to the specified groups (one group per row).
Multicast Group	The Class D IP address to which multicast datagrams are addressed. Note that a given Source Prefix may contain sources for several Multicast Groups.
InIntf	The parent interface for the (source, group) pair. A 'Pr' in this column indicates that a prune message has been sent to the upstream router (the From-Gateway for this Source Prefix in the DVMRP routing table).

OutIntf(s)	The child interfaces over which multicast datagrams for this (source, group) pair are forwarded. A 'p' in this column indicates that the router has received a prune message(s) from a (all) downstream router(s) on this port.

7.1.6 DVMRP Tree Building and Forwarding Summary

As we have seen, DVMRP enables packets to be forwarded away from a multicast source along the Reverse Path Multicasting (RPM) tree. The general name for this technique is Reverse Path Forwarding, and it is used in some other multicast routing protocols, as we shall see later.

Reverse Path Forwarding was described in Steve Deering's Ph.D. thesis, and was a refinement of early work on Reverse Path Broadcasting done by Dalal and Metcalfe in 1978. In our discussion of RPB, we saw that it was wasteful in its excessive usage of network resources, because all nodes received a (single) copy of each packet. No effort was made in RPB to prune the tree to only include branches leading to active receivers. After all, RPB was a broadcast technique, not a multicast technique.

Truncated RPB added a small optimization so that IGMP was used to tell if there were listeners on the LANs at the edge of the RPB tree; if there were no listeners, the packets were stopped at the leaf routers and not forwarded onto the edge LANs. Despite the savings of LAN bandwidth, TRPB still sends a copy of each packet to every router in the topology.

Reverse Path Multicasting, used in DVMRP, takes the group membership information derived from IGMP and sends control messages upstream (i.e., toward the source subnetwork) in order to prune the distribution tree. This technique trades off some memory in the participating routers (to store the prune information) in order to gain back the wasted bandwidth on branches that do not lead to interested receivers.

In DVMRP, the RPM distribution tree is created on demand to describe the forwarding table for a given source sending to a group. As described in section 7.1.5, the forwarding table indicates the expected inbound interface for packets from this source, and the expected outbound interface(s) for distribution to the rest of the group. Forwarding table entries are created when packets to a "new" (source, group) pair arrive at a DVMRP router. As each packet is received, its source and group are matched against the appropriate row of the forwarding table. If the packet was received on the correct inbound interface, it is forwarded downstream on the appropriate outbound interfaces for this group.

DVMRP's tree-building protocol is often called "broadcast-and-prune"

because the first time a packet for a new (source, group) pair arrives, it is transmitted towards all routers in the internetwork. Then the edge routers initiate prunes. Unnecessary delivery branches are pruned

within the round-trip time from the top of a branch to the furthest leaf router, typically on the order tens of milliseconds or less, thus, the distribution tree for this new (source, group) pair is quickly trimmed to serve only active receivers.

The check DVMRP routers do when a packet arrives at the router is called the "reverse-path check." The first thing a router must do upon receipt of a multicast packet is determine that it arrived on the "correct" inbound interface. For packets matching (source, group) pairs that this router has already seen, there will already be a forwarding table entry indicating the expected incoming interface. For "new" packets, DVMRP's routing table is used to compare the actual receiving interface with the one that is considered by the router to be on the shortest path back to the source. If the reverse-path check succeeds, the packet is forwarded only on those interfaces that the router considers "child" interfaces (with respect to the source of the packet); there may be some interfaces that, for a given source, are neither child nor incoming interfaces. Child interfaces attach to subnetworks which use this router as their previous-hop on the shortest path toward the source (router addresses on this subnetwork are used as a tie-breaker when there is more than one such router).

Once the incoming interface and valid downstream child interfaces for this (source, group) are determined, a forwarding table entry is created to enable quick forwarding of future packets for this (source, group) pair. A multicast packet must never be forwarded back toward its source: This would result in a forwarding loop.

7.2. Multicast Extensions to OSPF (MOSPF)

Version 2 of the Open Shortest Path First (OSPF) routing protocol is defined in RFC-1583. OSPF is an Interior Gateway Protocol (IGP) that distributes unicast topology information among routers belonging to a single OSPF "Autonomous System." OSPF is based on link-state algorithms which permit rapid route calculation with a minimum of routing protocol traffic. In addition to efficient route calculation, OSPF is an open standard that supports hierarchical routing, load balancing, and the import of external routing information.

The Multicast Extensions to OSPF (MOSPF) are defined in RFC-1584. MOSPF routers maintain a current image of the network topology through the unicast OSPF link-state routing protocol. The multicast extensions to OSPF are built on top of OSPF Version 2 so that a multicast routing capability can be incrementally introduced into an OSPF Version 2 routing domain. Routers running MOSPF will interoperate with non-MOSPF routers when forwarding unicast IP data traffic. MOSPF does not support tunnels.

7.2.1 Intra-Area Routing with MOSPF

Intra-Area Routing describes the basic routing algorithm employed by MOSPF. This elementary algorithm runs inside a single OSPF area and supports multicast forwarding when a source and all destination group members reside in the same OSPF area, or when the entire OSPF Autonomous System is a single area (and the source is inside that area...). The following discussion assumes that the reader is familiar with OSPF.

7.2.1.1 Local Group Database

Similar to all other multicast routing protocols, MOSPF routers use the Internet Group Management Protocol (IGMP) to monitor multicast group membership on directly-attached subnetworks. MOSPF routers maintain a "local group database" which lists directly-attached groups and determines the local router's responsibility for delivering multicast datagrams to these groups.

On any given subnetwork, the transmission of IGMP Host Membership Queries is performed solely by the Designated Router (DR). However, the responsibility of listening to IGMP Host Membership Reports is performed by not only the Designated Router (DR) but also the Backup Designated Router (BDR). Therefore, in a mixed LAN containing both MOSPF and OSPF routers, an MOSPF router must be elected the DR for the subnetwork. This can be achieved by setting the OSPF RouterPriority to zero in each non-MOSPF router to prevent them from becoming the (B)DR.

The DR is responsible for communicating group membership information to all other routers in the OSPF area by flooding Group-Membership LSAs. Similar to Router-LSAs and Network-LSAs, Group-Membership LSAs are only flooded within a single area.

7.2.1.2 Datagram's Shortest Path Tree

The datagram's shortest path tree describes the path taken by a multicast datagram as it travels through the area from the source subnetwork to each of the group members' subnetworks. The shortest path tree for each (source, group) pair is built "on demand" when a router receives the first multicast datagram for a particular (source, group) pair.

When the initial datagram arrives, the source subnetwork is located in the MOSPF link state database. The MOSPF link state database is simply the standard OSPF link state database with the addition of Group-Membership LSAs. Based on the Router- and Network-LSAs in the OSPF link state database, a source-based shortest-path tree is constructed using Dijkstra's algorithm. After the tree is built, Group-Membership LSAs are used to prune the tree such that the only remaining branches lead to subnetworks containing members of this group. The output of

these algorithms is a pruned source-based tree rooted at the datagram's source (Figure 15).

To forward multicast datagrams to downstream members of a group, each router must determine its position in the datagram's shortest path tree.

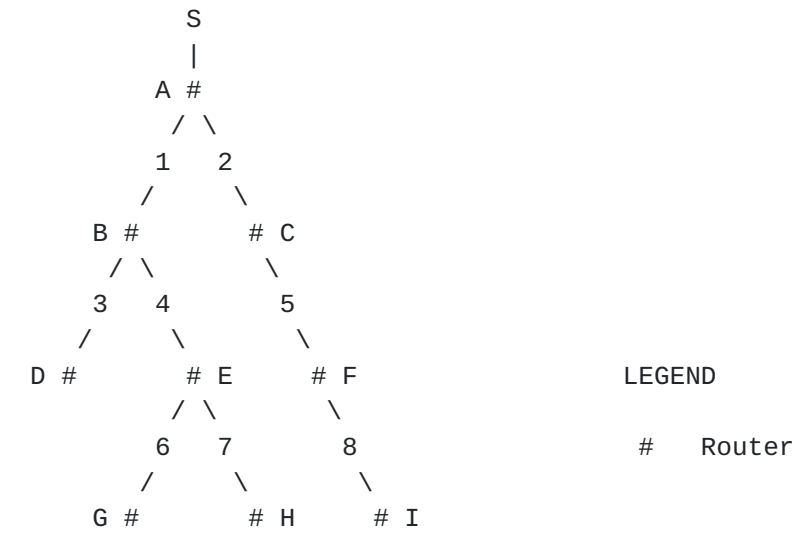


Figure 15. Shortest Path Tree for a (S, G) pair

Assume that Figure 15 illustrates the shortest path tree for a given (source, group) pair. Router E's upstream node is Router B and there are two downstream interfaces: one connecting to Subnetwork 6 and another connecting to Subnetwork 7.

Note the following properties of the basic MOSPF routing algorithm:

- For a given multicast datagram, all routers within an OSPF area calculate the same source-based shortest path delivery tree. Tie-breakers have been defined to guarantee that if several equal-cost paths exist, all routers agree on a single path through the area. Unlike unicast OSPF, MOSPF does not support the concept of equal-cost multipath routing.

- Synchronized link state databases containing Group-Membership LSAs allow an MOSPF router to build a source-based shortest-path tree in memory, working forward from the source to the group member(s). Unlike the DVMRP, this means that the first datagram of a new transmission does not have to be flooded to all routers in an area.

- The "on demand" construction of the source-based delivery tree has the benefit of spreading calculations over time, resulting

in a lesser impact for participating routers. Of course, this

may strain the CPU(s) in a router if many new (source, group) pairs appear at about the same time, or if there are a lot of events which force the MOSPF process to flush and rebuild its forwarding cache. In a stable topology with long-lived multicast sessions, these effects should be minimal.

7.2.1.3 Forwarding Cache

Each MOSPF router makes its forwarding decision based on the contents of its forwarding cache. Contrary to DVMRP, MOSPF forwarding is not RPF-based. The forwarding cache is built from the source-based shortest-path tree for each (source, group) pair, and the router's local group database. After the router discovers its position in the shortest path tree, a forwarding cache entry is created containing the (source, group) pair, its expected "upstream" incoming interface, and the necessary "downstream" outgoing interface(s). The forwarding cache entry is then used to quickly forward all subsequent datagrams from this source to this group. If a new source begins sending to a new group, MOSPF must first calculate the distribution tree so that it may create a cache entry that can be used to forward the packet.

Figure 16 displays the forwarding cache for an example MOSPF router. The elements in the display include the following items:

Dest. Group — A known destination group address to which datagrams are currently being forwarded, or to which traffic was sent "recently" (i.e., since the last topology or group membership or other event which (re-)initialized MOSPF's forwarding cache.

Source — The datagram's source host address. Each (Dest. Group, Source) pair uniquely identifies a separate forwarding cache entry.

Dest. Group	Source	Upstream	Downstream		TTL
224.1.1.1	128.1.0.2	11	12	13	5
224.1.1.1	128.4.1.2	11	12	13	2
224.1.1.1	128.5.2.2	11	12	13	3
224.2.2.2	128.2.0.3	12	11		7

Figure 16: MOSPF Forwarding Cache

Upstream — Datagrams matching this row's Dest. Group and

Source must be received on this interface.

Downstream	If a datagram matching this row's Dest. Group and Source is received on the correct Upstream interface, then it is forwarded across the listed Downstream interfaces.
TTL	The minimum number of hops a datagram must cross to reach any of the Dest. Group's members. An MOSPF router may discard a datagram if it can see that the datagram has insufficient TTL to reach even the closest group member.

The information in the forwarding cache is not aged or periodically refreshed: It is maintained as long as there are system resources available (e.g., memory) or until the next topology change. The contents of the forwarding cache will change when:

- The topology of the OSPF internetwork changes, forcing all of the shortest path trees to be recalculated. (Once the cache has been flushed, entries are not rebuilt. If another packet for one of the previous (Dest. Group, Source) pairs is received, then a "new" cache entry for that pair will be created then.)

- There is a change in the Group-Membership LSAs indicating that the distribution of individual group members has changed.

7.2.2 Mixing MOSPF and OSPF Routers

MOSPF routers can be combined with non-multicast OSPF routers. This permits the gradual deployment of MOSPF and allows experimentation with multicast routing on a limited scale.

It is important to note that an MOSPF router is required to eliminate all non-multicast OSPF routers when it builds its source-based shortest-path delivery tree. (An MOSPF router can determine the multicast capability of any other router based on the setting of the multicast-capable bit (MC-bit) in the Options field of each router's link state advertisements.) The omission of non-multicast routers may create a number of potential problems when forwarding multicast traffic:

- The Designated Router for a multi-access network must be an MOSPF router. If a non-multicast OSPF router is elected the DR, the subnetwork will not be selected to forward multicast datagrams since a non-multicast DR cannot generate Group-Membership LSAs for its subnetwork (because it is not running IGMP, so it won't hear IGMP Host Membership Reports).

- Even though there may be unicast connectivity to a destination, there may not be multicast connectivity. For example, the only

path between two points could require traversal of a non-multicast-capable OSPF router.

- The forwarding of multicast and unicast datagrams between two points may follow different paths, making some routing problems a bit more challenging to solve.

7.2.3 Inter-Area Routing with MOSPF

Inter-area routing involves the case where a datagram's source and some of its destination group members reside in different OSPF areas. It should be noted that the forwarding of multicast datagrams continues to be determined by the contents of the forwarding cache which is still built from the local group database and the datagram source-based trees. The major differences are related to the way that group membership information is propagated and the way that the inter-area source-based tree is constructed.

7.2.3.1 Inter-Area Multicast Forwarders

In MOSPF, a subset of an area's Area Border Routers (ABRs) function as "inter-area multicast forwarders." An inter-area multicast forwarder is responsible for the forwarding of group membership information and multicast datagrams between areas. Configuration parameters determine whether or not a particular ABR also functions as an inter-area multicast forwarder.

Inter-area multicast forwarders summarize their attached areas' group membership information to the backbone by originating new Group-Membership LSAs into the backbone area. Note that the summarization of group membership in MOSPF is asymmetric. This means that while group membership information from non-backbone areas is flooded into the backbone, but group membership from the backbone (or from any other non-backbone areas) is not flooded into any non-backbone area(s).

To permit the forwarding of multicast traffic between areas, MOSPF introduces the concept of a "wild-card multicast receiver." A wild-card multicast receiver is a router that receives all multicast traffic generated in an area. In non-backbone areas, all inter-area multicast forwarders operate as wild-card multicast receivers. This guarantees that all multicast traffic originating in any non-backbone area is delivered to its inter-area multicast forwarder, and then if necessary into the backbone area. Since the backbone knows group membership for all areas, the datagram can be forwarded to the appropriate location(s) in the OSPF autonomous system, if only it is forwarded into the backbone by the source area's multicast ABR.

7.2.3.2 Inter-Area Datagram's Shortest-Path Tree

In the case of inter-area multicast routing, it is usually impossible to build a complete shortest-path delivery tree. Incomplete trees are a fact of life because each OSPF area's complete topological and group

membership information is not distributed between OSPF areas.

Topological estimates are made through the use of wild-card receivers and OSPF Summary-Links LSAs.

If the source of a multicast datagram resides in the same area as the router performing the calculation, the pruning process must be careful to ensure that branches leading to other areas are not removed from the tree. Only those branches having no group members nor wild-card multicast receivers are pruned. Branches containing wild-card multicast receivers must be retained since the local routers do not know whether there are any group members residing in other areas.

If the source of a multicast datagram resides in a different area than the router performing the calculation, the details describing the local topology surrounding the source station are not known. However, this information can be estimated using information provided by Summary-Links LSAs for the source subnetwork. In this case, the base of the tree begins with branches directly connecting the source subnetwork to each of the local area's inter-area multicast forwarders. Datagrams sourced from outside the local area will enter the area via one of its inter-area multicast forwarders, so they all must be part of the candidate distribution tree.

Since each inter-area multicast forwarder is also an ABR, it must maintain a separate link state database for each attached area. Thus each inter-area multicast forwarder is required to calculate a separate forwarding tree for each of its attached areas.

7.2.4 Inter-Autonomous System Multicasting with MOSPF

Inter-Autonomous System multicasting involves the situation where a datagram's source or some of its destination group members are in different OSPF Autonomous Systems. In OSPF terminology, "inter-AS" communication also refers to connectivity between an OSPF domain and another routing domain which could be within the same Autonomous System from the perspective of an Exterior Gateway Protocol.

To facilitate inter-AS multicast routing, selected Autonomous System Boundary Routers (ASBRs) are configured as "inter-AS multicast forwarders." MOSPF makes the assumption that each inter-AS multicast forwarder executes an inter-AS multicast routing protocol which forwards multicast datagrams in a reverse path forwarding (RPF) manner. Since the publication of the MOSPF RFC, a term has been defined for such a router: Multicast Border Router. See section 9 for an overview of the MBR concepts. Each inter-AS multicast forwarder is a wildcard multicast receiver in each of its attached areas. This guarantees that each inter-AS multicast forwarder remains on all pruned shortest-path trees and receives all multicast datagrams.

The details of inter-AS forwarding are very similar to inter-area forwarding. On the "inside" of the OSPF domain, the multicast ASBR must conform to all the requirements of intra-area and inter-area

forwarding. Within the OSPF domain, group members are reached by the usual forward path computations, and paths to external sources are approximated by a reverse-path source-based tree, with the multicast ASBR standing in for the actual source. When the source is within the OSPF AS, and there are external group members, it falls to the inter-AS multicast forwarders, in their role as wildcard receivers, to make sure that the data gets out of the OSPF domain and sent off in the correct direction.

7.2.5 MOSPF Tree Building and Forwarding Summary

MOSPF builds a separate tree for each (source, group) combination. The tree is rooted at the source, and includes each of the group members as leaves. If the (M)OSPF domain is divided into multiple areas, the tree is built in pieces, one area at a time. These pieces are then glued together at the area border routers which connect the various areas.

Sometimes group membership of certain areas (or other ASes) is unknown. MOSPF forces the tree to extend to these areas and/or ASes by adding their area border routers/AS boundary routers to the tree as "wildcard multicast receivers".

Construction of the tree within a given area depends on the location of the source. If the source is within the area, "forward" costs are used, and the path within the area follows the "forward path" , that is, the same route that unicast packets would take from source to group member. If the source belongs to a different area, or a different AS, "reverse costs" are used, resulting in reverse path forwarding through the area. (Reverse path forwarding is less preferred, but is forced because OSPF Summary LSAs and AS-External LSAs only advertise costs in the reverse direction).

MOSPF's tree-building process is data-driven. Despite having Group LSAs present in each area's link state database, no trees are built unless multicast data is seen from a source to a group (of course, if the Link State Database indicates no Group LSAs for this group, then no tree is built since there must be no group members present in this area). So, despite MOSPF being a "Dense-mode" routing protocol, it is not based on broadcast-and-prune, but rather it is a so-called "explicit-join" protocol.

If a packet arrives for a (source, group) pair which has not been seen by this router before, the router executes the Dijkstra algorithm over the relevant links in the Link State Database. The Dijkstra algorithm outputs the "source-rooted" shortest-path tree for this (source, group), as described earlier. The router examines its position in the tree, caching the expected inbound interface for packets from this source, and listing the outbound interface(s) that lead to active downstream

receivers. Subsequent traffic is examined against this cached data. The traffic from the source must arrive on the correct interface to be processed further.

7.3 Protocol-Independent Multicast (PIM)

The Protocol Independent Multicast (PIM) routing protocols have been developed by the Inter-Domain Multicast Routing (IDMR) working group of the IETF. The objective of the IDMR working group is to develop one--or possibly more than one--standards-track multicast routing protocol(s) that can provide scalable multicast routing across the Internet.

PIM is actually two protocols: PIM - Dense Mode (PIM-DM) and PIM - Sparse Mode (PIM-SM). In the remainder of this introduction, any references to "PIM" apply equally well to either of the two protocols... there is no intention to imply that there is only one PIM protocol. While PIM-DM and PIM-SM share part of their names, and they do have related control messages, they are really two independent protocols.

PIM receives its name because it is not dependent on the mechanisms provided by any particular unicast routing protocol. However, any implementation supporting PIM requires the presence of a unicast routing protocol to provide routing table information and to adapt to topology changes.

PIM makes a clear distinction between a multicast routing protocol that is designed for dense environments and one that is designed for sparse environments. Dense-mode refers to a protocol that is designed to operate in an environment where group members are relatively densely packed and bandwidth is plentiful. Sparse-mode refers to a protocol that is optimized for environments where group members are distributed across many regions of the Internet and bandwidth is not necessarily widely available. It is important to note that sparse-mode does not imply that the group has a few members, just that they are widely dispersed across the Internet.

The designers of PIM-SM argue that DVMRP and MOSPF were developed for environments where group members are densely distributed, and bandwidth is relatively plentiful. They emphasize that when group members and senders are sparsely distributed across a wide area, DVMRP and MOSPF exhibit inefficiencies (but both are efficient in delivering packets). The DVMRP periodically sends multicast packets over many links that do not lead to group members, while MOSPF propagates group membership information across links that may not lead to senders or receivers.

7.3.1 PIM - Dense Mode (PIM-DM)

While the PIM architecture was driven by the need to provide scalable sparse-mode delivery trees, PIM also defines a new dense-mode protocol instead of relying on existing dense-mode protocols such as DVMRP and MOSPF. It is envisioned that PIM-DM would be deployed in resource rich environments, such as a campus LAN where group membership is relatively

dense and bandwidth is likely to be readily available. PIM-DM's control messages are similar to PIM-SM's by design.

PIM - Dense Mode (PIM-DM) is similar to DVMRP in that it employs the Reverse Path Multicasting (RPM) algorithm. However, there are several important differences between PIM-DM and the DVMRP:

- To find routes back to sources, PIM-DM relies on the presence of an existing unicast routing table. PIM-DM is independent of the mechanisms of any specific unicast routing protocol. In contrast, DVMRP contains an integrated routing protocol that makes use of its own RIP-like exchanges to build its own unicast routing table (so a router may orient itself with respect to active source(s)).

- Unlike the DVMRP, which calculates a set of child interfaces for each (source, group) pair, PIM-DM simply forwards multicast traffic on all downstream interfaces until explicit prune messages are received. PIM-DM is willing to accept packet duplication to eliminate routing protocol dependencies and to avoid the overhead inherent in determining the parent/child relationships.

For those cases where group members suddenly appear on a pruned branch of the delivery tree, PIM-DM--like DVMRP--employs graft messages to re-attach the previously pruned branch to the delivery tree.

7.3.1.1 PIM-DM Tree Building and Forwarding Summary

Dense-mode PIM builds source-based trees by default. It uses the RPM algorithm, which is what DVMRP uses: Thus, PIM-DM is a data-driven protocol that floods packets to the edges of the PIM-DM domain, and expects prunes to be returned on inactive branches. A minor difference from DVMRP is that PIM-DM floods packets for new (source, group) pairs on all non-incoming interfaces. PIM-DM trades off a bit of extra flooding traffic for a simpler protocol design.

Pruning in PIM-DM only happens via explicit Prune messages, which are multicast on broadcast links (if there are other routers present which hear a prune, and they still wish to receive traffic for this group to support active receivers that are downstream of them, then these other routers must multicast PIM-Join packets to ensure they remain attached to the distribution tree). Finally, PIM-DM uses a reliable graft mechanism to enable previously-sent prunes to be "erased" when new downstream group members appear after a prune had been sent.

Since PIM-DM uses RPM, it implements a reverse-path check on all packets which it receives. Again, this check verifies that received packets arrive on the interface that the router would use if it needed to send a packet toward the source's prefix. Since PIM-DM does not have its own routing protocol (as DVMRP does), it uses the existing unicast

routing protocol to locate itself with respect to the source(s) of multicast packets it has seen.

8. "SPARSE MODE" ROUTING PROTOCOLS

The most recent additions to the set of multicast routing protocols are called "sparse mode" protocols. They are designed from a different perspective than the "dense mode" protocols that we have already examined. Often, they are not data-driven, in the sense that forwarding state is set up in advance, and they trade off using bandwidth liberally (which is a valid thing to do in a campus LAN environment) for other techniques that are much more suited to scaling over large WANs, where bandwidth is scarce and expensive.

These emerging routing protocols include:

- Protocol Independent Multicast - Sparse Mode (PIM-SM), and
- Core-Based Trees (CBT).

While these routing protocols are designed to operate efficiently over a wide area network where bandwidth is scarce and group members may be quite sparsely distributed, this is not to imply that they are only suitable for small groups. Sparse doesn't mean small, rather it is meant to convey that the groups are widely dispersed, and thus it is wasteful to (for instance) flood their data periodically across the entire internetwork.

CBT and PIM-Sparse Mode (PIM-SM) have been designed to provide highly efficient communication between members of sparsely distributed groups--the type of groups that are likely to be prevalent in wide-area internetworks. The designers of these sparse-mode protocols' have observed that several hosts participating in a multicast session do not require periodically broadcasting their traffic across the entire internetwork.

Noting today's existing MBone scaling problems, and extrapolating to a future of ubiquitous multicast (overlaid with perhaps thousands of widely scattered groups), it is not hard to imagine that existing multicast routing protocols will experience scaling problems. To mitigate these potential scaling issues, PIM-SM and CBT are designed to ensure that multicast traffic only crosses routers that have explicitly joined a shared tree on behalf of group members.

8.1 Protocol-Independent Multicast - Sparse Mode (PIM-SM)

As described previously, PIM also defines a "dense-mode" or source-based tree variant. Again, the two protocols are quite unique, and other than control messages, they have very little in common. Note that although PIM integrates control message processing and data packet forwarding among PIM-Sparse and -Dense Modes, PIM-SM and PIM-DM must never run in the same region at the same time. Essentially, a region is a set of

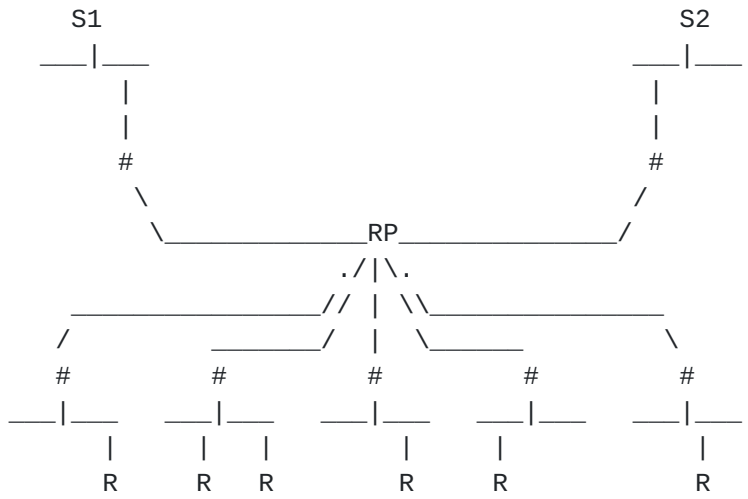
routers which are executing a common multicast routing protocol.

PIM-SM differs from existing dense-mode protocols in two key ways:

- Routers with adjacent or downstream members are required to explicitly join a sparse mode delivery tree by transmitting join messages. If a router does not join the pre-defined delivery tree, it will not receive multicast traffic addressed to the group.

  In contrast, dense-mode protocols assume downstream group membership and forward multicast traffic on downstream links until explicit prune messages are received. Thus, the default forwarding action of dense-mode routing protocols is to forward all traffic, while the default action of a sparse-mode protocol is to block traffic unless it has been explicitly requested.

- PIM-SM evolved from the Core-Based Trees (CBT) approach in that it employs the concept of a "core" (or rendezvous point (RP) in PIM-SM terminology) where receivers "meet" sources.



LEGEND

#	PIM Router
R	Multicast Receiver

Figure 17: Rendezvous Point

When joining a group, each receiver uses IGMP to notify its directly-attached router, which in turn joins the multicast delivery tree by sending an explicit PIM-Join message hop-by-hop toward the group's RP. A source uses the RP to announce its presence, and act as a conduit to members that have joined the group. This model requires sparse-mode

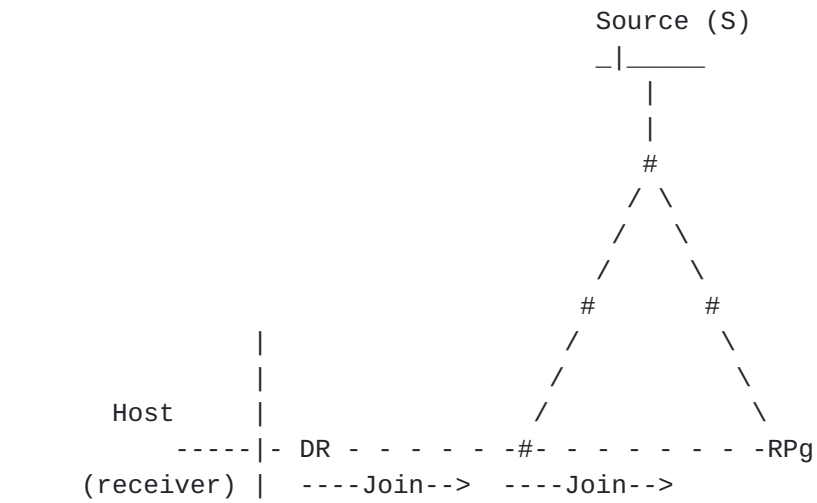
routers to maintain a small amount of state (the RP-set for the sparse-mode region) prior to the arrival of data.

There is only one RP-set per sparse-mode domain. By using a hash function, each PIM-SM router can map a group address uniquely to one of the members of the RP-set (to determine the group's RP). At any given time, each group has precisely one RP. In the event of the failure of an RP, a new RP-set is distributed which does not include the failed RP.

8.1.1 Directly Attached Host Joins a Group

When there is more than one PIM router connected to a multi-access LAN, the router with the highest IP address is selected to function as the Designated Router (DR) for the LAN. The DR sends Join/Prune messages toward the RP.

When the DR receives an IGMP Report message for a new group, it performs a deterministic hash function over the sparse-mode region's RP-set to uniquely determine the RP for the group.



LEGEND

PIM Router RPg Rendezvous Point for group g

Figure 18: Host Joins a Multicast Group

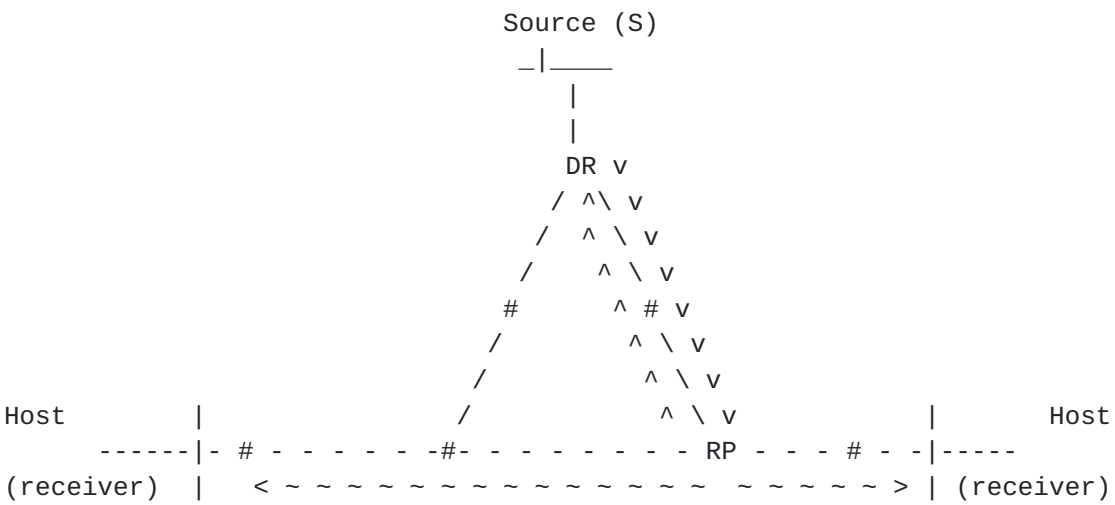
After performing the lookup, the DR creates a multicast forwarding entry for the (*, group) pair and transmits a unicast PIM-Join message toward the RP for this specific group. The (*, group) notation indicates an (any source, group) pair. The intermediate routers forward the unicast

PIM-Join message, creating a forwarding entry for the (*, group) pair

only if such a forwarding entry does not yet exist. Intermediate routers must create a forwarding entry so that they will be able to forward future traffic downstream toward the DR which originated the PIM-Join message.

8.1.2 Directly Attached Source Sends to a Group

When a source first transmits a multicast packet to a group, its DR forwards the datagram to the RP for subsequent distribution along the group's delivery tree. The DR encapsulates the initial multicast packets in PIM-SM-Register packets and unicasts them toward the group's RP. The PIM-SM-Register packets inform the RP of a new source. The RP may then elect to transmit PIM-Join messages back toward the source's DR to join this source's shortest-path tree, which will allow future unencapsulated packets to flow from this source's DR to the group's RP.



LEGEND

#	PIM Router	v v v	PIM-SM-Register
RP	Rendezvous Point	^ ^ ^	PIM-Join
		~ ~ ~	Resend to group members

Figure 19: Source sends to a Multicast Group

Unless the RP decides to join the source's SPT, rooted at the source's DR, the (source, group) state is not created in all the routers between source's DR and the RP, and the DR must continue to send the source's multicast IP packets to the RP as unicast packets encapsulated within

unicast PIM-SM-Register packets. The DR may stop forwarding multicast

packets encapsulated in this manner once it has received a PIM-Register-Stop message from the group's RP. The RP may send PIM-Register-Stop messages if there are no downstream receivers for a group, or if the RP has successfully joined the source's shortest-path tree (SPT).

8.1.3 Shared Tree (RP-Tree) or Shortest Path Tree (SPT)?

The RP-tree provides connectivity for group members but does not optimize the delivery path through the internetwork. PIM-SM allows routers to either a) continue to receive multicast traffic over the shared RP-tree, or b) subsequently create a source-based shortest-path tree on behalf of their attached receiver(s). Besides reducing the delay between this router and the source (beneficial to its attached receivers), the shared tree also reduces traffic concentration effects on the RP-tree.

A PIM-SM router with local receivers has the option of switching to the source's shortest-path tree (i.e., source-based tree) once it starts receiving data packets from the source. The change-over may be triggered if the data rate from the source exceeds a predefined threshold. The local receiver's last-hop router does this by sending a PIM-Join message toward the active source. After the source-based SPT is active, protocol mechanisms allow a Prune message for that source to be transmitted to the group's RP, thus removing this router from the this source's shared RP-tree. Alternatively, the DR may be configured to never switch over to the source-based SPT, or some other metric might be used to control when to switch to a source-based SPT.

==========

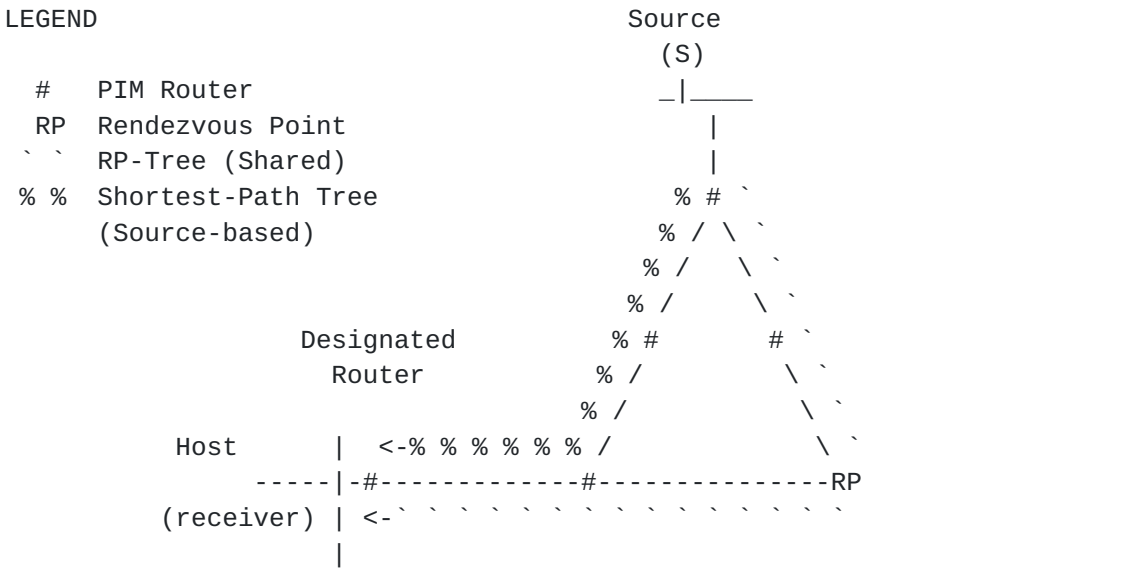


Figure 20: Shared RP-Tree and Shortest Path Tree (SPT)
==========

Besides a last-hop router being able to switch to a source-based tree, there is also the capability of the RP for a group to transition to a source's shortest-path tree. Similar controls (bandwidth threshold, administrative metrics, etc.) may be used at an RP to influence these decisions. The RP only joins the source's DR's SPT if local policy controls permit this.

8.1.4 PIM-SM Tree Building and Forwarding Summary

PIM-SM can use both source-based and shared trees. In fact, a given group may have some routers on its shared tree, and others on source-based trees, simultaneously. By default, PIM-SM uses shared trees rooted at the Rendezvous Point, but regardless of which tree type is in use, there is no broadcasting of any traffic. Interested receivers use IGMP to inform their local PIM-SM router(s), then the subnetwork's PIM-SM Designated Router then issues PIM-Join messages (on behalf of the receivers) toward the group's RP. These join messages establish forwarding state in the intermediate routers which is used in the future to make forwarding decisions. If a packet is received which has no pre-established forwarding state, it is dropped.

As each packet is received, it must match a pre-existing forwarding cache entry. If it does, then you know which interface on which to do the a reverse-path check. This is consistent with the forwarding technique used by PIM-DM, and similar to that used by DVMRP. The unicast routing table provides the necessary information to determine the best route toward the group's RP; the packet must have arrived on the interface this router would use to send traffic toward the group's RP. Note that the forwarding state which is created by PIM-SM is uni-directional (it allows traffic to flow from the source's DR toward the RP, not away from it).

8.2 Core Based Trees (CBT)

Core Based Trees is another multicast architecture that is based on a shared delivery tree. It is specifically intended to address the important issue of scalability when supporting multicast applications across the public Internet, and is also suitable for use within private intranetworks.

Similar to PIM-SM, CBT is protocol-independent. CBT employs the information contained in the unicast routing table to build its shared delivery tree. It does not care how the unicast routing table is derived, only that a unicast routing table is present. This feature allows CBT to be deployed without requiring the presence of any specific unicast routing protocol.

"Protocol independence" doesn't necessarily have to mean that multicast

paths are the same set of routers and links used by unicast routing, though it is easy to make this assumption (it may indeed hold true in

some cases). An underlying routing protocol could collect both unicast- and multicast- related information, so that unicast routes could be calculated based on the unicast information, and multicast routes based on the multicast information. If the path (set of intervening routers and links) used between any two network nodes is the same for multicast and for unicast, then it can be said that the unicast and multicast topologies overlap (for that set of paths).

Where multicast and unicast topologies do happen to overlap, multicast and unicast paths could be calculated from the one set of information (i.e., unicast). However, if the set of routers and links between two network nodes differs for multicast and unicast traffic, then the unicast and multicast topologies are incongruent for that network path. It's a matter of policy whether unicast and multicast topologies are aligned for any set of network links.

The current version of the CBT specification has adopted a similar "bootstrap" mechanism to that defined in the PIM-SM specification. It is an implementation choice whether a dynamic or static mechanism is used for discovering how groups map to cores. This process of discovering which core serves which group(s) is what is referred to as bootstrapping.

Each group has only one core, but one core might serve multiple groups. Use of the dynamic bootstrap mechanism is only applicable within a multicast region, not between regions. The advantage of the dynamic approach is that a region's CBT routers need less configuration. The disadvantage is that core placement could be particularly sub-optimal for some set of receivers. Manual placement means that each group's core can be "better" positioned relative to a group's members. CBT's modular design allows other core discovery mechanisms to be used if such mechanisms are considered more beneficial to CBT's requirements. For inter-domain RP/Core discovery, efforts are underway to standardize (or at least separately specify) a common mechanism, the intent being that any shared tree protocol could implement this common interdomain discovery architecture using its own protocol message types.

In a significant departure from PIM-SM, CBT has decided to maintain its scaling characteristics by not offering the option of optimizing delay by shifting from a Shared Tree (e.g., PIM-SM's RP-Tree) to a Shortest Path Tree (SPT). The designers of CBT believe that this is a critical decision since when multicasting becomes widely deployed, the need for routers to maintain large amounts of state information will become the overpowering scaling factor. Thus CBT's state information (i.e., its forwarding cache) consists of: (group, incoming interface, {outgoing interface list}). Forwarding decisions are made by using the group's destination address as the search key into the forwarding cache.

Finally, unlike PIM-SM's shared tree state, CBT state is bi-directional. Data may therefore flow in either direction along a branch. Thus, data

from a source which is directly attached to an existing tree branch need not be encapsulated.

8.2.1 Joining a Group's Shared Tree

A host that wants to join a multicast group issues an IGMP host membership report. This message informs its local CBT-aware router(s) that it wishes to receive traffic addressed to the multicast group. Upon receipt of an IGMP host membership report for a new group, the local CBT router issues a JOIN_REQUEST which is processed hop-by-hop, creating transient join state (incoming interface, outgoing interface) in each router traversed.

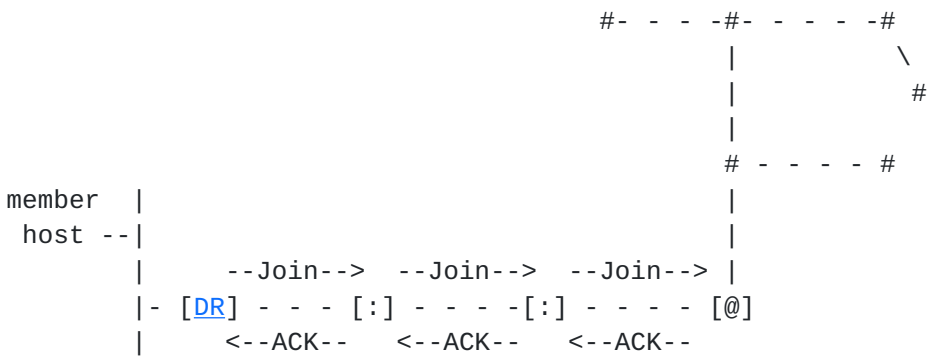
If the JOIN_REQUEST encounters a router that is already on the group's shared tree before it reaches the core router, then that router issues a JOIN_ACK hop-by-hop back toward the sending router. If the JOIN_REQUEST does not encounter an on-tree CBT router along its path towards the core, then the core router is responsible for responding with a JOIN_ACK. In either case, each intermediate router that forwards the JOIN_REQUEST towards the core is required to create a transient "join state." This transient "join state" includes the multicast group, and the JOIN_REQUEST's incoming and outgoing interfaces. This information allows an intermediate router to forward returning JOIN_ACKs along the exact reverse path to the CBT router which initiated the JOIN_REQUEST.

As the JOIN_ACK travels towards the CBT router that issued the JOIN_REQUEST, each intermediate router creates new "active state" for this group. New branches are established by having the intermediate routers remember which interface is upstream, and which interface(s) is(are) downstream. Once a new branch is created, each child router monitors the status of its parent router with a keepalive mechanism, the CBT "Echo" protocol. A child router periodically unicasts a ECHO_REQUEST to its parent router, which is then required to respond with a unicast ECHO_REPLY message.

If, for any reason, the link between an on-tree router and its parent should fail, or if the parent router is otherwise unreachable, the on-tree router transmits a FLUSH_TREE message on its child interface(s) which begins tearing down all the downstream branches for this group. Each leaf router is then responsible for re-attaching itself to the group's core, thus rebuilding the shared delivery tree. Leaf routers only re-join the shared tree if they have at least one directly-attached group member.

The Designated Router (DR) for a given broadcast-capable subnetwork is elected by CBT's "Hello" protocol. It functions as the only upstream router for all groups using that link. The DR is not necessarily the best next-hop router to every core for every multicast group. The

implication is that it is possible for the DR to receive a JOIN_REQUEST over a LAN, but the DR may need to redirect the JOIN_REQUEST back across



LEGEND

[DR] Local CBT Designated Router
[:] CBT Router
[@] Core Router
CBT Router that is already on the shared tree

Figure 21: CBT Tree Joining Process

the same link to the best next-hop router toward a given group's core. Data traffic is never duplicated across a link, only JOIN_REQUESTs, which should be an inconsequential volume of traffic.

8.2.2 Data Packet Forwarding

When a JOIN_ACK is received by an intermediate router, it either adds the interface over which the JOIN_ACK was received to an existing group's forwarding cache entry, or creates a new entry for the multicast group if one does not already exist. When a CBT router receives a data packet addressed to any multicast group, it simply forwards the packet over the outgoing interfaces specified by the group's forwarding cache entry.

8.2.3 Non-Member Sending

Similar to other multicast routing protocols, CBT does not require that the source of a multicast packet be a member of the multicast group. However, for a multicast data packet to reach the active core for the group, at least one CBT-capable router must be present on the non-member sender's subnetwork. The local CBT-capable router employs IP-in-IP encapsulation and unicasts the data packet to the active core for delivery to the rest of the multicast group. Thus, every CBT-capable router in each CBT region needs a list of corresponding <core, group>

mappings

8.2.4 CBT Tree Building and Forwarding Summary

CBT trees are constructed based on forward paths from the source(s) to the core. Any group only has exactly one active core, as you recall. As CBT is an explicit-join protocol, no routers forward traffic to a group unless there are pre-established active receivers for that group within the CBT domain.

Trees are built using each CBT router's unicast routing table, by forwarding JOIN_REQUESTs toward a group's core. The resulting non-source-specific state is bi-directional, a feature unique to CBT trees. If any group member needs to send to the group, zero extra forwarding state needs to be established...every possible receiver is already also a potential sender, with no penalty of extra state (in the form of (source, group) state), in the routers.

8.2.5 CBT Multicast Interoperability

Multicast interoperability is currently being defined. Work is underway in the IDMR working group to describe the attachment of stub-CBT and stub-PIM domains to a DVMRP backbone. Future work will focus on developing methods of connecting non-DVMRP transit domains to a DVMRP backbone.

CBT interoperability will be achieved through the deployment of domain border routers (BRs) which enable the forwarding of multicast traffic between the CBT and DVMRP domains. The BR implements DVMRP and CBT on different interfaces and is responsible for forwarding data across the domain boundary.

The BR is also responsible for exporting selected routes out of the CBT domain into the DVMRP domain. While the CBT stub domain never needs to import routes, the DVMRP backbone needs to import routes to any sources of traffic which are inside the CBT domain. The routes must be imported so that DVMRP can perform its reverse-path check.

==========

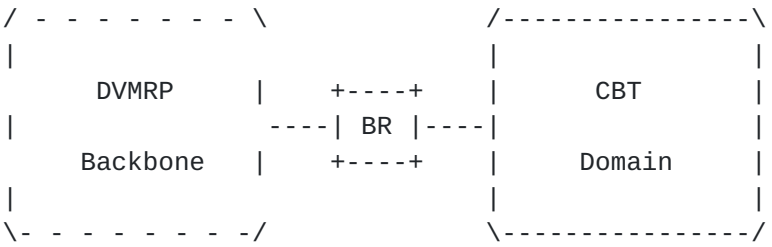


Figure 22: Domain Border Router

9. MULTICAST IP ROUTING: RELATED TOPICS

To close this overview of multicast IP routing technology, we first examine multicast routing protocol interoperability, then turn to expanding-ring searches, which may not be equally-effective depending on which multicast routing protocol is being used.

9.1 Interoperability Framework for Multicast Border Routers

In late 1996, the IETF IDMR working group began discussing a formal structure that would describe the way different multicast routing protocols should interact inside a multicast border router (MBR). The work-in-progress can be found in the following internet draft: <draft-thaler-interop-00.ps>, or its successor. The draft covers explicit rules for the major multicast routing protocols that existed at the end of 1996: DVMRP, MOSPF, PIM-DM, PIM-SM, and CBT, but applies to any potential future multicast routing protocol(s) as well.

The IDMR standards work will focus on this generic inter-protocol MBR scheme, rather than having to write 25 documents, 20 detailing how each of those 5 protocols must interwork with the 4 others, plus 5 detailing how two disjoint regions running the same protocol must interwork.

9.1.1 Requirements for Multicast Border Routers

In order to ensure reliable multicast delivery across a network with an arbitrary mixture of multicast routing protocols, some constraints are imposed to limit the scope of the problem space.

Each multicast routing domain, or region, may be connected in a "tree of regions" topology. If more arbitrary inter-regional topologies are desired, a hierarchical multicast routing protocol (such as, H-DVMRP) must be employed, because it carries topology information about how the regions are interconnected. Until this information is available, we must consider the case of a tree of regions with one centrally-placed "backbone" region. Each pair of regions is interconnected one or more MBR(s).

A MBR is responsible for injecting a default route into its "child regions," and also injecting subnetwork reachability information into its "parent region," optionally using aggregation techniques to reduce the volume of the information while preserving its meaning. MBRs which comply with <draft-thaler-interop-00.ps> have other characteristics and duties, including:

- The MBR consists at least two active routing components, each an instance of some multicast routing protocol. No assumption is made about the type of routing protocol (e.g., broadcast-and-prune or explicit-join; distance-vector or link-state; etc.) any component

runs, or the nature of a "component". Multiple components running the same protocol are allowed.

- A MBR forwards packets between two or more independent regions, with one or more active interfaces per region, but only one component per region.

- Each interface for which multicast is enabled is "owned" by exactly one of the components at a time.

- All components share a common forwarding cache of (S,G) entries, which are created when data packets are received, and can be deleted at any time. The component owning an interface is the only component that may change forwarding cache entries pertaining to that interface. Each forwarding cache entry has a single incoming interface (iif) and a list of outgoing interfaces (oiflist).

9.2. ISSUES WITH EXPANDING-RING SEARCHES

Expanding-ring searches may be used when an end-station wishes to find the closest example of a certain kind of server. One key assumption is that each of these servers is equivalent in the service provided: Ask any of them a question pertinent to that service and you should get the same answer. For example, such a service is the DNS. The searching client sends a query packet with the IP header's TTL field set to 1. This packet will only reach its local subnetwork. If a response is not heard within some timeout interval, a second query is emitted, this time with the TTL set to 2. This process of sending and waiting for a response, while incrementing the TTL, is what an expanding-ring search is, fundamentally. Expanding-ring searches can facilitate resource-discovery or auto-configuration protocols.

Another key assumption is that the multicast infrastructure provides the ability to broadcast from a source equally well in all directions. This usually implies that, at a minimum, all routers in an internetwork support a multicast routing protocol.

There are two classes of routing protocols to consider when discussing expanding-ring searches: DVMRP, MOSPF, and PIM-DM make up one class, while PIM-SM and CBT comprise the other.

DVMRP supports expanding-ring searches fairly well for a reasonable number of sources. The downside of using DVMRP is that there is source-specific state kept across the entire DVMRP internetwork. As the number of sources increases, the amount of state increases linearly. Due to DVMRP's broadcast-and-prune nature, the tree for each source will quickly converge to reach all receivers for a given group (limited to receivers within the packet's TTL). If we assume that all routers in your internetwork speak DVMRP, these TTL-scoped searches will have the desired result: As the TTL is incremented, the packets will cross successively further routers, radiating away from the source subnetwork.

MOSPF supports expanding-ring searches particularly well. It also keeps source-specific state, so has the same scaling issues as DVMRP with

respect to state (forwarding cache) increasing linearly with the number of sources. MOSPF's unique capability is that it knows the topology of its local area. One by-product of this knowledge is that for a given (source, group) pair, each MOSPF router knows the minimum TTL needed to reach the closest group member. If a packet's TTL is not at least this large, it need not be forwarded. This conserves bandwidth that would otherwise be wasted by several iterations of successively larger TTLs. However, since MOSPF is an explicit-join protocol, any servers wishing to be found must join the search group in advance. Otherwise, MOSPF's trees will not include these subnetworks.

PIM-DM is very similar to DVMRP in the context of expanding-ring searches. If we continue to assume that all routers in a given internetwork support the multicast routing protocol, then RPM (used by PIM-DM and the DVMRP) will ensure that sources' traffic is broadcast across the entire internetwork (limited only by the packet's initial TTL), with no forwarding loops.

Shared-tree protocols do not have properties that necessarily lend themselves to supporting expanding-ring searches. PIM-SM, by default, does not build source-based trees. Consider the case of a sender on a leaf subnet in a PIM-SM domain. Multicast packets sent with TTL=1 will only reach end-stations on the local subnetwork, but TTL=2 packets will be tunneled inside PIM-SM-Register packets destined for the RP. Once at the RP, the PIM-SM-Register wrapper is removed, exposing the multicast packet, which should now have TTL=1 (the DR should have decremented the TTL before forwarding the packet). The RP can now forward the original packet over its attached downstream interfaces for this group. Since the TTL=1, this is as far as they will go. Future packets, with incrementally-higher TTLs, will radiate outward from the RP along any downstream branches for this group. Thus, the search will locate resources that are closest to the RP, not the source (unless the RP and the source happen to be very close together).

CBT does not really have this problem, at least in the latest version, since it does not need to encapsulate packets to get them to the group's core. Also, CBT's state is bi-directional, so any receiver can also be a sender with no tree-setup penalty. CBT, because it is a shared-tree protocol, isn't as good as DVMRP, MOSPF, or PIM-DM for expanding-ring searches, but it is better than PIM-SM: CBT tends to have more symmetric distances, and "closer" on the core-based tree is more correlated with "closer" in terms of network hops. However, CBT is not perfect for these searches. The effectiveness of a CBT search depends on the density of branch points of the group's distribution tree in the immediate vicinity of the source. If we assume that all routers are also CBT routers, then the search can be quite effective.

10. REFERENCES

10.1 Requests for Comments (RFCs)

1075 "Distance Vector Multicast Routing Protocol," D. Waitzman, C. Partridge, and S. Deering, November 1988.

1112 "Host Extensions for IP Multicasting," Steve Deering, August 1989.

1583 "OSPF Version 2," John Moy, March 1994.

1584 "Multicast Extensions to OSPF," John Moy, March 1994.

1585 "MOSPF: Analysis and Experience," John Moy, March 1994.

1700 "Assigned Numbers," J. Reynolds and J. Postel, October 1994. (STD 2)

1812 "Requirements for IP version 4 Routers," Fred Baker, Editor, June 1995

2117 "Protocol Independent Multicast-Sparse Mode (PIM-SM): Protocol Specification," D. Estrin, D. Farinacci, A. Helmy, D. Thaler; S. Deering, M. Handley, V. Jacobson, C. Liu, P. Sharma, and L. Wei, July 1997.

2189 "Core Based Trees (CBT version 2) Multicast Routing," A. Ballardie, September 1997.

2200 "Internet Official Protocol Standards," Jon Postel, Editor, June 1997. (STD 1)

2201 "Core Based Trees (CBT) Multicast Routing Architecture," A. Ballardie, September 1997.

10.2 Internet-Drafts

"Core Based Tree (CBT) Multicast Border Router Specification for Connecting a CBT Stub Region to a DVMRP Backbone," <draft-ietf-idmr-cbt-dvmrp-00.txt>, A. J. Ballardie, March 1997.

"Distance Vector Multicast Routing Protocol," <draft-ietf-idmr-dvmrp-v3-04.ps>, T. Pusateri, February 19, 1997.

"Internet Group Management Protocol, Version 2," <draft-ietf-idmr-igmp-v2-06.txt>, William Fenner, January 22, 1997.

"Internet Group Management Protocol, Version 3," <draft-cain-igmp-00.txt>, Brad Cain, Ajit Thyagarajan, and Steve Deering, Expired.

"Protocol Independent Multicast Version 2, Dense Mode Specification," <draft-ietf-idmr-pim-dm-spec-05.ps>, S. Deering, D. Estrin, D. Farinacci, V. Jacobson, A. Helmy, and L. Wei, May 21, 1997.

"Protocol Independent Multicast-Sparse Mode (PIM-SM): Motivation and Architecture," <draft-ietf-idmr-pim-arch-04.ps>, S. Deering, D. Estrin, D. Farinacci, V. Jacobson, C. Liu, and L. Wei, November 19, 1996.

"PIM Multicast Border Router (PMBR) specification for connecting PIM-SM domains to a DVMRP Backbone," <draft-ietf-mboned-pmbr-spec-00.txt>, D. Estrin, A. Helmy, D. Thaler, February 3, 1997.

"Administratively Scoped IP Multicast," <draft-ietf-mboned-admin-ip-space-03.txt>, D. Meyer, June 10, 1997.

"Interoperability Rules for Multicast Routing Protocols," <draft-thaler-interop-00.txt>, D. Thaler, November 7, 1996.

See the IDMR home pages for an archive of specifications:

<URL:http://www.cs.ucl.ac.uk/ietf/public_idmr/>
<URL:http://www.ietf.org/html.charters/idmr-charter.html>

10.3 Textbooks

Comer, Douglas E. Internetworking with TCP/IP Volume 1 Principles, Protocols, and Architecture Second Edition, Prentice Hall, Inc. Englewood Cliffs, New Jersey, 1991

Huitema, Christian. Routing in the Internet, Prentice Hall, Inc. Englewood Cliffs, New Jersey, 1995

Stevens, W. Richard. TCP/IP Illustrated: Volume 1 The Protocols, Addison Wesley Publishing Company, Reading MA, 1994

Wright, Gary and W. Richard Stevens. TCP/IP Illustrated: Volume 2 The Implementation, Addison Wesley Publishing Company, Reading MA, 1995

10.4 Other

Dalal, Y. K., and Metcalfe, R. M., "Reverse Path Forwarding of Broadcast Packets", Communications of the ACM, 21(12):1040-1048, December 1978.

Deering, Steven E. "Multicast Routing in a Datagram Internetwork," Ph.D. Thesis, Stanford University, December 1991.

Ballardie, Anthony J. "A New Approach to Multicast Communication in a Datagram Internetwork," Ph.D. Thesis, University of London, May 1995.

"Hierarchical Distance Vector Multicast Routing for the MBone," Ajit Thyagarajan and Steve Deering, Proceedings of the ACM SIGCOMM, pages 60-66, October, 1995.

11. SECURITY CONSIDERATIONS

As with unicast routing, the integrity and accuracy of the multicast routing information is important to the correct operation of the multicast segments of the Internet. Lack of authentication of routing protocol updates can permit an adversary to inject incorrect routing data and cause multicast routing to break or flow in unintended directions. Some existing multicast routing protocols (e.g., MOSPF) do support cryptographic authentication of their protocol exchanges. More detailed discussion of multicast routing protocol security is left to the specifications of those routing protocols.

Lack of authentication of IGMP can permit an adversary to inject false IGMP messages on a directly attached subnet. Such messages could cause unnecessary traffic to be transmitted to that subnet (e.g., via a forged JOIN) or could cause desired traffic to not be transmitted to that subnet (e.g., via a forged LEAVE). If this is considered to be an issue, one could use the IP Authentication Header [RFC-1825, RFC-1826] to provide cryptographic authentication of the IGMP messages. The reader should consult the IGMPv2 specification for additional information on this topic.

Security issues in multicast data traffic are beyond the scope of this document.

12. ACKNOWLEDGEMENTS

This RFC would not have been possible without the encouragement of Mike O'Dell and the support of David Meyer and Joel Halpern. Also invaluable was the feedback and comments from the IETF MBoneD and IDMR working groups. A number of people spent considerable time commenting on and discussing this paper with the authors, and deserve to be mentioned by name: Kevin Almeroth, Ran Atkinson, Tony Ballardie, Steve Casner, Jon Crowcroft, Steve Deering, Bill Fenner, Hugh Holbrook, Cyndi Jung, John Moy, Shuching Shieh, Dave Thaler, and Nair Venugopal. If we neglected to mention anyone here, please accept our sincerest apologies.

13. AUTHORS' ADDRESSES

Tom Maufer

3Com Corporation
5400 Bayfront Plaza
Mailstop 5247
P.O. Box 58145
Santa Clara, CA 95052-8145

Phone: +1 408 764-8814
Email: <maufer@3Com.com>

Chuck Semeria

3Com Corporation
5400 Bayfront Plaza
Mailstop 2218
P.O. Box 58145
Santa Clara, CA 95052-8145

Phone: +1 408 764-7201
Email: <semeria@3Com.com>