

6

IP Multicast Routing

Multicast is a real-time, network-level information distribution technology. It does not need any central server to distribute information at the application level. Like many other IP technologies, multicast was originally designed in a university. It grew from an overlay network called the mBone (Figure 6.1) which is built on top of regular Internet links. Today, multicast seems to have reached a critical level of maturity which makes it capable of supporting commercial services such as television broadcast over IP, real-time financial data distribution, and videoconferencing. These applications will soon trigger a need for IP multicast-enabled intranets.

6.1 Introduction

The chapter explains the advantages of network-level data distribution and describes the protocols currently used, and their limits. There is also a description of some widely used applications. As multicast is an evolving technology, we also cover the current work at IETF regarding group address allocation and multicast interdomain routing.

6.2 When to use multicast routing

6.2.1 A real-time technology

There are already many techniques that are used to distribute information to many recipients on the Internet. They were developed to solve specific problems that were encountered during the development of the Internet:

- The domain name system (DNS) is used to distribute the mapping of domain names to IP addresses. DNS defines an efficient caching and replication mechanism for use between DNS servers.

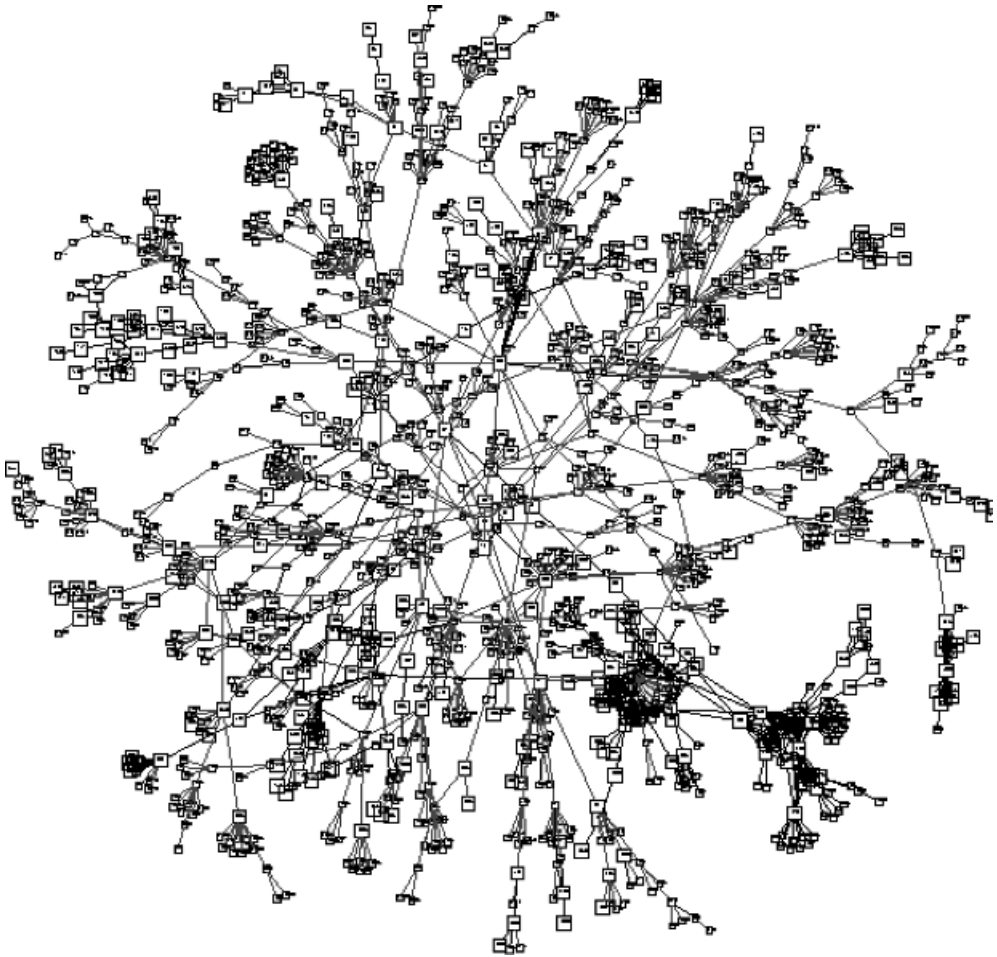


Figure 6.1 The mBone as of August 5, 1996. Reproduced from the University of California at Berkeley.

- NNTP, the Network News Transfer Protocol, is used to send newsgroup messages to news servers worldwide.
- IRC, the Internet Relay Chat, is a text chat protocol optimized to immediately send any sentence typed by any participant of a forum to all other relevant chat servers, which in turn send this sentence to all members of the forum that they host.
- Even HTTP, the protocol used to transfer web documents, was designed to let cache servers know how long they can keep a page in memory, in order to minimize unnecessary network traffic.

These techniques are very efficient at what they do, but they share a common characteristic: they are not real time. Because they duplicate and distribute information at the application

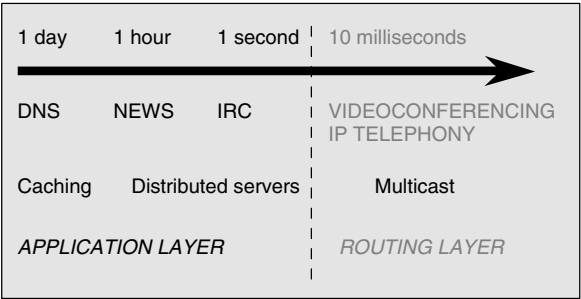


Figure 6.2 Achievable information transmission delays according to distribution technology used.

layer, classic information distribution techniques are unable to handle real-time information (i.e., information that must be distributed in less than 100 ms or so, see Figure 6.2).

Videoconferencing and television over IP are the primary applications of IP multicast, but there are many other situations in which several computers need to share the same information with very low latency: interactive gaming or financial applications are also very likely to use IP multicast once it is widely available.

6.2.2 Network efficiency

The network efficiency of IP multicast is best demonstrated by an example. We can take the example of an IRC forum, with just one server. For this application each client opens a socket on a central server (or a set of replicated servers), which takes care of duplicating and sending all incoming messages back to the forum members.

The simplified IP network shown in Figure 6.3 shows a ‘packet storm’ caused by a single packet sent from client ‘a’ to the reflector. Several copies of the same packet are sent simultaneously over multiple links. The reflector has to be a powerful machine, since it has to handle a separate connection for every client, and the network connections to the

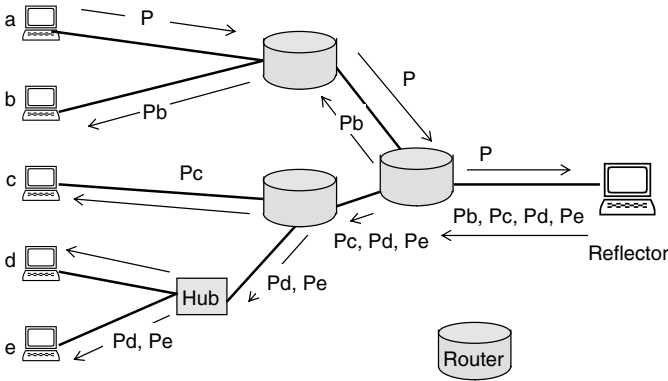


Figure 6.3 Multi-unicast is inefficient.

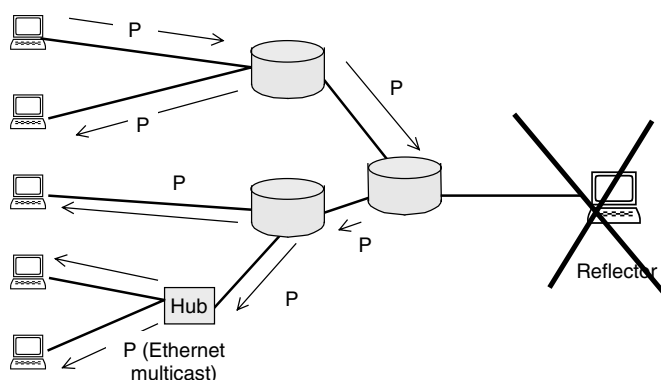


Figure 6.4 Multicast optimizes the distribution of information.

reflector must be able to carry all the generated traffic, which is proportional to the number of clients. A more scalable solution would ideally send only one copy of each packet over each link and would not need a special machine to handle all the work: this is exactly what IP multicast is doing, as illustrated in Figure 6.4. The drawing also illustrates that IP multicast is supported natively by some transport networks, in this case an Ethernet hub.

6.2.3 Resource discovery

Another application for multicast is the discovery of resources on a network. Many applications today rely on broadcasts (sending information to all hosts on a network of computers linked together by means of a network layer like Ethernet) of an interrogation message to find network resources. The Windows[®] operating system is one of them. Broadcast is fine when just a few workstations share a small LAN, but in bigger networks where hundreds of workstations are connected using hubs and switches it becomes a real problem. Because network managers want to avoid broadcast storms as much as possible, they usually configure their routers to not forward broadcasts across subnets. This limits the practical usefulness of broadcast discovery to just the subnet of the broadcasting host.

Multicast is one possible solution to these limitations of broadcast; there are other useful approaches (e.g., the IEEE 802.1 WG defined the notion of VLANs for distributed working groups). Multicast is a way of distributing information to a group which can easily span several subnets and yet reach only the hosts that have requested to be members of the group. Moreover, multicast can be configured to carry out expanding ring searches, so a host can query its immediate neighborhood for a resource without flooding the universe in the first place. The H.323 protocol uses this type of resource discovery to find gatekeepers on the network.

6.3 The multicast framework

6.3.1 Multicast address, multicast group

A ‘multicast IP address’ format has been introduced in IPv4 and IPv6 to support multicast applications, in addition to the existing unicast (pointing to a single destination) and

broadcast (pointing to all hosts on a subnet) addresses. Multicast addresses should not be confused with anycast addresses, which have been added in IPv6: a packet sent to an anycast address must reach one and only one host in a group, while a packet sent to a multicast address must reach all members of the group identified by the multicast address.

In IPv4, a multicast address is a class D address, which ranges from 224.0.0.0 to 239.255.255.255 (all addresses starting with the bit pattern '1110'). Addresses 224.0.0.0 to 224.0.0.255 are reserved for multicast-routing protocols. With the remaining addresses, combined with a port number (from 1,024 to 65 535) in the case of UDP multicast, there are still more than 16 000 billion possibilities for distinct multicast conversations. However, only the IP address part is relevant when building the distribution tree, so applications using distinct ports must share the same distribution trees. In IPv6, multicast addresses will have a high-order octet equal to FF.

There is the same difference between a regular email address and a mailing list address as between a unicast address and a multicast address (Figure 6.5). Clients who subscribe to a particular multicast address will receive all datagrams sent with this multicast address in the destination address field.

A multicast group is a set of hosts that subscribed to the same multicast address. The subscription is done using a protocol called IGMP (Internet Group Membership Protocol). These hosts are called the group members. A group is completely dynamic: at any time a machine can leave or join a group. There is no restriction to the number or location of members in the group.

Note: A client is not required to be a member of a group to send a message to its members. In fact, there is only one significant difference between a mailing list and a multicast group. In the first case, the complete list of members is known to a central server. For multicast, the routers in the network only know if they have at least one member on each interface, without knowing who the members are.

Since groups are completely dynamic, multicast addresses need to be obtained dynamically. The main issue is to choose an address that is not already in use. On the mBone, the addresses already in use can be obtained via the SDR application (see Section 6.7.3.2), but some applications simply choose a random address. The second issue is to make this address known to potential listeners: here again it is possible to use SDR (this has the

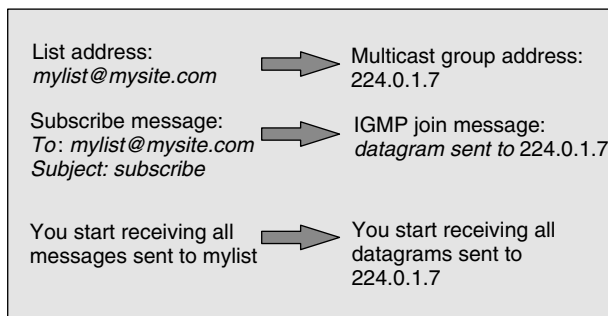


Figure 6.5 Mailing list address versus multicast group address.

advantage of letting everyone know that you are using this address), but a simple web page also serves the purpose if it is known to the potential audience.

A permanent group is just a group with a well-known address (registered by the **Internet Assigned Numbers Authority for IANA**) which is used for a particular application. It does not imply that there is some permanent member in that group. Table 6.1 lists some well-known groups.

Note: TCP cannot be used for multicast communications, and multicast datagrams have to be standard UDP or RAW datagrams, which are delivered to group members with no guarantee. Other reliable transmission mechanisms can be implemented on top of UDP.

6.3.2 Multicast on ethernet

In addition to reserved class D IP addresses, the IANA owns a block of Ethernet addresses reserved for IP multicast, which in hexadecimal begins with 01:00:5E (the first byte of any Ethernet address must be 01 to specify a multicast address). The IANA allocates half of this block for mapping class D IP multicast addresses to IEEE-802 multicast addresses; so, the Ethernet addresses corresponding to IP multicasting are in the range 01:00:5E:00:00:00 through 01:00:5E:7f:ff:ff.

There is no one-to-one mapping. The reason for this can be explained by a bit of history: when Steve Deering first designed IP multicast, he figured out that he would need to buy 16 blocks of 24 bits from IEEE to map all IP multicast addresses. Each block was worth \$2,000, so he was only allowed to use half of a 24-bit block, which accounts for the 23 bits we have today.

This allocation allows for 23 bits in the Ethernet address to correspond to the IP multicast group ID. The mapping places the low-order 23 bits of the multicast group ID into these 23 bits of the Ethernet address (Figure 6.6). Since the upper 5 bits of the multicast address are ignored in this mapping, there is no one-to-one relationship: 32 different multicast group IDs map to each Ethernet address.

Because there is no one-to-one mapping between Ethernet and IP multicast addresses, an Ethernet card can receive and forward to the device driver wrong packets. The device driver or the IP stack of the host must filter out these datagrams by checking the IP

Table 6.1 Some well-known multicast address groups

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 MOSPF routers	224.0.0.5
Routing Information Protocol (RIP)—Version 2	224.0.0.9
Network Time Protocol (NTP)	224.0.1.1
Audio news	224.0.1.7
IETF audio	224.0.1.11
IETF video	224.0.1.12

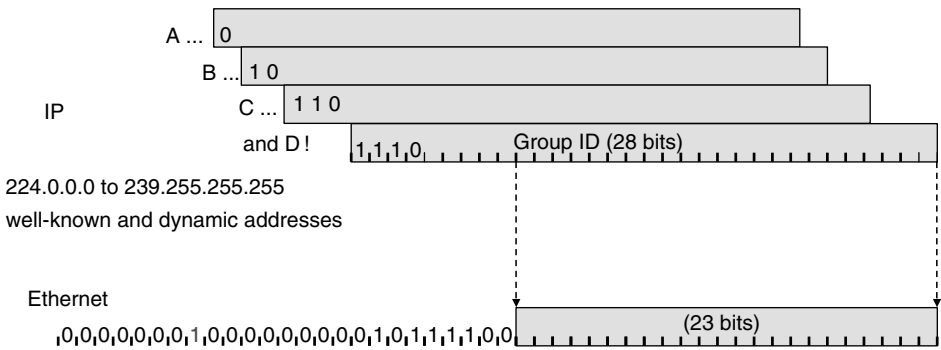


Figure 6.6 Mapping of IP multicast addresses to Ethernet multicast addresses.

destination address. The receiving processes must notify their IP layers that they want to receive datagrams destined for a given multicast address, and the device driver must enable reception of these multicast frames. This process is handled by joining a multicast group.

IP multicasting on a single physical Ethernet network is simple. The sending process specifies a destination IP address that is a multicast address and then the device driver converts this address to the corresponding Ethernet address and sends it.

6.3.3 Group membership protocol

6.3.3.1 IGMPv1

The Internet Group Membership Protocol (IGMP) version 1 is specified in RFC 1112. In the same way as a special form of email is sent to the list server to subscribe to the list, a host sends a group membership protocol datagram to the group IP multicast address in order to become a member of a multicast group. IGMP has been assigned protocol number 2 (RFC 1700).

When a host first subscribes to a multicast group, a couple of IGMP reports are sent to the group address to which the host subscribes with a TTL of 1 (Figure 6.7). Since multicast routers promiscuously receive all multicast traffic (the network interface forwards all packets to the device driver), they get informed of the new member. Because of the TTL, an IGMP message is never forwarded out of the subnet.

On each link, a multicast router is elected to be the ‘querier’ and periodically (every minute, typically) sends an IGMP query message to the all-hosts group (224.0.0.1) with a TTL of 1 (Figure 6.8). All hosts on directly connected subnets are supposed to issue an answer along with an IGMP report sent to each group address to which it belongs. To avoid a synchronized storm of messages, these reports are sent after a random delay. When a host hears a report for a group and is also a member of that group, it resets the timer and keeps silent to avoid duplicate messages. The router will consider that there is no member left for group *G* on a link if it doesn’t hear reports for group *G* after several queries on this link.

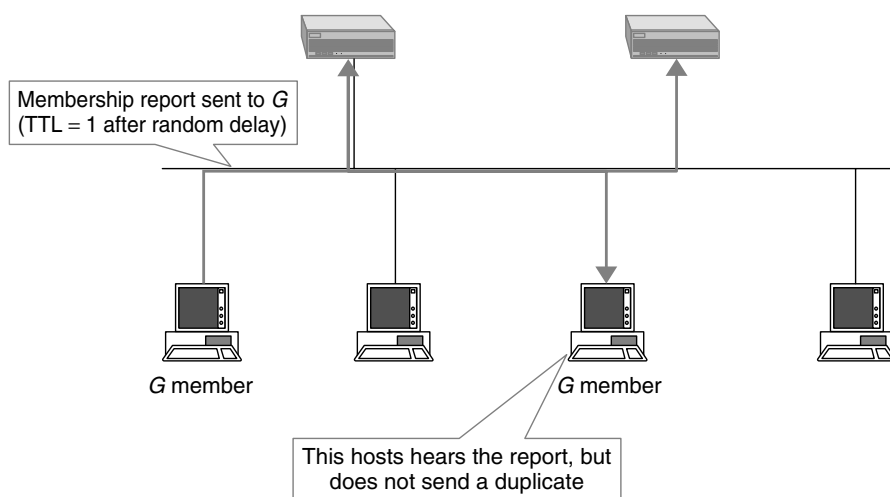


Figure 6.7 Avoiding unnecessary membership reports. When first joining a group, two reports or more are sent without waiting for a query.

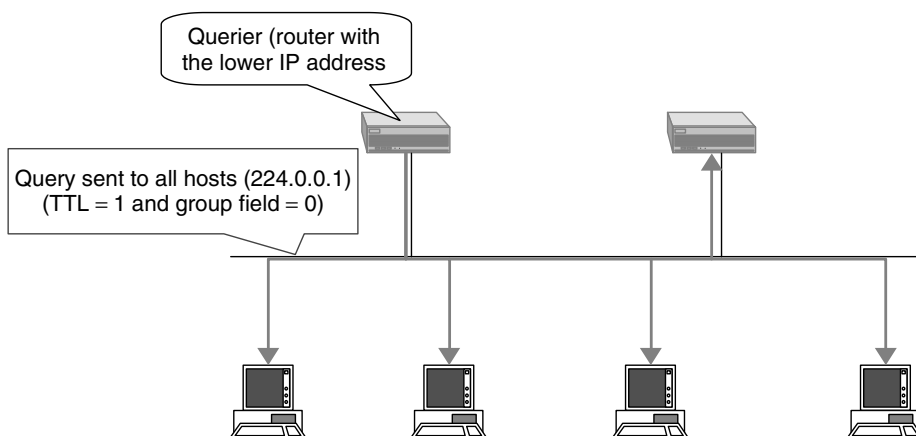


Figure 6.8 Periodic group membership queries by the querier router. Queries are sent every 60–90 s.

In the IGMPv1 format (Figure 6.9), message type 1 is used for queries and message type 2 is used for reports. The group address is either the multicast group concerned by the report or 0 in queries.

Note: IGMP only operates over broadcast LANs or point-to-point links, but there are some ways to extend the subscription mechanism over NBMA (non-broadcast, multiple access) networks, the ‘MARS’ protocol is an example of such a solution over ATM networks.

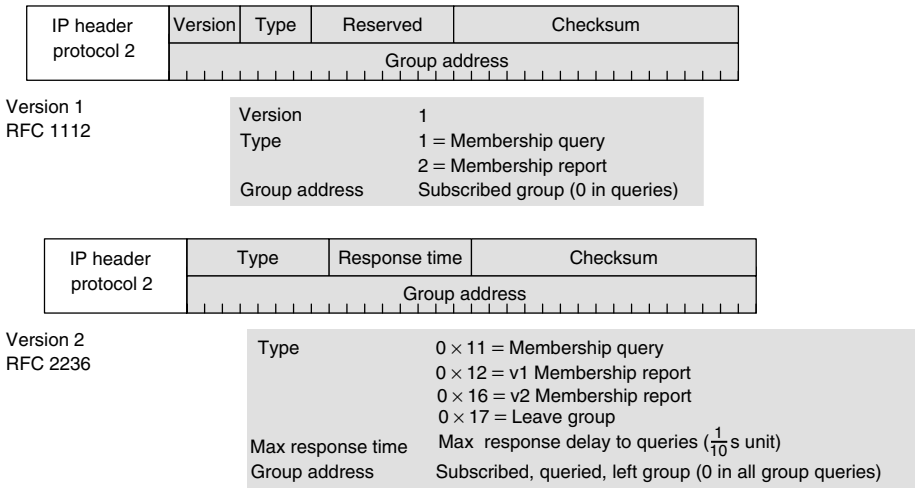


Figure 6.9 IGMPv1 and IGMPv2 message format.

6.3.3.2 IGMPv2

In IGMPv1, a router considers a group has no members left if it does not receive IGMP reports addressed to the group after a number of queries. In the meantime it will keep forwarding useless and bandwidth-consuming datagrams. In IGMPv2, an additional ‘leave group’ message has been defined to reduce the latency of hosts leaving a group (Figure 6.9). IGMPv2 is specified by RFC 2236 and is backward-compatible with v1.

The message fields ‘type’ and ‘version’ have been merged into a new 8-bit-type field (0x11 membership query, 0x12 v1 membership report, 0x16 v2 membership report, 0x17 leave group). The group address now indicates either the group being queried or reported to, or the one that has left. It is left to 0 to query all groups.

The reserved space has been allocated to indicate a maximal response delay in tenths of a second. The ‘leave’ message for a group is sent by a leaving host only if this host is the last one to have effectively sent a report membership for that group (otherwise it knows that there still are other members on the LAN). The querier router then sends a couple of group-specific queries with a small max response time to check no one else is still a member. If no report is heard for the group, then the router considers there are no more members on the LAN.

The querier election for IGMPv2 is very simple: initially all routers send queries and then only the router with the smallest IP address keeps sending queries. If the other routers do not hear queries for some time they restart the election process.

6.3.3.3 IGMPv3

IGMPv3, defined in RFC 3376, adds source selection possibilities, such as listening to some sources only or to all but a set of unwanted sources. This can be used, for instance,

to exclude from large conferences some users who send background noise (e.g., ones who do not know how to switch off their microphones). This also helps to prevent ‘denial of service’ attacks where the hacker sends a stream conflicting with the original session on the same multicast group and port. Because of this, some IGMP query and report messages have been extended to include a list of sources and a new IGMPv3 report type (0x22) has been introduced.

6.4 Controlling scope in multicast applications

6.4.1 Scope versus initial TTL

Like any other IP packet, a multicast datagram has a **TTL** (time to live) field. The TTL is decremented at each hop. When the TTL reaches 0, the packet is discarded by routers. For a unicast packet, this TTL is always set to a rather high value (127, typically) and is just used to prevent routing loops. The TTL field of a multicast datagram is also decremented at each multicast router. But, in addition of preventing routing loops, it is also an indication of how large the scope of the datagram is. If the IP multicast sender is considered to be like a radio station, the initial value of TTL defines the power of the emitter. The larger the TTL, the larger the range that can be reached (Figure 6.10). Therefore, multicast datagrams are usually sent with a small initial TTL.

The TTL can therefore be used as some basic form of ‘power control’ for a multicast session. A multicast session sent using a TTL of 2 can only span a disk centered on the sender with a diameter of 4 routers. Increasing the TTL to 6 would expand this diameter to 12 multicast routers. The broadcast area depends on the source.

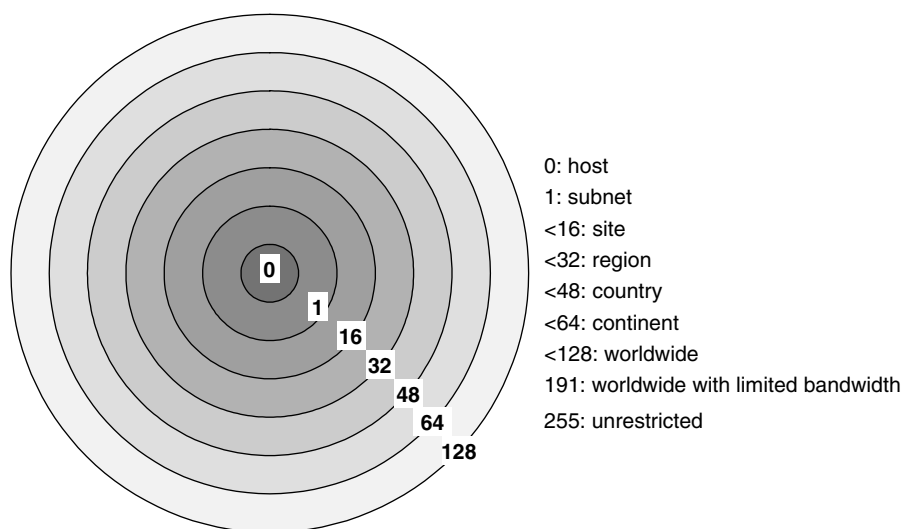


Figure 6.10 Classic TTL conventions.

6.4.2 TTL threshold

The TTL can also be used to set a virtual administrative boundary to a domain which does not depend on the source. All multicast interfaces can be configured to only forward packets having a TTL greater than a preset value (Figure 6.11). If an administrative domain can be approximately defined by a disk of diameter D , then setting the minimal forwarding threshold of all edge routers higher than D will prevent all sessions originating in the domain with a TTL of D to propagate to the outside world. Such sessions with an initial TTL of D will cover the whole domain but stay within the boundary of the edge routers.

This scheme also applies to nested domains (e.g., an internal subdomain could be configured with a threshold of 16 and the parent domain would then have a TTL of 32).

This method of limiting the scope of a multicast broadcast using TTL has a serious limitation: it does not allow administrative domains to overlap. For instance let us take the case of a company that has an engineering department A and an accounting department B, two bookkeepers are in charge of the engineering department and belong to both domains. We want to be able to make engineering-only conferences, accounting-only conferences, and company-wide conferences from any desktop in the relevant domains.

In the set-up shown in Figure 6.12 a conference sent from domain A with a TTL of 16 will stay in domain A. A conference with a TTL of 32 will be company-wide. But

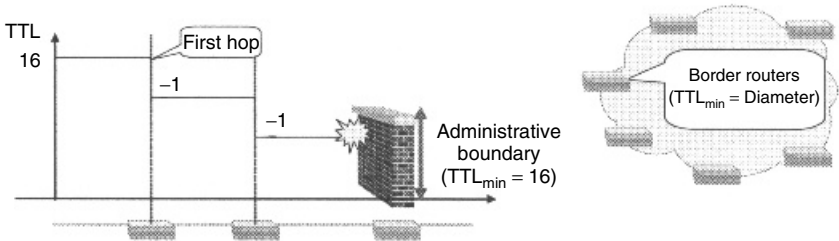


Figure 6.11 Using a TTL threshold to restrict multicast packets to a given domain.

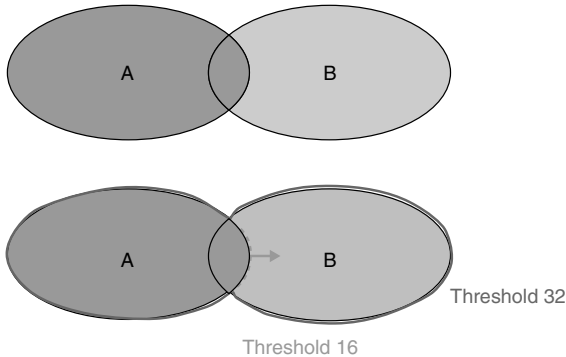


Figure 6.12 TTL threshold cannot be used with overlapping domains.

how can we make a conference for domain B only? If we set the outgoing threshold of the remaining common interfaces (left) to $X > 16$, then it will be impossible to initiate an 'A-only' conference from the bookkeepers' desktops (an initial TTL ≤ 16 would stay in the intersection domain, an initial TTL > 16 would leak in domain B). A threshold X below 16 creates the same impossibility for B-only conferences.

The multicast address range 239.0.0.0 to 239.255.255.255 (administratively scoped addresses) has been reserved to allow administrators to have better control over the scope of a session. Administrators can now configure all edge routers to not forward some addresses in this range. All sessions sent using a multicast address in this range will stay within the domain, regardless of the initial TTL. Overlapping multicast domains can now be configured simply by using different administratively scoped addresses in each of the domains.

Administrative scope is bidirectional: it prevents all 239.x.x.x traffic from getting out and getting in. This is useful since many site administrators on the mBone forget to set the administrative scope and still use software that is set to send 239.x.x.x datagrams.

6.5 Building the multicast delivery tree

With IP multicast, routers are responsible for duplicating the packets and sending them to appropriate interfaces. But, which interfaces are they? In fact the construction of the multicast delivery tree is the most complex issue of the multicast technology. Several techniques can be used, the most common are discussed below.

In the following text we will call a 'source router' any router directly connected to a subnetwork with an active source station.

6.5.1 Flooding and spanning tree

The simplest way to send a packet to every member of a group is flooding. In this technique each router of the IP network replicates every inbound multicast packet to all interfaces except the inbound interface. If the same packet arrives more than once, it is discarded. This is simple and robust (hence its use in some military networks), but clearly not scalable.

An improvement of the flooding algorithm is to select just a subset of Internet routers, but a subset that can still reach any destination. This subset should form a 'spanning tree' of interconnected routers, in which two distinct routers are interconnected by one and only one active path (Figure 6.13). This topology ensures there will be no routing loop, making it unnecessary to detect duplicate packets and making flooding much more efficient. Unfortunately, it is computationally difficult to build a spanning tree for large networks. There are two main types of spanning trees: shared trees and source-rooted trees.

6.5.2 Shared trees

Shared tree techniques use only one spanning tree for the group, independently of the source. A simple way to build a common spanning tree is to choose a 'rendezvous' point. Then all routers willing to receive the datagrams sent to the group send a message toward

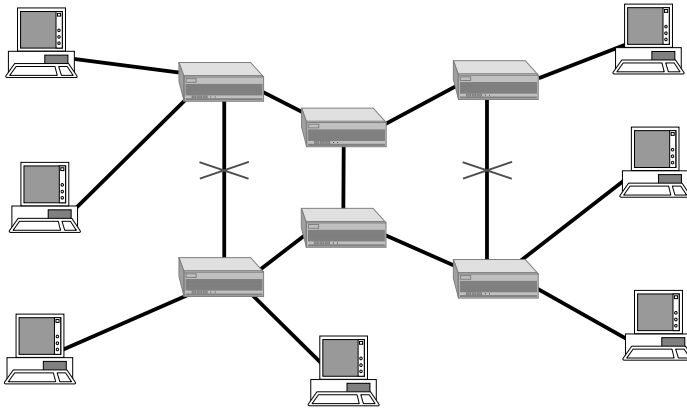


Figure 6.13 Spanning tree (there is exactly one path between any pair of nodes).

the rendezvous point, and each multicast router seeing this message on its way marks the interface from which it arrived and the outgoing interface. Now, any multicast datagram received at the outgoing interface will be copied to all other marked interfaces.

For a source router, sending a datagram to the group is just a matter of sending an encapsulated datagram to the rendezvous point, which unwraps it and forwards a copy to all of its marked interfaces.

6.5.3 Source-based trees

Some algorithms build a different tree for each source router. When a host sends a datagram to the group, the datagram will be duplicated according to the spanning tree rooted at the host's router (Figure 6.14). This leads to more efficient paths and shorter delivery delays.

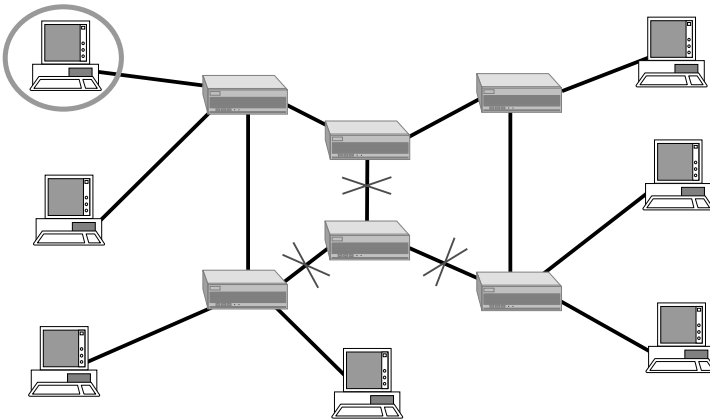


Figure 6.14 Source-based tree.

6.5.3.1 Reverse path broadcasting and truncated reverse path broadcasting

RPB (reverse path broadcasting) is a technique used to build source-based spanning trees. For each source, if the packet arrives on the link that the router believes to be the shortest way back toward the source (this information is derived from the protocol's own routing table in the case of **DVMRP** or from the unicast-routing table in the case of **PIM**), then the router duplicates the packet and forwards it to every interface except the originating one. Otherwise (i.e., if the packet comes from a link that is not the shortest way back to the source), the packet is dropped (Figure 6.15).

The algorithm in Figure 6.15 has one main limitation: it includes all routers and subnets in the tree, even if some of them are not part of the destination multicast group.

A possible enhancement of RPB is **truncated reverse path broadcasting (TRPB)**: here routers use the information obtained with IGMP to avoid sending multicast datagrams to leaf subnets in which no host is a member of the destination multicast group. However, the delivery tree between routers still makes no use of IGMP information, even though some parts of the tree might be useless.

DVMRPv1 (Distance Vector Multicast Routing Protocol), the original mBone-routing protocol, used the TRPB forwarding algorithm. The DVMRP multicast-routing protocol is very similar to the RIP unicast-routing protocol, except that it tracks distances to the source, not the destination.

6.5.3.2 Reverse path multicasting

RPM builds source-based trees that span only subnets with group members and builds routers along the shortest path to subnets with group members. The first packet is forwarded using the TRPB algorithm, but if edge routers see that none of their leaf subnets is a member of the destination group, they send a 'prune' message to the parent router.

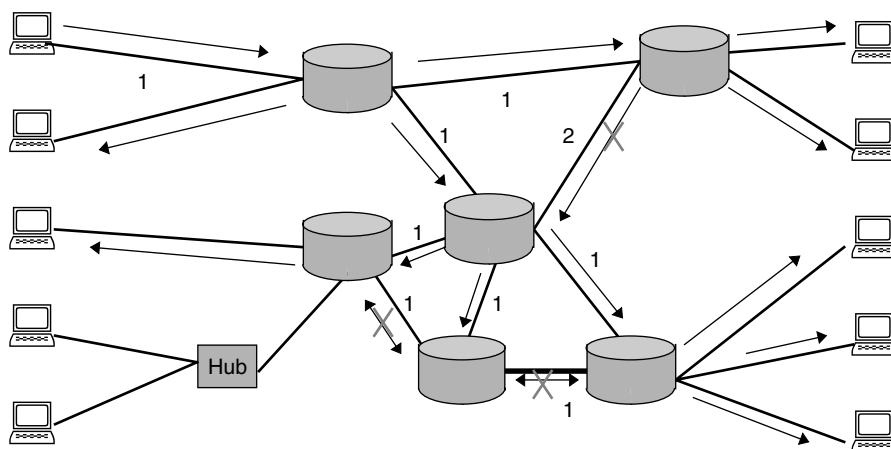


Figure 6.15 Distribution tree with RPB.

The parent router stores this information and disables this child interface for this source and this group. If all child interfaces are disabled for a given source and group, then this router itself sends a prune message upstream.

In order to allow dynamic group expansion, prune information has a limited lifetime, and therefore the network is periodically flooded again with TRPB. RPM is a big improvement over simple TRPB, but it requires routers to store a lot of prune information (for each active [source, group] pair) and periodic flooding wastes some bandwidth. RPM is well suited for networks with a large proportion of edge subnets which have members of multicast groups: it is a ‘dense-mode’ multicast-routing algorithm.

6.6 Multicast-routing protocols

6.6.1 Dense- and sparse-mode protocols

Multicast-routing techniques fall in two broad categories: sparse-mode protocols and dense-mode protocols. Sparse-mode protocols are optimized for large networks where only a small portion of all connected hosts are members of each group. Dense-mode protocols are optimized for networks where most hosts are members of active multicast groups. This is not necessarily small networks (e.g., at an exchange between large ISPs, it is very likely that there will be at least one member in each ISP domain for all active groups).

Technically, sparse-mode protocols tend to use a shared tree, and a router needs to subscribe to a group to become a member. Dense-mode protocols tend to use source-rooted trees and include by default all multicast routers in the distribution tree. Routers need to send prune messages if they are not interested.

The most popular sparse-mode protocols are PIM-SM and CBT. The most popular dense-mode protocols are DVMRPv3 and PIM-DM.

6.6.1.1 DVMRPv3

DVMRPv3 (Distance Vector Multicast Routing Protocol) is a routing protocol that uses an RPM algorithm to forward multicast packets. It is the dominant protocol of the mBone.

As we saw in Section 6.5.3.1, when a router *R* running an RPM algorithm receives a multicast datagram, it needs to know:

- whether the packet was received on the interface closest to the source (**reverse path forwarding**, or **RPF, check**) from the multicast topology perspective. If it is not, then the packet should have been received first by the interface closer to the source, so this packet is probably a duplicate and must be dropped. Note that in most cases all links on the network are not multicast-enabled, so the interface closest to the source from the unicast topology perspective and the interface closest to the source for the multicast topology perspective will often differ. For this reason, DVMRP runs its own routing protocol in order to take multicast topology into account.

- whether the source of this datagram is closer to R or closer to R neighbor routers. If neighbor routers are closer, they will receive the datagram first, so there is no need to forward the current packet to these routers.

In unicast-routing protocols, such as RIP, each router advertises its best route *from* the router *to* each destination for the unicast topology. The result is that each router knows the unicast distance *from* it *to* each destination.

Here, what we really want to know in order to build an optimized distribution tree is the distance *from* the source *to* the router in the *multicast* topology. This is very often the same, but not always, as in the case of asymmetric links or when using tunnels. All current multicast-routing protocols including DVMRP assume that links are symmetric, so the link symmetry issue is currently ignored. DVMRP solves the issue of multicast-specific topology by using its own routing protocol running over multicast-enabled interfaces.

For each directly attached subnet S , a DVMRP router R advertises the distance from S to R to each neighbor router N_i (in the case of Figure 6.16 just one hop). When N receives the notification that S can reach R in h hops, it first checks whether any other router Z has sent a message saying that S is closer to it. (If this is the case, the accessibility notification from R is not forwarded). Otherwise, N will send a message to each of its neighbors saying that S can reach N in $h + d$ hops, where d is the administrative distance

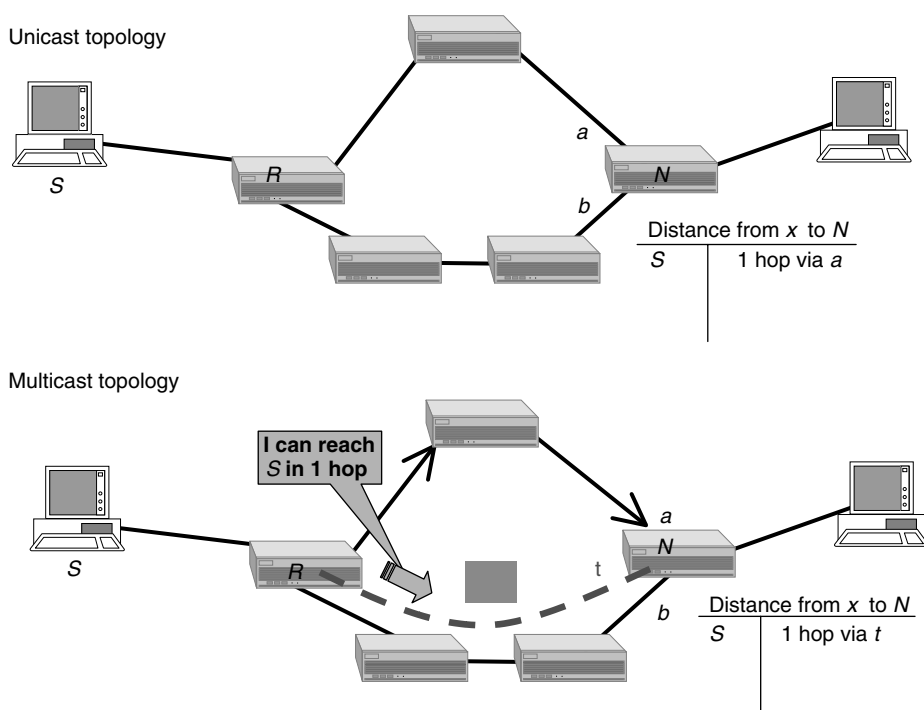


Figure 6.16 Because of tunnel t , the routers see a different topology at unicast and multicast level.

associated with the interface connected to *R*. The interface can be a physical interface or a virtual tunnel interface as in Figure 6.16.

A DVMRP routing table might look like Table 6.2. DVMRP also builds a group-specific forwarding table (Table 6.3) since the routing table does not include group membership information. This table includes by default all interfaces connected to neighbor DVMRP routers (including virtual tunnel interfaces). After prune messages have been received some interfaces are pruned for certain groups (Figure 6.17). On interfaces with directly attached hosts, the forwarding information is based on IGMP queries and reports. Prune states have a lifetime of about 2 hours on the mBone. The number of prunes that routers

Table 6.2 A DVMRP routing table

Source prefix	Subnet mask	From gateway	Metric	Status	Entry lifetime (s)
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

Table 6.3 DVMRP forwarding table

Source subnet prefix	Multicast group	TTL In interface (prunes sent)	Out interface(s) (prunes received)
128.1.0.0	224.1.1.1	200 1	2–3
	224.2.2.2	100 1	2–3
	224.3.3.3	250 1	2
128.2.0.0	224.1.1.1	150 2	2–3

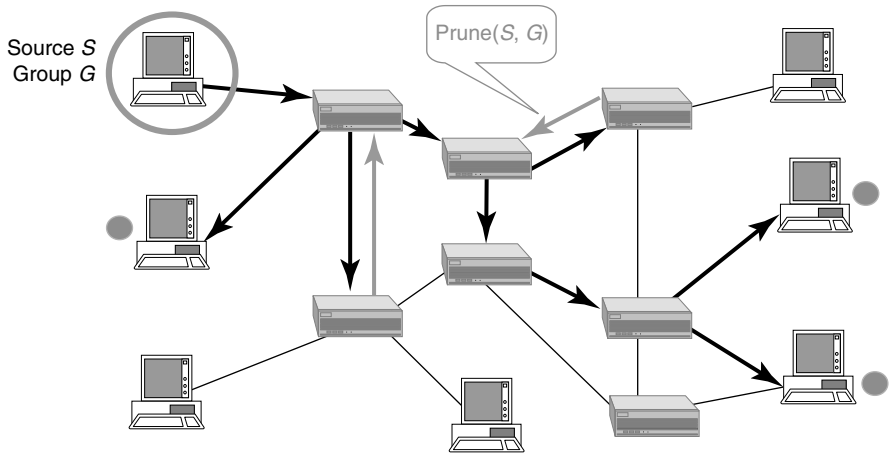


Figure 6.17 Use of the prune message. Subnets with at least one host willing to receive packets of group *G*.

need to maintain (per source, group, and interface) is the main limitation of the scalability of DVMRP. This is the paradox of DVMRP: as the number of listeners increases for a source, the amount of state required in the router decreases. So DVMRP is really a dense-mode protocol!

DVMRPv3 also has a notion of ‘graft’ messages. These graft messages (for each active [source, group] pair) are sent by a router to indicate that it is willing to reattach to a multicast tree for which it had previously sent a prune message (Figure 6.18).

All messages exchanged by DVMRP routers are encapsulated in IP datagrams with protocol number 2 (IGMP) and IGMP packet type 0x13.

Some further improvements of DVMRP are underway, such as **CIDR**-like aggregation. The main issue with the scalability of DVMRP is the periodic flooding that occurs when prune states expire. All DVMRP routers will receive unwanted multicast traffic until they have returned a prune. However, measurements made on the mBone show that this is not yet a real problem. Figure 6.19 is a graph of flooding activity for two pruned sessions (one audio and one video) which can be found on <http://ganef.cs.ucla.edu/~mbone/tunnel.html>. The graph in Figure 6.19 shows that most of the time the session is pruned back immediately after the first packet of the session reaches the router; so, flooding activity is really minimal! The aggregate flood/prune rate for all sessions typically never exceeds 40 packets/s.

6.6.2 Other protocols

6.6.2.1 MOSPF

6.6.2.1.1 Description of operation in a single MOSPF area

The multicast extension to OSPF is described in RFC 1584. **MOSPF** uses the link state information built by OSPF to calculate a shortest path tree on the fly for each

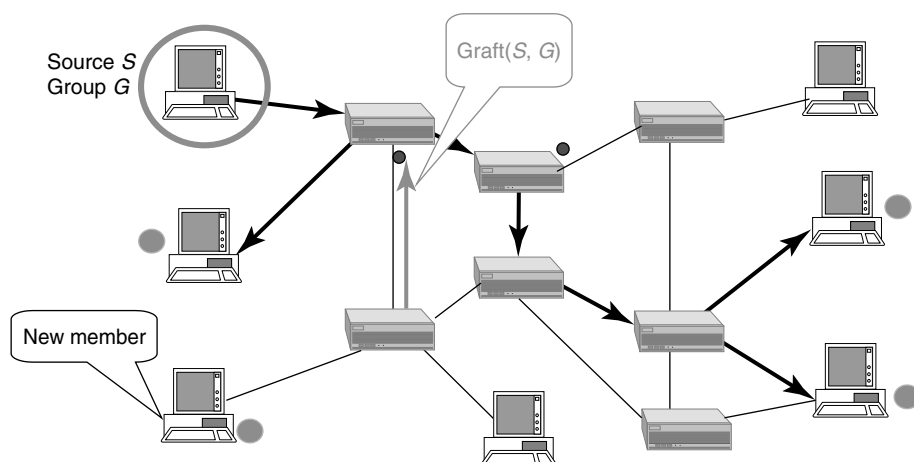


Figure 6.18 Use of the graft message.

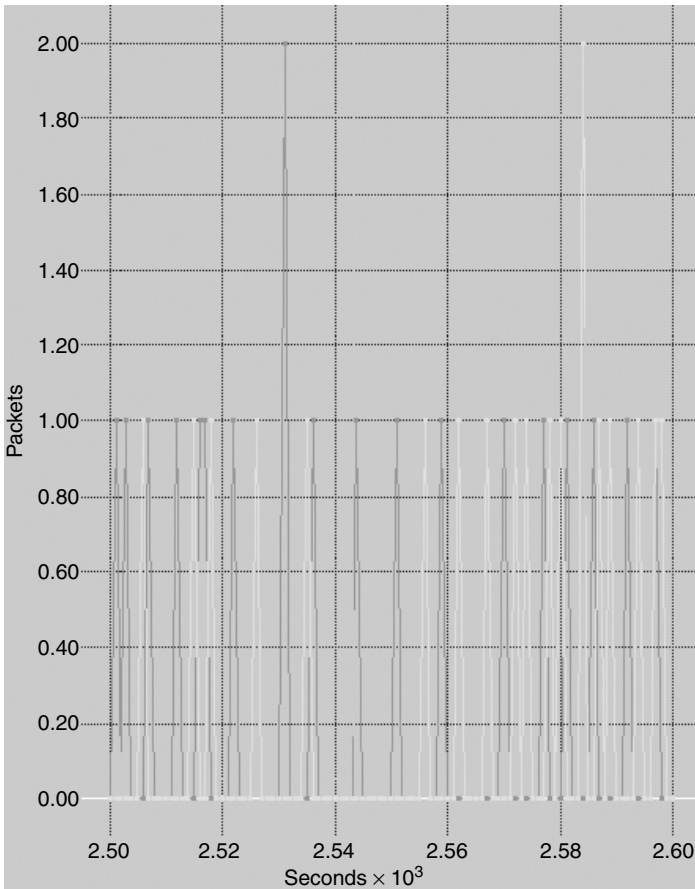


Figure 6.19 Periodic flooding on a typical mBone access.

[source, group] pair. The router knows the multicast topology because link-state advertisements (LSAs) comprise a multicast-capable bit (Figure 6.20), so the tree spans only MOSPF routers.

In addition to the regular OSPF-routing table, each MOSPF router maintains a group membership table. On each subnet, one or two MOSPF routers maintain multicast group memberships in a local group database using IGMP: the designated router (DR) performs IGMP queries on each subnet, and both the DR and the backup designated router (BDR) listen to IGMP host membership reports. The DR then floods the entire OSPF area with ‘group membership link-state advertisements’.

Since each MOSPF router has all the necessary information locally, the multicast tree built using Dijkstra’s algorithm only spans subnetworks that have members of the group, so it does not have to be pruned. This is the major difference with DVMRP (i.e., DVMRP floods the networks whenever there is a new multicast flow and when the prune state expires).

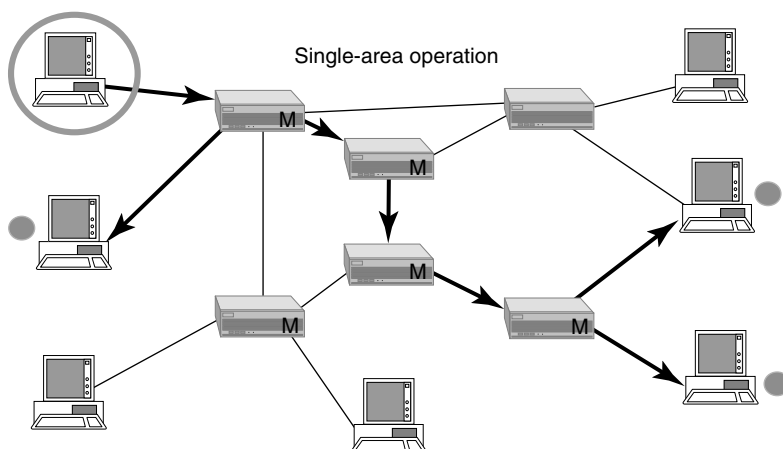


Figure 6.20 MOSPF distribution tree.

6.6.2.1.2 Inter-area routing

In OSPF, **area border routers (ABRs)** are used to forward datagrams outside the OSPF area (Figure 6.21). In MOSPF, some are also configured to act as inter-area multicast forwarders. An inter-area multicast forwarder sends new group membership LSAs to the backbone area for each group that has at least one member within the local OSPF area. The inter-area multicast forwarder is a ‘wild card multicast receiver’ for the local OSPF area (i.e., it receives all multicast traffic generated within that OSPF area and decides whether to forward it to the backbone based on the LSAs received from the backbone).

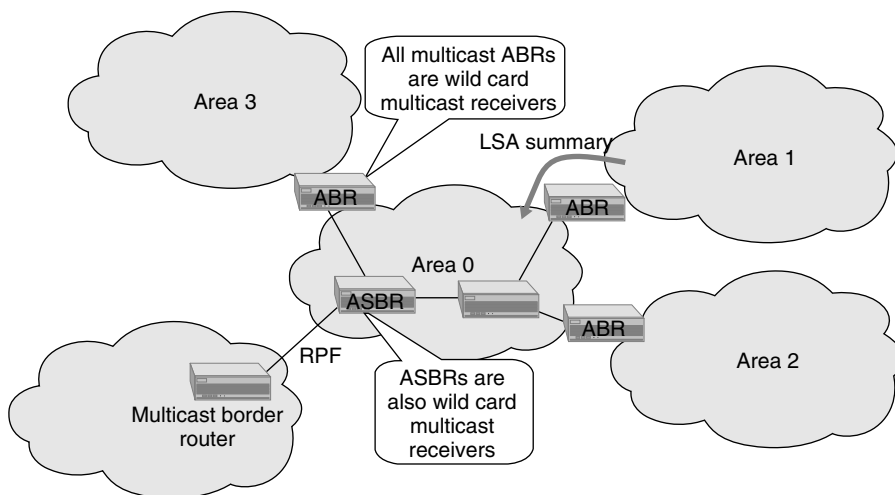


Figure 6.21 MOSPF with multiple areas. Area multicast border router and AS boundary routers handle inter-area and inter-AS multicasting.

6.6.2.2 PIM

The Inter-Domain Multicast Routing working group of the IETF is tasked with developing a set of standards describing multicast-routing protocols. For the moment the working group has defined PIM (protocol-independent multicast), which comes in two flavors: PIM dense mode and PIM sparse mode. PIM-DM and PIM-SM must be used in separate multicast domains; however, packet forwarding and control messages operate seamlessly between the two.

6.6.2.2.1 PIM-DM

PIM-DM (dense mode) relies on the routing tables established by any unicast-routing protocol. This topology information is used to find the route back to the source and build a spanning tree using the reverse path multicasting algorithm. PIM-DM forwards the multicast packets to all downstream interfaces (flooding) until a prune message is received (Figure 6.22). By comparison, DVMRP determines ‘child’ interfaces (i.e., interfaces that are known to be on the shortest path back to the source from the downstream router).

PIM-DM also uses graft messages to reattach a pruned part of the delivery tree if a new member joins the group.

6.6.2.2.2 PIM-SM

PIM-SM (sparse mode) is specified in RFC 2362. By design, PIM-SM is suited for WAN nets that have limited bandwidth and scarce group members. With this constraint, it is impossible to use flooding; so, DVMRP would not scale well.

With PIM-SM, designated routers must explicitly join a group by sending a ‘join’ message to a rendezvous point (RP) for that group (Figure 6.23). There is only one RP per group; this is determined among candidate routers by a deterministic hash function of the group address. Each multicast router in the path of the join message to the RP creates a forwarding entry for that group.

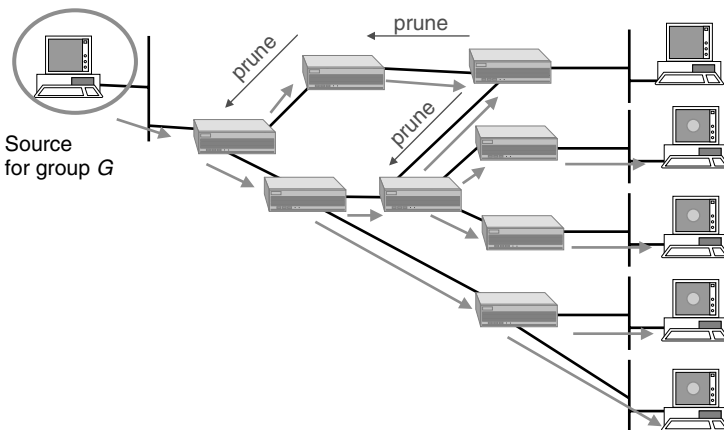


Figure 6.22 PIM-DM also uses prune messages.

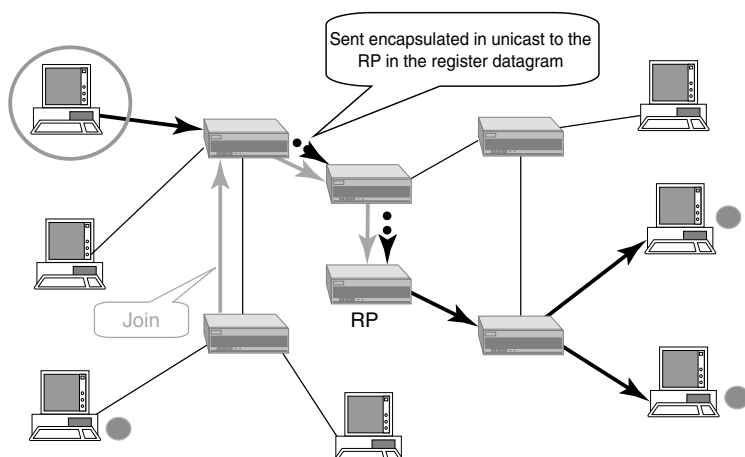


Figure 6.23 PIM-SM uses a rendezvous point for each group.

The first packet of a new multicast stream is sent encapsulated in a unicast ‘register’ packet to the RP. Each router in the path of this register packet creates a forwarding entry so that future multicast datagrams for this group can be sent unencapsulated.

If the traffic from a source exceeds a certain threshold, the last hop router has the option (it is in no way mandatory) to stop using the RP for that source and build a source-based shortest path tree by sending a join message toward the source of the stream. Once the tree is built, the last hop router sends a prune message for that source to the rendezvous point (Figure 6.24).

6.6.2.3 Core-based trees

CBTs (RFC 2201) have been designed to be used in the context of very large networks, where scalability issues can prevent the use of other RPM-based multicast techniques.

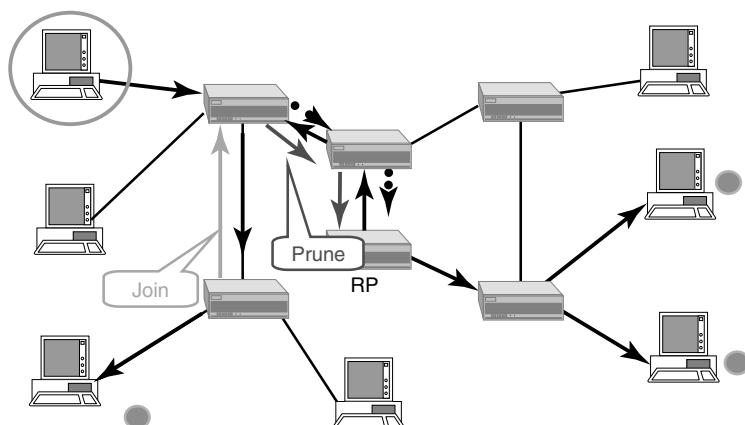


Figure 6.24 PIM-SM allows last-hop routers to switch to a source-based tree.

CBTs use a bidirectional shared tree. Many features are identical to PIM-SM, including the notion of a rendezvous point (called a core router) and an election mechanism. However, the CBT design does not allow shortcuts to be established; this was presented by CBT designers as a feature to preserve the scalability features of CBT. Because the tree to the rendezvous point is bidirectional, routers that are already attached to a group do not need to encapsulate their multicast messages sent to the same group.

6.7 The mBone

6.7.1 An experimental network that triggered the deployment of commercial multicast networks

The mBone started as an experimental network with just 40 subnets in 4 countries in 1992; by January 1998 there were nearly 6,000 subnets. It was composed of islands of multicast routers interconnected by tunnels over regular Internet links, in which case multicast datagrams are conveyed on the tunnels as IP over IP datagrams (protocol 4).

Today, many service providers offer multicast support as a commercial service, not just for experiments, either for their own TV over IP offerings or for corporations looking for efficient broadcast over IP capabilities.

6.7.2 Routing protocols and topology

Most routers run DVMRP (MOSPFv2 does not handle tunnels), but the islands themselves may run MOSPF, PIM, or CBT. The mBone was structured around main nodes, often universities or research labs, which in turn offered multicast connectivity to smaller networks. Figure 6.25 shows the main nodes of the mBone in France back in 1996.

6.7.3 mBone applications

Today, many commercial multicast applications exist; however, for a first contact with multicast, the tools developed for use on the mBone offer a useful introduction. These applications help us to better understand the issues and limitations of SDP for use in VoIP. SDP was really designed for multicast conferences.

6.7.3.1 Videoconferencing with RTP on multicast networks

On unicast networks, RTP can be used for point-to-point communications, but it requires a mixer or multi-unicast for multipoint communications. On a multicast network, such as the mBone, RTP and RTCP packets can be broadcast to all participants, and mixing is done locally by the receiving software.

their video streams. They can also dynamically change the audio codec used, since the codec used can be learned from the value of the RTP packet payload type.

6.7.3.2 SDR

SDR (session directory) is a tool based on the Session Announcement Protocol (SAP). It is used to list announced sessions on the mBone and could be used to advertise new sessions (Figure 6.26). Depending on the version, the SDR tool can also launch and automatically configure some multicast applications from SAP data.

SDR uses 239.255.255.255 for local scope groups and 224.2.127.254 for global scope groups. For administratively scoped groups, the highest address in the scoped range should be used. Any UDP port is suitable, but the tradition is to use port 9875.

SAP is a simple text-based protocol, most of whose data fields are self-explanatory. For the media description portion, it uses the Session Description Protocol (SDP). For more details about SDP, refer chapter 3 in *IP Telephony: Deploying Voice-over-IP Protocols*. The multicast origin of SDP in a send once, receive many and loose coupling context explains its shortcoming when used in duplex, one-to-one, interactive applications such

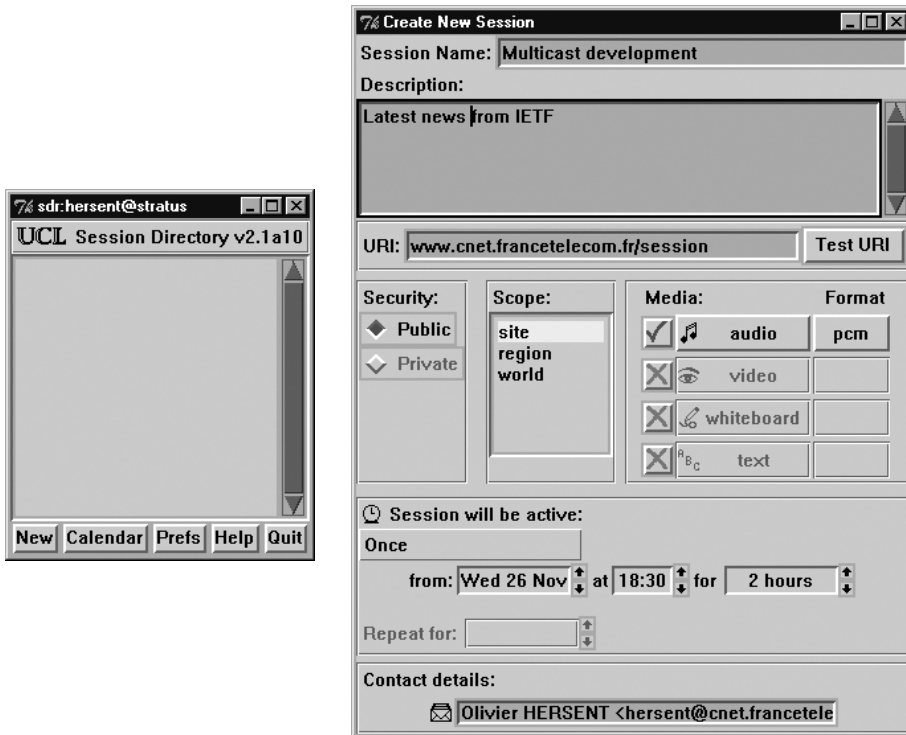


Figure 6.26 The SDR tool. SDR is used for address assignment, scoping session advertisement, and automatic application launching.

a SIP-based videoconferences: SIP had to add the offer/answer model to SDP, which was not used originally. Listed below is an example announcement using SAP with SDP encoding:

```
SAP: 596 bytes
version: 0
message type: 0
encrypt: 0
compress: 0
auth length: 0
msgid: 8192
address: 130.240.64.20
v=0
```

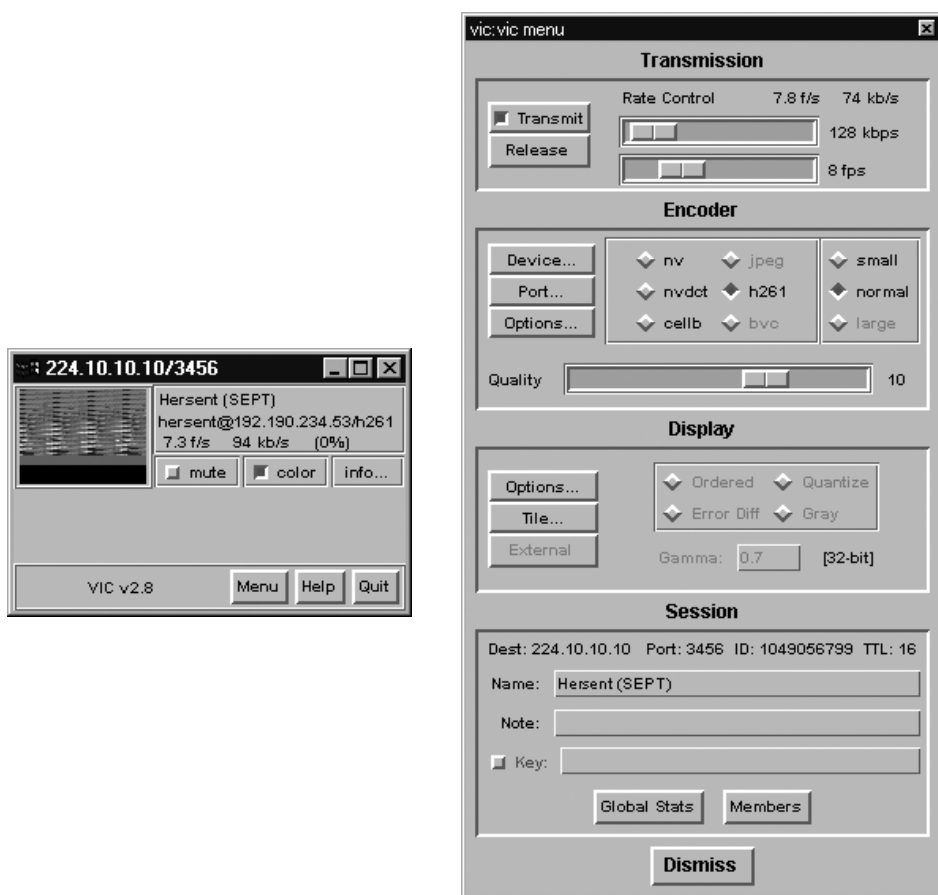


Figure 6.27 Some of the tools that were commonly used on the mBone: VIC video tool, VAT audio tool, wide-band shared whiteboard, and network text editor.

```
o=demo 3066564173 3066564269 IN IP4 130.240.64.67
s=Places all over the world
i=Low bandwidth video (10 kb/s) with views from all over the
world. It is probably wise to limit the overall bandwidth
to 100 kb/s (that is, a maximum of ten 10 kb/s streams).
Audio is primarily for feedback for the senders of video.
e=John Doe <Doe@mydomain.org>
c=IN IP4 224.2.172.238/127
t=0 0
a=tool:mStar 1.0beta1
a=type:broadcast
m=video 51482 RTP/AVP 31
c=IN IP4 224.2.172.238/127
m=audio 20154 RTP/AVP 0
c=IN IP4 224.2.213.113/127
a=rtppred1:5
a=pptime:40
a=rtppred2:5
a=rtppmap:121 red/8000
```

6.7.3.3 VIC and VAT

The VIC and VAT tools are among the very first interactive audio and video applications (Figure 6.27). The limited number of hosts connected to the mBone are mostly confined to universities. Still, even today the VIC and VAT tools are much better suited for large-scale conferencing and broadcasting than most commercial applications.

6.8 MULTICAST issues on non-broadcast media

6.8.1 Bridged LANs

Modern LANs use bridges to reduce the number of collisions. A bridge forwards a packet only to the segment on which the machine with the destination MAC address has been detected. Packets with multicast MAC addresses are traditionally forwarded on all interfaces, which is wasteful. There are several solutions to improve the situation.

6.8.2 IGMP snooping

This solution requires the bridge to inspect all multicast frames in order to decode IGMP reports. This allows the bridge to discover where the receivers are. In addition it decodes router messages like IGMP queries, DVMRP probes, and MOSPF and PIM hellos to discover the position of multicast routers (connected multicast routers need to receive all multicast traffic).

Because hosts never send duplicate IGMP reports, the bridge does not forward a report heard on one segment to another segment in order to see which hosts are receivers on each segment.

This solution has some potential drawbacks: because it relies on the content of IP multicast messages, it does not work for non-IP multicasts and even for IP it may stop working for new IP multicast algorithms. In addition, the inspection of all multicast frames possibly has an impact on performance.

6.8.3 Cisco group management protocol (CGMP)

There is currently no public specification of this proprietary protocol. The idea is to let the router add forwarding entries to the bridge's tables. The router sends CGMP control messages to the bridges. The bridge datagram-forwarding mechanism is left untouched only multicast MAC addresses are added to the forwarding tables for the segments on which the router has detected a member of the multicast group.

6.8.4 IEEE GMRP

GMRP (GARP¹ Multicast Registration Protocol), defined by IEEE 802.1p, is analogous to IGMP at the MAC layer. Hosts wanting to receive frames with a particular multicast MAC address send a GMRP message to the bridge. The bridge propagates this information to the other bridges.

Therefore, a host compatible with GMRP must, after sending the IGMP message to the IP layer, send a GMRP message at the MAC layer.

6.9 Conclusion

VoIP was born on multicast networks with the help of tools like VIC and VAT. Since then VoIP has grown independently on unicast networks, adding to protocols like UDP the missing features required to fully support telephony.

Now that residential VoIP networks increasingly frequently include video on demand and television over IP offerings ('triple play'), VoIP has come face to face with multicast again.

We believe that the combination of VoIP and multicast-enabled tools will open a whole new range of applications for education, remote learning, reporting, and gaming. In the coming years, we will get used to communicating using video and will generate more and more video content (3 G phones, etc.): the combination of multicast and VoIP will enable us to interact and communicate more efficiently using video content.

¹ GARP stands for Generic Attribute Registration Protocol (formerly Group Address Resolution Protocol).

6.10 References

- IGMP version 1: RFC 1112.
- IGMP version 2: W. Fenner, *Internet Group Management Protocol, Version 2*. RFC 2236, November 1997.
- DVMRP: original RFC 1065, 1075, DVMRPv3 *Distance Vector Multicast Routing Protocol* <draft-ietf-idmr-dvmrp-v3-06.txt>.
- MOSPF: RFC 1584, 1585.
- CBT: A. Ballardie, *Core based trees (CBT version 2) multicast routing*: Protocol specification. RFC 2189, September 1997.
- PIM-SM: Estrin, Farinacci, Helmy, Thaler, Deering, Handley, Jacobson, Liu, Sharma, and Wei. *Protocol independent multicast-sparse mode (PIM-SM)*: Protocol Specification. RFC 2117, June 1997.
- Domain interoperability: *Interoperability Rules for Multicast Routing Protocols*
- <draft-thaler-multicast-interop-02.txt> (Exp Sept 98).
- Border Multicast Protocol: *Border Gateway Multicast Protocol* <draft-ietf-idmr-gum-01.txt>.
- HDVMP: A.S. Thyagarajan and S.E. Deering. Hierarchical distance-vector multicast routing for the MBone. In: *Proceedings of the ACM SIGCOMM*, pp. 60–66, October 1995.
- MBGP: *Border Gateway Multicast Protocol* <draft-ietf-idmr-gum-02.txt> (Thaler, Estrin, and Meyer).
- MASC: *Multicast-Address-Set advertisement and Claim mechanism* <draft-ietf-idmr-masc-00.txt> (Estrin, Handley, and D. Thaler).
- Multicast address allocation extensions to the Dynamic Host Configuration Protocol <draft-ietf-dhc-mdhcp-03.txt> (S. Patel).
- Multicast address allocation extensions options <draft-ietf-dhc-multopt-02.txt> (S. Patel).
- AAP: Multicast Address Allocation Protocol <draft-handley-aap-00.txt> (Handley).
- IPv6: R. Hinden and S. Deering, *IPv6 Multicast Address Assignments*, <draft-ietf-ipngwg-multicast-assgn-04.txt>, July 1997.
- BGP4+: T. Bates, R. Chandra, D. Katz, and Y. Rekhter, *Multiprotocol Extensions for BGP-4*, RFC 2283, February 1998.
- BGP4 Y. Rekhter and T. Li, *A Border Gateway Protocol 4 (BGP-4)*, RFC 1771, March 1995.
- Multicast and firewalls: <draft-finlayson-mcast-firewall-00.txt> *IP Multicast and Firewalls*.
- Multicast: S. Deering, *Host Extensions for IP Multicasting*, RFC 1112, August 1989.
- Firewalls: N. Freed, and K. Carosso, *An Internet Firewall Transparency Requirement*, Work-in-Progress, Internet-Draft <draft-freed-firewall-req-02.txt>, December 1997.
- RTP: H. Schulzrinne, S. Casner, R. Frederick, and V. Jacobson, *RTP: A Transport Protocol for Real-Time Applications*, RFC 1889, January 1996.
- SAP: M. Handley, *Session Announcement Protocol*, Work-in-Progress, Internet-Draft <draft-ietf-mmusic-sap-00.txt,ps>.
- Administrative scoping: D. Meyer, *Administratively Scoped IP Multicast*, Work-in-Progress, Internet-Draft <draft-ietf-mboned-admin-ip-space-04.txt>, November 1997.
- Domain-wide reports: B. Fenner, *Domain Wide Multicast Group Membership Reports*, Work-in-Progress, Internet-Draft <draft-ietf-idmr-membership-reports-00.txt>, November, 1997.
- UMTp: R. Finlayson, *The UDP Multicast Tunneling Protocol*, Work-in-Progress, Internet-Draft <draft-finlayson-umtp-02.txt>, February 1998.
- SOCKS: M. Leech, M. Ganis, Y. Lee, R. Kuris, D. Koblas, and L. Jones, *SOCKS Protocol Version 5*, RFC 1928, April 1996.
- SOCKS: D. Chouinard, *SOCKS V5 UDP and Multicast Extensions*, Work-in-Progress, Internet-Draft <draft-chouinard-aft-socksv5-mult-00.txt>, July 1997.