Multicast and Streaming Media

Lecture date: 07-Nov-2000 by: Bruce Maggs

scribe notes by: Simon Goldsmith

Contents

1 Multicast

Multicasting is when one source sends a message to a specific set of receivers. There are a variety of applications for this operation, with different applications requiring different amounts of bandwidth and accepting various degrees of latency. Delivery of a high bandwidth stream of data to multiple users requires an efficient mechanism.

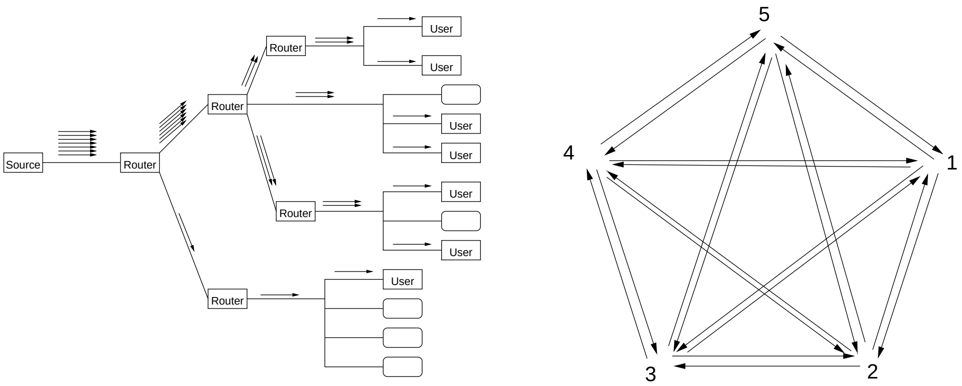

Figure 1: (left) Multicast via Unicast and (right) all to all Multicast via Unicast

1.1 The naive approach

The naive approach to multicast would be for the source to simply send a distinct copy of the message to each receiver. This approach is terribly inefficient in terms of the total number of messages sent, and can easily saturate network links (either the link coming out from the sender or some intermediate link on the way to the many receivers). In the case of an all-to-all multicast (e.g. video conference), this approach becomes even worse - the number of messages sent is quadratic in the number of participants.

1.2 Broadcast

Both IP and Ethernet support the related operation of broadcast. In a broadcast, one source sends a message to all receivers. One solution to multicast is to broadcast the message to all hosts on the subnet / ethernet using IP and ethernet broadcast addresses.

- IP subnet broadcast addresses: subnet mask + all 1 bits e.g. 128.2.255.255
- Ethernet MAC-layer broadcast address: FF:FF:FF:FF:FF:FF

Multicast via broadcast can reduce traffic on a single network (as compared to the naive approach), but becomes inefficient when most hosts do not want to participate. Even a host that does not wish to receive the broadcasted message must read each broadcast packet and identify it as unwanted broadcast traffic before discarding it. Needless to say, requiring hosts which do not wish to participate in a multicast to have to do work as a result of it is an undesirable property.

1.3 Efficient Multicast - Defined

Ideally, a multicast protocol should ensure that

- each network link carries at most one copy of the multicasted message, and
- only those hosts that want the message get it (said another way: no host outside the multicast group receives the message)

The paths taken by multicast messages should always have a tree structure - i.e., one source, many destinations, no cycles. A cycles in the path structure can never help delivery. Avoiding cycles in constructing the paths from the source to the receivers is something that has to be carefully built into the multicast protocol. A tree structure avoids cycles and prevents repeated transmission of the same message across one link.

1.4 Broadcast vs. Multicast

Broadcast is when the source sends a message to all hosts on an ethernet or subnet. Multicast is when the source sends a message to a specific subset of hosts.

Both IP and Ethernet also support multicast through the use of special multicast addresses:

- IP Class D addresses (binary 11110...) are reserved for multicast.
- Ethernet type 0800 (as specified in packet header) denotes a multicast packet. These packets use hardware addresses 01:00:5E:00:00:00 - 01:00:5E:7F:FF:FF

A few words on the use of multicast addresses: One class D (multicast) IP address is used by many hosts distributed throughout the Internet. All members of a multicast group use the same address. The routers involved may make some attempt at redundancy checking - to make sure two multicast sources are not using the same broadcast address - but such checking is not well supported in practice.

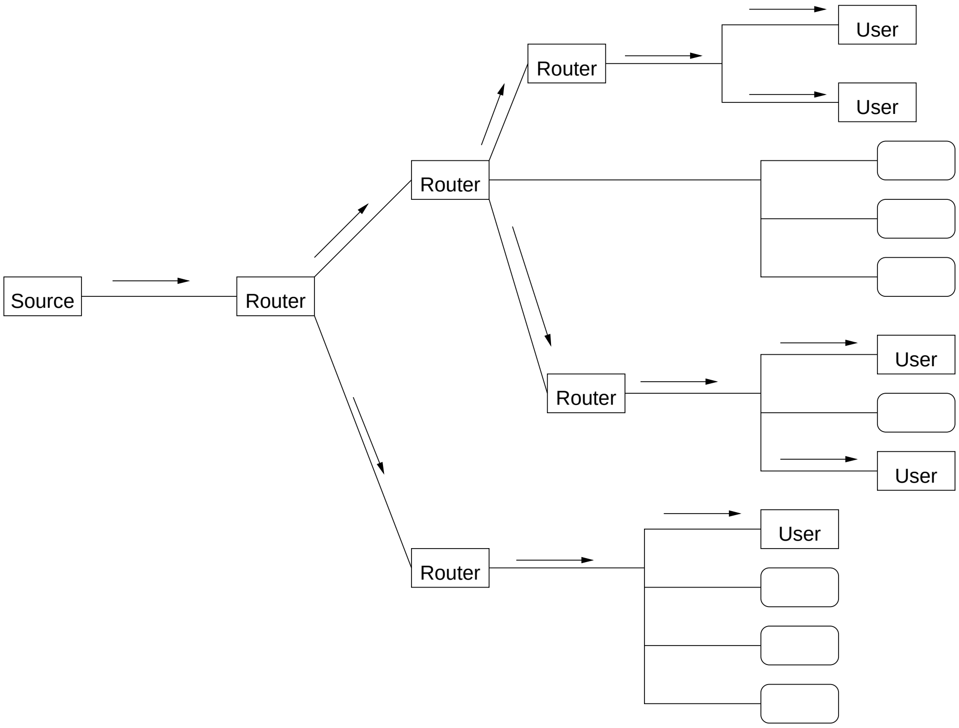

Figure 2: ideal multicast

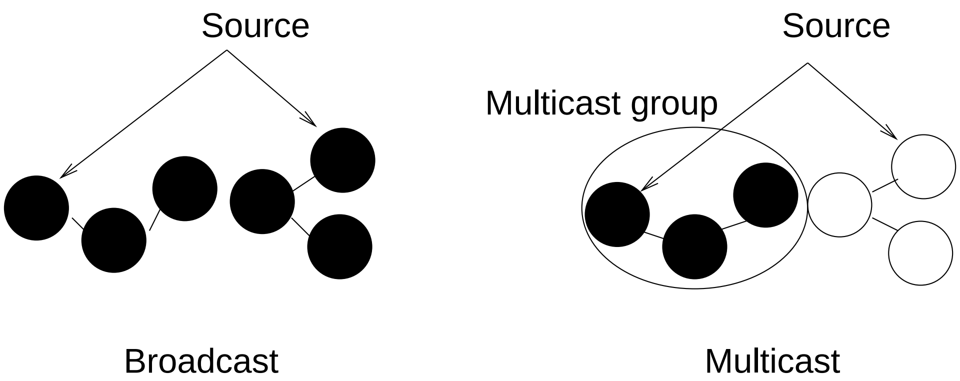

Figure 3: Broadcast versus Multicast

1.5 MBONE

MBONE is an experimental multicast backbone - many ISPs now support it. Since not all routers are multicast enabled, MBONE routers must sometimes tunnel data over ordinary unicast IP.

1.6 Building the Multicast Tree

Multicast addresses cannot be advertised (via BGP) like normal addresses - it would make a mess. Many hosts share a multicast address. In order to efficiently route multicast traffic, routers need to know to copy packets if they are a link where the multicast tree fans out. They must also ensure that the paths that the packets of a multicast message take are indeed acyclic (i.e., the paths form a tree). If multicast packets traveled in a cycle and were copied at each step, it is not hard to imagine an exponential blowup that would bring the routers in the cycle to their knees. Only subnetworks with group members (receivers) and subnetworks with routers on the shortest path from the source to those subnetworks should receive multicast traffic.

We will now build up to a discussion of Distance Vector Multicast Routing Protocol (DVMRP), the multicast routing protocol for MBONE.

1.6.1 RPB

The Reverse Path Broadcasting (RPB) algorithm is a means by which a sender of a multicast message can notify all hosts on the network of the message's availability. The idea behind RPB is for each router to ensure tree structure by only passing messages along shortest paths from receivers back to the source. (It is not difficult to convince yourself that such a set of paths must form a tree with the source as the root.) Let A and B be two arbitrary routers which are connected by a link ab. The following rules are sufficient to ensure tree structure:

- never forward a message back on the same link it arrived on
- each message is sent on each link at most once
- A forwards a packet to B iff B's next hop to the source is A

This third rule bears closer inspection. For A to know whether to forward a packet to B, it must know if it is B's next hop to the source. Normal routing tables (built from e.g. BGP) do not contain this information. Thus, for the purposes of RPB, neighboring routers advertise their next hop to the source to each other. This additional knowledge guarantees that each router will receive packets from the source from exactly one of their neighbors (their next hop to the source).

Note that the path in the multicast tree from the source to some router A will be the shortest path *from A to the source* and not necessarily vice versa. This is because A only has enough information available locally to know if a packet is coming in on its shortest path to the source - not if a packet is coming in on the shortest path from the source to it.

1.6.2 RPM

Suppose a source has a stream available that it would like to multicast. It sends out an initial message, which RPB will ensure makes it to every potentially interested host without any duplication. From here hosts will reply (or not) indicating whether or not they want to receive the multicast message. From these replies RPB can then construct a multicast tree, but since RPM is a refinement of RPB, we'll look at RPM's way of constructing a tree.

Reverse Path Multicasting (RPM) can be thought of as RPB plus prune messages. A prune message is a way for a node in the multicast tree to tell its parent that it does not wish to receive any more multicast packets (from a particular multicast source). So after the initial notification, any router which does not wish to receive the multicast message will send a prune message to its parent in the multicast tree (and thus receive no more traffic for that message). If at any time a router finds that no host in its subnet wants a multicast message and no routers are depending on it to send the message packets on to them, it will send a prune message to its parent and leave the multicast group.

RPB and RPM are said to be receiver oriented multicast schemes. The building of the multicast tree is not managed by the source, but rather by the routers along the way. Building of the tree starts locally when a node decides whether to send a prune message or not.

1.6.3 DVMRP

RPM is an algorithm for building a multicast tree. Distance Vector Multicasting Routing Protocol (DVMRP) is a routing protocol that implements RPM. DVMRP is a distance vector protocol (like RIP) - i.e., paths through the network are assigned weights based on a metric.

The notion of shortest path for multicast routing is not fully captured by simple hop count. In particular, tunnels are considered to be more costly than links directly connecting multicast enabled routers. Furthermore, some tunnels have better metric values than others. Each tunnel through MBONE has a metric value assigned to it and a time-to-live (TTL) threshold. This TTL threshold limits the scope of multicast (i.e. a packet with TTL $<$ threshold is dropped). Routers in MBONE run DVMRP or MOSPF.

DVMRP Grafting is a facility that allows a host or router to rejoin the multicast group shortly after it has sent a prune message to drop out. Routers keep the state for a multicast session around for a while after sending a prune message to facilitate DVMRP Grafting.
Some in class questions:
Q: When / how fast is the tree built?
A: The tree is built rather quickly after the source announces that it has a stream.

Q: Is there any sort of central authority to regulate multicast sessions?
A: Basically, no. Routers will try to make sure two multicast sources aren't using the same multicast address, but that is not well supported. All you need to start multicasting is a connection to a multicast enabled router. So generating a large amount of traffic by multicasting is fairly easy.

2 Quality of Service (QoS) Strategies

For applications such as live streaming audio (and/or video), voice over IP or video conferencing, it is desirable for a receiver to have some guaranteed bandwidth - a reservation. Such performance guarantees are referred to as Quality of Service (QoS) guarantees.

2.1 RSVP

One particular QoS strategy is RSVP (Resource reSerVation Protocol) which uses a dynamic resource reservation strategy.

Overview of RSVP:

- receivers make requests for resources
- routers honor these requests if possible and, based on the requests made of them, make their own requests up the multicast tree (this is where the rules (below) come in) – said another way, the routers join the requests as the tree joins.

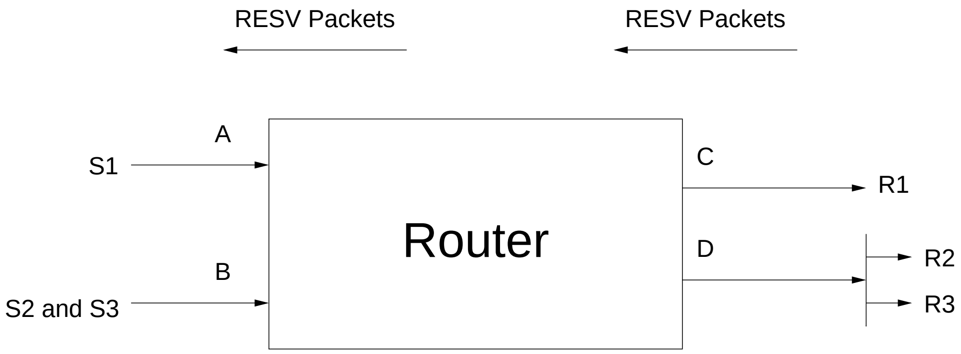

Figure 4: A router joining RESV messages. This router is connected to three sources, S1 via link A and S2 and S3 via link B. It is also connected to three receivers, R1 via link C and R2 and R3 via link D. It combines RESV messages from links C and D to make its own RESV requests up links A and B.

RSVP is a receiver based protocol like IP multicast (ST-11 is an example of a sender based protocol). Reservations start from the receivers and work their way through the network toward the source.

The source of a multicast group sends a Path message periodically to announce that it has a stream available. Each interested host then sends an RESV message in response to reserve resources (e.g., a path with fixed bandwidth and latency) Each router (node) in the multicast tree merges RESV requests where applicable. When possible, the router will provide the maximum request. RESV messages are sent point-to-point between routers. The RSVP reservations are soft and time out if not refreshed.

There are three possible rules for router to use to join RESV requests, depending on the nature of the multicast. In the examples below R is some unit of resources, e.g., bandwidth of 128 Kbps.

- **Wildcard Filter**: Used when only one source transmits at a time, e.g., audio conferencing. For every source, pass up the largest request overall (since all requests are for all sources).

Receives RESV requests from child	Reserves	Sends RESV requests to parent
C (*,3R)	C (*,3R)	A (*,3R)
D (*, 2R),(*, R)	D (*,2R)	B (*,3R)

- **Fixed Filter**: Used when receivers want to receive multiple simultaneous streams - e.g., video conference. For each source, pass up the largest request for that source.

Receives RESV requests from child	Reserves	Sends RESV requests to parent
C (S1,4R),(S2,5R)	C (S1,4R),(S2,5R)	A (S1,4R)
D (S1,3R),(S3,R),(S1,R)	D (S1,3R),(S3,R)	B (S2,5R),(S3,R)

- **Shared Explicit**: Used to receive a stream with one speaker at a time with restrictions on speakers - i.e., some receivers do not want to hear some speakers.

Receives RESV requests from child	Reserves	Sends RESV requests to parent
C ((S1,S2),R)	C ((S1,S2),R)	A (S1,4R)
D ((S1,S3),4R),(S2,2R)	D ((S1,S2,S3),4R)	B ((S2,S3),4R)

Question: Can RSVP handle the following situation: A video stream is available at different qualities, one at 400 kbps and one at 56 kbps. If one host (who is connected via modem) requests 56 kbps and another requests 400 kbps, the router servicing the request should treat the two as independent. The rules above, however, don't appear to allow for the resource requests for a single source to be added together.

Answer: No. It would have to treat the 56 kbps stream and the 400 kbps stream as separate multicast sources. There is a technology called SureStream (by RealNetworks) that does try to solve a similar problem though. Each client negotiates with the server and based on the result, the server sends a stream at the right quality. A trick with the encoding of the stream stored on the server makes this possible. As we'll see later though, this is based on a different network architecture.

2.2 IGMP

Internet Group Membership Protocol (IGMP) is a means by which a host can join a multicast group. IGMP is

Steps in joining a group

- new client joins
  - host sends IGMP request to nearest router
  - a branch is added to the multicast tree with the new client as a leaf
  - the new client begins to receive data

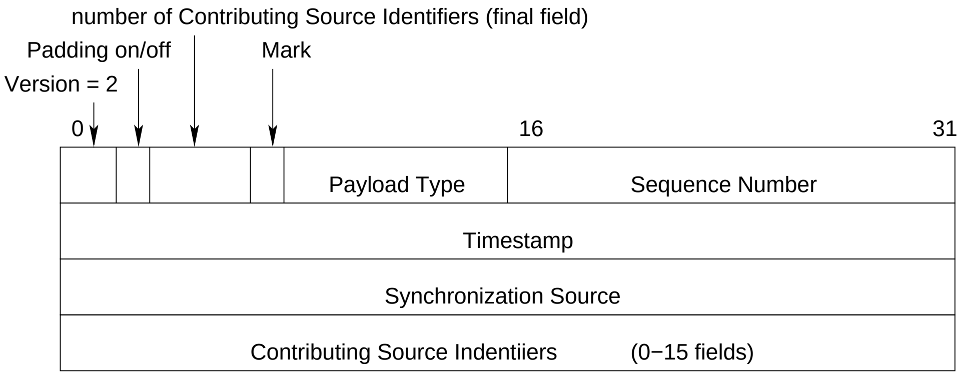

Figure 5: RTP packet header

- client reserves resources via RSVP
  - source sends a Path message
  - new client responds with a RESV

3 Transport Protocols for Streaming Media

3.1 Real Time Streaming Protocol (RTSP)

Real Time Streaming Protocol (RTSP) is analogous to http. It is used by most major players of streaming media. It negotiates the parameters of streaming and allows commands like rewind, fast forward, play, stop and pause to be performed on a stream.

3.2 Real Time Transport Protocol (RTP)

Real Time Transport Protocol (RTP) carries the actual data. It adds timing data so that multiple real-time streams can be synchronized. Typically, RTP is encapsulated by UDP, but sometimes it uses TCP. In multicasting, RTP and RTCP (mentioned below) use the path established by RSVP.

See figure 5. The version of RTP packets is always 2. The payload type indicates what format of data is enclosed in the packet (e.g., mpeg video, mpeg video with audio). The sequence number allows packets to be put back in order if they are received out of order (like TCP). The timestamp allows an application to correlate packets from different streams. The Synchronization Source is a random number generated by the source;

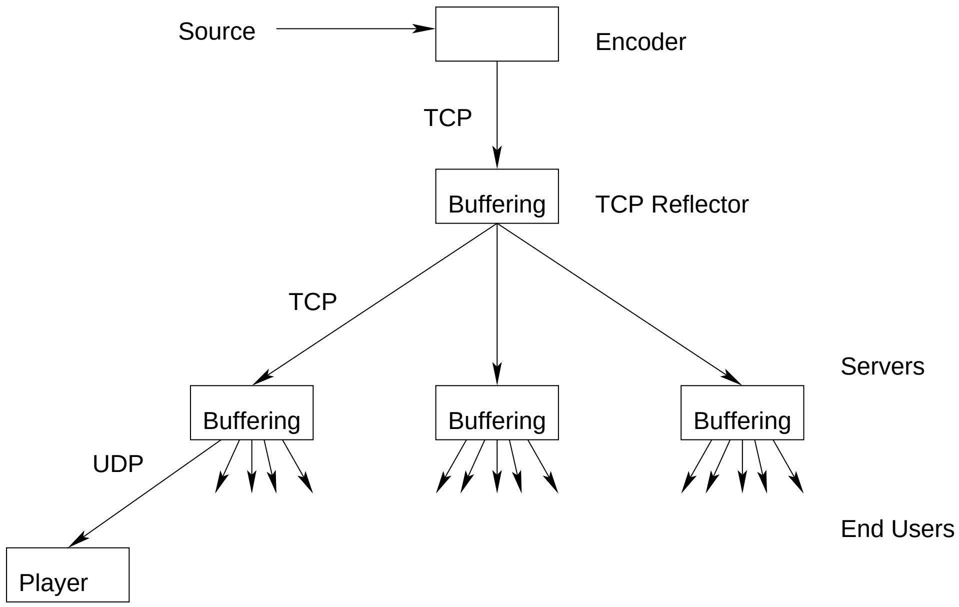

Figure 6: Typical Distribution of a Live Stream

it is used for the duration of the session. Contributing Source Identifiers are added by mixer software when it combines several streams.

3.3 Real Time Control Protocol (RTCP)

Real Time Control Protocol (RTCP) controls RTP. It gives to senders feedback on reception quality and performance. It also serves to keep Synchronization Source nonces unique.

4 Streaming in the Real World

4.1 Typical Distribution of a Live Stream

The multicast tree is mostly fixed - certain services own networks that are set up as in figure 6. All major streaming media software vendors provide reflector software with their streaming media server software. End users connect to their favorite server and receive the stream from there. Recording equipment at the source sends data to an encoder. The encoder encodes the data into the format in which it will be distributed (e.g., mpeg) and sends it via TCP to the reflector (a reflector is just a server running reflector

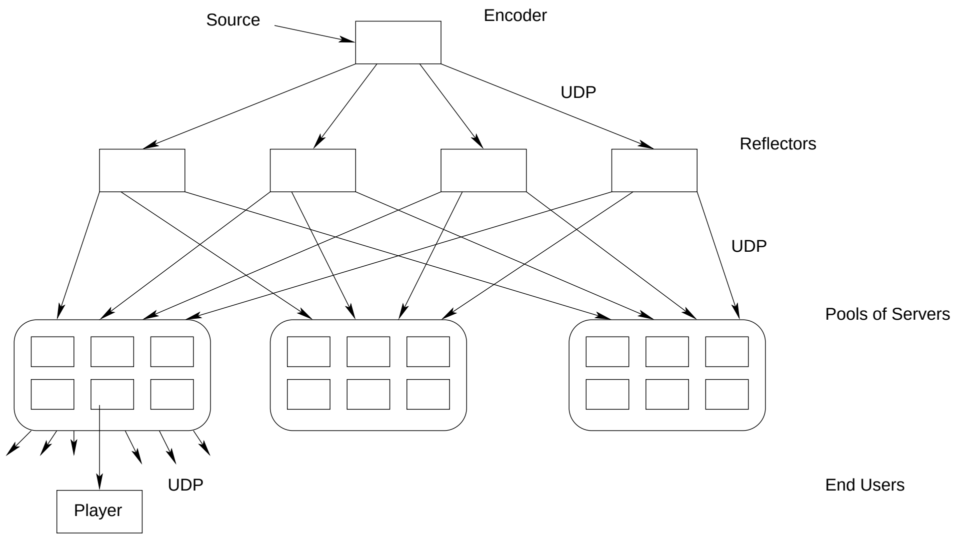

Figure 7: Akamai Distribution of a Live Stream

software). The reflector buffers some of the stream and passes it to a number of servers also via TCP. The servers also buffer the stream. The use of TCP and the buffering at the reflector and the servers ensure that packet loss during these early stages will not be passed down to the end users. The final leg of the streams journey to the end user, from the server to the client machine, is via UDP. For live streaming, once a packet is lost, there is no point in resending it.

Problems with this method:

- no fault tolerance
- substantial buffering required
- the buffering can lead to painful synchronization problems
- not quite true multicast

4.2 Akamai Distribution of a Live Stream

Again, the multicast tree is mostly fixed. The source sends the stream to the encoder which encodes it and passes it along to several reflectors. Each of the reflectors sends the stream to many groups of servers - this redundancy is to ensure that the stream gets to the servers quickly and without error. Within these groups the stream is broadcast so that each server has it.

End users connect to their local bank of servers to receive the stream. All communication is done via UDP - errors between the encoder and the servers should be taken care of by the redundancy, so the no retransmission should be necessary.

The duplication of the message as it is sent to the servers represents a slight inefficiency, but this inefficiency is amortized over the hundreds of users that each bank of servers will send the stream to.