Multicast technology: past, present, future

C. Pham

University Claude Bernard Lyon 1
LIP/INRIA RESO

Congduc.Pham@ens-lyon.fr
www710.univ-lyon1.fr/~cpham

Sunday, February 6\textsuperscript{th} 2005, DFMA 05
How multicast can change the way people use the Internet?

Everybody's talking about multicast! Really annoying! What is it exactly?
Purpose of this tutorial

- Multicast on the Internet has been introduced in 1986
- Provides a comprehensive overview of multicast technologies that span almost 20 years of communication networks
- Present the current technologies and concerns
- Show future directions in multicasting
Outline

- « the past »
  - Only protocols
  - One-size-fits-all philosophy

- « the present »
  - Simpler communication models
  - New concerns for congestion control and fairness

- « the future »
  - Interoperability with new network technologies
  - New approaches, not necessarily based on IP multicast
Layout explanation

- Body text

Indicates a new chapter

Indicates which chapter the slide belongs to

Introduction
This tutorial will...

- explain how multicast can change the way people use the Internet
- present the main technologies behind multicast, both at the routing and transport level
- show what are the main problems and how they can be solved
- state on the current deployment of multicast technologies and the problems encountered for large scale deployment
From unicast...

- Sending same data to many receivers via unicast is inefficient
- Popular WWW sites become serious bottlenecks
...to multicast on the Internet.

- Not n-unicast from the sender perspective
- Efficient one to many data distribution
- Towards low latency, high bandwidth
New applications for the Internet

Think about...

- high-speed www
- video-conferencing
- video-on-demand
- interactive TV programs
- remote archival systems
- tele-medecine, white board
- high-performance computing, grids
- virtual reality, immersion systems
- distributed interactive simulations/gaming...
A whole new world for multicast...
The delivery models (1)

- **model 1: streaming (e.g. for audio/video)**
  - multimedia data requires efficiency due to its size
  - requires real-time, semi-reliable delivery
The delivery models (2)

- **model 2: push delivery**
  - synchronous model where delivery is started at $t_0$
  - usually requires a fully reliable delivery, limited number of receivers
  - **Ex:** synchronous updates of software

![Diagram](image-url)
The delivery models (3)

- **model 3: on-demand delivery**
  - popular content (video clip, software, update, etc.) is continuously distributed in multicast
  - users arrive at any time, download, and leave
  - possibility of millions of users, no real-time constraint

![Diagram showing delivery models]

**Introduction**
A very simple example in figures

- File replication (PUSH) with ftp
  - 10MBytes file
  - 1 source, n receivers (replication sites)
  - 512KBits/s upstream access
  - n=100
    - \(T_x = 4.55\) hours
  - n=1000
    - \(T_x = 1\) day 21 hours 30 mins!
A real example: LHC (DataGrid)

Bunch crossing per 25 nsecs.
100 triggers per second
Event is ~1 MByte in size

Tier 1
~ 4 TIPS

France Regional Center
UK Regional Center
Italy Regional Center
Fermilab

Tier 2
~622 Mbits/sec

CERN Computer Center > ~20 TIPS

Tier 3
~622 Mbits/sec

Tier 0

1 TIPS = 25,000 SpecInt95
PC (1999) = ~15 SpecInt95

~100 MBytes/sec

Tier 1
~ 4 TIPS

Tier 2

Tier 3

Physics data cache
Workstation

Physicists work on analysis “channels”.
Each institute has ~10 physicists working on one or more channels
Data for these channels should be cached by the institute server

Introduction
Reliable multicast: a big win for grids

Data replications
Code & data transfers, interactive job submissions
Data communications for distributed applications (collective & gather operations, sync. barrier)
Databases, directories services

SDSC IBM SP
1024 procs
5x12x17 =1020

NCSA Origin Array
256+128+128
5x12x(4+2+2) =480

CPlant cluster
256 nodes

Multicast address group 224.2.0.1
Wide-area interactive simulations

display

computer-based sub-marine simulator

battle field simulation

INTERNET

(x,y,z)

human in the loop flight simulator

Introduction
The challenges of multicast

SCALABILITY - SECURITY - TCP Friendliness - MANAGEMENT

Introduction
Part I
« The past »

Basic of the IP multicast model
Early group management
IP multicast routing
First steps in reliability
Multicast BONE at the ENS Lyon

Low bandwidth video (10–25 kb/s) with views from all over the world.

Contact Details
- Format: H.261
- Proto: RTP
- Addr: 224.2.172.238
- Port: 51482
- TTL: 127
- Key: 0

Heard from 128.253.115.224 at 21 Mar 2003 15:57 CET

Join | Invite | Record | Dismiss

Basics IP multicast
MBone tools - RAT

- The Robust Audio Tool (RAT) is an open-source audio conferencing and streaming application that allows users to participate in audio conferences over the internet. These can be between two participants directly, or between a group of participants on a common multicast group.
**MBone tools - VIC**

- **VIC** is a video conferencing application developed by the Network Research Group at the LBNL in collaboration with the University of California, Berkeley.
MBone tools - WBD

- WBD is a **shared whiteboard** compatible with the LBL whiteboard, **WB**. It was originally written by Julian Highfield at Loughborough University and has since been modified by Kristian Hasler at UCL.
MBone - Advertising sessions

- **SDR** is a session directory tool designed to allow the advertisement and joining of multicast conferences on the Mbone. It was originally modelled on **sd** written by Van Jacobson at LBNL.
A look back in history of multicast

- **History**
  - Long history of usage on shared medium networks
  - Resource discovery: ARP, Bootp.

1973: Ethernet
1983: ARP (RFC 826)
1985: Bootp (RFC 951)
1986: Deering's work on IP multicast (RFC 966, 988, 1054, 1112)

Basics IP multicast
The Internet group model

- multicast/group communications means...
  - $1 \rightarrow n$ as well as $n \rightarrow m$
- a group is identified by a class D IP address (224.0.0.0 to 239.255.255.255)
  - abstract notion that does not identify any host!

Basics IP multicast
The group model is an open model

- anybody can belong to a multicast group
  - no authorization is required
- a host can belong to many different groups
  - no restriction
- a source can send to a group, no matter whether it belongs to the group or not
  - membership not required
- the group is dynamic, a host can subscribe to or leave at any time
- a host (source/receiver) does not know the number/identity of members of the group
Example: video-conferencing

The user's perspective

Multicast address group 224.2.0.1

What's behind the scene?

- peering point
- access router
- Internet router

Basics IP multicast

224.2.0.1
Receivers must be able to subscribe to groups, need group management facilities
A communication tree must be built from the source to the receivers
Branching points in the tree must keep multicast state information
Inter-domain routing must be reconsidered for multicast traffic
Need to consider non-multicast clouds

IP multicast TODO list

good luck…
unicast island

multicast island

incremental deployment

groups management

session advertising

tree construction

address allocation

duplication engine

forwarding state

routing

Basics IP multicast
Multicast and the TCP/IP layered model

- Application
- Security
- Reliability mgmt
- Congestion control
- Other building blocks

Socket layer
- TCP
- UDP
- ICMP
- IP/ IP multicast
- IGMP
- Device drivers

Higher-level services
Multicast routing

Basics IP multicast
The two sides of IP multicast

- **local-area multicast**
  - use the potential diffusion capabilities of the physical layer (e.g. Ethernet)
  - efficient and straightforward

- **wide-area multicast**
  - requires to go through multicast routers, use IGMP/multicast routing/... (e.g. DVMRP, PIM-DM, PIM-SM, PIM-SSM, MSDP, MBGP, BGMP, MOSPF, etc.)
  - routing in the same administrative domain is simple and efficient
  - inter-domain routing is complex, not fully operational
IP Multicast Architecture

Service model

Host-to-router protocol

Multicast routing protocols

Basics IP multicast
Part I
« The past »

Basic of IP multicast model
Early group management
IP multicast routing
First steps in reliability
Internet Group Management Protocol (RFC 1112)

- IGMP: “signaling” protocol to establish, maintain, remove groups on a subnet.
- Objective: keep router up-to-date with group membership of entire LAN
  - Routers need not know who all the members are, only that members exist
- Each host keeps track of which mcast groups are subscribed to
IGMP: subscribe to a group (1)

224.0.0.1 reach all multicast host on the subnet

Early group mngt
IGMP: subscribe to a group (2)

sends Report for 224.2.0.1

somebody has already subscribed for the group

Early group mngt
IGMP: subscribe to a group (3)

Early group mngt
Data distribution example

Early group mngt
IGMP
Join

Low bandwidth video (10–25 kb/s) with views from all over the world.

Contact Details
Format: H.261  Proto: RTP  Addr: 224.2.172.238  Port: 51482  TTL: 127  Key:
Heard from 128.253.115.224 at 21 Mar 2003 15:57 CET

Ethereal
Frame 2520 (46 on wire, 46 captured)
Ethernet II
Internet Protocol
Internet Group Management Protocol
   Version: 1
      Type: 6 (Host response (v2))
      Unused: 0x00
   Checksum: 0x5d0e
   Group address: 224.2.172.238 (224.2.172.238)
IGMP: leave a group (1)

Host 1

Sends Leave (IGMPv2) for 224.2.0.1 at 224.0.0.2

Host 2

Host 3

224.0.0.2 reach the multicast enabled router in the subnet

Early group mngt
IGMP: leave a group (2)

Early group mngt
**IGMP: leave a group (3)**

Hey, I'm still here!

---

**Early group mngt**
IGMP: leave a group (4)

Early group mngt
**IGMP: leave a group (5)**

- Host 1
- Host 2
- Host 3

Sends IGMP Query for 244.5.5.5 224.2.0.1

Early group mngt
IGMP Leave

Ethereal screen capture

<table>
<thead>
<tr>
<th>No.</th>
<th>Time</th>
<th>Source</th>
<th>Destination</th>
<th>Protocol</th>
<th>Info</th>
</tr>
</thead>
<tbody>
<tr>
<td>758</td>
<td>19.219191</td>
<td>venezia.cri2000.ens-1 224.2.172.238</td>
<td>UDP</td>
<td>Source port: 1060 Destination</td>
<td></td>
</tr>
<tr>
<td>759</td>
<td>19.220437</td>
<td>venezia.cri2000.ens-1 ALL-ROUTERS.MCAST.NET ICMP</td>
<td>Leave group (v2)</td>
<td></td>
<td></td>
</tr>
<tr>
<td>760</td>
<td>19.220531</td>
<td>venezia.cri2000.ens-1 ALL-ROUTERS.MCAST.NET ICMP</td>
<td>Leave group (v2)</td>
<td></td>
<td></td>
</tr>
<tr>
<td>761</td>
<td>19.220835</td>
<td>switch-giga.ens-lyon. ALL-SYSTEMS.MCAST.NET ICMP</td>
<td>Router query</td>
<td></td>
<td></td>
</tr>
<tr>
<td>762</td>
<td>19.221013</td>
<td>switch-giga.ens-lyon. ALL-SYSTEMS.MCAST.NET ICMP</td>
<td>Router query</td>
<td></td>
<td></td>
</tr>
<tr>
<td>763</td>
<td>19.238200</td>
<td>stinky.dc.luth.se 224.2.172.238</td>
<td>UDP</td>
<td>Source port: 1161 Destination</td>
<td></td>
</tr>
<tr>
<td>767</td>
<td>19.265582</td>
<td>leajh.kr.apan.net 224.2.172.238</td>
<td>UDP</td>
<td>Source port: 1253 Destination</td>
<td></td>
</tr>
<tr>
<td>768</td>
<td>19.270547</td>
<td>rgt-451-pc02.wmin.ac. SAP.MCAST.NET SAP</td>
<td>Announcement (v1)</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
IGMP: leave a group (5)

Host 1

Host 2

Host 3

Sends IGMP Query for 244.5.5.5

224.2.0.1

Early group mngt
OK, now I can express local interest, so what?

Early group mngt
Does all paths lead to Roma?
Before going further...

- **Multicast on Ethernet LAN**
  - How can an end-host get link-layer (MAC) packets?

- **Review of Ethernet filtering**
  - By default, the Ethernet device listens on:
    - its (Ethernet) MAC address fixed in a PROM
    - The broadcast MAC address FF:FF:FF:FF:FF:FF
  - Other Ethernet addresses must be explicitly programmed into the driver
  - For multicast, one must listen at:
    - the Ethernet-equivalent of 224.0.0.1 (all multicast host in the LAN)
    - The Ethernet-equivalent address on which multicast sessions are advertised
Mapping of IP multicast address

- A MAC address is built from a mapping of IP multicast addr (Deering88)

![Diagram showing the mapping of IP multicast address to MAC address](image)

- Group bit
- 32:1 ratio
- Organizationally Unique Identifier (OUI, see RFC 1700 Assigned Number)
- Special OUI for IETF: 0x01-00-5E

LAN multicast address

Early group mngt
Part I
« The past »

Basic of IP multicast model
Early group management
IP multicast routing
First steps in reliability
IP multicast routing

- Find a tree (dedicated, shared) between the source(s) and the receivers

- Dense Mode
  - Assumes that there are many many receivers willing to get multicast traffic
  - Uses the «push» model: everybody can receive

- Sparse Mode
  - Assumes that the number of receivers is small
  - Uses the «pull» model: requires an explicit query from the receivers.
Dense mode protocols, DVMRP

- The Ancestor: DVMRP (Distance Vector Multicast Routing)
- Based on Reverse Path Forwarding (RPF)

A multicast router forwards packets received from a link which is on the shortest path to the source, and drops other packets.
DVMRP… (cont')

- resulting multicast distribution tree
  - different sources lead to diff. trees
    ⇒ improves load distribution on the links
  - creates a spanning tree...

IP multicast routing
DVMRP… (cont')

- add “flood and prune” algorithm to dynamically update the tree

**step 1: flood the Internet (only limited by the packet’s TTL)**

**step 2: prune useless branches**

**IP multicast routing**
DVMRP... (cont')

flooding/pruning is done periodically to update the tree
  - required to discover new receivers and remove branches to receivers who left the session

limitations:
  - creates signaling load (PRUNE message)
  - periodically creates important traffic (flooding)
  - all routers keep some state for all the multicast groups in use in the Internet
DVMRP deployment

- Large scale deployment of DVMRP in the MBONE (multicast backbone) since 1992
- Tunnels are set up to link “multicast islands” through unicast areas
Multicast tunnelling illustrated

IP multicast routing
The early MBone with tunnels

Mixing tunnels and native multicast

DVMRP on Linux: the mrouted daemon

<table>
<thead>
<tr>
<th>Vif</th>
<th>Name</th>
<th>Local-Address</th>
<th>Subnet</th>
<th>M</th>
<th>Thr</th>
<th>Rate</th>
<th>Flags</th>
<th>Group Host (Time Left)</th>
<th>IGMP Querier</th>
<th>Nbr Bitmaps</th>
<th>Pkts/Bytes In</th>
<th>Pkts/Bytes Out</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>eth0</td>
<td>193.253.175.161</td>
<td>subnet: 193.253.175.128/26</td>
<td>1</td>
<td>1</td>
<td>0</td>
<td>leaf</td>
<td>239.2.11.73 193.253.175.135 (0:03:47)</td>
<td>IGMP querier: 193.253.175.129 up 50:21:15 last heard 0:00:40</td>
<td>Nbr Bitmaps: 0x0000000000000000</td>
<td>Pkts/Bytes In: 772010/38687700</td>
<td>Pkts/Bytes Out: 0/0</td>
</tr>
<tr>
<td>1</td>
<td>eth1</td>
<td>193.253.175.249</td>
<td>subnet: 193.253.175.248/30</td>
<td>1</td>
<td>1</td>
<td>0</td>
<td>querier leaf</td>
<td>224.0.0.4 193.253.175.249 (0:02:29)</td>
<td>(this system)</td>
<td>Nbr Bitmaps: 0x0000000000000000</td>
<td>Pkts/Bytes In: 0/0</td>
<td>Pkts/Bytes Out: 7935/10870820</td>
</tr>
<tr>
<td>2</td>
<td>eth2</td>
<td>193.253.175.253</td>
<td>subnet: 193.253.175.252/30</td>
<td>1</td>
<td>1</td>
<td>0</td>
<td>querier leaf</td>
<td>224.0.0.4 193.253.175.253 (0:02:33)</td>
<td>(this system)</td>
<td>Nbr Bitmaps: 0x0000000000000000</td>
<td>Pkts/Bytes In: 0/0</td>
<td>Pkts/Bytes Out: 0/0</td>
</tr>
</tbody>
</table>

IP multicast routing
DVMRP summary

- it works but... this is far from perfect
  - periodical flooding creates a heavy load on routers/links
  - each multicast router must keep some forwarding state for each group
  - tunneling quickly became anarchic
  - this is a flat architecture (the same protocol is used everywhere)

- conclusion: “dense mode protocols” like DVMRP are not scalable enough for WAN multicast routing
  - dense mode assumes a dense distribution of receivers, wrong in practice!
DVMRP uses Source-based Trees

Source Shivkumar Kalyanaraman

IP multicast routing
Moving to a Shared Tree

Source Shivkumar Kalyanaraman

IP multicast routing
Shared vs. Source-Based Trees

- **Source-based trees**
  - Shortest path trees - low delay, better load distribution
  - More state at routers (per-source state)
  - Efficient in dense-area multicast

- **Shared trees**
  - Higher delay (bounded by factor of 2), traffic concentration
  - Choice of core/RP affects efficiency
  - Per-group state at routers
  - Efficient for sparse-area multicast
Sparse mode protocols

- **The newcomers: PIM-SM/MSDP/MBGP**
  - **PIM-SM**: Protocol Independent Multicast - Sparse Mode
  - **MSDP**: Multicast Source Discovery Protocol
  - **MBGP**: Multi-protocol Border Gateway Protocol

- **domain** ≡ **site, or ISP network**
  - similar to “autonomous systems” of unicast routing

- **intra-domain mcast routing uses PIM-SM**
- **inter-domain mcast routing requires MBGP**
- **the discovery of sources in other domains requires MSDP**
PIM-SM Protocol Overview

- Basic protocol steps
  - Shared trees are unidirectional
  - Routers with local members Join toward Rendezvous Point (RP) to join shared tree
  - Routers with local sources encapsulate data in Register messages to RP
  - Routers with local members may initiate data-driven switch to source-specific shortest path trees

- PIM v.2 Specification (RFC 2362)
PIM-SM: Build Shared Tree

Shared tree after R1, R2 join

Join message toward RP

Source 1

(*) ,G

Rendezvous Point (RP)

(*) ,G

Source Shivkumar Kalyanaraman

IP multicast routing
Source router unicasts encapsulated data packet to RP in Register

Source 1

(*)G

Source router unicasts encapsulated data packet to RP in Register

RP de-capsulates, forwards down shared tree

Source Shivkumar Kalyanaraman
RP Send Join to High Rate Source

Source 1

(S1,G)

RP

Receiver 1

Receiver 2

Receiver 3

Source Shivkumar Kalyanaraman

Shared tree

SP tree

Join message toward S1

IP multicast routing
Build Source-Specific Distribution Tree

Build source-specific tree for high data rate source

Source Shivkumar Kalyanaraman

IP multicast routing
PIM-SM... (cont')

- Moving to a per-source tree is efficient for bulk data transfer, but has a higher cost in case of multiple sources.
- One tree per source versus a single shared tree.

---

**Source**

- **RP**
- **Source**
- **Receiver**

*From shared tree...*
PIM-SM on Internet routers

- PIM-SM is implemented on all major Internet routers (CISCO, JUNIPER, Alcatel AVICI, PROCKET...)
- A linux package exists, see http://netweb.usc.edu/pim/ (I haven’t tried it yet)
Example: PIM-SM on VTHD

[Diagram of VTHD: Multicast within the VPNs]

FTR&D Caen
FTR&D Grenoble
FTR&D Lannion
FTR&D Issy
FTR&D Sophia

IP multicast routing

Source doc VTHD
**Configuration on CISCO routers**

- **Enabling PIM**

  ```
  ip multicast-routing distributed
  !
  interface XX/XX
  ip pim sparse-dense-mode
  !
  ```

  For each interface

- **Declaring the RP**

  ```
  ip pim rp-address w.x.y.z
  ```

  IP addr of the RP
Part I
« The past »

Basic of IP multicast model
Early group management
Early IP multicast routing

First step in reliability
The Wild Wild Web

heterogeneity, link failures, congested routers packet loss, packet drop, bit errors...

UDP data

Reliability
Reliability Models

- Reliability => requires redundancy to recover from uncertain loss or other failure modes.
- Two types of redundancy:
  - Spatial redundancy: independent backup copies
    - Forward error correction (FEC) codes
    - Problem: requires huge overhead, since the FEC is also part of the packet(s) it cannot recover from erasure of all packets
  - Temporal redundancy: retransmit if packets lost/error
    - Lazy: trades off response time for reliability
    - Design of status reports and retransmission optimization important
Temporal Redundancy Model

- Packets
  - Sequence Numbers
  - CRC or Checksum

- Status Reports
  - ACKs
  - NAKs
  - SACKs
  - Bitmaps

- Retransmissions
  - Packets
  - FEC information

Reliability
End-to-end reliability models

- **Sender-reliable**
  - Sender detects packet losses by gap in ACK sequence
  - Easy resource management

- **Receiver-reliable**
  - Receiver detect the packet losses and send NACK towards the source
Challenge: scalability (1)

- many problems arise with 10,000 receivers...

- Problem 1: scalable control traffic
  - ACK every 2 packets (à la TCP)...oops, 10000ACKs / 2 pkt!
  - NAK (negative ack) only if failure... oops, if pkt is lost close to the source, 10000 NAKs!

source implosion!
Challenge: scalability (2)

- problem 2: scalable repairs/exposure
  - receivers may receive several times the same packet
A piece of the solutions (1)

- solutions to problem 1: scalable control traffic
  - solution 1: feedback suppression at the receivers
    - each node picks a random backoff timer
    - send the NAK at timeout if loss not corrected
  - solution 2: proactive FEC (forward error correction)
    - send data plus additional FEC packets
    - any FEC packet can replace any lost data packet
  - solution 3: use a tree of intelligent routers/servers
    - use a tree for ACK aggregation and/or NAK suppression
    - PGM, ARM, DyRAM

Reliability
A piece of the solutions (2)

- solutions to problem 2: scalable repairs
  - solution 1: use TTL-scoped retransmissions
    - repair packets have limited scope
  - solution 2: use proactive/reactive FEC
    - proactive: always send data + FEC
    - reactive: in case of retransmission, send FEC
  - solution 3: use a tree of retransmission servers
    - a receiver can be a retransmission server if he has the requested data
Scalable Reliable Multicast

Floyd et al., 1995

- Receiver-reliable, NACK-based
- NACK local suppression
  - Delay before sending
  - Based on RTT estimation
  - Deterministic + Stochastic
- Every member may multicast NACK or retransmission
- Periodic session messages
  - Sequence number: detection of loss
  - Estimation of distance matrix among members
SRM Request Suppression

from Haobo Yu, Christos Papadopoulos
SRM Request Suppression

from Haobo Yu, Christos Papadopoulos

Reliability
SRM Request Suppression

from Haobo Yu, Christos Papadopoulos

Reliability
SRM Request Suppression

Each node picks a random backoff timer

from Haobo Yu, Christos Papadopoulos
SRM Request Suppression

Each node picks a random backoff timer.

Source: Haobo Yu, Christos Papadopoulos
SRM Request Suppression

Each node picks a random backoff timer.

Reliability from Haobo Yu, Christos Papadopoulos
SRM Request Suppression

from Haobo Yu, Christos Papadopoulos
SRM Request Suppression

from Haobo Yu, Christos Papadopoulos

Reliability
SRM Request Suppression

from Haobo Yu, Christos Papadopoulos
SRM Request Suppression

from Haobo Yu, Christos Papadopoulos
SRM Request Suppression

from Haobo Yu, Christos Papadopoulos

Reliability
Deterministic Suppression

\[ \text{distance} = (T_4 - T_3 + T_2 - T_1) / 2 \]

\[ \text{Delay} = C_1 \times d_{S,R} \]

from Haobo Yu, Christos Papadopoulos
Simple TTL-scoped of repairs

- use the TTL field of IP packets to limit the scope of the repair packet
Summary: reliability problems

- What is the problem of loss recovery?
  - feedback (ACK or NACK) implosion
    - ACK/NACK aggregation based on timers are approximative!
  - replies/repairs duplications
    - TTL-scoped retransmissions are approximative!
  - Heterogeneity of receivers (crying baby, congestion control)
  - difficult adaptability to dynamic membership changes

- Design goals
  - reduce the feedback traffic
  - reduce recovery latencies
  - improve recovery isolation