CS551
Multicast Routing
Bill Cheng

http://merlot.usc.edu/cs551-f12
Why Study Multicast?

- Want to send information to a group of people
  - allows you to send one packet and let the network make copies to everyone

- Anonymous addressing
  - don’t have to keep track of individual users
  - don’t worry about changes in group membership
  - but:
    - some applications want to know how many or who are in the group
    - not all users need all the information all the time (e.g., in retransmissions, some users need not see the retransmissions)
Multicast Routing

- Unicast: one source to one destination
- Multicast: one source to many destinations
- Two main functions:
  - Efficient data distribution
  - Logical naming of a group (anonymous group addressing)
Unicast
Unicast (Cont...)
Unicast (Cont...)
Unicast (Cont...)

Src

Diagram showing a source (Src) connected to multiple destinations through intermediate nodes.
Unicast (Cont...)

![Diagram showing a source node (Src) connected to multiple destination nodes through intermediate nodes.]
Multicast
Multicast (Cont...)
Multicast (Cont...)
Multicast (Cont...)
Multicast (Cont...)
Multicast (Cont...)
Multicast State

Router:
- learn of the existence of multicast groups (advertisement)
- identify links with group members
- establish state to route packets

replicate packets on appropriate interfaces
Bandwidth Reduction

Applications:
- broadcast, teleconference, etc.

Caveats
- reliability? how do we do failure/loss recovery in multicast?
- different users with different start times?
  - solutions include starting people in the middle (like TV), caching (like Tivo), multiple versions of stream
Logical Naming / Anonymous Addressing

- Application level multicast: mailing lists
- Single address maps to logically related set of destinations
- Convenience
- Scaling: single name/address as group grows, changes
- Special case: *anycast*
  - find me any member of a group, don’t care which one
    (e.g., mailing a letter, any post office will do)
Common Problems in Multicast

- Scalability
  - number of sources
  - number of receivers
  - geographic/network distance (*sparse* vs. *dense* groups)

- message (NACK) *implosion*

- adaption to many receivers (some receivers are slower than others, some have higher lost rates, etc.)
Common Techniques in Multicast

- **Soft state**
  - rather than reliable send and ACK, send periodically

- **Response after randomized delay**
  - delay may be biased to favor certain hosts responding

- **Suppression of duplicate responses**
  - listen to others responses: if they say the same as what you are planning to say, you don’t need to say it

- **These techniques are used in many places**
Multicast groups

- Members are the intended receivers
- Senders may or may not be members
- Destination address is class D IP address
  - globally known portion of address space
- Hosts may belong to many groups
- Hosts may send to many groups
- Support dynamic creation of groups, dynamic membership, dynamic sources
Groups can have different scope

LAN (local scope)

Campus/admin scoping

TTL scoping must be used with caution

Concept of scope important to multipoint protocols and applications (later...)
Example applications

- Broadcast audio/video
- Push-based systems
- Software distribution
- Web-cache updates
- Teleconferencing (audio, video, shared whiteboard, text editor)
- Multi-player games
- Server/service location
- Distributed applications
- Sensor networks?
Other Parts Of The Architecture

- Multicast address allocation (later)
- Assume address is advertised
- Avoid collisions as much as possible
  - Mcast address must be unique in space and time
- Use randomization
- Can’t have highly used address space
- Multiple multicast groups per conference...different app streams, different layers...more later
Some Concepts

- Application level multicast
- Network level multicast
- Aside: active networks
Application-level Multicast

- an overlay link
- a underlay link
Application-level Multicast

Src

Diagram showing a tree structure with nodes and arrows indicating multicast flow.
Application-level Multicast (Cont...)

Src
Application-level Multicast (Cont...)

Src

[Diagram of multicast tree]
Why Not Just Use Application-level Multicast?

- Cons of application-level multicast:
  - duplicate data on links
  - cannot use link-level multicast
  - does not have direct access to unicast routing and does not know about network topology

- Pros of application-level multicast
  - can deploy applications today and does not need help from ISP
Components of the IP Multicast Architecture

- **service model**
- host-to-router protocol (IGMP)
- multicast routing protocols (various)
IP Multicast Service Model (RFC-1112)

- Each group identified by a single IP address
- Groups may be of any size
- Members of groups may be located anywhere in the Internet
- Members of groups can join and leave at will
- Senders need not be members
IP Multicast Service Model (Cont...)

- Group membership not known explicitly
- Network builds multicast distribution tree to send data
  - responsibility of designated router on same LAN as host (and other routers in the network)
- Analogy:
  - each multicast address is like a radio frequency, on which anyone can transmit, and to which anyone can tune-in
IP Multicast Addresses

Class D IP addresses:

```
  1 1 1 0
  . . . .
  group ID
```

in "dotted decimal" notation: 224.0.0.0 -- 239.255.255.255

Two administrative categories:

- "well-known" multicast addresses, assigned by IANA for example, 224.0.0.1 is "all hosts" and 224.0.0.2 is "all routers"
- "transient" multicast addresses, assigned and reclaimed dynamically
Address Allocation

- Outside the scope of this class, but...
- Initially, random allocation
  - odds of collision are low
  - but are they? (recall the birthday paradox)
- Later: more careful schemes
  - Ex. see "The MASC/BGMP Architecture for Multicast Routing", SIGCOMM '98
- (What about hierarchy?)
**IP Multicast Service -- Sending**

- Uses normal IP-Send operation, with an IP multicast address specified as the destination

- Must provide sending application a way to:
  - specify outgoing network interface, if >1 available
  - specify IP time-to-live (TTL) on outgoing packet
  - enable/disable loopback if the sending host is a member of the destination group on the outgoing interface
Join-IP-Multicast-Group (group-address, interface)

Leave-IP-Multicast-Group (group-address, interface)

receive multicast packets for joined groups via normal IP-Receive operation

note: neither sender nor receiver know group size or membership (recall the radio frequency analogy)
Multicast Scope Control:
(1) TTL Expanding-ring Search

to reach or find a nearby subset of a group
Multicast Scope Control:
(2) Administrative TTL Boundaries

To keep multicast traffic within an administrative domain, e.g., for privacy or resource reasons.

TTL threshold set on interfaces to these links, greater than the diameter of the admin. domain.
Multicast Scope Control:
(3) Administratively-scoped Addresses

- RFC 1112 -- "send with my company/country/continent/etc."
- uses address range 239.0.0.0/8
- supports overlapping (not just nested) domains

The rest of the Internet

An administrative domain

Address boundary set on interfaces to these links
The MBone

MBone = Multicast Backbone

An "interconnected" set of multicast capable routers, providing the IP multicast service in the Internet
- some use native multicast
- others tunnel multicast between themselves

Can be thought of as a virtual network, overlaid on the Internet
Components of the MBone

- **H**: host/router
- **R**: MBone router
- **---**: physical link
- **-----**: tunnel
- **------**: part of MBone

---

Copyright © William C. Cheng
MBone Tunnels

- a method for sending multicast packets through multicast-ignorant routers

- IP multicast packet is encapsulated in a unicast packet addressed to far end of tunnel:

  
  | IP header, dest = unicast | IP header, dest = multicast | transport header and data...

- a tunnel acts like a virtual point-to-point link

- each end of tunnel is manually configured with unicast address of the other end