Reliable Multicast

Srinidhi Varadarajan
Introduction

• Need to reliably deliver multicast data from a group of senders to a group of receivers.
• Highly application dependent. Not all applications require strong semantics

**Multicast Choices**
- In order delivery vs out of order delivery. Applications using idempotent operations don’t need in order delivery
- Replicated data vs. single data source

• Highly desirable to use underlying IP multicast framework
Reliability & Multicast: Design Choices

- Unicast reliability has two choices:
  - Sender retransmits on timeout e.g. TCP
  - Receiver requests retransmission e.g. NETBLT

- Sender retransmissions are not suitable for reliable multicast.
  - Receivers can cause ACK implosion
  - Sender needs to track changing state of multiple receivers
  - No concept of single congestion window, RTT estimate. Different receivers may be on different bottleneck links
Reliability & Multicast

- Specification of transmission unit.
  - Shared communication state: e.g. TCP seq. numbers
  - Application data units, e.g. block 5 of file xyz.

- Shared communication state migrates poorly.
  - Sequence numbers are arbitrary. Receivers that join late have no idea what they’ve missed.
Data has unique persistent names:
- Globally unique source ID and locally unique page id.
- Ids are persistent.

Senders multicast data to the entire group.

 Receivers detect loss by finding gaps in the sequence number space.

 On detecting a gap, a receiver waits for a random amount of time (dependent on its distance to the sender) and multicasts a retransmission request.

 Any node can respond to the retransmission request. A sender waits a random amount of time (dependent on its distance to the receiver) and multicasts the requested packet.

Senders discard duplicate requests within a certain interval.
SRM Framework

- Multicast group members periodically send session messages.
  - Session messages are bandwidth controlled, using a mechanism similar to RTCP.
  - Session messages are used to indicate highest sequence number and timestamps. Timestamps are used to estimate round-trip delays.

- Random timers are uniformly distributed in the range:
  - $[C_1d_{r,s}, C_1+C_2d_{r,s}]$

- Adaptive adjustment of timer constants is possible. Increased request delay can be traded off against lower probability of duplicate requests/responses.
Recommended Reading

- “A Reliable and Efficient Protocol for Broadband Broadcast Networks”, Eramilli and Singh, SIGCOMM 88
- “RMTP: A Reliable Multicast Transport Protocol”, Lin and S. Paul, INFOCOM 96