Some RTP Design Principles

Steve Casner
7 November 2006

Why Implement RTCP?

- RFC 3550 says: "The primary function is to provide feedback on the quality of the data distribution. This is an integral part of the RTP's role as a transport protocol and is related to the flow and congestion control functions of other transport protocols (see Section 10 on the requirement for congestion control)."

- Alan Clark tells us that RTCP XR is crucial for VoIP systems

- Also: CNAME-SSRC mapping, counting participants, dynamic timers, lip sync for multiple media, SDES

- More recently: Feedback for retransmission, codec control

=> Generally, for conveying session meta information

Why Keep Control and Data Separate

This was a design evolution from RTPv1 to RTPv2 for the following motivations:

- There may be separate consumers:
  - Different processes
  - Different hardware (DSP vs. controller)

- For multicast, can have third-party monitors for RTCP without processing the data

- Network treatment may be different

- Integrated layer processing argument for minimizing the number of multiplexing points [Clark & Tennenhouse]

Evolution

- Recognizing other constraints, RTP & RTCP mux is proposed
- Other RTP and RTCP adjustments have already been made:
  - RTCP timing rules relaxed for unicast and feedback profile
  - Relaxed even/odd port rule
  - SSRC collision rules for address changes
- Architectural ideas to keep in mind when considering changes:
  - Information in the data header should be there to support the processing of the data in that packet.
  - Security setup affects the channel, but not the media.