Multiplexing Multiple Media Types In a Single RTP Session
draft-lennox-rtcweb-rtp-media-type-mux-00

Jonathan Lennox jonathan@vidyo.com
Jonathan Rosenberg jdrosen@skype.net
Problem Statement

• We want to be able to send RTP sources of multiple media types over a single transport flow
  – For the reasons Magnus has mentioned

• Saying we want to send multiple RTP sessions is assuming the solution...
Solution

• Just send sources of multiple media types in one media session.
• No new RTP-level standardization work needed (just ignore one SHOULD in RFC 3550).
• The SDP-level standardization work is roughly equivalent for every transport mechanism.
  – The BUNDLE group semantics is probably the right approach, unless we want to jettison backward compatibility.
Source purposes

• Receivers need to know what a source is for, before receiving it.

• In some cases (media type mux), PT is sufficient.

• In some cases (e.g., CLUE) you need specific information about every source.

• Session-scoped purposes are probably actually a somewhat minor case, given source mux.
Non-req: pure-transport translators

• Pure-transport translators can’t reliably work for RTP negotiated with offer/answer.
  – Each receiver picks its own PT values, supported codecs, session bandwidth, SRTP keys.
  – So you have to re-write RTP headers, can’t forward unknown RTCP extensions, and may need to terminate and re-send RTCP.

• Thus, requirements to support pure-transport translation between mux and non-mux aren’t very compelling.
Source-mux optimizations

• Want to avoid self-reporting and cross-reporting in RTCP
  – Reporting about your own sources, or duplicate reports from all of your sources about remote ones.
  – Uses quadratic amounts of RTCP bandwidth sending redundant or useless information.
  – Only really useful for pure-transport translators.

• This isn’t specific to media type mux, so probably should be in a separate draft.