Scenario

Video Conferencing

Lots of video sources (10-100s)

Many Sources: where do they go in SDP?

Option A: Multiple Sources per m= line Mostly defined in RFC5576 Scales to many (10-100) sources Successfully used in production applications

Option B: Multiple m= lines with multiple transports

Requires lots of network transports

Typically used in limited cases (audio + video + slides)

Doesn't scale to many (10-100) sources

Option C: Multiple m= lines with BUNDLE Might work???

A, C: Many similarities

- Single RTP session
 - Shared
 - ICE Credentials
 - Crypto params
 - Codecs
- SSRC-based multiplexing
 - a=ssrc used for demux
 - a=ssrc msid: used to identify MediaStreamTrack

Problems with lots of m= lines

- Difficult signaling scenarios
 - Both sides can't add/remove sources at the same time (glare)
 - Can't request resolution while sources are being removed (glare)
 - Answerer can't specify more sources than offerer
- Requires BUNDLE
 - BUNDLE has its own issues; are new m= lines BUNDLEd or not?
 - BUNDLE requires duplication of ICE, crypto, codec attributes
 - Need to do RFC5576 a=ssrc attributes anyway.
- Unproven, especially for video conferencing
 - Lots of **unknowns**, and unknown unknowns (probably)

Advantages of multiple sources per m= line

- Avoids problems with many m= lines
 - Concurrent or partial changes work
 - Only adds or removes an a=ssrc attribute, unencumbered by m-line indexes
 - Other control options possible (CLUE channel, COP, etc)
 - Doesn't require BUNDLE in order to be scalable (i.e. works now)
 - RFC5576 already in use (fewer unknowns)
- Is compatible with other standards and proposals
 - BUNDLE or MMT
 - CLUE or XCON or COP or whatever comes next
 - We can make a decision now without waiting for BUNDLE vs MMT, CLUE, etc.

Non-problems

- Necessary per-source attributes are mostly already defined in RFC5576; draft proposed for the remaining ones
- BUNDLE can be used to multiplex audio and video, but not needed for same-media sources

Proposal

- 1. Multiple sources are encoded in SDP in the RFC5576 style (one m=line per media type).
- Each MediaStreamTrack is identified by a=ssrc and MSID attribute.
- 3. Works with either BUNDLE or MMT.

We don't have to wait. We can decide now.

Example (BUNDLE)

. . . a=group:BUNDLE c1 c2 m=audio 10000 RTP/AVP 0 a=mid:c1 a=candidate:0 1 UDP 2113601791 192.0.2.240 51091 typ host a=candidate:1 1 UDP 1694194431 198.51.100.32 51091 typ srflx raddr 192.0.2.240 rport 51091 a=rtpmap:0 PCMU/8000 a=ssrc:11111 msid:abc 0 a=ssrc:22222 msid:def 0 m=video 10000 RTP/AVP 31 a=mid:c2 a=candidate:0 1 UDP 2113601791 192.0.2.240 51091 typ host a=candidate:1 1 UDP 1694194431 198.51.100.32 51091 typ srflx raddr 192.0.2.240 rport 51091 a=rtpmap:31 H261/90000 a=ssrc:33333 msid:abc 1 a=ssrc:44444 msid:ghi 0

Example (MMT)

. . .

m=anymedia 10000 RTP/AVP 0 31 a=candidate:0 1 UDP 2113601791 192.0.2.240 51091 typ host a=candidate:1 1 UDP 1694194431 198.51.100.32 51091 typ srflx raddr 192.0.2.240 rport 51091 a=rtpmap:0 PCMU/8000 a=rtpmap:31 H261/90000 a=mmtype:0 audio a=mmtype:31 video a=ssrc:11111 msid:abc 0 a=ssrc:22222 msid:def 0 a=ssrc:33333 msid:abc 1 a=ssrc:44444 msid:ghi 0