

A/B Testing

Plan A and Plan B

Plan A

Plan A summary

- One m-line per RTP stream
 - 1 or more lines per capture
 - Separate lines for simulcast and layering if present
 - Maybe separate lines for FEC
- Demux on SSRC
 - a=ssrc to provide mapping of line to stream
 - Use non-overlapping PT where a=ssrc not specified
- Uses draft-ietf-mmusic-msid for application-layer identification

Plan A summary (cont.)

- a=bundle-only
 - Marks an m-line as not wanted unless bundling
 - Limits transport parameter allocation in offers
- Explicit simulcast grouping
 - a=group:SIMULCAST groups variant encodings from the same capture
 - Order is significant (lowest quality first)
 - Simulcast streams signaled based on capabilities (max bandwidth, frame rate)
- Layered grouping similar to (or same as) simulcast
 - Dependent layers appear in order after the layers they depend upon

Plan A Legacy Interoperation

- a=bundle-only streams have a zero port
 - Ignored by legacy endpoints as if they were negotiated off
- a=ssrc is optional

Plan B

Plan B summary

- One m-line per payload configuration
 - Multiple captures and streams on the same line
 - a=ssrc lines required for every stream
 - Grouping based on a=ssrc-group
 - Separated on tuple of *media type* + (*payload types*) + *content* (RFC4796)
- Only streams from offerer toward answerer are negotiated in each offer/answer exchange
 - Offer contains offerer SSRCs only
 - Answer contains a=remote-ssrc 'recv' parameter to accept/reject streams
 - A reversed offer/answer exchange provides return path media (1.5RTT full exchange)
 - Enables “glareless” addition of streams to a session
- Demux on SSRC only

Plan B summary (cont.)

- New `MediaStreamTrack.content` attribute in API
 - Controls placement of streams into separate m-lines
- Uses `draft-ietf-mmusic-msid` for application-layer identification
- Simulcast grouping
 - Similar grouping semantic based on `a=ssrc-group`
 - No explicit ordering
 - Explicit resolution used to identify sizes
- Scalable coding is mentioned but no specific solution is provided
 - Assume simulcast grouping is reused?

Plan B Legacy Interoperation

- a=ssrc is mandatory
- Uses a default stream for legacy peers
 - Default stream selected by Plan B endpoint
 - Similar end result to a=bundle-only...
 - ...except that a=bundle-only allows for >1 default
- No glareless add

Differences Summary

Plan A

- Negotiation @ m-line
 - Zero port on m-line
 - All streams treated the same
- Directions negotiated using sendrecv/sendonly/recvonly
 - Full negotiation in 1 RTT
 - No “glareless” add
- Demux on SSRC, PT
- Depends on bundling
 - a=bundle-only limits #transports
 - Control over legacy fallback

Plan B

- Negotiation @ a=ssrc
 - a=ssrc:... recv:off
 - Splitting m-lines controlled by API
- Only offerer→answerer streams negotiated in offer/answer
 - Full negotiation in 1.5 RTT
 - “glareless” stream addition
- Demux on SSRC
- Less affected by bundling
 - Transport limit inherent
 - Only one stream for fallback