# RTP Multiple Stream Sessions and Simulcast

draft-westerlund-avtcore-multistream-and-simulcast-00

Magnus Westerlund and Bo Burman
Ericsson

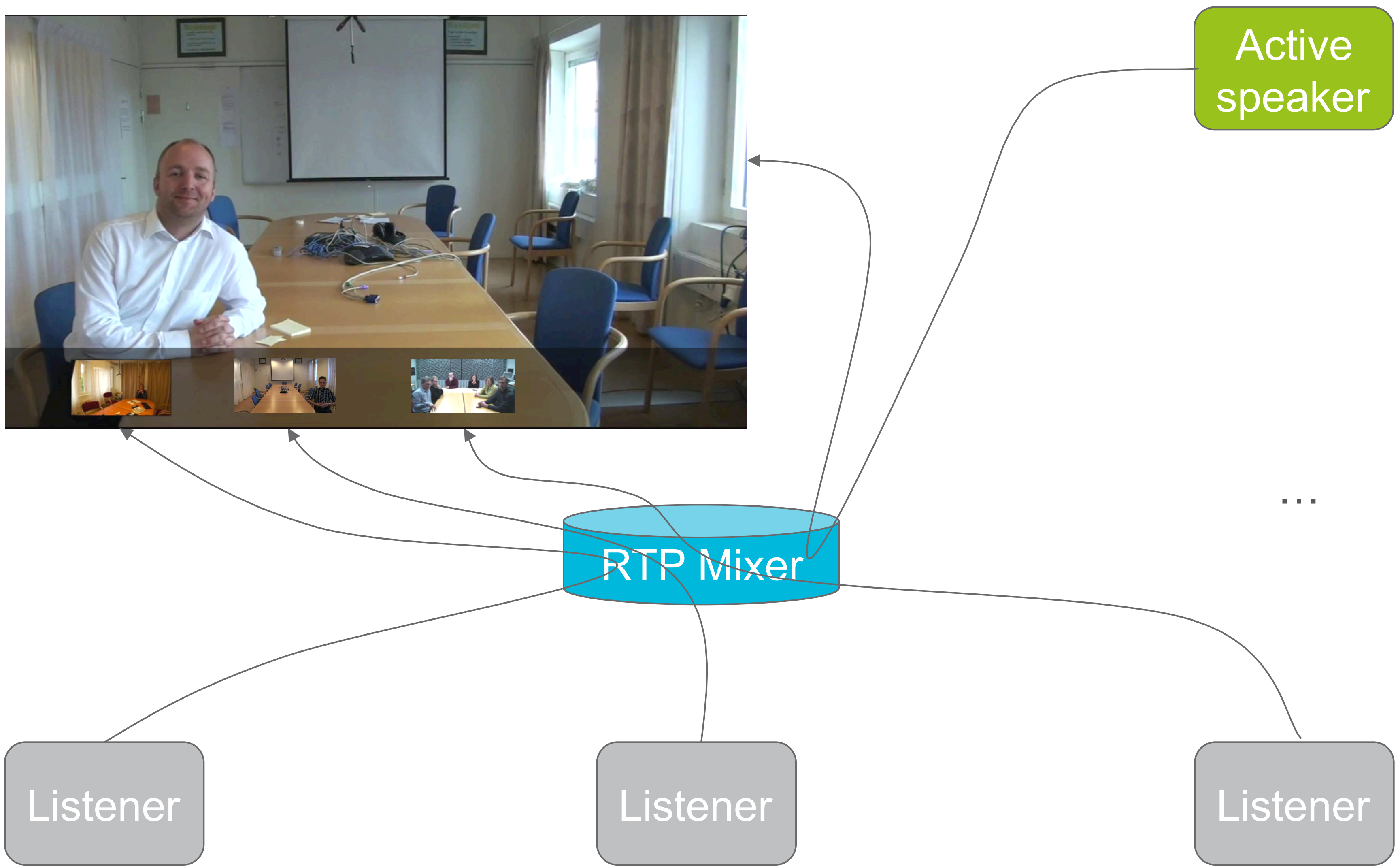
# IPR Disclosure

- Telefonaktiebolaget LM Ericsson (publ)'s has made a Statement about IPR related to this draft in https://datatracker.ietf.org/ipr/1592/

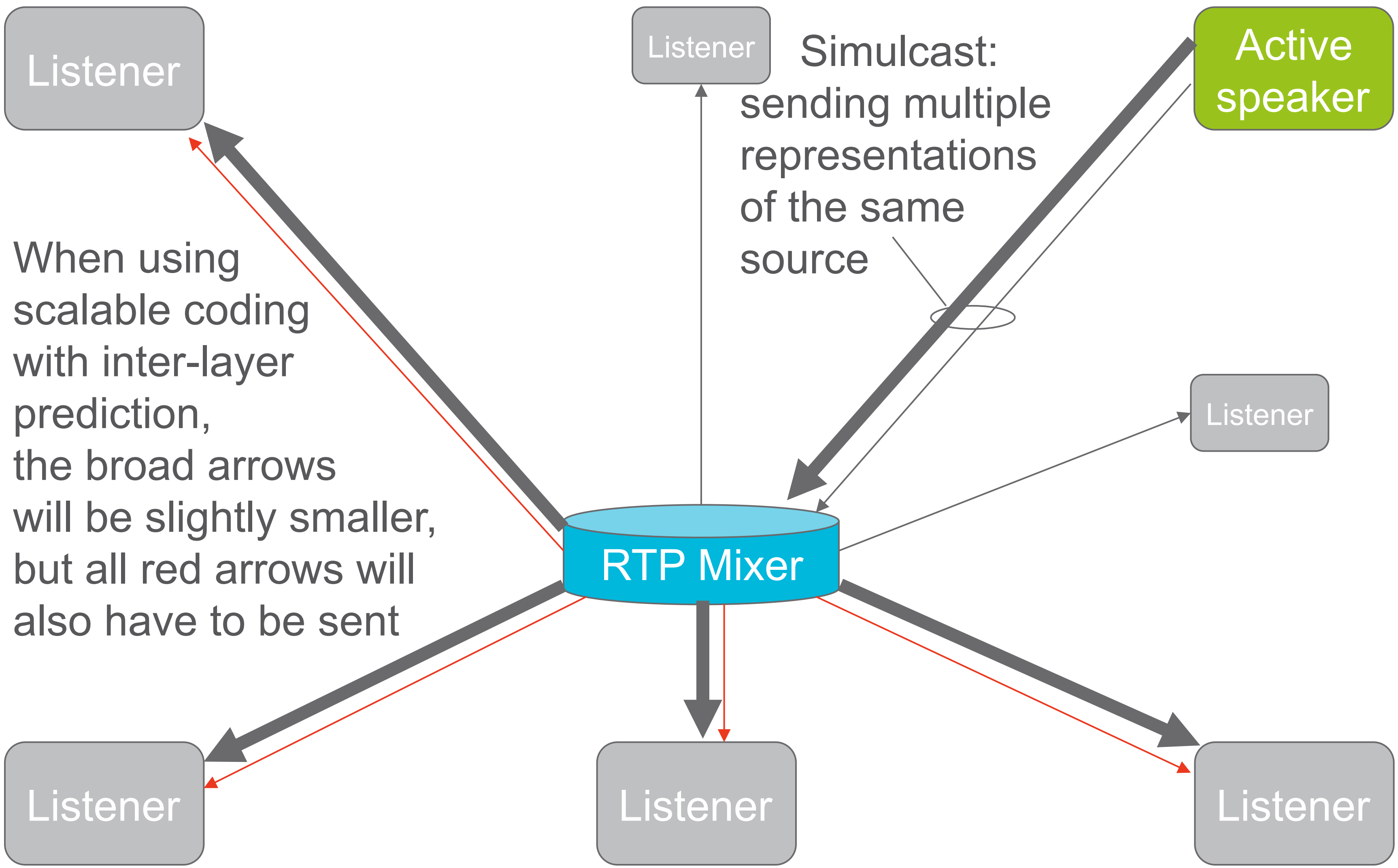
# Outline

- Problem Descriptions
  - A Target Scenario
  - Simulcast
  - Multiple Streams in Advanced RTP usages
  - Bandwidth Signaling
  - Codec Control / Optimization
- Problem Summary
- Extensions
  - Multiple Streams signaling
  - Bandwidth SDP attribute
  - Simulcast Grouping
  - SRCNAME SDES item
  - Codec Control
- Way Forward
  - Architecture document
  - Extensions

# A Target Scenario



# Simulcast

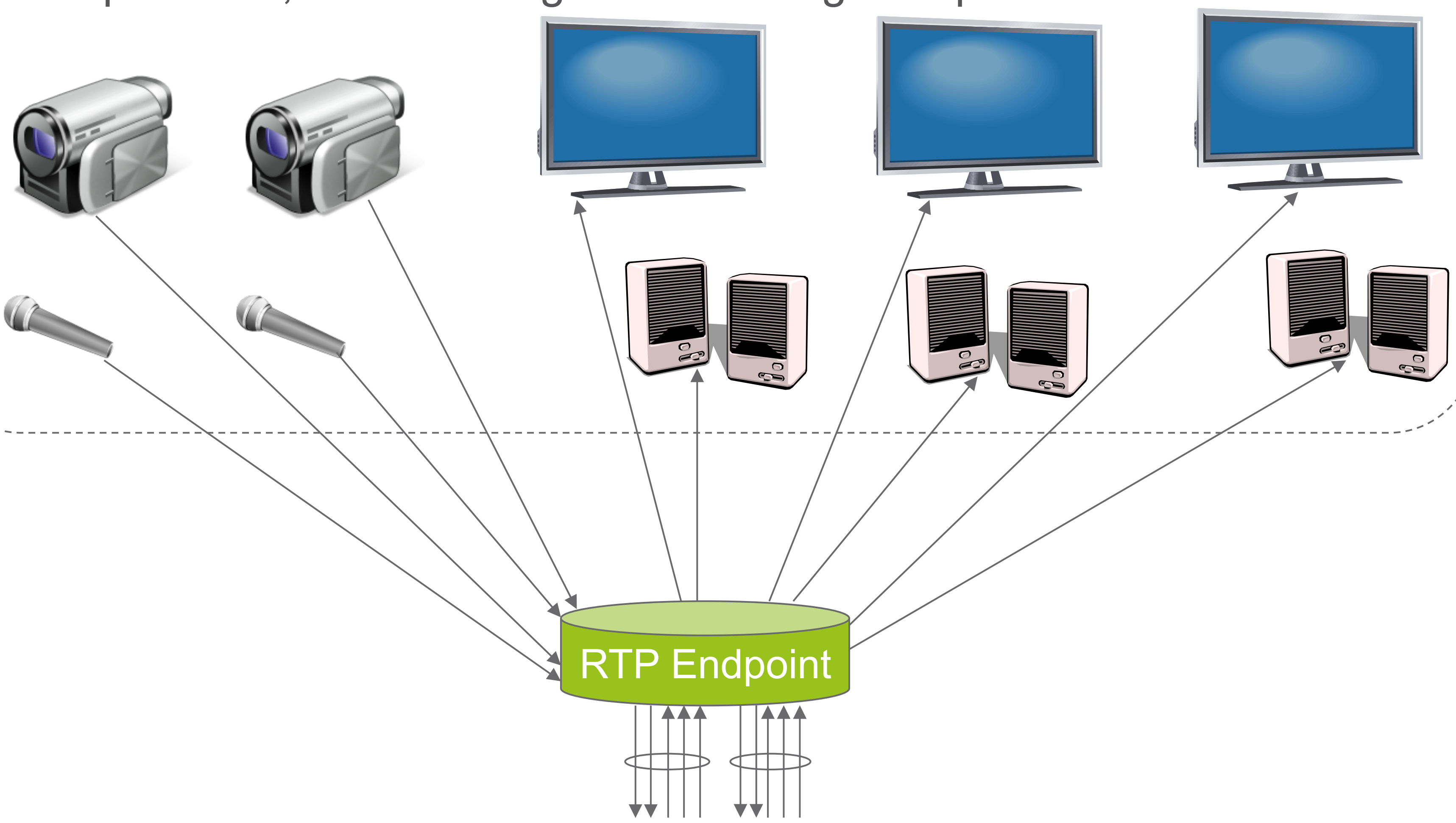

Listener
Listener
Simulcast:
sending multiple
representations
of the same
source
Active
speaker
When using
scalable coding
with inter-layer
prediction,
the broad arrows
will be slightly smaller,
but all red arrows will
also have to be sent
Listener
RTP Mixer
Listener
Listener
Listener

# Simulcast and Scalable Encoding

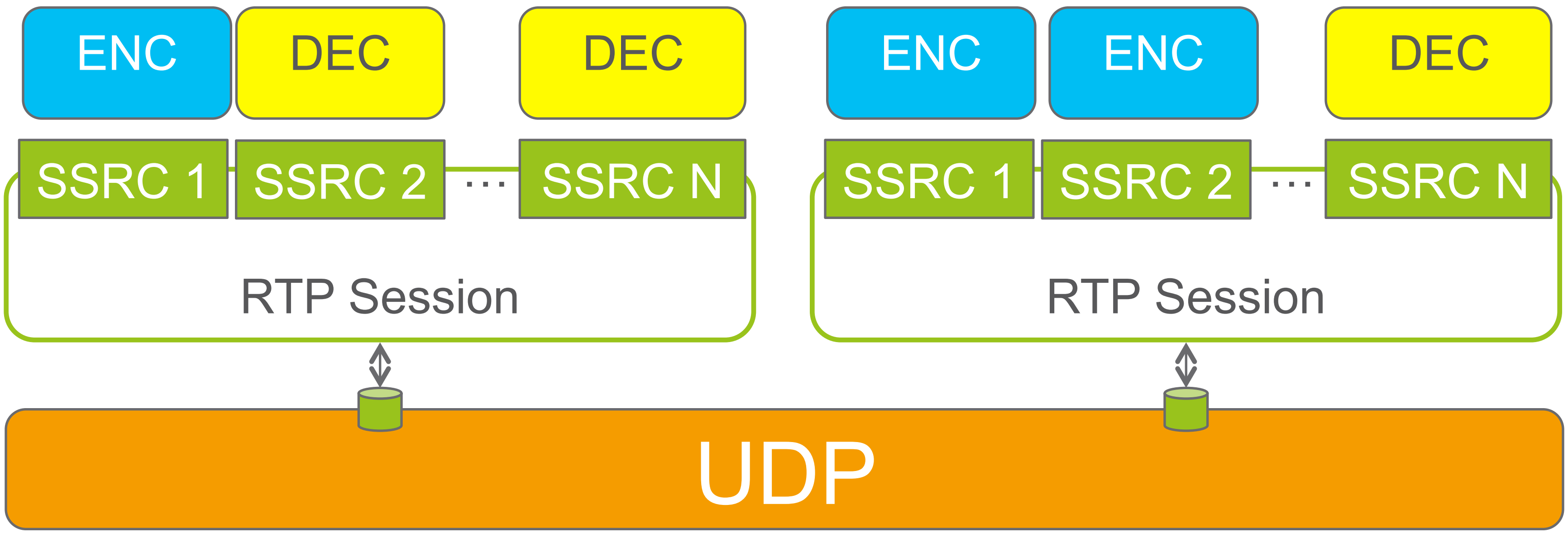
- Simulcast is both an alternative and complementary to Scalable Encoding
- The trade-offs when it comes to efficiency are different
  - SVC encoding is more efficient in sender to mixer path
  - Simulcast is more efficient in mixer to receiver path
  - Combining scalable encoding with simulcast for best of both worlds
- Simulcast is codec agnostic
- Simulcast can be done for other purposes
  - Provide two different encodings for interoperability
  - Provide redundancy for robustness

# Multiple Streams

A sample client, both sending and receiving multiple video streams

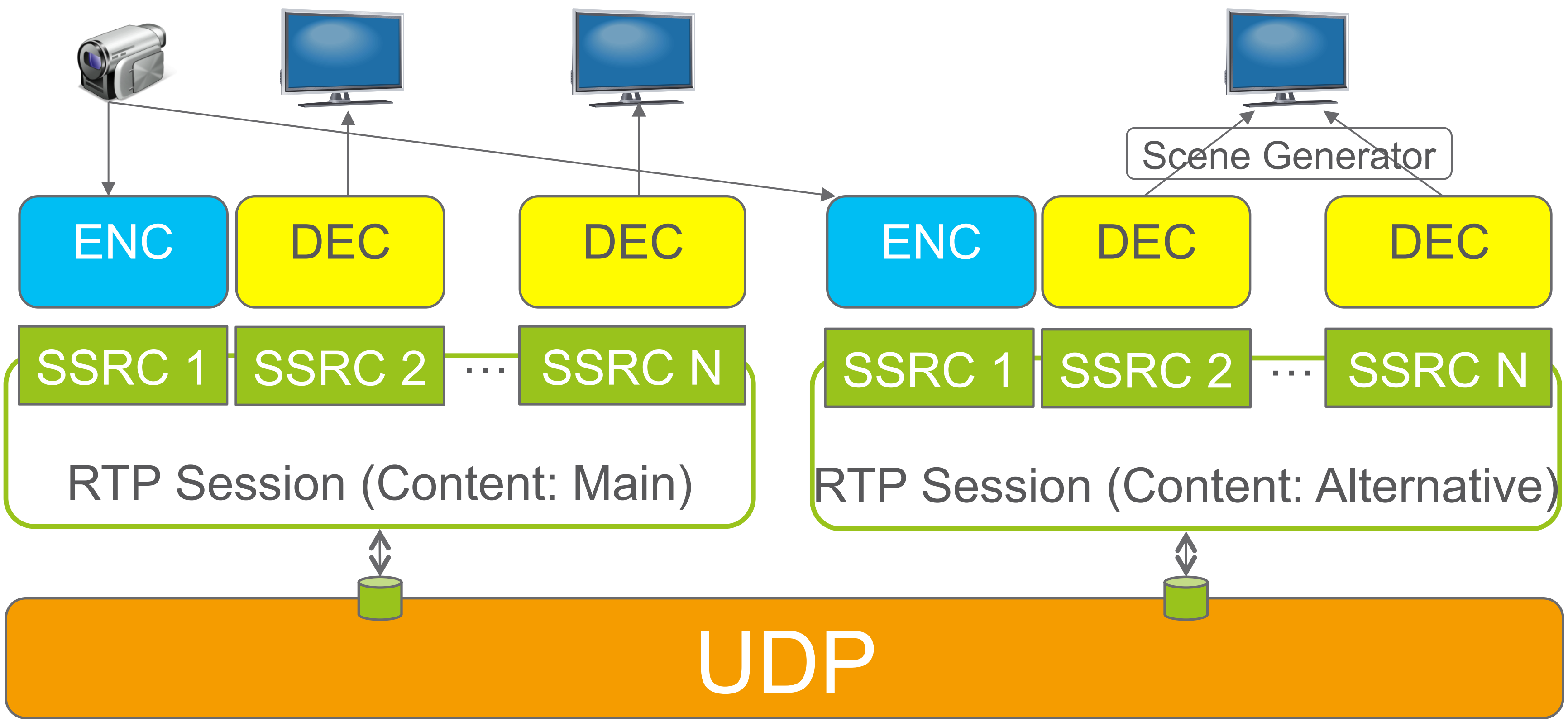


# Multi-Stream Layering




- An RTP Session can contain 1..N SSRCs
- An RTP Session is identified by a lower layer identifier, such as a UDP port or five tuples
- A multimedia session contains one or more RTP sessions

# Multi-Stream Simulcast Layering



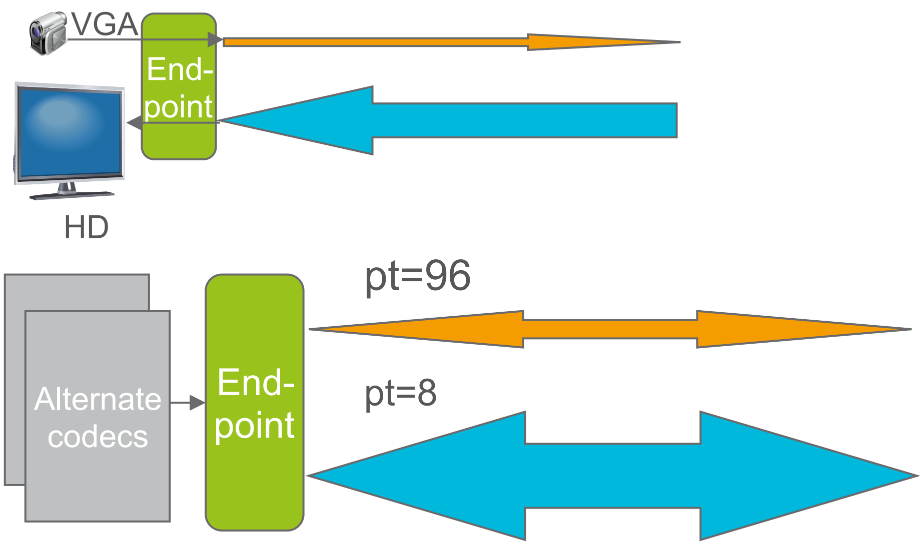

- A single source can be simulcasted as different versions
  - Same actual source in several sessions;
    *don't want to force new semantics into SSRC value*
- Several sources can be rendered at the same device

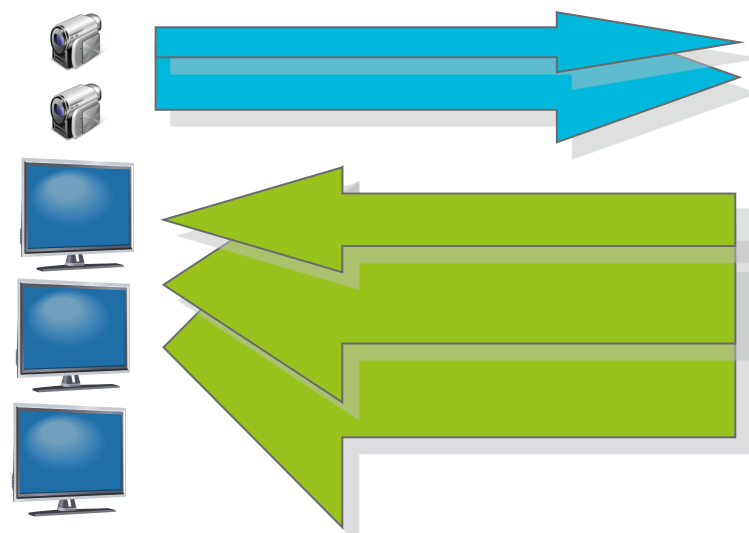
# Multi-Stream Issues

- More advanced use cases than point to point VoIP:
  - Video conferencing
  - Telepresence
  - IPTV
  - Etc.
- This can result in multiple media streams
  - Is the end-point capable of handling multiple simultaneous media streams of the same media type?
    - Legacy capabilities is likely one SSRC per direction
  - When should additional media streams be in the same RTP session, when in a new session?
  - When streams have relations, how to express that for:
    - Retransmission
    - Redundancy
    - Simulcast

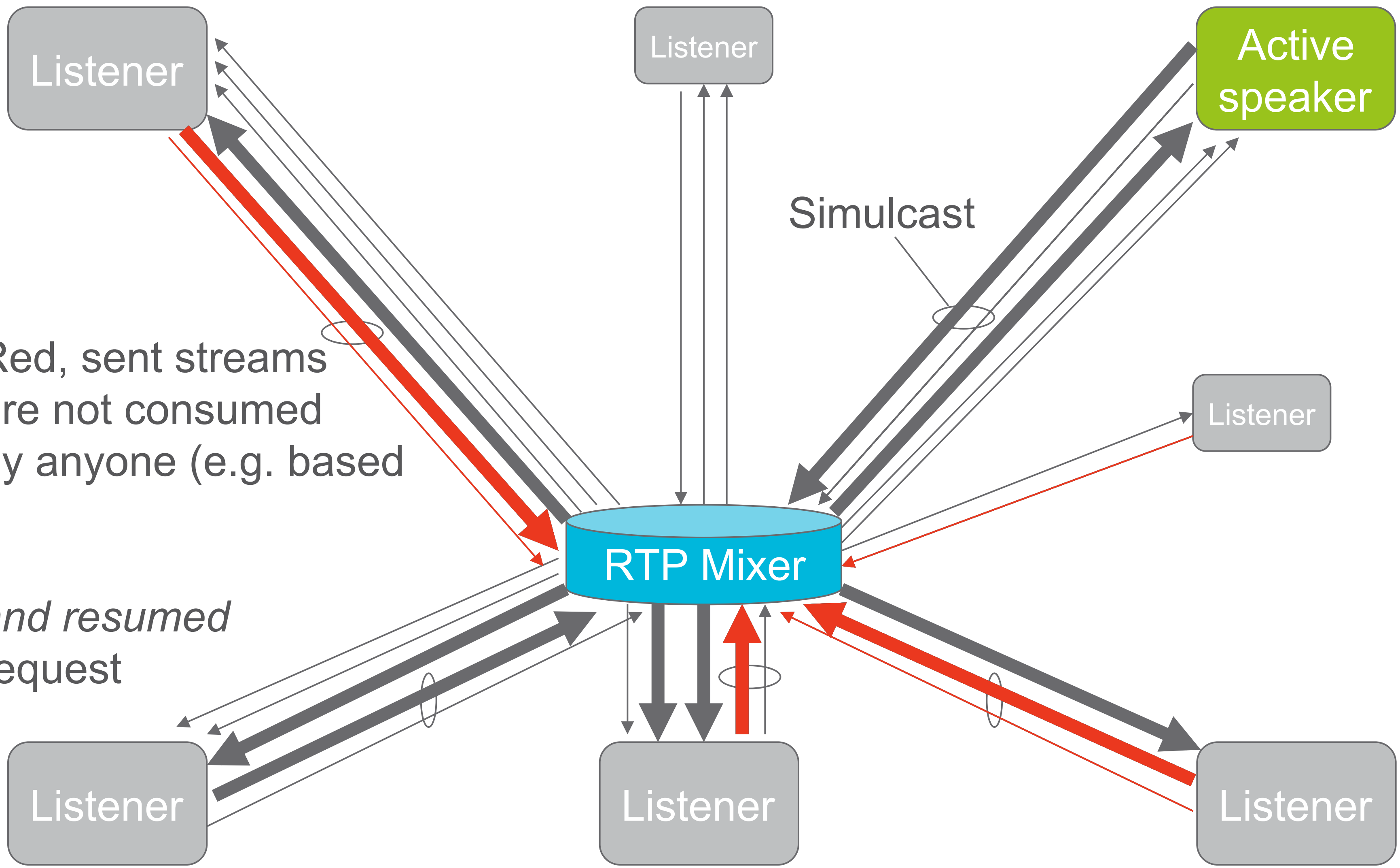
# Bandwidth Signaling

- Current SDP Bandwidth signaling insufficient in handling:
  - Asymmetric bandwidth capabilities in the path
  - Asymmetric bandwidth usage inherent from application
  - When different Payload Types have different bandwidth ranges
  - When multi-stream applications use multiple streams in each direction
  - The allowed burstiness of media sources is not explicit



# Codec Control / Optimization




Red, sent streams are not consumed by anyone (e.g. based

*and resumed* request

NOTE! Just as valid for scalable coding!

# Problem Summary

- We have a general RTP architecture clarity issue
  - We need to clarify multiple SSRCs in one RTP session
  - We need to discuss when appropriate to use multiple RTP sessions
  - We need to create common principles for streams that aren’t independent, but have common source.
- We need to do this for all reasonable topologies

- Multiple SSRCs in an RTP session appear to need signaling support to avoid legacy issues
- Simulcast is good tool, we need signaling and RTP association mechanisms to make it work
- Bandwidth configuration and capability declaration in asymmetric usages and encodings needs to be improved
- Scalable Codecs and Simulcast needs additional Codec Control tools to optimize sessions

# Proposed Extensions (1/4)

- Multiple Streams Signaling
  - Separated directions
    - a=max-send-ssrc:96 2
    - a=max-recv-ssrc:96 5
  - Both payload specific and payload agnostic
    - a=max-recv-ssrc:98 6
    - a=max-recv-ssrc:99 4
    - a=max-recv-ssrc:* 8

# Proposed Extensions (2/4)

› Bandwidth Signaling

  – b= line not possible to extend with sufficient new semantics
  – Per direction and payload type (also payload agnostic)

  – Per source
    › a=bw:recv pt=96 SMT:tb=64000:320
    › a=bw:recv pt=97 SMT:tb=12200:128

  – Entire media level aggregate
    › a=bw:send pt=* AMT:tb=384000:512

  – Allow for future needed semantics to be defined

# Proposed Extensions (3/4)

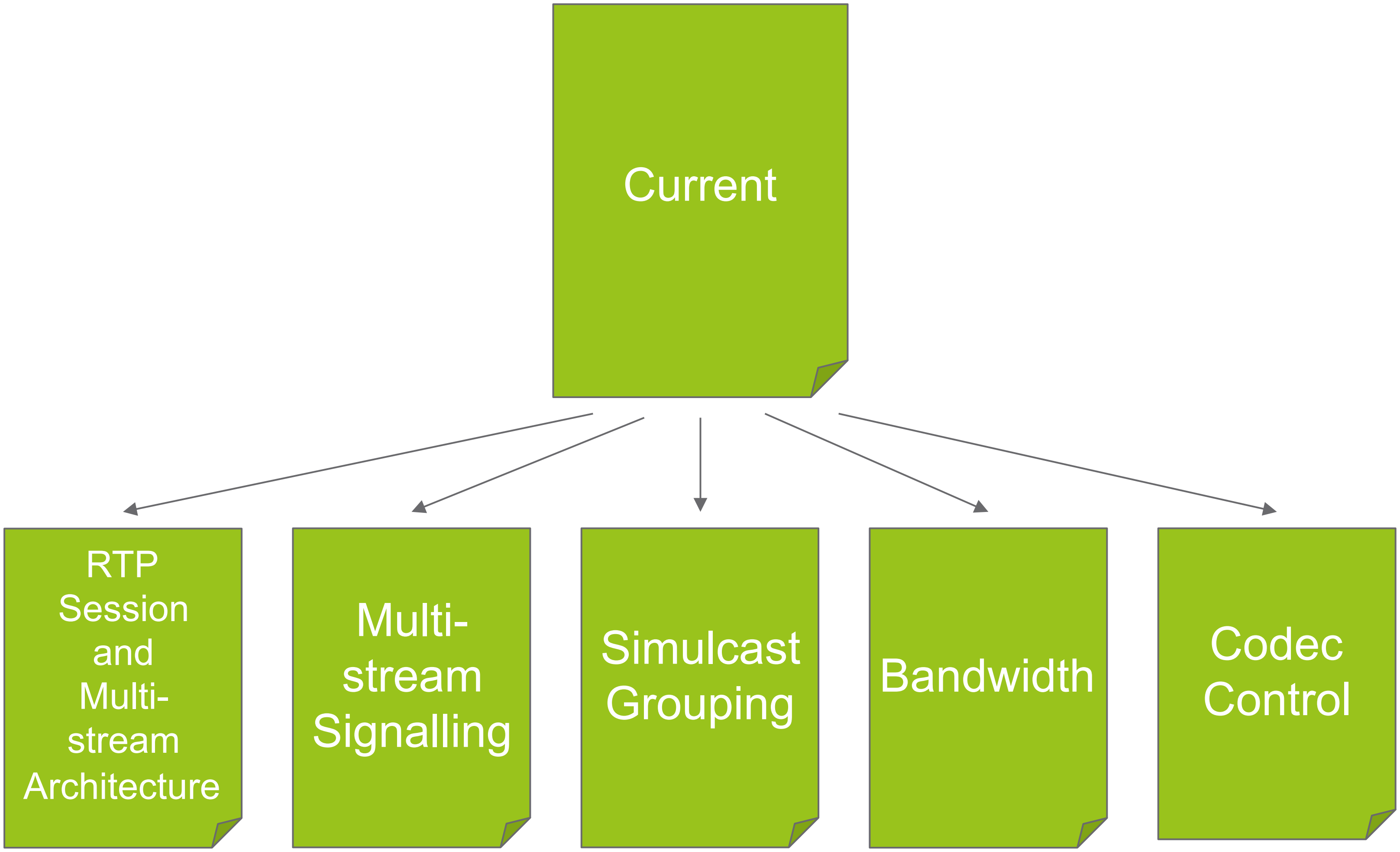
- Simulcast Grouping in SDP
  - Different semantics between directions
  - a=group:SCS 1 2 3 … (SimulCast Send intention)
  - a=group:SCR 4 5 … (SimulCast Receive capability / acknowledge)

- Simulcast Source Identification in RTP
  - SDES CNAME is defined as unique per endpoint, not per source
  - *New* SDES SRCNAME unique per actual media source
    - Indicate which streams are alternative encodings to each other

# Proposed Extensions (4/4)

- Codec Control extensions is forthcomming

# Anticipated Document Split



# Proposal for Going Forward

- That AVTCORE takes on the general Architecture questions:
  - Make it clear when appropriate to use multiple streams within an RTP session
  - How should one use RTP sessions and SSRCs when having alternative, complementary or redundant streams
- That the various extensions are submitted to the appropriate WG as individual pieces for progressing:
  - AVTCORE:
    - Architecture
    - Multi-stream Signaling
  - AVTEXT:
    - Simulcast Group Signaling
    - Codec Control
  - MMUSIC:
    - Bandwidth Signaling

ERICSSON