

Network Working Group  
Internet-Draft  
Intended status: Standards Track  
Expires: April 25, 2014

M. Westerlund  
B. Burman  
Ericsson  
S. Nandakumar  
Cisco  
October 22, 2013

**Using Simulcast in RTP Sessions**  
**draft-westerlund-avtcore-rtp-simulcast-03**

Abstract

In some application scenarios it may be desirable to send multiple differently encoded versions of the same Media Source in independent Source Packet Streams. This is called Simulcast. This document discusses the best way of accomplishing Simulcast in RTP and how to signal it in SDP. A solution is defined by making three extensions to SDP, and using RTP/RTCP identification methods to relate RTP Source Packet Streams. The first SDP extension consists of two new session level SDP attributes that express capability to send or receive Simulcast Source Packet Streams, respectively. The second SDP extension introduces an SDP media level attribute that groups and identifies a selected set of media level parameters for a specific direction, called a media configuration. The third SDP extension describes how to group such media configurations on SDP session or media level for Simulcast purposes.

Status of This Memo

This Internet-Draft is submitted in full conformance with the provisions of [BCP 78](#) and [BCP 79](#).

Internet-Drafts are working documents of the Internet Engineering Task Force (IETF). Note that other groups may also distribute working documents as Internet-Drafts. The list of current Internet-Drafts is at <http://datatracker.ietf.org/drafts/current/>.

Internet-Drafts are draft documents valid for a maximum of six months and may be updated, replaced, or obsoleted by other documents at any time. It is inappropriate to use Internet-Drafts as reference material or to cite them other than as "work in progress."

This Internet-Draft will expire on April 25, 2014.

Copyright Notice

Copyright (c) 2013 IETF Trust and the persons identified as the document authors. All rights reserved.

This document is subject to [BCP 78](http://trustee.ietf.org/license-info) and the IETF Trust's Legal Provisions Relating to IETF Documents (<http://trustee.ietf.org/license-info>) in effect on the date of publication of this document. Please review these documents carefully, as they describe your rights and restrictions with respect to this document. Code Components extracted from this document must include Simplified BSD License text as described in Section 4.e of the Trust Legal Provisions and are provided without warranty as described in the Simplified BSD License.

## Table of Contents

<a href="#">1.</a>	<a href="#">Introduction</a>	<a href="#">3</a>
<a href="#">2.</a>	<a href="#">Definitions</a>	<a href="#">3</a>
<a href="#">2.1.</a>	<a href="#">Terminology</a>	<a href="#">3</a>
<a href="#">2.2.</a>	<a href="#">Requirements Language</a>	<a href="#">4</a>
<a href="#">3.</a>	<a href="#">Use Cases</a>	<a href="#">4</a>
<a href="#">3.1.</a>	<a href="#">Reaching a Diverse Set of Receivers</a>	<a href="#">5</a>
<a href="#">3.2.</a>	<a href="#">Application Specific Media Source Handling</a>	<a href="#">6</a>
<a href="#">3.3.</a>	<a href="#">Receiver Adaptation in Multicast/Broadcast</a>	<a href="#">7</a>
<a href="#">3.4.</a>	<a href="#">Receiver Media Source Preferences</a>	<a href="#">7</a>
<a href="#">4.</a>	<a href="#">Requirements</a>	<a href="#">8</a>
<a href="#">5.</a>	<a href="#">Proposed Solution Overview</a>	<a href="#">9</a>
<a href="#">6.</a>	<a href="#">Proposed Signaling</a>	<a href="#">10</a>
<a href="#">6.1.</a>	<a href="#">Simulcast Capability</a>	<a href="#">11</a>
<a href="#">6.1.1.</a>	<a href="#">Declarative Use</a>	<a href="#">12</a>
<a href="#">6.1.2.</a>	<a href="#">Offer/Answer Use</a>	<a href="#">12</a>
<a href="#">6.2.</a>	<a href="#">Media Configuration</a>	<a href="#">13</a>
<a href="#">6.2.1.</a>	<a href="#">Simulcast Limitations</a>	<a href="#">16</a>
<a href="#">6.2.2.</a>	<a href="#">Declarative Use</a>	<a href="#">17</a>
<a href="#">6.2.3.</a>	<a href="#">Offer/Answer Use</a>	<a href="#">17</a>
<a href="#">6.3.</a>	<a href="#">Grouping Simulcast Configurations</a>	<a href="#">18</a>
<a href="#">6.3.1.</a>	<a href="#">Declarative Use</a>	<a href="#">19</a>
<a href="#">6.3.2.</a>	<a href="#">Offer/Answer Use</a>	<a href="#">19</a>
<a href="#">6.4.</a>	<a href="#">Relating Simulcast Versions</a>	<a href="#">20</a>
<a href="#">6.5.</a>	<a href="#">Two-Phase Negotiation</a>	<a href="#">20</a>
<a href="#">6.6.</a>	<a href="#">Signaling Examples</a>	<a href="#">21</a>
<a href="#">6.6.1.</a>	<a href="#">Unified Plan Client</a>	<a href="#">21</a>
<a href="#">6.6.2.</a>	<a href="#">Multi-Transport Client</a>	<a href="#">24</a>
<a href="#">6.6.3.</a>	<a href="#">Multi-Source Client</a>	<a href="#">26</a>
<a href="#">7.</a>	<a href="#">Network Aspects</a>	<a href="#">28</a>
<a href="#">8.</a>	<a href="#">IANA Considerations</a>	<a href="#">29</a>
<a href="#">9.</a>	<a href="#">Security Considerations</a>	<a href="#">29</a>
<a href="#">10.</a>	<a href="#">Contributors</a>	<a href="#">29</a>
<a href="#">11.</a>	<a href="#">Acknowledgements</a>	<a href="#">30</a>



<a href="#">12. References</a>	<a href="#">30</a>
<a href="#">12.1. Normative References</a>	<a href="#">30</a>
<a href="#">12.2. Informative References</a>	<a href="#">31</a>
<a href="#">Appendix A. Discussion on Receiver Diversity</a>	<a href="#">32</a>
Authors' Addresses	<a href="#">34</a>

## [1. Introduction](#)

Most of today's multiparty video conference solutions make use of centralized servers to reduce the bandwidth and CPU consumption in the endpoints. Those servers receive Source Packet Streams from each participant and send some suitable set of possibly modified streams to the rest of the participants, which usually have heterogeneous capabilities (screen size, CPU, bandwidth, codec, etc). One of the biggest issues is how to perform stream adaptation to different participants' constraints with the minimum possible impact on video quality and server performance.

Simulcast is the act of simultaneously sending multiple different versions of the same media content, e.g. the same video source encoded with different video encoder types or image resolutions. This can be done in several ways and for different purposes. This document focuses on the case where it is desirable to provide a Media Source as multiple Source Packet Streams over RTP [[RFC3550](#)] towards an intermediary so that the intermediary can provide the wanted functionality by selecting which Source Packet Stream to forward to other participants in the session, and more specifically how the identification and grouping of the involved Source Packet Streams are done. From an RTP perspective, Simulcast is a specific application of the aspects discussed in RTP Multiplexing Guidelines [[I-D.ietf-avtcore-multiplex-guidelines](#)].

The purpose of this document is to describe a few scenarios where it is motivated to use Simulcast, and propose a suitable solution for signaling and performing RTP Simulcast.

## [2. Definitions](#)

### [2.1. Terminology](#)

This document makes use of the terminology defined in RTP Taxonomy [[I-D.lennox-raiarea-rtp-grouping-taxonomy](#)]. In addition, the following terms are used:

Media Configuration: A specific set of parameter values applied on the encoding and packetization process that creates a specific Source Packet Stream. In SDP, the applicable parameter values are described by the joint set of "rtpmap" parameters, "fmt"p"



parameters, and the "config-id" ([Section 6.2](#)) parameters, including extensions.

## **2.2. Requirements Language**

The key words "MUST", "MUST NOT", "REQUIRED", "SHALL", "SHALL NOT", "SHOULD", "SHOULD NOT", "RECOMMENDED", "MAY", and "OPTIONAL" in this document are to be interpreted as described in [RFC 2119](#) [[RFC2119](#)].

## **3. Use Cases**

Many use cases of Simulcast as described in this document relate to a multi-party Communication Session where one or more central nodes are used to adapt the view of the Communication Session towards individual Participants, and facilitate the Media Transport between Participants. Thus, these cases targets the RTP Mixer topology defined in [[RFC5117](#)] ([Section 3.4](#): Topo-Mixer), further elaborated and extended with other topologies in [[I-D.ietf-avtcore-rtp-topologies-update](#)] ([Section 3.6](#) to 3.9).

There are two principle approaches for an RTP Mixer to provide this adapted view of the Communication Session to each receiving Participant:

- o Transcoding (decoding and re-encoding) received Source Packet Streams with characteristics adapted to each receiving Participant. This often include mixing or composition of Media Sources from multiple Participants into a mixed Media Source originated by the RTP Mixer. The main advantage of this approach is that it achieves close to optimal adaptation to individual receiving Participants. The main disadvantages are that it can be very computationally expensive to the RTP Mixer and typically also degrades media Quality of Experience (QoE) such as end-to-end delay for the receiving Participants.
- o Switching a subset of all received Source Packet Streams or sub-streams to each receiving Participant, where the used subset is typically specific to each receiving Participant. The main advantages of this approach are that it is computationally cheap to the RTP Mixer and it has very limited impact on media QoE. The main disadvantage is that it can be difficult to combine a subset of received Source Packet Streams into a perfect fit to the resource situation of a receiving Participant.

The use of Simulcast is relates to the latter approach, where it is more important to reduce the load on the RTP Mixer and/or minimize QoE impact than to achieve an optimal adaptation of resource usage.



A multicast/broadcast case where the receivers themselves select the most appropriate simulcast version and tune in to the right transport to receive that version is also considered ([Section 3.3](#)). This enables large receiver populations with heterogeneity where it comes to capabilities and the use network paths bandwidth.

In this section, an "RTP switch" is used as a common short term for the terms "switching RTP mixer", "source projecting middlebox", and "video switching MCU" as discussed in [\[I-D.ietf-avtcore-rtsp-topologies-update\]](#).

### **3.1. Reaching a Diverse Set of Receivers**

The Media Sources provided by a sending Participant potentially need to reach several receiving Participants that differ in terms of available resources. A discussion on that topic is included in [Appendix A](#). The receiver resources that typically differ include, but are not limited to:

**Codec:** This includes codec type (such as SDP MIME type) and can include codec configuration options (e.g. SDP fmp parameters). A couple of codec resources that differ only in codec configuration will be "different" if they are somehow not "compatible", like if they differ in video codec profile, or the transport packetization configuration.

**Sampling:** This relates to how the Media Source is sampled, in spatial as well as in temporal domain. For video streams, spatial sampling affects image resolution and temporal sampling affects video frame rate. For audio, spatial sampling relates to the number of audio channels and temporal sampling affects audio bandwidth. This may be used to suit different rendering capabilities or needs at the receiving endpoints, as well as a method to achieve different transport capabilities, bitrates and eventually QoE by controlling the amount of source data.

**Bitrate:** This relates to the amount of bits spent per second to transmit the Media Source as an Source Packet Stream, which typically also affects the Quality of Experience (QoE) for the receiving user.

Letting the sending Participant create a Simulcast of a few differently configured Source Packet Streams per Media Source can be a good trade-off when using an RTP switch as middlebox, instead of sending a single Source Packet Stream and using an RTP Mixer to create individual transcodings to each receiving Participant.





This requires that the receiving Participants can be categorized in terms of available resources and that the sending Participant can choose a matching configuration for a single Source Packet Stream per category and Media Source.

For example, assume for simplicity a set of receiving Participants that differ only in that some have support to receive Codec A, and the others have support to receive Codec B. Further assume that the sending participant can send both Codec A and B. It can then reach all receivers by creating two Simulcasted Source Packet Streams from each Media Source; one for Codec A and one for Codec B.

In another simple example, a set of receiving Participants differ only in screen resolution; some are able to display video with at most 360p resolution and some support 720p resolution. A sending Participant can then reach all receivers by creating a Simulcast of Source Packet Streams with 360p and 720p resolution for each sent video Media Source.

In more elaborate cases, the receiving Participants differ both in available Sampling and Bitrate, and maybe also Codec, and it is up to the RTP switch to find a good trade-off in which Simulcasted stream to choose for each intended receiver. It is also the responsibility of the RTP switch to negotiate a good fit of Simulcast streams with the sending Participant.

The maximum number of Simulcasted Source Packet Streams that can be sent is mainly limited by the amount of processing and uplink network resources available to the sending Participant.

### **3.2. Application Specific Media Source Handling**

The application logic that controls the Communication Session may include special handling of some Media Sources. It is for example commonly the case that the media from a sending Participant is not sent back to itself.

It is also common that a currently active speaker Participant is shown in larger size or higher quality than other Participants (the Sampling or Bitrate aspects of [Section 3.1](#)). Not sending the active speaker media back to itself means there is some other Participant's media instead that receive special handling towards the active speaker; typically the previous active speaker. This way, the previously active speaker is needed both in larger size (to current active speaker) and in small size (to the rest of the Participants), which can be solved with a Simulcast from the previously active speaker to the RTP switch.



### **3.3. Receiver Adaptation in Multicast/Broadcast**

When using Broadcast or Multicast technology to distribute real-time media streams to large populations of receivers there can still be significant heterogeneity among the receiver population. This can depend on several factors:

**Network Bandwidth:** The network paths to individual receivers will have variations in the bandwidth. Thus putting different limits on the supported bit-rates that can be received.

**Endpoint Capabilities:** The endpoint's hardware and software can have varying capabilities in relation to screen resolution, decoding capabilities, and supported media codecs.

To handle these variations, a transmitter of real-time media may want to apply Simulcast to its Source Packet Streams and provide a set of media configurations, enabling the receivers to select the best fit from these sets themselves. The endpoint capabilities will usually result in a single initial choice. However, the network bandwidth can vary over time, which requires a client to continuously monitor its reception to determine if the received media streams still fit within the available bandwidth. If not, another Simulcast media configuration containing a thinner set of Source Packet Streams will have to be chosen.

When one uses IP multicast, the level of Simulcast granularity that the receiver can select from is by choosing different multicast addresses. Thus, different Simulcast versions need to be put on different Media Transports using different multicast addresses. If these Simulcast versions are described using SDP, they need to be part of different SDP media descriptions, as SDP binds to transport on media description level. To enable more than the initial choice to function well, there is a need to enable correct mapping of Source Packet Streams in one Simulcast media configuration to a corresponding Source Packet Stream in another Simulcast media configuration on another multicast group.

### **3.4. Receiver Media Source Preferences**

The application logic that controls the Communication Session may allow receiving Participants to apply preferences to the characteristics of the Source Packet Stream they receive, for example in terms of the aspects listed in [Section 3.1](#). Sending a Simulcast of Source Packet Streams is one way of accommodating receivers with conflicting or otherwise incompatible preferences.



#### **4. Requirements**

The following requirements need to be met to support the use cases in previous sections:

REQ-1: Identification. It must be possible to identify a set of simulcasted Source Packet Streams as originating from the same Media Source:

REQ-1.1: In SDP signaling.

REQ-1.2: On RTP/RTCP level.

REQ-2: Transport usage. The solution must work when distributing different Simulcast versions on:

REQ-2.1: Same Media Transport and RTP session.

REQ-2.2: Different Media Transports and RTP sessions.

REQ-3: Capability negotiation. It must be possible that:

REQ-3.1: Sender can express capability of sending simulcast.

REQ-3.2: Receiver can express capability of receiving simulcast.

REQ-3.3: Sender can express maximum number of Simulcast versions that can be provided.

REQ-3.4: Receiver can express maximum number of Simulcast versions that can be received.

REQ-3.5: Sender can detail the characteristics of the Simulcast versions that can be provided.

REQ-3.6: Receiver can detail the characteristics of the Simulcast versions that it prefers to receive.

REQ-4: Distinguishing features. It must be possible to have different Simulcast versions use different values for any combination of:

REQ-4.1: Codec. This includes both codec type and configuration options for both codec and RTP packetization. It also includes different layers from a scalable codec, but only as long as those layers are possible to identify on RTP level.

REQ-4.2: Bitrate of Source Packet Stream.



REQ-4.3: Sampling in spatial as well as in temporal domain.

REQ-5: Compatibility. It must be possible to use Simulcast in combination with other RTP mechanisms that generate additional Source Packet Streams:

REQ-5.1: RTP Retransmission [[RFC4588](#)].

REQ-5.2: RTP Forward Error Correction [[RFC5109](#)].

REQ-6: Interoperability. The solution must also be able to use in:

REQ-6.1: Interworking with non-simulcast legacy clients using a single Media Source per media type.

REQ-6.2: WebRTC "Unified Plan" environment.

## 5. Proposed Solution Overview

Signaling Simulcast is about negotiating between media sender and receiver what the different Simulcast versions should be, how to identify them in terms of Source Packet Streams, and how to inter-relate those Source Packet Streams.

The proposed solution consists of:

- o Signaling Simulcast capability in an optional, pre-stage Offer/Answer:
  - \* Separate send and receive Simulcast capabilities as SDP session level attributes.
  - \* Media properties that are supported as base for different Simulcast versions are listed as parameters that are also possible to rank.
  - \* Early indication of maximum number of available encoding/decoding resources on SDP media level.
- o Including detailed information for the Simulcast in a main Offer/Answer:
  - \* Including Simulcast capability indications, as described above, being kept from the pre-stage Offer/Answer, if any.
  - \* Defining and labeling of the media configuration for each Simulcast version to be sent or received.





- \* The media configuration for a Simulcast version can include acceptable parameter ranges for parameters that are most likely used to distinguish Simulcast versions.
  - \* Indicating the use of Simulcast, separately per direction, by grouping the defined media configurations, not individual streams, that will constitute the Simulcast.
  - \* Allowing that any one of the media configurations in a specific Simulcast is signaled inactive from the start of the session. This is defined as equivalent to the affected Source Packet Stream being in PAUSED state [[I-D.westerlund-avtext-rtp-stream-pause](#)].
  - \* Adding and/or modifying SDP media descriptions as needed to accommodate the negotiated Simulcast streams.
  - \* Parameter limits to the aggregate of media configurations are signaled by existing SDP attributes on session and media description level.
  - \* Including media level indication of maximum number of available encoding/decoding resources on SDP media level. They MAY be modified compared to the pre-stage Offer/Answer, if any.
  - \* Identifying which Source Packet Stream corresponds to which media configuration by including the configuration label as part of the SDES item SRCNAME [[I-D.westerlund-avtext-rtcp-sdes-srcname](#)] information include in the RTP and RTCP packets. The optional mechanism for source specific signalling defined in SRCNAME could be used to let Simulcast sender pre-announce such a relationship before sending the Source Packet Stream.
- o Adding Simulcast information to the Source Packet Stream:
- \* Identifying Source Packet Streams from same Media Source using the new RTCP SDES Item SRCNAME [[I-D.westerlund-avtext-rtcp-sdes-srcname](#)], and as described there including the possibility to send the same information as an RTP Header Extension [[RFC5285](#)].
  - \* Using PAUSE/RESUME [[I-D.westerlund-avtext-rtp-stream-pause](#)] functionality to temporarily turn individual Simulcast versions on or off.

## 6. Proposed Signaling



This section further details the signaling solution outlined above ([Section 5](#)).

### 6.1. Simulcast Capability

There are numerous media properties that can be varied to construct a set of Simulcast versions. A Simulcast enabled endpoint could also support Simulcast based on several of those properties. As long as those properties are relatively independent and if each Simulcast version need explicit definition in the SDP, this would lead to an exponential number of Simulcast version candidates and a very long SDP that is likely also hard to interpret. There is thus a need to limit the Simulcast version candidates included in the SDP to cover as small set of properties as possible.

If a legacy endpoint not supporting Simulcast were to be presented with an SDP including media descriptions for a set of Simulcast versions, it may not know how to correctly handle or interpret these "surplus" media descriptions.

Based on the functionality that Simulcast is intended to achieve, it should be clear that the reasons to send Simulcast versions are not the same as to receive Simulcast versions, seen from a single endpoint.

For these reasons, it is proposed to define two new SDP session level attributes, "a=sim-send-cap" and "a=sim-recv-cap", which explicitly signal support for Simulcast media transmission and Simulcast media reception, respectively, for that media description. "a=sim-send-cap" and "a=sim-recv-cap" MAY be used independently and simultaneously. These attributes are also proposed to have parameters indicating the media properties used to create the Simulcast versions, and their preferred ranking. The meaning of the attributes on SDP media level is undefined and MUST NOT be used.

```
simulcast-cap    = "a="( "sim-send-cap:" / "sim-recv-cap:" )
                  cap-prop-list
cap-prop-list    = cap-prop-entry *(WSP cap-prop-entry)
cap-prop-entry   = cap-prop ["=" q-value]
cap-prop         = "rtpmap"
                  / "fmt"
                  / "imageattr"
                  / "framerate"
                  / token ; for future extensions
q-value          = ( "0" "." 1*2DIGIT )
                  / ( "1" "." 1*2("0") )
                  ; Values between 0.00 and 1.00
; WSP and DIGIT defined in [RFC5234]
```



; token defined in [[RFC4566](#)]

Figure 1: ABNF for Simulcast Capability

The media property values are taken from existing (and could be extended to cover other or future) SDP attributes that express media properties that can be varied to create different Simulcast versions:

rtpmap: Differences in codec type, sampling rate (see [Section 4](#)), and number of channels.

fmt: Differences in codec-specific encoding parameters.

imageattr: Differences in video resolution and aspect ratio [[RFC6236](#)].

framerate: Differences in framerate.

The optional q-value expresses the relative preference to base a Simulcast version on that media property, with 1.00 meaning maximum (100%) preference and 0.00 meaning no (0%) preference. Several media properties can share the same q-value, in which case they are equally preferred. Not including any q-value for a media property value SHALL default to a q-value of 1.00.

The list of media properties is made extensible, to allow introducing additional dimensions for Simulcast versions.

#### **[6.1.1.1](#). Declarative Use**

When used as a declarative media description, sim-recv-cap indicates the configured end-point's required capability to recognize and receive a specified set of Source Packet Streams as Simulcast streams. In the same fashion, sim-send-cap requests the end-point to send a specified set of Source Packet Streams as Simulcast streams. sim-recv-cap and sim-send-cap MAY be used independently and at the same time and they need not specify the same capability properties.

#### **[6.1.1.2](#). Offer/Answer Use**

An offerer wanting to use Simulcast SHALL include either one or both of those attributes, depending on in which direction(s) Simulcast is both supported and desirable. An offerer that receives an answer without "a=sim-send-cap" or "a=sim-recv-cap" MUST NOT define or use any Simulcast alternatives in that direction to the answerer.



An answerer that does not understand the concept of Simulcast will also not know those attributes and will remove them in the SDP answer, as defined in existing SDP Offer/Answer procedures. An answerer that does understand the attributes and that wants to support Simulcast in the indicated direction SHALL reverse directionality of the attribute; "sim-send-cap" becomes "sim-recv-cap" and vice versa, and include it in the answer.

An offerer that intends to send Simulcast alternatives and thus includes "a=sim-send-cap", MUST also include at least one media property parameter that it intends to use to construct the Simulcast alternatives, but it MAY include more media property parameters. Including multiple media property parameters in "a=sim-send-cap" SHALL be interpreted as an offer to send Simulcast versions covering all combinations thereof, but MAY be further restricted by other information in the SDP such as for example the number of simulcast-related media descriptions in the SDP or use of max-ssrc signaling [[I-D.westerlund-mmusic-max-ssrc](#)].

An offerer that is capable of receiving Simulcast alternatives and thus includes "a=sim-recv-cap", MUST also include at least one media property parameter that it is willing to use as discriminator between received Simulcast alternatives, but MAY include more media property parameters. Including multiple media property parameters in "a=sim-recv-cap" SHALL be interpreted as an offer to receive Simulcast versions covering all combinations thereof, but MAY be further restricted by other information in the SDP such as for example the number of simulcast-related media descriptions in the SDP or use of max-ssrc signaling [[I-D.westerlund-mmusic-max-ssrc](#)].

An answerer that either lacks the capability or does not desire to use Simulcast versions based on a certain media property parameter in a specific direction MUST remove such media property parameter from "a=sim-send-cap" or "a=sim-recv-cap". The answerer MUST NOT add any media property parameters that were not included in the offer.

An answerer SHOULD take the offerer's q-values into account when choosing which media configurations ([Section 6.2](#)) to include in the answer and how to group them ([Section 6.3](#)) into the resulting Simulcast(s).

## **6.2. Media Configuration**

Media that constitutes a Simulcast version has certain desirable characteristics that is meant to suit one category of diverse receivers ([Section 3.1](#)). A receiver that is willing to receive Simulcast streams must be given sufficient means to express what it is capable of and desires to receive. A sender that is willing to





send Simulcast streams must similarly be given sufficient means to express what it is capable of and desires to send.

An obvious candidate to express those characteristics is the media format in an SDP media description, defined by the `rtpmap` and `fntp` attributes, which is typically mapped to an RTP Payload Type. Some of the most interesting characteristics for Simulcast purposes are however not included in `rtpmap` or `fntp`, but are instead defined as separate attributes. Some of those individual attributes are possible to directly relate to a defined media format and could form a configuration together with the media format, but some attributes cannot be related to a specific media format and using the existing media format as a common identifier for a media configuration is not fully sufficient.

The act of Simulcast is trying to handle senders and receivers belonging to the vast multi-dimensional parameter space of "media configuration" by sub-dividing that parameter space into manageable and meaningful sub-sets. Communication between a sender and a receiver can be established successfully only when the actually sent media configuration (sub-set) fits within the receiver's available media configuration sub-set. At the same time, practical and implementation aspects often limits the size of those sub-sets. When that receiver or sender sub-set is either too small or is not known, the probability of successful communication decreases significantly. To increase the probability of finding a match between sender and receiver media configurations, it is essential that a media configuration can be a set instead of a single point in the parameter space, i.e. include parameter listings and/or ranges instead of single values.

Therefore, it is proposed to define a new media level SDP attribute, "a=config-id", which has relate the needed parameter types and the corresponding value ranges that together constitute a Simulcast media configuration. Each SDP media description MAY contain zero or more config-id attributes. The meaning of the attribute on SDP session level is undefined and MUST NOT be used.

```
configuration      = "a=config-id:" config-id WSP config-dir
                    WSP config-list
config-id          = token
config-dir         = "send"
                    / "recv"
config-list        = config-entry *(WSP config-entry)
config-entry       = "pt" "=" pt-value *("," pt-value)
                    / image-attr
                    / "framerate" "=" fr-param
                    / "b" "=" bw-mod ":" bw-value *1("-" bw-value)
```



```

        / ext-config-id [ "=" ext-config-value ]
        ; for future ext
image-attr    = "imageattr" "=" resolution-list
resolution-list = resolution-set *("," resolution-set)
ext-config-id  = token
ext-config-value = non-ws-string
pt-value      = 1*3DIGIT ; could be made more strict
resolution-set = "[" "x=" xyrange "," "y=" xyrange *key-values "]"
key-values     = ( "," key-value )
key-value      = ( "sar=" srange )
               / ( "par=" prange )
               / ( "q=" qvalue )
onetone       = "1" / "2" / "3" / "4" / "5"
               / "6" / "7" / "8" / "9"
xyvalue       = onetone *5DIGIT
step          = xyvalue
xyrange       = ( "[" xyvalue ":" [ step ":" ] xyvalue "]" )
               / ( "[" xyvalue 1*( "," xyvalue ) "]" )
               / ( xyvalue )
spvalue       = ( "0" "." onetone *3DIGIT )
               / ( onetone "." 1*4DIGIT )
srange        = ( "[" spvalue 1*( "," spvalue ) "]" )
               / ( "[" spvalue "-" spvalue "]" )
               / ( spvalue )
prange        = ( "[" spvalue "-" spvalue "]" )
qvalue        = ( "0" "." 1*2DIGIT )
               / ( "1" "." 1*2("0") )
fr-param      = fr-value *("," fr-value)
               / fr-value "-" fr-value
fr-value      = 1*3DIGIT [ "." 1*2DIGIT ]
bw-mod        = "AS"
               / "TIAS"
               / token ; for future extensions
bw-value      = 1*DIGIT
; WSP, DQUOTE and DIGIT defined in RFC5234
; token and non-ws-string defined in RFC4566

```

Figure 2: ABNF for Media Configuration

A media configuration is thus identified by:

**config-id:** A token that identifies the media configuration, which MUST be unique across all media configurations and media descriptions in the SDP.

**config-dir:** The direction for the stream(s) receiving the media configuration, as seen from the part issuing the SDP.



The media configuration MUST contain at least one and MAY contain more of the below media configuration entries. Each entry type MUST NOT appear more than once in every media configuration.

pt: A comma-separated list of media formats, RTP payload types, which MUST be defined within the same media description as config-id. This describes the allowed set of codecs or codec configurations for this media configuration. MUST be present in every media configuration.

imageattr: An OPTIONAL listing of preferred image resolutions for this media configuration. MUST NOT be used with other than video and image media types. An imageattr media configuration entry MUST NOT conflict with any "a=imageattr" attribute present in the same media description.

framerate: An OPTIONAL range or enumeration of preferred framerates for this media configuration. MUST NOT be used with other than video media types. The high end of the range MUST be equal to or larger than the low end. An enumerating framerate media configuration entry MUST include the value of the "a=framerate" attribute, if any. A framerate range media configuration entry MUST include the "a=framerate" value in the range.

b: An acceptable bandwidth range for this media configuration. Either one of the defined bandwidth modifiers MAY be used, which MUST share semantics with corresponding bandwidth modifiers from the SDP bandwidth attribute. The bandwidth value MUST be interpreted as defined by the bandwidth modifier. The high end of the range MUST be equal to or larger than the low end. The high end of the range MUST NOT exceed the bandwidth parameter in the same media description, if any. The sum of bandwidth range low ends for all media configurations within a media description MUST NOT exceed the value of that media description's bandwidth parameter. MUST be present in every media configuration.

Media configuration entry types "pt" and "b" MUST be supported by all implementations of this specification. Otherwise, an implementation MAY ignore any media configuration entry types that are not understood. A media configuration MAY be re-used to describe more than a single Source Packet Stream.

#### **6.2.1. Simulcast Limitations**

The Session and Media level attributes and parameters outside of individual media configurations (a=config-id) provides limitations on the set of media configurations in simultaneous use. For example a media description bandwidth limitation using b=AS would apply on all



the Packet Streams sent within the scope of that media description, thus forcing the sum of the media configuration bandwidth in use to share that available bandwidth. Don't forget other Packet Streams such as RTP retransmission or FEC flows that also needs to be included.

There exist a number of different limitations, and this section does not intend to be complete. The payload formats and their configurations can offer limitations, for example video profile and levels imposes a joint limit on bit-rate, frame-rate and resolution. The bandwidth parameters on session and media description level apply according to their semantics and their level. Packetization limitations, e.g. maxptime, as well as recommendations apply to all the configurations within the scope where this parameter is defined.

It is important to note that limits, such as bandwidth expressed within a media configuration are not limited by the media description values. First of all, the sum of bit-rates across all media configurations in a media description can be greater than the media description limit as not all configurations may be in simultaneous use. For example, only a single configuration can be enabled, which is then allowed to consume the full outer limit. Secondly, the media configuration directionality needs to be taken into account, for example that SDP receiver limitations are not applied to the sender configuration.

#### **6.2.2. Declarative Use**

When used as a declarative media description, config-id with recv parameter indicates the configured end-point's required media configuration to receive a specified set of Source Packet Streams as Simulcast streams. In the same fashion, config-id with send parameter requests the end-point to use the specified media configuration when sending a specified set of Source Packet Streams as Simulcast streams.

#### **6.2.3. Offer/Answer Use**

An offerer wanting to use Simulcast in a specific direction SHALL use config-id to describe the media configurations to use in that direction in the Offer.

An answerer receiving a config-id media configuration for a specific direction, accepting to use that media configuration SHALL include a corresponding media configuration with the reverse direction in the Answer. The config-id identification value MUST be kept between the Offer and the Answer. An answerer not accepting to use a specific media configuration SHALL remove it from the Answer.





The Answer MUST keep exactly the same media configuration types in a media configuration as were present in the corresponding media configuration in the Offer.

The answerer MAY remove values from enumerations and MAY reduce ranges of media configuration entries in the Answer. If the reduced media configuration entry relates to the answerer's send direction, negotiation is complete and no further action is needed. If the reduced media configuration relates to the answerer's receive direction, the offerer SHOULD send another Offer where that related, send direction media configuration is reduced at least to the level in the previous Answer, but MAY be reduced even more, and MAY be removed entirely.

### 6.3. Grouping Simulcast Configurations

A set of media configurations ([Section 6.2](#)) is needed to describe a Simulcast. Each Source Packet Stream in the Simulcast share the same Media Source, but have different media configurations. Thus, the actual grouping of media configurations is what defines a specific Simulcast. It is proposed to define two new media level and session level SDP attributes, "a=sim-send" and "a=sim-recv", which uses config-id values to group media configurations for the purpose of Simulcast transmission and reception, respectively. "a=sim-send" and "a=sim-recv" MAY be used independently and simultaneously. They MAY be used on session level to group media configurations when different Simulcast encodings of a Media Source are to be sent in different Media Transports and RTP sessions. They MAY also be used on media level to group media configurations when different Simulcast encodings of a Media Source are to be sent based on the same media description and thus use the same Media Transport and RTP session. When used on media level, the Simulcast direction MAY conflict with the general media description direction, but a conflict MUST be interpreted as the Simulcast being effectively inhibited. For example, sim-send in a recvonly media description means that no Simulcast Source Packet Streams are sent.

```

simulcast           = "a="( "sim-send:" / "sim-recv:" ) config-id-list
config-id-list      = config-item *(WSP config-item)
config-item         = config-id [":" config-param-list]
config-id           = token
config-param-list   = config-param *("," config-param)
config-param        = "inactive"
                    / token ["=" param-value] ; for future extension
param-value         = 1*(value-char)
                    / DQUOTE non_ws_string DQUOTE
value-char          = token-char / %x28 / %x29 / %x2F / %x3A-3C
                    / %x3E-40 / %x5B-5D ; VCHAR except "=" and ",", "
```



```
; WSP and VCHAR defined in [RFC5234]  
; token, token-char and non_ws_string defined in [RFC4566]
```

Figure 3: ABNF for Simulcast Configuration Grouping

The config-id identification of a media configuration MUST be defined by a "config-id" attribute in any of the media descriptions that are part of the SDP.

#### **[6.3.1.](#) Declarative Use**

When used as a declarative media description, sim-recv indicates the configured end-point's required ability to receive Source Packet Streams with the specified set of media configurations as Simulcast streams. In the same fashion, sim-send requests the end-point to send Source Packet Streams with the specified set of media configurations as Simulcast streams.

The configuration parameter "inactive" SHALL be interpreted as the related Source Packet Stream is in PAUSED state [[I-D.westerlund-avtext-rtp-stream-pause](#)] at the start of the session, and applicable RTP level procedures from that specification SHALL be applied.

#### **[6.3.2.](#) Offer/Answer Use**

An offerer wanting to send a set of Source Packet Streams as Simulcast streams includes sim-send in the Offer to describe which media configurations to use for that Simulcast. Similarly, an offerer wanting to receive a set of Source Packet Streams as Simulcast streams includes sim-recv in the Offer to describe which media configurations to use for that Simulcast.

An answerer receiving sim-send, accepting to receive those media configurations as Simulcasted Source Packet Streams SHALL include sim-recv with the accepted media configurations in the Answer. Similarly, an answerer receiving sim-recv, accepting to send those media configurations as Simulcasted Source Packet Streams SHALL include sim-send with the accepted media configurations in the Answer. An answerer MAY remove media configurations from sim-send or sim-recv included in the Answer compared to the ones included in the sim-send or sim-recv in the Offer. The answerer MUST NOT add any media configurations to sim-send or sim-recv in the Answer that were not in the corresponding ones in the Offer.

An "inactive" parameter present in the Offer MUST be kept in the Answer. The Answer MAY add an "inactive" parameter to any of the



media configurations. An "inactive" parameter on a media configuration in "sim-recv" is equivalent to a PAUSE (or in some cases, an equivalent TMMBR 0) message [[I-D.westerlund-avtext-rtp-stream-pause](#)] being sent for the received Source Packet Stream at the start of the session, and applicable RTP level procedures from that specification SHALL be applied. An "inactive" parameter on a media configuration in "sim-send" is equivalent to the related Source Packet Stream being in PAUSED state at the start of the session, and applicable RTP level procedures SHALL be applied.

The number of different Source Packet Streams used for a Simulcast related to a single media description MUST NOT exceed the number of listed media configurations in the corresponding sim-recv in that media description sent by the media receiver.

#### **[6.4.](#) Relating Simulcast Versions**

To ensure that Simulcast Packet Streams can be related correctly on RTP level, SDES SRCNAME [[I-D.westerlund-avtext-rtcp-sdes-srcname](#)] MUST be used to label Simulcast versions belonging to the same Media Source. The RTP Header Extension option of that specification MAY be used with Simulcast.

The SRCNAME identifier for Simulcast MUST contain a first part that uniquely identifies the Media Source within a given CNAME, followed by a single "." (period) and the config-id as defined above ([Section 6.2](#)).

The SRCNAME parameter to source-specific signaling [[RFC5576](#)] ("a=ssrc") MAY be used for Source Packet Streams in the send direction to relate SRCNAME to SSRC already in the SDP.

#### **[6.5.](#) Two-Phase Negotiation**

The new "a=sim-send-cap" and "a=sim-recv-cap" attributes MAY be included in the SDP as an optional pre-stage in a two-phased approach, where the pre-stage involves a first SDP Offer/Answer procedure that only establishes Simulcast capability at both the offerer and the answerer. This has the additional advantage to avoid sending media descriptions related to Simulcast to an endpoint that does not support simulcast. In case two Offer/Answer procedures are already used for other reasons, it will not incur any significant extra signaling round-trips. Such other two-phase techniques include use of SIP OPTIONS, SIP UPDATE [[RFC3311](#)] with reliable provisional responses, and BUNDLE [[I-D.ietf-mmusic-sdp-bundle-negotiation](#)].



Thus, when using the pre-stage Offer/Answer, it SHOULD NOT include any simulcast-grouped media descriptions, which SHOULD then instead be added in a main Offer/Answer phase. When using the pre-stage Offer/Answer, half a signaling round-trip time can sometimes be saved if main phase is initiated by the Simulcast receiver, meaning that the endpoint that included "a=sim-recv" in the pre-stage SDP is the offerer in the main phase. If both endpoints are Simulcast receivers, it does not matter which endpoint sends the main Offer, using regular Offer/Answer rules to handle any race conditions.

It is not possible to use any pre-stage to establish capability with declarative SDP, in which case it SHALL be by-passed, using only the main phase directly.

## 6.6. Signaling Examples

These examples are for a case of client to video conference service using a centralized media topology with an RTP mixer.

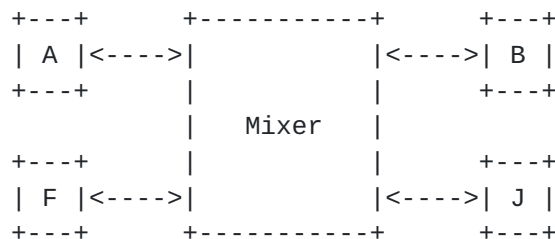


Figure 4: Four-party Mixer-based Conference

### 6.6.1. Unified Plan Client

Alice is calling in to the mixer with a Simulcast-enabled Unified Plan client capable of a single Media Source per media type. The only difference to a non-Simulcast client is capability to send video resolution [[RFC6236](#)] ("imageattr") and framerate based Simulcast. Alice uses a pre-stage Offer, which looks like:

```

v=0
o=alice 2362969037 2362969040 IN IP4 192.0.2.156
s=Simulcast Enabled Unified Plan Client
t=0 0
c=IN IP4 192.0.2.156
b=AS:665
a=sim-send-cap:imageattr framerate
m=audio 49200 RTP/AVP 96 8
b=AS:145
a=rtpmap:96 G719/48000/2
a=rtpmap:8 PCMA/8000

```





```
m=video 49300 RTP/AVP 97
b=AS:520
a=rtpmap:97 H264/90000
a=fmtp:97 profile-level-id=42c01e
a=imageattr:97 send [x=640,y=360] [x=320,y=180] \
    recv [x=640,y=360] [x=320,y=180]
```

Figure 5: Unified Plan Simulcast Pre-Stage Offer

In this pre-stage, the only thing in the SDP that indicates Simulcast capability is the line in the video media description containing the "sim-send-cap" attribute, which also indicates that sent Simulcast versions can differ in video resolution and/or framerate.

The Answer from the server indicates both that it too is Simulcast capable and that it would prefer to use video resolution ("imageattr") based Simulcast, but that it supports both video resolution and framerate. Should it not have been Simulcast capable, the "a=sim-recv-cap" line would not have been present and communication would have started with the media negotiated in the SDP.

```
v=0
o=server 823479283 1209384938 IN IP4 192.0.2.2
s=Answer to Simulcast Enabled Unified Plan Client
t=0 0
c=IN IP4 192.0.2.43
b=AS:665
a=sim-recv-cap:imageattr=1.0 framerate=0.8
m=audio 49200 RTP/AVP 96
b=AS:145
a=rtpmap:96 G719/48000/2
m=video 49300 RTP/AVP 97
b=AS:520
a=rtpmap:97 H264/90000
a=fmtp:97 profile-level-id=42c01e
a=imageattr:97 send [x=640,y=360] [x=320,y=180] \
    recv [x=640,y=360] [x=320,y=180]
```

Figure 6: Unified Plan Simulcast Pre-Stage Answer

Since the server is the Simulcast media receiver, it immediately initiates another Offer/Answer including details on the Simulcast versions. The server also keeps the "sim-recv-cap" as explicit Simulcast capability indication in this main Offer/Answer. Note that the "non-simulcast" media can be started already now, before the main



Offer/Answer, with the only restriction that the Simulcast functionality is not yet established.

```
v=0
o=server 823479283 1209384938 IN IP4 192.0.2.2
s=Server Inviting Simulcast Enabled Unified Plan Client
t=0 0
c=IN IP4 192.0.2.43
b=AS:825
a=sim-recv-cap:imageattr=1.0 framerate=0.8
m=audio 49200 RTP/AVP 96
b=AS:145
a=rtpmap:96 G719/48000/2
m=video 49300 RTP/AVP 97
b=AS:2200
a=rtpmap:97 H264/90000
a=fmtp:97 profile-level-id=42c01e
a=config-id:a recv pt=97 imageattr=[x=640,y=360],[x=1280,y=720] \
    framerate=25-60 b=AS:500-2500
a=config-id:b recv pt=97 imageattr=[x=320,y=180],[x=640,y=360] \
    framerate=25-60 b=AS:150-500
a=config-id:c recv pt=97 imageattr=[x=256,y=144],[x=320,y=180] \
    framerate=10-30 b=AS:100-250
a=sim-recv:a b c
```

Figure 7: Unified Plan Simulcast Main Offer

The server chooses to structure the Answer according to Unified Plan and has added three config-id lines in the video media description, one for each Simulcast media configuration that it is prepared to receive. Each media configuration refers to a defined media format, and lists a set of preferred video resolutions as well as a range of acceptable framerates, concluded by a bandwidth range. It also includes the sim-recv attribute for those three media configurations, indicating that the Simulcast it is prepared to receive in this media description can include one or more of those media configurations.

Alice's Answer is:

```
v=0
o=alice 2362969037 2362969040 IN IP4 192.0.2.156
s=Final answer from Simulcast Enabled Unified Plan Client
t=0 0
c=IN IP4 192.0.2.156
b=AS:825
a=sim-send-cap:imageattr framerate
m=audio 49200 RTP/AVP 96
```



```

b=AS:145
a=rtpmap:96 G719/48000/2
m=video 49300 RTP/AVP 97
b=AS:520
a=rtpmap:97 H264/90000
a=fmtp:97 profile-level-id=42c01e
a=config-id:b send pt=97 imageattr=[x=640,y=360] \
    framerate=25-30 b=AS:150-400
a=config-id:c send pt=97 imageattr=[x=320,y=180] \
    framerate=10-12.5 b=AS:100-150
a=sim-send:b c:inactive
a=ssrc:31053821 cname=SDIe93850aQFid9P srcname=1.b
a=ssrc:43298172 cname=SDIe93850aQFid9P srcname=1.c
a=imageattr:97 send [x=640,y=360] [x=320,y=180] \
    recv [x=640,y=360] [x=320,y=180]

```

Figure 8: Unified Plan Simulcast Main Answer

The Simulcast capability, `sim-send-cap`, is kept from Alice's previous Offer. One of the media configurations from the server Offer, `config-id:a`, is not acceptable to Alice's client for some reason and is removed from the Answer. The resulting Simulcast, described by `sim-send`, thus contains two media configurations, `b` and `c`, where `c` is initially set to "inactive" that effectively means it is paused from the start of the session. The media configuration parameter value ranges are in some cases reduced, which makes a more precise definition of what will actually be sent. This Answer SDP also includes a specification of the SSRC values that will be sent and what media configurations those SSRC will carry, by including the `srcname` parameter. The first part of `srcname`, before the ".", is the Media Source identification. Both SSRC share the same Media Source identification, since they are part of the same Simulcast. The second part, after the ".", is the `config-id` of the media configuration sent with that SSRC.

#### [6.6.2.](#) Multi-Transport Client

Bob is calling in to the mixer with a Simulcast-enabled client, like Alice's capable of a single Media Source per media type, but also capable of sending Source Packet Streams as Simulcast versions on separate Media Transports. In this example, Bob's client knows that the server is capable of Simulcast and does not use any pre-stage Offer, but goes straight to the main Offer.

```

v=0
o=bob 94572932847 3429478298 IN IP4 192.0.2.93
s=Offer from Simulcast Enabled Multi-Transport Client

```



```
t=0 0
c=IN IP4 192.0.2.93
b=AS:825
a=sim-send-cap:imageattr=1.0 framerate=0.9
a=sim-send:x y
m=audio 50138 RTP/AVP 101
b=AS:145
a=rtpmap:101 G719/48000/2
m=video 50226 RTP/AVP 118
b=AS:500
a=rtpmap:118 H264/90000
a=fmtp:118 profile-level-id=42c01e
a=config-id:x send pt=118 imageattr=[x=320,y=180],[x=640,y=360] \
    framerate=25-50 b=AS:200-500
a=ssrc:3929384298 cname=Nsdko390en828Fkn srcname=M.x
a=imageattr:118 send [x=640,y=360] [x=320,y=180] \
    recv [x=640,y=360] [x=320,y=180]
m=video 50228 RTP/AVP 119
b=AS:150
a=config-id:y send pt=119 imageattr=[x=256,y=144],[x=320,y=180] \
    framerate=12.5-25 b=AS:100-200
a=ssrc:1923419284 cname=Nsdko390en828Fkn srcname=M.y
a=imageattr:119 send [x=320,y=180] [x=256,y=144]
a=sendonly
```

Figure 9: Multi-Transport Simulcast Main Offer

As can be seen from above, this Offer uses sim-send on session level and has split the Simulcast media configurations on two media descriptions, in order to be able to use separate Media Transports and enable differentiated treatment of the two Simulcast streams.

The server accepts this structure to the Answer:

```
v=0
o=server 283479882 9384298374 IN IP4 192.0.2.2
s=Server Answering Simulcast Enabled Multi-Transport Client
t=0 0
c=IN IP4 192.0.2.45
b=AS:825
a=sim-recv-cap:imageattr framerate
a=sim-recv:x y
m=audio 49200 RTP/AVP 96
b=AS:145
a=rtpmap:96 G719/48000/2
m=video 49300 RTP/AVP 118
b=AS:500
```





```
a=rtpmap:118 H264/900000
a=fmtp:118 profile-level-id=42c01e
a=config-id:x recv pt=118 imageattr=[x=640,y=360] \
    framerate=25-50 b=AS:350-500
a=imageattr:118 send [x=640,y=360] [x=320,y=180] \
    recv [x=640,y=360] [x=320,y=180]
m=video 49300 RTP/AVP 119
b=AS:150
a=rtpmap:119 H264/900000
a=fmtp:119 profile-level-id=42c01e
a=config-id:y recv pt=119 imageattr=[x=256,y=144] \
    framerate=12.5-25 b=AS:120-150
a=imageattr:119 recv [x=320,y=180] [x=256,y=144]
a=recvonly
```

Figure 10: Multi-Transport Simulcast Main Answer

### 6.6.3. Multi-Source Client

Fred is calling in to the same conference as in the examples above with a three-camera, three-display system, thus capable of handling three separate Media Sources in each direction, where each Media Source is also Simulcast-enabled in the send direction. Fred's client is a Unified Plan client, restricted to a single Media Source per media description.

```
v=0
o=fred 238947129 823479223 IN IP4 192.0.2.125
s=Offer from Simulcast Enabled Multi-Source Client
t=0 0
c=IN IP4 192.0.2.125
b=AS:825
a=sim-send-cap:imageattr=1.0 framerate=0.5

m=audio 49200 RTP/AVP 98
b=AS:145
a=rtpmap:98 G719/48000/2

m=video 49600 RTP/AVP 100
b=AS:3500
a=rtpmap:100 H264/900000
a=fmtp:100 profile-level-id=42c02a
a=config-id:1h send pt=100 imageattr=[x=1920,y=1080] \
    framerate=30-60 b=AS:2000-3500
a=config-id:1m send pt=100 imageattr=[x=1280,y=720] \
    framerate=15-60 b=AS:1000-2000
a=config-id:1l send pt=100 imageattr=[x=640,y=360] \
```



```
    framerate=10-60 b=AS:200-1000
a=sim-send:1h 1m 1l
a=ssrc:2397234521 cname=EkeS32892Fe029DK srcname=1.1h
a=ssrc:1023894789 cname=EkeS32892Fe029DK srcname=1.1m
a=ssrc:4029284928 cname=EkeS32892Fe029DK srcname=1.1l
a=imageattr:100 send [x=1920,y=1080] [x=1280,y=720] [x=640,y=360] \
    recv [x=1920,y=1080] [x=1280,y=720] [x=640,y=360]

m=video 49600 RTP/AVP 100
b=AS:3500
a=rtpmap:100 H264/900000
a=fmtp:100 profile-level-id=42c02a
a=config-id:2h send pt=100 imageattr=[x=1920,y=1080] \
    framerate=30-60 b=AS:2000-3500
a=config-id:2m send pt=100 imageattr=[x=1280,y=720] \
    framerate=15-60 b=AS:1000-2000
a=config-id:2l send pt=100 imageattr=[x=640,y=360] \
    framerate=10-60 b=AS:200-1000
a=sim-send:2h 2m 2l
a=ssrc:2301017618 cname=EkeS32892Fe029DK srcname=2.2h
a=ssrc:639711316 cname=EkeS32892Fe029DK srcname=2.2m
a=ssrc:3293473905 cname=EkeS32892Fe029DK srcname=2.2l
a=imageattr:100 send [x=1920,y=1080] [x=1280,y=720] [x=640,y=360] \
    recv [x=1920,y=1080] [x=1280,y=720] [x=640,y=360]

m=video 49600 RTP/AVP 100
b=AS:3500
a=rtpmap:100 H264/900000
a=fmtp:100 profile-level-id=42c02a
a=config-id:3h send pt=100 imageattr=[x=1920,y=1080] \
    framerate=30-60 b=AS:2000-3500
a=config-id:3m send pt=100 imageattr=[x=1280,y=720] \
    framerate=15-60 b=AS:1000-2000
a=config-id:3l send pt=100 imageattr=[x=640,y=360] \
    framerate=10-60 b=AS:200-1000
a=sim-send:3h 3m 3l
a=ssrc:4115355057 cname=EkeS32892Fe029DK srcname=3.3h
a=ssrc:3196538337 cname=EkeS32892Fe029DK srcname=3.3m
a=ssrc:3757973912 cname=EkeS32892Fe029DK srcname=3.3l
a=imageattr:100 send [x=1920,y=1080] [x=1280,y=720] [x=640,y=360] \
    recv [x=1920,y=1080] [x=1280,y=720] [x=640,y=360]
```

Figure 11: Fred's Multi-Source Simulcast Main Offer



The three media descriptions for video are essentially the same, except values that needs to be unique are provided unique values. The above also assumes that BUNDLE will be used across these three video media description to create a common RTP session.

## 7. Network Aspects

Simulcast is in defined as the act of sending multiple alternative encodings of the same underlying media source. When transmitting multiple independent streams that originate from the same source, it could potentially be done in several different ways using RTP. A general discussion on considerations for use of the different RTP multiplexing alternatives can be found in Guidelines for Multiplexing in RTP [[I-D.ietf-avtcore-multiplex-guidelines](#)]. Discussion and clarification on how to handle multiple streams in an RTP session can be found in [[I-D.ietf-avtcore-rtp-multi-stream](#)].

The network aspects that are relevant for Simulcast are:

Quality of Service: When using Simulcast it might be of interest to prioritize a particular Simulcast version, rather than applying equal treatment to all versions. For example, lower bit-rate versions may be prioritized over higher bit-rate versions to minimize congestion or packet losses in the low bit-rate versions. Thus, there is a benefit to use a Simulcast solution that supports QoS as good as possible. By separating Simulcast versions into different RTP sessions and send those RTP sessions over different Media Transports, a Simulcast version can be prioritized by existing flow based QoS mechanisms. When using unicast, QoS mechanisms based on individual packet marking are also feasible, which do not require separation of Simulcast versions into different RTP sessions to apply different QoS.

NAT/FW Traversal: Using multiple RTP sessions will incur more cost for NAT/FW traversal unless they can re-use the same transport flow, which can be achieved by either one of multiplexing multiple RTP sessions on a single lower layer transport [[I-D.westerlund-avtcore-transport-multiplexing](#)] or Multiplexing Negotiation Using SDP Port Numbers [[I-D.ietf-mmusic-sdp-bundle-negotiation](#)]. If flow based QoS with any differentiation is desirable, the cost for additional transport flows is likely necessary.

Multicast: Multiple RTP sessions will be required to enable combining Simulcast with multicast. Different Simulcast versions have to be separated to different multicast groups to allow a multicast receiver to pick the version it wants, rather than receive all of them. In this case, the only reasonable



implementation is to use different RTP sessions for each multicast group so that reporting and other RTCP functions operate as intended.

## **8. IANA Considerations**

This document requests that five new attributes, `sim-send-cap`, `sim-recv-cap`, `sim-send`, `sim-recv`, and `config-id`. It is also requested to make a new registry of defined parameters taken from existing SDP attributes for `sim-send-cap`, `sim-recv-cap`, and `config-id`.

Formal registrations to be written.

## **9. Security Considerations**

The Simulcast capability and configuration attributes and parameters are vulnerable to attacks in signaling.

A false inclusion of Simulcast attributes may result in generation of a second phase SDP that potentially contains a large number of non-supported media descriptions expressing Simulcast alternatives. A correct SDP implementation will however be able to reject any non-supported media descriptions and the effect from that should be limited.

A hostile removal of the Simulcast attributes will result in skipping any second phase Offer/Answer and that Simulcast is not used.

The Simulcast grouping semantics are vulnerable to attacks in the signalling. Changing the set of media configurations that are used in a Simulcast will impact the number of Source Packet Streams.

A hostile removal of Simulcast grouping will prevent streams from being interpreted as Simulcast, which obviously prevents use of the Simulcast functionality. It will also risk that intended Simulcast streams are instead presented as separate, independent streams to a receiver.

Neither of the above will likely have any major consequences and can be mitigated by signaling that is at least integrity and source authenticated to prevent an attacker to change it.

## **10. Contributors**

Morgan Lindqvist and Fredrik Jansson, both from Ericsson, have contributed with important material to the first versions of this document.





## [11.](#) Acknowledgements

## [12.](#) References

### [12.1.](#) Normative References

- [I-D.westerlund-avtext-rtcp-sdes-srcname]  
Westerlund, M., "RTCP Source Description Item SRCNAME to Label Individual Media Sources", [draft-westerlund-avtext-rtcp-sdes-srcname-03](#) (work in progress), October 2013.
- [I-D.westerlund-avtext-rtp-stream-pause]  
Akram, A., Burman, B., Grondal, D., and M. Westerlund, "RTP Media Stream Pause and Resume", [draft-westerlund-avtext-rtp-stream-pause-03](#) (work in progress), October 2012.
- [I-D.westerlund-mmusic-max-ssrc]  
Holmberg, C., Westerlund, M., and F. Jansson, "Multiple Synchronization Sources (SSRC) in SDP Media Descriptions", [draft-westerlund-mmusic-max-ssrc-02](#) (work in progress), September 2013.
- [RFC2119] Bradner, S., "Key words for use in RFCs to Indicate Requirement Levels", [BCP 14](#), [RFC 2119](#), March 1997.
- [RFC3311] Rosenberg, J., "The Session Initiation Protocol (SIP) UPDATE Method", [RFC 3311](#), October 2002.
- [RFC3550] Schulzrinne, H., Casner, S., Frederick, R., and V. Jacobson, "RTP: A Transport Protocol for Real-Time Applications", STD 64, [RFC 3550](#), July 2003.
- [RFC4566] Handley, M., Jacobson, V., and C. Perkins, "SDP: Session Description Protocol", [RFC 4566](#), July 2006.
- [RFC4568] Andreassen, F., Baugher, M., and D. Wing, "Session Description Protocol (SDP) Security Descriptions for Media Streams", [RFC 4568](#), July 2006.
- [RFC5109] Li, A., "RTP Payload Format for Generic Forward Error Correction", [RFC 5109](#), December 2007.
- [RFC5234] Crocker, D. and P. Overell, "Augmented BNF for Syntax Specifications: ABNF", STD 68, [RFC 5234](#), January 2008.
- [RFC5285] Singer, D. and H. Desineni, "A General Mechanism for RTP Header Extensions", [RFC 5285](#), July 2008.



- [RFC5576] Lennox, J., Ott, J., and T. Schierl, "Source-Specific Media Attributes in the Session Description Protocol (SDP)", [RFC 5576](#), June 2009.
- [RFC5888] Camarillo, G. and H. Schulzrinne, "The Session Description Protocol (SDP) Grouping Framework", [RFC 5888](#), June 2010.
- [RFC6236] Johansson, I. and K. Jung, "Negotiation of Generic Image Attributes in the Session Description Protocol (SDP)", [RFC 6236](#), May 2011.

## **12.2. Informative References**

- [I-D.ietf-avtcore-multiplex-guidelines]  
Westerlund, M., Perkins, C., and H. Alvestrand,  
"Guidelines for using the Multiplexing Features of RTP to Support Multiple Media Streams", [draft-ietf-avtcore-multiplex-guidelines-01](#) (work in progress), July 2013.
- [I-D.ietf-avtcore-rtp-multi-stream]  
Lennox, J., Westerlund, M., Wu, W., and C. Perkins,  
"Sending Multiple Media Streams in a Single RTP Session", [draft-ietf-avtcore-rtp-multi-stream-01](#) (work in progress), July 2013.
- [I-D.ietf-avtcore-rtp-topologies-update]  
Westerlund, M. and S. Wenger, "RTP Topologies", [draft-ietf-avtcore-rtp-topologies-update-00](#) (work in progress), April 2013.
- [I-D.ietf-mmusic-sdp-bundle-negotiation]  
Holmberg, C., Alvestrand, H., and C. Jennings,  
"Multiplexing Negotiation Using Session Description Protocol (SDP) Port Numbers", [draft-ietf-mmusic-sdp-bundle-negotiation-05](#) (work in progress), October 2013.
- [I-D.lennox-raiarea-rtp-grouping-taxonomy]  
Lennox, J., Gross, K., Nandakumar, S., and G. Salgueiro,  
"A Taxonomy of Grouping Semantics and Mechanisms for Real-Time Transport Protocol (RTP) Sources", [draft-lennox-raiarea-rtp-grouping-taxonomy-03](#) (work in progress), October 2013.
- [I-D.westerlund-avtcore-transport-multiplexing]  
Westerlund, M. and C. Perkins, "Multiple RTP Sessions on a Single Lower-Layer Transport", [draft-westerlund-avtcore-transport-multiplexing-06](#) (work in progress), August 2013.



- [RFC3264] Rosenberg, J. and H. Schulzrinne, "An Offer/Answer Model with Session Description Protocol (SDP)", [RFC 3264](#), June 2002.
- [RFC3569] Bhattacharyya, S., "An Overview of Source-Specific Multicast (SSM)", [RFC 3569](#), July 2003.
- [RFC4588] Rey, J., Leon, D., Miyazaki, A., Varsa, V., and R. Hakenberg, "RTP Retransmission Payload Format", [RFC 4588](#), July 2006.
- [RFC5117] Westerlund, M. and S. Wenger, "RTP Topologies", [RFC 5117](#), January 2008.
- [RFC5245] Rosenberg, J., "Interactive Connectivity Establishment (ICE): A Protocol for Network Address Translator (NAT) Traversal for Offer/Answer Protocols", [RFC 5245](#), April 2010.
- [RFC6190] Wenger, S., Wang, Y., Schierl, T., and A. Eleftheriadis, "RTP Payload Format for Scalable Video Coding", [RFC 6190](#), May 2011.

## **Appendix A. Discussion on Receiver Diversity**

Receiver diversity can be handled in a number of different ways, each with its own advantages and disadvantages. In that, there are relations between RTP Mixer processing requirement, bandwidth usage on uplink from sending Participant to RTP Mixer, bandwidth usage on downlink from RTP Mixer to receiving Participant, and media Quality of Experience at the receiving Participant.

The following is a listing of possible approaches:

1. Lowest Common Denominator: Create a single Source Packet Stream per Media Source and, assuming that everyone can receive a "simple" stream, adapt the characteristics of that Source Packet Stream already at the sending Participant to the lowest common denominator among all receiving Participants. Let the RTP Mixer forward this single Source Packet Stream to all receiving Participants. The advantages are low bandwidth usage on both uplink and downlink and low RTP Mixer processing requirements. The disadvantage is that the least capable receiver and/or network path dictates the (low) QoE for everyone else.
2. Individual Transcoding: Create a single Source Packet Stream per Media Source with characteristics governed by resources available to the sending Participant and the network path to the RTP Mixer.



Let the RTP Mixer transcode (decode and re-encode) that into individual Source Packet Streams for each receiving Participant, governed by the RTP Mixer resources, receiving Participant resources, and the network path to that Participant. The advantages are adapted although overall slightly lowered QoE (due to transcoding) to each Participant and optimised bandwidth usage on both uplink and downlink. The disadvantage is (very) high RTP Mixer processing requirements.

3. Individual Simulcast: Create individual Source Packet Streams of each Media Source to each receiving Participant, constituting a complete individual Simulcast. Let the RTP Mixer forward each individual Source Packet Stream to the targeted receiving Participant. The advantages are low RTP Mixer processing and optimised downlink bandwidth. The disadvantage is (very) high uplink bandwidth.
4. Grouped Simulcast: For each Media Source, create a "suitable" logical grouping of receiving Participants in sub-groups with respect to available receiver resources, for example the resources listed above ([Section 3.1](#)). Create a set of Source Packet Streams for this Media Source with well-chosen characteristics, where each Source Packet Stream in the set is a good-enough fit to the receiving sub-group of Participants. This set of Source Packet Streams constitutes a Simulcast of the Media Source. The size of the set and the characteristics of each Source Packet Stream can be adjusted to cater for various restrictions in the sending Participant, receiving Participants in the sub-group, and network path(s) to the Participants in the sub-group. Let the RTP Mixer forward the same Source Packet Stream to all Participants in a sub-group, for all Source Packet Streams and sub-groups. The advantages are low RTP Mixer processing, near optimum QoE, and near optimum downlink bandwidth. The disadvantages are high uplink bandwidth and arguably that downlink bandwidth and QoE are optimum only for a sub-group and not per individual receiving Participant.

A summary of the advantages and disadvantages of the above four principle alternatives is given below (Table 1):

Method	Mixer CPU	Uplink	Downlink	QoE
1	Low	Low	Low	Low
2	Very high	Optimum	Optimum	Near optimum
3	Low	Very high	Optimum	Optimum
4	Low	High	Near optimum	Near optimum





Table 1: Receiver Diversity Handling Comparison

The authors of this document believes that alternative 4, the Grouped Simulcast, can be a good tradeoff whenever supported by sufficient uplink resources.

## Authors' Addresses

Magnus Westerlund  
Ericsson  
Farogatan 6  
SE-164 80 Kista  
Sweden

Phone: +46 10 714 82 87  
Email: [magnus.westerlund@ericsson.com](mailto:magnus.westerlund@ericsson.com)

Bo Burman  
Ericsson  
Farogatan 6  
SE-164 80 Kista  
Sweden

Phone: +46 10 714 13 11  
Email: [bo.burman@ericsson.com](mailto:bo.burman@ericsson.com)

Suhas Nandakumar  
Cisco  
170 West Tasman Drive  
San Jose, CA 95134  
USA

Email: [snandaku@cisco.com](mailto:snandaku@cisco.com)

