Network Working Group   B. Burman
Internet-Draft   M. Westerlund
Intended status: Standards Track   Ericsson
Expires: July 24, 2015   S. Nandakumar
                                                             M. Zanaty
                                                                 Cisco
                                                      January 20, 2015

Using Simulcast in SDP and RTP Sessions
draft-ietf-mmusic-sdp-simulcast-00

Abstract

   In some application scenarios it may be desirable to send multiple
   differently encoded versions of the same media source in independent
   RTP 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 an extension to SDP, and using
   RTP/RTCP identification methods to relate RTP streams belonging to
   the same media source. The SDP extension consists a new media level
   SDP attribute that express capability to send and/or receive
   simulcast RTP streams. One part of the RTP/RTCP identification
   method is included as a reference to a separate document, since it is
   useful also for other 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 July 24, 2015.

Copyright Notice

Table of Contents

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 RTP streams from each
participant and send some suitable set of possibly modified RTP
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 RTP stream adaptation to
different participants' constraints with the minimum possible impact
on both video quality and server performance.

Simulcast is defined in this memo as the act of simultaneously
sending multiple different encoded streams of the same media source,
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 encoded streams over
RTP [RFC3550] towards an intermediary so that the intermediary can
provide the wanted functionality by selecting which RTP stream to
forward to other participants in the session, and more specifically
how the identification and grouping of the involved RTP 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
SDP signaling and performing RTP simulcast.

2. Definitions

2.1. Terminology

This document makes use of the terminology defined in RTP Taxonomy
[I-D.ietf-avtext-rtp-grouping-taxonomy], RTP Topology [RFC5117] and
RTP Topologies Update [I-D.ietf-avtcore-rtp-topologies-update]. In
addition, the following terms are used:

RTP Mixer: An RTP middle node, 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).

RTP Switch: A common short term for the terms "switching RTP mixer",
   "source projecting middlebox", and "video switching MCU" as
   discussed in [I-D.ietf-avtcore-rtp-topologies-update].

Simulcast version: One encoded stream from the set of encoded
   streams that constitutes the simulcast for a single media source.

Simulcast version alternative: One encoded stream being encoded in
   one of possibly multiple alternative ways to create a simulcast
   version.

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 type of
topology.

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 RTP 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 RTP 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 RTP streams into a perfect fit to the resource situation
   of a receiving participant.

The use of simulcast 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 selects the
most appropriate simulcast version and tune in to the right media
transport to receive that version is also considered (Section 3.3) .
This enables large, heterogeneous receiver populations, when it comes
to capabilities and the use of network path bandwidth resources.

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. 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 fmtp 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 RTP 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 RTP streams per media source can be a good
tradeoff when using an RTP switch as middlebox, instead of sending a
single RTP 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 RTP 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 RTP 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
RTP 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 RTP 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 that instead has to 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 end point'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 a media source and provide it as a set of
different encoded streams, enabling the receivers to select the best
fit from this set themselves. The end point 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 RTP streams still
fit within the available bandwidth. If not, another set of encoded
streams from the ones offered in the simulcast will have to be
chosen.

When using IP multicast, the level of granularity that the receiver
can select from is decided by its ability to choose 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.

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 RTP stream they receive, for example in terms
of the aspects listed in Section 3.1. Sending a simulcast of RTP
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 RTP 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 using:

REQ-2.1: Legacy SDP with separate media transports per SDP media
   description.

REQ-2.2: Bundled SDP media descriptions.

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 codec parameters, as
   can be expressed by SDP format values and RTP payload types.

REQ-5: Compatibility. It must be possible to use simulcast in
   combination with other RTP mechanisms that generate additional RTP
   streams:

REQ-5.1: RTP Retransmission [RFC4588].

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

REQ-5.3: Related payload types such as audio Comfort Noise and/or
   DTMF.

REQ-6: Interoperability. The solution must be possible 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 with a single media
        source per SDP media description.

5. Proposed Solution Overview

   The proposed solution consists of signaling simulcast capability and
   configurations in SDP [RFC4566]:

   o An offer or answer can contain a number of simulcast versions,
      separate for send and receive directions.

   o An offer or answer can contain multiple, alternative simulcast
      versions in the same fashion as multiple, alternative codecs can
      be offered in a media description.

   o Currently, a single media source per SDP media description is
      assumed, which makes the solution work in an Unified Plan
      [I-D.roach-mmusic-unified-plan] context (although different from
      what is currently defined there), both with and without BUNDLE
      grouping.

   o The codec configuration for each simulcast version is expressed in
      terms of existing SDP formats (and typically RTP payload types).
      Some codecs may rely on codec configuration based on general
      attributes that apply for all formats within a media description,
      and which could thus not be used to separate different simulcast
      versions. This memo makes no attempt to address such
      shortcomings, but if needed instead encourages that a separate,
      general mechanism is defined for that purpose.

   o It is possible, but not required to use source-specific signaling
      [RFC5576] with the proposed solution.

6. Proposed Solution

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

6.1. Simulcast Capability

   Simulcast capability is expressed as a new media level SDP attribute,
   "a=simulcast". For each desired direction (send/recv/sendrecv), the
   simulcast attribute defines a list of simulcast versions (separated
   by semicolons), each of which is a list of alternative RTP payload
   types (separated by commas) for that simulcast version. The meaning
   of the attribute on SDP session level is undefined and MUST NOT be
   used. There MUST be at most one "a=simulcast" attribute per media
   description. The ABNF [RFC5234] for this attribute is:

```
simulcast-attribute = "a=simulcast" 1*3( WSP sc-dir-list )
sc-dir-list         = sc-dir WSP sc-fmt-list *( ";" sc-fmt-list )
sc-dir              = "send" / "recv" / "sendrecv"
sc-fmt-list         = sc-fmt *( "," sc-fmt )
sc-fmt              = fmt
; WSP defined in [RFC5234]
; fmt defined in [RFC4566]
```


                      Figure 1: ABNF for Simulcast

   There are separate and independent sets of parameters for simulcast
   in send and receive directions. When listing multiple directions,
   each direction MUST NOT occur more than once.

   Attribute parameters are grouped by direction and consist of a
   listing of SDP format tokens (usually corresponding to RTP payload
   types), which describe the simulcast versions to be used. The number
   of (non-alternative, see below) formats in the list sets a limit to
   the number of supported simulcast versions in that direction. The
   order of the listed simulcast versions in the "send" direction is not
   significant. The order of the listed simulcast versions in the
   "recv" direction expresses a preference which simulcast versions that
   are preferred, with the leftmost being most preferred, if the number
   of actually sent simulcast versions have to be reduced for some
   reason.

   Formats that have explicit dependencies [RFC5583] to other formats
   (even in the same media description) MAY be listed as different
   simulcast versions.

   Alternative simulcast versions MAY be specified as part of the
   attribute parameters by expressing each simulcast version format as a
   comma-separated list of alternative values. In this case, all
   combinations of those alternatives MUST be supported. The order of
   the alternatives within a simulcast version is not significant; codec
   preference is expressed by format type ordering on the m-line, using
   regular SDP rules.

   A simulcast version can use a codec defined such that the same RTP
   SSRC can change RTP payload type multiple times during a session,
   possibly even on a per-packet basis. A typical example can be a
   speech codec that makes use of Comfort Noise [RFC3389] and/or DTMF
   [RFC4733] formats. In those cases, such "related" formats MUST NOT
   be listed explicitly in the attribute parameters, since they are not
   strictly simulcast versions of the media source, but rather a
   specific way of generating the RTP stream of a single simulcast
   version with varying RTP payload type. Instead, only a single codec

format MUST be used per simulcast version or simulcast version
alternative (if there are such). The codec format SHOULD be the
codec most relevant to the media description, if possible to
identify, for example the audio codec rather than the DTMF. What
codec format to choose in the case of switching between multiple
equally "important" formats is left open, but it is assumed that in
the presence of such strong relation it does not matter which is
chosen.

Use of the redundant audio data [RFC2198] format could be seen as a
form of simulcast for loss protection purposes, but is not considered
conflicting with the mechanisms described in this memo and MAY
therefore be used as any other format. In this case the "red"
format, rather than the carried formats, SHOULD be the one to list as
a simulcast version on the "a=simulcast" line.

   Editor's note: Consider adding the possibility to put an RTP
   stream in "paused" state [I-D.ietf-avtext-rtp-stream-pause] from
   the beginning of the session, possibly starting it at a later
   point in time by applying RTP/RTCP level procedures from that
   specification.

6.1.1. Declarative Use

When used as a declarative media description, a=simulcast "recv"
direction formats indicates the configured end point's required
capability to recognize and receive a specified set of RTP streams as
simulcast streams. In the same fashion, a=simulcast "send" direction
requests the end point to send a specified set of RTP streams as
simulcast streams. The "sendrecv" direction combines "send" and
"recv" requirements, using the same format values for both.

If simulcast version alternatives are listed, it means that the
configured end point MUST be prepared to receive any of the "recv"
formats, and MAY send any of the "send" formats for that simulcast
version.

6.1.2. Offer/Answer Use

An offerer wanting to use simulcast SHALL include the "a=simulcast"
attribute in the offer. An offerer that receives an answer without
"a=simulcast" MUST NOT use simulcast towards the answerer. An
offerer that receives an answer with "a=simulcast" not listing a
direction or without any formats in a specified direction MUST NOT
use simulcast in that direction.

An answerer that does not understand the concept of simulcast will
also not know the attribute and will remove it in the SDP answer, as

defined in existing SDP Offer/Answer [RFC3264] procedures. An
answerer that does understand the attribute and that wants to support
simulcast in an indicated direction SHALL reverse directionality of
the unidirectional direction parameters; "send" becomes "recv" and
vice versa, and include it in the answer. If the offered direction
is "sendrecv", the answerer MAY keep it, but MAY also change it to
"send" or "recv" to indicate that it is only interested in simulcast
for a single direction. Note that, like all other use of SDP format
tags for the send direction in Offer/Answer, format tags related to
the simulcast send direction in an offer ("send" or "sendrecv") are
placeholders that refer to information in the offer SDP, and the
actual formats that will be used on the wire (including RTP Payload
Format numbers) depends on information included in the SDP answer.

An offerer listing a set of receive simulcast versions and/or
alternatives in the offer MUST be prepared to receive RTP streams for
any of those simulcast versions and/or alternatives from the
answerer.

An answerer that receives an offer with simulcast containing an
"a=simulcast" attribute listing alternative formats for simulcast
versions MAY keep all the alternatives in the answer, but it MAY also
choose to remove any non-desirable alternatives per simulcast version
in the answer. The answerer MUST NOT add any alternatives that were
not present in the offer.

An answerer that receives an offer with simulcast that lists a number
of simulcast versions, MAY reduce the number of simulcast versions in
the answer, but MUST NOT add simulcast versions.

An offerer that receives an answer where some simulcast version
alternatives are kept MUST be prepared to receive any of the kept
send direction alternatives, and MAY send any of the kept receive
direction alternatives from the answer. This is similar to the case
when the answer includes multiple formats on the m-line.

An offerer that receives an answer where some of the simulcast
versions are removed MAY release the corresponding resources (codec,
transport, etc) in its receive direction and MUST NOT send any RTP
streams corresponding to the removed simulcast versions.

The media formats and corresponding characteristics of encoded
streams used in a simulcast SHOULD be chosen such that they are
different. If this difference is not required, RTP duplication
[RFC7104] procedures SHOULD be considered instead of simulcast.

Note: The inclusion of "a=simulcast" or the use of simulcast does
not change any of the interpretation or Offer/Answer procedures
for other SDP attributes, like "a=fmtp".

6.2. Relating Simulcast Versions

As long as there is only a single media source per SDP media
description, simulcast RTP streams can be related on RTP level
through the RTP payload type, as specified in the SDP "a=simulcast"
attribute (Section 6.1) parameters. When using BUNDLE
[I-D.ietf-mmusic-sdp-bundle-negotiation] to use multiple SDP media
descriptions to specify a single RTP session, there is an
identification mechanism that allows relating RTP streams back to
individual media descriptions, after which the above RTP payload type
relation can be used.

6.3. Signaling Examples

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

```
        +---+      +-----------+      +---+
        | A |<---->|           |<---->| B |
        +---+      |           |      +---+
                   |   Mixer   |
        +---+      |           |      +---+
        | F |<---->|           |<---->| J |
        +---+      +-----------+      +---+
```
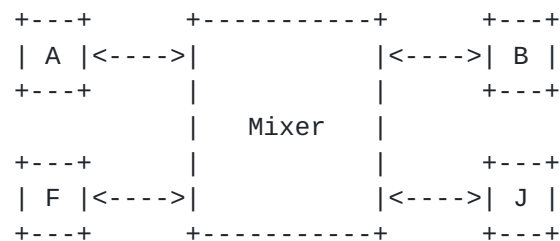
Figure 2: Four-party Mixer-based Conference

6.3.1. 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
client can send a simulcast of 2 video resolutions and frame rates:
HD 1280x720p 30fps and thumbnail 320x180p 15fps. Alice's Offer:

```
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
m=audio 49200 RTP/AVP 0
a=rtpmap:0 PCMU/8000
m=video 49300 RTP/AVP 97 98
a=rtpmap:97 H264/90000
a=rtpmap:98 H264/90000
a=fmtp:97 profile-level-id=42c01f; max-fs=3600; max-mbps=108000
a=fmtp:98 profile-level-id=42c00b; max-fs=240; max-mbps=3600
a=imageattr:97 send [x=1280,y=720] recv [x=1280,y=720]
a=imageattr:98 send [x=320,y=180] recv [x=320,y=180]
a=simulcast send 97;98 recv 97
```

                 Figure 3: Unified Plan Simulcast Offer

The only thing in the SDP that indicates simulcast capability is the
line in the video media description containing the "simulcast"
attribute. The included format parameters indicates that sent
simulcast versions can differ in video resolution and framerate.

The Answer from the server indicates that it too is simulcast
capable. Should it not have been simulcast capable, the
"a=simulcast" 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
m=audio 49672 RTP/AVP 0
a=rtpmap:0 PCMU/8000
m=video 49674 RTP/AVP 97 98
a=rtpmap:97 H264/90000
a=rtpmap:98 H264/90000
a=fmtp:97 profile-level-id=42c01f; max-fs=3600; max-mbps=108000
a=fmtp:98 profile-level-id=42c00b; max-fs=240; max-mbps=3600
a=imageattr:97 send [x=1280,y=720] recv [x=1280,y=720]
a=imageattr:98 send [x=320,y=180] recv [x=320,y=180]
a=simulcast recv 97;98 send 97
```

                 Figure 4: Unified Plan Simulcast Answer

Since the server is the simulcast media receiver, it reverses the
direction of the "simulcast" attribute.

6.3.2. Multi-Source Client

Fred is calling in to the same conference as in the example above
with a two-camera, two-display system, thus capable of handling two
separate media sources in each direction, where each media source is
simulcast-enabled in the send direction. Fred's client is a Unified
Plan client, restricted to a single media source per media
description.

The first two simulcast versions for the first media source use
different codecs, H264-SVC [RFC6190] and H264 [RFC6184]. These two
simulcast versions also have a temporal dependency. Two different
video codecs, VP8 [I-D.ietf-payload-vp8] and H264, are offered as
alternatives for the third simulcast version for the first media
source.

The second media source is offered with three different simulcast
versions. All video streams of this second media source are loss
protected by RTP retransmission [RFC4588].

Fred's client is also using BUNDLE to send all RTP streams from all
media descriptions in the same RTP session on a single media
transport. There are not so many RTP payload types in this example
that there is any risk of running out of payload types, but for the
sake of making an example, it is assumed that one of the payload
types cannot be kept unique across all media descriptions.
Therefore, the SDP makes use of the mechanism (work in progress) in
BUNDLE that identifies which media description an RTP stream belongs
to (a new RTCP SDES item and RTP header extension [RFC5285] type
carrying the a=mid value). That identification will make it possible
to identify unambiguously also on RTP level which media source it is
and thus what the related simulcast versions are, even though two
separate RTP streams in the joint RTP session share RTP payload type.

```
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
a=group:BUNDLE foo bar zen

m=audio 49200 RTP/AVP 99
a=mid:foo
a=rtpmap:99 G722/8000
```

```
   m=video 49600 RTP/AVP 100 101 102 103
   a=mid:bar
   a=rtpmap:100 H264-SVC/90000
   a=rtpmap:101 H264/90000
   a=rtpmap:102 H264/90000
   a=rtpmap:103 VP8/90000
   a=fmtp:100 profile-level-id=42400d; max-fs=3600; max-mbps=108000; \
       mst-mode=NI-TC
   a=fmtp:101 profile-level-id=42c00d; max-fs=3600; max-mbps=54000
   a=fmtp:102 profile-level-id=42c00d; max-fs=900; max-mbps=27000
   a=fmtp:103 max-fs=900; max-fr=30
   a=imageattr:100 send [x=1280,y=720] recv [x=1280,y=720]
   a=imageattr:101 send [x=1280,y=720] recv [x=1280,y=720]
   a=imageattr:102 send [x=640,y=360] recv [x=640,y=360]
   a=imageattr:103 send [x=640,y=360] recv [x=640,y=360]
   a=depend:100 lay bar:101
   a=extmap:1 urn:ietf:params:rtp-hdrext:sdes:mid
   a=simulcast sendrecv 100;101 send 103,102

   m=video 49602 RTP/AVP 96 103 97 104 105 106
   a=mid:zen
   a=rtpmap:96 VP8/90000
   a=fmtp:96 max-fs=3600; max-fr=30
   a=rtpmap:104 rtx/90000
   a=fmtp:104 apt=96;rtx-time=200
   a=rtpmap:103 VP8/90000
   a=fmtp:103 max-fs=900; max-fr=30
   a=rtpmap:105 rtx/90000
   a=fmtp:105 apt=103;rtx-time=200
   a=rtpmap:97 VP8/90000
   a=fmtp:97 max-fs=240; max-fr=15
   a=rtpmap:106 rtx/90000
   a=fmtp:106 apt=97;rtx-time=200
   a=extmap:1 urn:ietf:params:rtp-hdrext:sdes:mid
   a=simulcast send 97;96;103
```

                Figure 5: Fred's Multi-Source Simulcast Offer

   Note: Empty lines in the SDP above are added only for readability
   and would not be present in an actual SDP.

7. Network Aspects

   Simulcast is in this memo defined as the act of sending multiple
   alternative encoded streams 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. The proposed
   solution can be extended to support this functionality with an
   optional mid: prefix before the RTP payload types of a simulcast
   version, to describe simulcast across multiple media descriptions.

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. The proposed solution can be extended to support this
   functionality with an optional mid: prefix before the RTP payload
   types of a simulcast version, to describe simulcast across
   multiple media descriptions.

8. IANA Considerations

   This document requests to register a new attribute, simulcast.

   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 the "a=simulcast" attribute may result in
   simultaneous transmission of multiple RTP streams that would
   otherwise not be generated. The impact is limited by the media
   description joint bandwidth, shared by all simulcast versions
   irrespective of their number. There may however be a large number of
   unwanted RTP streams that will impact the share of the bandwidth
   allocated for the originally wanted RTP stream.

   A hostile removal of the "a=simulcast" attribute will result in
   simulcast not being used.

   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. Robert Hansen, from Cisco, contributed significantly to
   subsequent versions.

11. Acknowledgements

12. References

12.1. Normative References

   [RFC2119] Bradner, S., "Key words for use in RFCs to Indicate
              Requirement Levels", BCP 14, RFC 2119, March 1997.

   [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.

[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.

[RFC7104]   Begen, A., Cai, Y., and H. Ou, "Duplication Grouping
            Semantics in the Session Description Protocol", RFC 7104,
            January 2014.

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-03 (work in progress), October 2014.

[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-06 (work in progress),
            October 2014.

[I-D.ietf-avtcore-rtp-topologies-update]
            Westerlund, M. and S. Wenger, "RTP Topologies", draft-
            ietf-avtcore-rtp-topologies-update-05 (work in progress),
            November 2014.

[I-D.ietf-avtext-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-ietf-avtext-
            rtp-grouping-taxonomy-04 (work in progress), January 2015.

[I-D.ietf-avtext-rtp-stream-pause]
            Akram, A., Even, R., and M. Westerlund, "RTP Stream Pause
            and Resume", draft-ietf-avtext-rtp-stream-pause-05 (work
            in progress), October 2014.

[I-D.ietf-mmusic-sdp-bundle-negotiation]
            Holmberg, C., Alvestrand, H., and C. Jennings,
            "Negotiating Media Multiplexing Using the Session
            Description Protocol (SDP)", draft-ietf-mmusic-sdp-bundle-
            negotiation-15 (work in progress), January 2015.

[I-D.ietf-payload-vp8]
          Westin, P., Lundin, H., Glover, M., Uberti, J., and F.
          Galligan, "RTP Payload Format for VP8 Video", draft-ietf-
          payload-vp8-13 (work in progress), October 2014.

[I-D.roach-mmusic-unified-plan]
          Roach, A., Uberti, J., and M. Thomson, "A Unified Plan for
          Using SDP with Large Numbers of Media Flows", draft-roach-
          mmusic-unified-plan-00 (work in progress), July 2013.

[I-D.westerlund-avtcore-transport-multiplexing]
          Westerlund, M. and C. Perkins, "Multiplexing Multiple RTP
          Sessions onto a Single Lower-Layer Transport", draft-
          westerlund-avtcore-transport-multiplexing-07 (work in
          progress), October 2013.

[RFC2198] Perkins, C., Kouvelas, I., Hodson, O., Hardman, V.,
          Handley, M., Bolot, J., Vega-Garcia, A., and S. Fosse-
          Parisis, "RTP Payload for Redundant Audio Data", RFC 2198,
          September 1997.

[RFC3264] Rosenberg, J. and H. Schulzrinne, "An Offer/Answer Model
          with Session Description Protocol (SDP)", RFC 3264, June
          2002.

[RFC3389] Zopf, R., "Real-time Transport Protocol (RTP) Payload for
          Comfort Noise (CN)", RFC 3389, September 2002.

[RFC4588] Rey, J., Leon, D., Miyazaki, A., Varsa, V., and R.
          Hakenberg, "RTP Retransmission Payload Format", RFC 4588,
          July 2006.

[RFC4733] Schulzrinne, H. and T. Taylor, "RTP Payload for DTMF
          Digits, Telephony Tones, and Telephony Signals", RFC 4733,
          December 2006.

[RFC5117] Westerlund, M. and S. Wenger, "RTP Topologies", RFC 5117,
          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.

   [RFC5583] Schierl, T. and S. Wenger, "Signaling Media Decoding
              Dependency in the Session Description Protocol (SDP)", RFC
              5583, July 2009.

   [RFC6184] Wang, Y., Even, R., Kristensen, T., and R. Jesup, "RTP
              Payload Format for H.264 Video", RFC 6184, May 2011.

   [RFC6190] Wenger, S., Wang, Y., Schierl, T., and A. Eleftheriadis,
              "RTP Payload Format for Scalable Video Coding", RFC 6190,
              May 2011.

   [RFC6236] Johansson, I. and K. Jung, "Negotiation of Generic Image
              Attributes in the Session Description Protocol (SDP)", RFC
              6236, May 2011.

Appendix A. Changes From Earlier Versions

   NOTE TO RFC EDITOR: Please remove this section prior to publication.

A.1. Modifications Between Individual Version -00 and WG Version -00

   o Added this appendix.

Authors' Addresses

   Bo Burman
   Ericsson
   Kistavagen 25
   SE-164 80 Stockholm
   Sweden

   Phone: +46 10 714 13 11
   Email: bo.burman@ericsson.com

   Magnus Westerlund
   Ericsson
   Farogatan 6
   SE-164 80 Stockholm
   Sweden

   Phone: +46 10 714 82 87
   Email: magnus.westerlund@ericsson.com

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

Email: snandaku@cisco.com

Mo Zanaty
Cisco
170 West Tasman Drive
San Jose, CA 95134
USA

Email: mzanaty@cisco.com