Describing multiple RTP media streams in SDP
draft-even-mmusic-multiple-streams-02.txt

Abstract

   This document describes issues when describing multiple RTP streams
   in a single RTP session using SDP and considers the different RTP
   topologies that should be supported. The document looks at current
   solutions and provides paths toward addressing the issues.

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 August 21, 2013.

Copyright Notice

Table of Contents

1. Introduction

   Communication systems can send and receive multiple RTP media
   streams. The streams can be multiple video streams from the same
   source/camera representing, for example, different resolutions
   (simulcast, scalable video (SVC)) or repair streams (FEC). They can
   be different streams from the same endpoint but from different
   cameras, for example a Telepresence system sending two views of the
   room from two different cameras. They can also be multiple streams
   from separate original endpoints, sent by a middlebox.

   RTP [RFC3550] and [I-D.ietf-avtcore-multi-media-rtp-session] allow
   the multiplexing of multiple media of the same and different types
   (video with video and audio with video] in a single RTP session
   identified by a single transport address. The RTP streams are
   identified by their synchronization source identifiers (SSRC).

   SIP offer answer [RFC3264] uses SDP [RFC4566] to negotiate RTP
   [RFC3550] media streams. This document discusses the capabilities
   and limitations of SDP when describing SSRC multiplexed streams.

   When looking at the following offer

   m=video 10000 RTP/AVP 31 32

   a=rtpmap:31 H261/90000

   a=rtpmap:32 MPV/90000

   What does it mean one RTP session is offered with H.261 or MPV codecs
   for the same content, or one RTP session is offered with H.261 and
   MPV codecs each with different content?

   This offer should really mean "arbitrarily many streams, with
   potentially different content, any of which could use either H.261 or
   MPV, potentially switching dynamically between them." Now how do we
   provide enough information in SDP to allow the receiver to get a
   better understanding of what the offer is.

   Reading some text from RFC3264 it may look like a Media stream is
   defined as a single media instance

   "The offer will contain zero or more media streams (each media stream
   is described by an "m=" line and its associated attributes)."

   "In all cases, the formats in the "m=" line MUST be listed in order
   of preference, with the first format listed being preferred. In this
   case, preferred means that the recipient of the offer SHOULD use the

format with the highest preference that is acceptable to it."

"For each "m=" line in the offer, there MUST be a corresponding "m="
line in the answer. The answer MUST contain exactly the same number
of "m=" lines as the offer. This allows for streams to be matched up
based on their order"

Since SDP and [RFC3264] offer/answer describe RTP sessions, SDP's
term "media stream" is poorly chosen. Careful reading reveals that a
single SDP "media stream" can be used by arbitrarily many RTP
streams. (Indeed, historically this was the case in SAP, the first
usage of SDP, which was used to describe loosely-coupled RTP
multicast sessions with arbitrarily many participants.)

The logic of RFC3264 about the preference does not work if you have
multiple RTP streams in the same m-line unless the same preference
applies to all the RTP streams. So when we look at solution we will
also need to clarify the text in RFC3264 and most probably will need
to have the right terminology for RTP session, media session across
the different documents.

SDP [RFC4566] is used to describe the multimedia session. The basic
model uses a two level hierarchy, consisting of session level and
media level.

SDP support of multiplexing multiple media streams in one RTP session
based on the RTP stream SSRC does not provide sufficient capabilities
to allow each of the multiplexed RTP streams identified by SSRC to
have unique attributes, for example different bandwidth.
Furthermore, when an offer has multiple payload type in a single
media level descriptor (m-line), this is identified as option to
receive all this payload types multiplexed.

SDP provides a framework to define grouping relations between SDP
media streams [RFC5888]. This framework specifies the grouping based
on the SDP media session and not on RTP stream.

Some tools for supporting RTP stream level attributes per RTP streams
as well as support for simulcast were proposed and this document will
look at them. It was not a major problem so far since most endpoints
are using a single audio and video stream and are using SDP media
level descriptors (m-lines) to describe each of the streams. Some of
the existing implementation when offering multiple payload types in a
single m-line are doing a second offer/answer exchange offering only
one of the payload types removing the rest in order to indicate that
they can only receive one media type encoding at a time. Every
change of media type requires an offer / answer exchange.

Currently both RTCweb and CLUE WGs have interest in better support
for multiplexing either multiple RTP media streams from the same type
or different types. The work in
[draft-ietf-mmusic-sdp-bundle-negotiation-01] and
[draft-holmberg-mmusic-sdp-mmt-negotiation-00] provides two different
directions for initial bundling support options for SDP negotiation
of multiplexing different media types but the problem of identifying
different RTP streams with different attributes is still not fully
solved. There is a dependency between what will be the bundling
approach and the solution for describing individual RTP streams
attributes.

This document discusses the different RTP topologies and describes
existing tools and see what they provide and how they can be extended
to provide better SDP support for SSRC multiplexed RTP streams while
supporting the different topologies.

2. Terminology

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 RFC2119[RFC2119] and
indicate requirement levels for compliant RTP implementations.

3. RTP topologies for CLUE

The typical RTP topologies used by Telepresence systems specify
different behaviors for RTP and RTCP distribution. A number of RTP
topologies are described in
[I-D.westerlund-avtcore-rtp-topologies-update]. The CLUE WG
direction is to be able to support the relevant topologies including
point-to-point, as well as media mixers, media- switching mixers, and
source-projection mixers.

In the point-to-point topology, one peer communicates directly with a
single peer over unicast. There can be one or more RTP sessions, and
each RTP session can carry multiple RTP streams identified by their
SSRC. All SSRCs will be recognized by the peers based on the
information in the RTCP SDES report that will include the CNAME and
SSRC of the sent RTP streams. In some cases, a video conferencing
system with multiple video sources in a point-to-point may
nonetheless have RTP which is best described by one of the mixer
topologies below. For example, it can produce composed or switched
RTP streams to be used by a receiving system with fewer displays than
the sender has sources.

In the Media Mixer topology, the peers communicate only with the
mixer. The mixer provides mixed or composed media streams, using its
own SSRC for the sent streams. There are two cases here. In the
first case the mixer may have separate RTP sessions with each peer
(similar to the point to point topology) terminating the RTCP
sessions on the mixer; this is known as Topo-RTCP-Terminating MCU in
[I-D.westerlund-avtcore-rtp-topologies-update]. In the second case,
the mixer can use a conference-wide RTP session similar to
[I-D.westerlund-avtcore-rtp-topologies-update] Topo-mixer or Topo-
Video-switching. The major difference is that for the second case,
the mixer uses conference-wide RTP sessions, and distributes the RTCP
reports to all the RTP session participants, enabling them to learn
all the CNAMEs and SSRCs of the participants and know the
contributing source or sources (CSRCs) of the original streams from
the RTP header. In the first case, the Mixer terminates the RTCP and
the participants cannot know all the available sources based on the
RTCP information. The conference roster information including
conference participants, endpoints, media and media-id (SSRC) can be
available using the conference event package [RFC4575] element.

In the Media-Switching Mixer topology, the peer to mixer
communication is unicast with mixer RTCP feedback. It is
conceptually similar to a composing mixer as described in the
previous paragraph, except that rather than composing or mixing
multiple sources, the mixer provides one or more conceptual sources
selecting one source at a time from the original sources. The Mixer
creates a conference-wide RTP session by sharing remote SSRC values
as CSRCs to all conference participants.

In the Source-Projection Mixer (SPM) topology, the peer to mixer
communication is unicast with RTCP mixer feedback. Every potential
sender in the conference has a source which is "projected" by the
mixer into every other session in the conference; thus, every
original source is maintained with an independent RTP identity to
every receiver, maintaining separate decoding state and its original
RTCP SDES information. However, RTCP is terminated at the mixer,
which might also perform reliability, repair, rate adaptation, or
transcoding on the stream. Senders' SSRCs may be renumbered by the
mixer. The sender may turn the projected sources on and off at any
time, depending on which sources it thinks are most relevant for the
receiver; this is the primary reason why this topology must act as an
RTP mixer rather than as a translator, as otherwise these disabled
sources would appear to have enormous packet loss. Source switching
is accomplished through this process of enabling and disabling
projected sources, with the higher-level semantic assignment of
reason for the RTP streams assigned externally.

When looking at SSRC multiplexing we can see that in various

topologies, the SSRC behavior may be different:

1. The SSRCs are static (assigned by the MCU/Mixer), and there is an
    SSRC for each media capture encoding defined in the CLUE
    protocol. Source information may be conveyed using CSRC, or, in
    the case of topo-RTCP-Terminating MCU, is not conveyed.

2. The SSRCs are dynamic, representing the original source and are
    relayed by the Mixer/MCU to the participants.

In the source projecting mixer (SPM) topology, the number of sources
and their SSRCs may change dynamically. An example is a video
conference that starts with 4 participants and the (SPM) forwards the
video RTP streams from 3 of them to all participants. Later 10 more
participants join the conference and the SPM will forward 9 video
sources to each participant. The projected streams keep their
original SSRCs and each participant may get different streams relayed
by the SPM. The SPM creates a separate RTP session with each
participant and will convey the origin of the media using RTCP SDES
information. In this case the number of RTP streams and the sources
they are coming from may change dynamically. This will be a
challenge if we will need to explicitly provide in the SDP all the
sources in the initial offer, and change it whenever a party joins or
leaves. There is also a scaling issue to explicitly list all the
sources for large conferences.

4. Review of current directions in MMUSIC, AVText and AVTcore

This section provides an overview of the RFCs and drafts that tries
to provide more information about RTP streams based on their SSRC and
can be helpful to assign attribute to individual RTP streams that are
multiplexed to a single transport address.

When looking at the available tools based on current work in MMUSIC,
AVTcore and AVText for supporting SSRC multiplexing at the SDP level
the following documents are considered to be relevant.

SDP Source attribute [RFC5576] mechanisms to describe specific
attributes of RTP sources based on their SSRC. This document defines
a mechanism to describe RTP sources, identified by their
synchronization source(SSRC) identifier, in SDP, to associate
attributes with these sources, and to express relationships among
individual sources. It also defines a number of new SDP attributes
that apply to individual sources ("source-level" attributes),
describes how a number of existing media stream ("media-level")
attributes can also be applied at the source level, and establishes
IANA registries for source-level attributes and source grouping

semantics. This mechanism provides an extensible framework but that
implies that there will be a need to specify source level attributes
and probably to change the IANA procedure for attribute registration
adding the requirement to specify if it is also source level
attribute (currently [RFC4566] requires for type of attribute to
specify (session level, media level, or both))

[I-D.westerlund-mmusic-max-ssrc] a signaling solution for how to use
multiple SSRCs within one RTP session. This document also defines
two new SDP attributes, "max-send-ssrc" and" max-recv-ssrc". The
attributes allows an entity to, for a given media description,
indicate sending and receiving capabilities of multiple media
sources, based on codec usage . Since the number of payload type
numbers in an SDP m-line specifies that all these payloads can be
received this draft provides a way to specify how many can be sent
and received simultaneously. Still if there are more payload type
numbers in the m-line it is still implies that a receiver must be
able to receive any subset at any given time but no more than max-
ssrc.

A proposed solution to support simulcast is defined in
[I-D.westerlund-avtcore-rtp-simulcast]. Simulcast is an application
usage where multiple media streams derived from the same media source
may be sent simultaneously. The document discusses the best way of
accomplishing this in RTP using a session-based solution. The
document describes a solution where each stream from the unicast
stream will use a separate RTP session. Section 4.2 of the document
looks at using a single RTP session using RFC5576 [RFC5576] and the
proposed source name attribute specified in
[I-D.westerlund-avtext-rtcp-sdes-srcname]. Another way for a single
seesion support may be by using a different payload type numbers but
section 4.1 of [I-D.westerlund-avtcore-rtp-simulcast] discourages
such usage.

[I-D.westerlund-avtext-rtcp-sdes-srcname] provides an extension that
may be send in SDP, as an RTCP SDES information or as an RTP header
extension that uniquely identifies a single media source. It defines
a hierarchical order of the SRCNAME parameter that can be used, for
example, to describe multiple resolutions from the same source (see
section 5.1 of [I-D.westerlund-avtcore-rtp-simulcast]). Still all
the examples are using RTP session multiplexing and there is no
description of using a single RTP session. This can probably be
addressed using bundle with separate m-line for each resolution.

Other documents that discusses the media source issue and may be
required as part of the solution includes:

[I-D.lennox-mmusic-sdp-source-selection] specifies how participants

in a multimedia session can request a specific source from a remote
party.

[I-D.westerlund-avtext-codec-operation-point] extends the codec
control messages by specifying messages that let participants
communicate a set of codec configuration parameters.

Negotiation of generic image attributes in SDP [RFC6236] provides the
means to negotiate the image size. The image attribute can be used
to offer different image parameters like size but in order to offer
multiple RTP streams with different resolutions it does it using
separate RTP session for each image option.

5. Requirements from a solution

When two or more endpoints with multiple sources communicate with
each other and requires multiplexing multiple media types in the same
RTP session,the following requirements should be addressed:

o It should be possible to group relationships among sources of an
   RTP session

o It should be possible to identify streams among sources of an RTP
   session

o It should be possible to indicate the maximum number of Sources
   and Receivers

o It should be possible to negotiate Codec Configuration parameters

o It should be possible to request the transmission of specific
   sources

o It should be possible to indicate the priority of transmission of
   sources

o It should be possible to indicate support of multiple media type
   multiplexing

o It should be possible to support receipt of multiple RTP sources
   without explicit per-source signaling or negotiation.

o It must be possible for a multimedia session to use multiple
   transport flows for a given media type where it is considered
   valuable (for example, for distributed media, or differential
   quality-of-service).

o It must be possible for a source to be placed into a switched RTP
   session even if the source is a "late joiner", i.e. was added to
   the conference after the receiver requested the switched source.

o It must be possible for a receiver to identify the actual source
   that is currently being mapped to a switched media stream, and
   correlate it with out-of-band information such as rosters.

o If a given source is being sent on the same transport flow (media
   track) for more than one reason (e.g. if it corresponds to more
   than one RTCwen Mediastream at once), it should be possible for a
   sender to send only one copy of the source.

o On the network, media flows should, as much as possible, look and
   behave like currently-defined usages of existing protocols;
   established semantics of existing protocols must not be redefined.

o The solution should seek to minimize the processing burden for
   boxes that distribute media to decoding hardware.

o If multiple sources from a single synchronization context are
   being sent simultaneously, it must be possible for a receiver to
   associate and synchronize them properly.

[6](#). **SDP limitations and proposed solution**

   As the default behavior, Group relationship among sources of an RTP
   session can be indicated by extending the Session Description
   Protocol (SDP) Grouping Framework [RFC5888]. However the Session
   Description Protocol (SDP) Grouping Framework is limited to one media
   description per SFP m-line and does not support multiplexing multiple
   media types in one RTP session. Alternatively, Group relationship
   among sources of an RTP session can be implicitly indicated using
   hierarchical order of the SRCNAME parameter defined in
   [I-D.westerlund-avtext-rtcp-sdes-srcname].

   Normally each RTP stream in the multiplexed RTP streams is identified
   by its SSRC. However in some cases, one media stream may include
   multiple sub-stream sharing the same properties, e.g., scalable
   media. In such cases, the capability to identify each sub-stream
   among sources of an RTP session is required.
   [I-D.westerlund-avtext-codec-operation-point] provides a means for
   sub-stream identification. However such means are too much codec
   specific and used together with codec control messages.

   It is clear that the current SDP does not provide enough tools to
   address all requirements. There are a couple of options to represent

multiple media streams, The major two options are:

o Use multiple m-line each describing a single RTP stream

o Use multiple m-lines each describing one or multiple RTP streams.

6.1. single RTP stream

When using a single RTP stream in each m-line provides an easy way to
describe the streams. This solution requires that all RTP media
streams MUST be declared explicitly in the initial offer or in a
later one before they can be used. As discussed above this solution
does not scale well when doing a multipoint conference using the
Source Projecting Mixer when the conference include multiple
participants and each participant may get a different subset of all
streams. Supporting this use case may require a lot of SIP re-
invites.

6.2. One or multiple RTP streams

For this option each m-line will specify the maximum number of SSRC
that can be sent or received using this m-line. So similar streams
can be added or removed implicitly without requiring more signaling.
The solution will use the maxssrc attribute to specify how many RTP
streams can be sent or received. If there is no maxssrc parameter it
will imply a single RTP stream is specified. This option allows the
SDP to use a single RTP stream per m-line if there is a need to
specify specific attribute that cannot be described in a single
m-line and bundle the m-lines.

7. Acknowledgements

Place Holder

8. IANA Considerations

TBD

9. Security Considerations

TBD.

10. References

10.1. Normative References

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

10.2. Informative References

   [I-D.ietf-avtcore-multi-media-rtp-session]
               Westerlund, M., Perkins, C., and J. Lennox, "Multiple
               Media Types in an RTP Session",
               draft-ietf-avtcore-multi-media-rtp-session-01 (work in
               progress), October 2012.

   [I-D.lennox-mmusic-sdp-source-selection]
               Lennox, J. and H. Schulzrinne, "Mechanisms for Media
               Source Selection in the Session Description Protocol
               (SDP)", draft-lennox-mmusic-sdp-source-selection-04 (work
               in progress), March 2012.

   [I-D.westerlund-avtcore-rtp-simulcast]
               Westerlund, M., Burman, B., Lindqvist, M., and F. Jansson,
               "Using Simulcast in RTP sessions",
               draft-westerlund-avtcore-rtp-simulcast-01 (work in
               progress), July 2012.

   [I-D.westerlund-avtcore-rtp-topologies-update]
               Westerlund, M. and S. Wenger, "RTP Topologies",
               draft-westerlund-avtcore-rtp-topologies-update-01 (work in
               progress), October 2012.

   [I-D.westerlund-avtext-codec-operation-point]
               Westerlund, M., Burman, B., and L. Hamm, "Codec Operation
               Point RTCP Extension",
               draft-westerlund-avtext-codec-operation-point-00 (work in
               progress), March 2012.

   [I-D.westerlund-avtext-rtcp-sdes-srcname]
               Westerlund, M., Burman, B., and P. Sandgren, "RTCP SDES
               Item SRCNAME to Label Individual Sources",
               draft-westerlund-avtext-rtcp-sdes-srcname-01 (work in
               progress), July 2012.

   [I-D.westerlund-mmusic-max-ssrc]
               Holmberg, C., Westerlund, M., Burman, B., and F. Jansson,
               "Multiple Synchronization Sources (SSRC) in SDP Media
               Descriptions", draft-westerlund-mmusic-max-ssrc-00 (work
               in progress), September 2012.

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

   [RFC4575]   Rosenberg, J., Schulzrinne, H., and O. Levin, "A Session
               Initiation Protocol (SIP) Event Package for Conference
               State", RFC 4575, August 2006.

   [RFC4796]   Hautakorpi, J. and G. Camarillo, "The Session Description
               Protocol (SDP) Content Attribute", RFC 4796,
               February 2007.

   [RFC5104]   Wenger, S., Chandra, U., Westerlund, M., and B. Burman,
               "Codec Control Messages in the RTP Audio-Visual Profile
               with Feedback (AVPF)", RFC 5104, February 2008.

   [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.

   [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.

Authors' Addresses

    Roni Even
    Huawei Technologies
    Tel Aviv,
    Israel

    Email: roni.even@mail01.huawei.com

    Jonathan Lennox
    Vidyo, Inc.
    433 Hackensack Avenue
    Seventh Floor
    Hackensack, NJ   07601
    US

    Email: jonathan@vidyo.com

    Qin Wu
    Huawei Technologies

    Email: bill.wu@huawei.com