

AVT
Internet-Draft
Intended status: Informational
Expires: April 30, 2009

T. Schierl
Fraunhofer HHI
J. Lennox
Vidyo
October 27, 2008

Multi-Session and Multi-Source Transmission in the Real-Time Transport
Protocol (RTP)
draft-schierl-avt-rtp-multi-session-transmission-00

Status of this Memo

By submitting this Internet-Draft, each author represents that any applicable patent or other IPR claims of which he or she is aware have been or will be disclosed, and any of which he or she becomes aware will be disclosed, in accordance with [Section 6 of BCP 79](#).

Internet-Drafts are working documents of the Internet Engineering Task Force (IETF), its areas, and its working groups. Note that other groups may also distribute working documents as Internet-Drafts.

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."

The list of current Internet-Drafts can be accessed at <http://www.ietf.org/ietf/lid-abstracts.txt>.

The list of Internet-Draft Shadow Directories can be accessed at <http://www.ietf.org/shadow.html>.

This Internet-Draft will expire on April 30, 2009.

Abstract

In this draft, we discuss problems related to multi-session and multi-source transmission using the Real-Time Transport Protocol (RTP). Most of the input to this draft is taken from email discussion. Multi-session and multi-source transmission is motivated by media data which allows for different transport layer treatment of parts of the media. This is typically the case for layered media. Multi-session transmission is when media data from a single media source is split over multiple RTP sessions. Single-session multi-source transmission (from now on just called "multi-source transmission") is when data from a single media source is sent as

Internet-Draft

RTP Multi-Session Transmission

October 2008

several RTP streams in the same RTP session. The main problems discussed are the mechanisms used for data alignment and source correlation. This draft gives further an overview of payload formats using multi-sessions/multi-source transmission and highlights other transport related issues. The draft concludes with recommendations for the discussed problems.

Table of Contents

1.	Introduction	4
2.	Definitions	4
3.	Terminology	5
4.	Existing Users of Multi-Session and Multi-Source Transmission	5
4.1.	Progressive Video with Hybrid (PVH)	5
4.2.	H.264 Scalable Video Coding (SVC)	6
4.3.	H.264 Multi-View Coding (MVC)	6
4.4.	G.718: Embedded Variable Bit-Rate (EV-VBR) Speech/Audio Codec	6
4.5.	MPEG Surround	7
4.6.	RTP Forward Error Correction	7
4.7.	RTP Retransmission	7
5.	Topology Overview	8
6.	Requirements for multi-session transmission	8
6.1.	Requirements on Data Alignment	8
6.2.	Requirements on Source Correlation	9
7.	Review of techniques for Data Alignment	9
7.1.	NTP Timestamp Alignment using RTCP Sender Report (SR) Packets	9
7.1.1.	Identified problems	10
7.2.	Review of other potential techniques for Data Alignment	12
7.2.1.	RTP Timestamp Alignment	12
7.2.2.	Initial RTP Timestamp or RTP Timestamp Offset Signaling	12
7.2.3.	CCM message - need NTP update	13
7.2.4.	Multiple early RTCP SRs	13
7.2.5.	Codec-Specific Mechanisms	13
7.2.6.	RTP header extension	14
8.	Review of techniques for Source Correlation	14
8.1.	Source Correlation using CNAME in SDES	14
8.2.	Review of other potential techniques for Source Correlation	15

8.2.1.	Single SSRC Space	15
8.2.2.	SSRC Groups	15
8.2.3.	CNAME in Source Attributes	16
8.2.4.	Application-specific Inference of Association	16
9.	Summary of RTP solution for Data Alignment and Source	

	Correlation	16
9.1.	Data Alignment in RTP	16
9.2.	Source Correlation in RTP	16
9.3.	Dependency signaling	17
10.	Recommendations	17
11.	Other transport related issues for multi-session transmission	18
11.1.	Inter-session Jitter	18
11.2.	Inter-session Interleaving	18
12.	Security Considerations	18
13.	IANA Considerations	18
14.	References	18
14.1.	Normative References	18
14.2.	Informative References	19
Appendix A.	Acknowledgements	20
	Authors' Addresses	21
	Intellectual Property and Copyright Statements	22

1. Introduction

Multi-session transmission is when media data from a single media source is split over multiple Real-Time Transport Protocol (RTP) [[RFC3550](#)] sessions. This is usually done because different transport layer treatment is desired for different aspects of the media source, e.g., different multicast groups or different traffic classes. If the traffic is being sent using multicast routing, this is often known as "layered multicast."

Single-session multi-source transmission (from now on just called "multi-source transmission") is when data from a single media source is sent as several RTP streams in the same RTP session. In this case, the streams need to be treated differently by RTP (e.g. with separate RTCP statistics, or selective forwarding by RTP translators) but do not need different transport characteristics. This is often referred to as "SSRC multiplexing", after the synchronization source identifier (SSRC) which distinguishes sources in an RTP session.

Such techniques are often used for "layered" or "embedded" codecs (the former term is typically used for video, the latter for audio). A lower-bitrate, and often lower-complexity, stream (known as the "base"), often backward-compatible with older codecs, provides basic media quality, while one or more additional streams (known as "enhancements") provide richer media or otherwise provide an enhanced user experience. Various layered and embedded codecs are discussed in [Section 4](#).

Multi-session and multi-source transmission are also used for stream

robustness. Both RTP Forward Error Correction [[RFC5109](#)] and RTP Retransmission [[RFC4588](#)] use multi-session transmission, and the latter can optionally use multi-source transmission as well.

For both multi-session and multi-source transmission, two issues arise: how streams are correlated, i.e. how receivers determine which base and enhancement streams carry data for the same media source; and how streams are aligned, i.e. how receivers determine which packets of the base stream are associated with which packets of the enhancement stream.

[2.](#) Definitions

multi-session transmission: In multi-session transmission, media data from a single media source is split over multiple RTP sessions. The term "layered multicast" is equivalent to multi-session transmission for sessions using multicast addresses.

multi-source transmission: In multi-source transmission, data from a single media source is sent as several RTP streams in the same RTP session. The sources contained in an RTP session are identified by their synchronization source identifiers (SSRCs) or, if combined by a RTP mixer, by their contributing source identifiers (CSRCs), as defined in RTP [[RFC3550](#)].

associated multimedia streams: Associated multimedia streams are independent media sources from the same session participant, e.g. audio and video sources, or multiple cameras from a single participant. Each source can have an independent media clock, reflecting the device that captured the media. For live media, these clocks will often drift relative to each other, over and above their often inherently-different clock rates. In RTP, each stream has separate initial RTP timestamps and sequence numbers. Related sources are associated using the RTCP Canonical Name (CNAME) Source Description (SDES) field. A common time base may be computed using NTP timestamps, based on information carried in RTCP Sender Report (SR) packets. The sources are typically synchronized ("lip-synced") by receivers when rendered, based on the computed NTP timestamps.

Data Alignment: Assembling data of the same media frame which is transferred in different sessions or as different sources in the

same session as part of a layered media. The assembly of the media frame must be achieved before decoding, otherwise the decoding process typically fails or may be only possible at a reduced quality.

Source Correlation: The logical association of RTP streams transferred as multiple separate sessions or as multiple sources in the same session to one layered media.

[3.](#) 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 [RFC 2119](#) [[RFC2119](#)].

[4.](#) Existing Users of Multi-Session and Multi-Source Transmission

[4.1.](#) Progressive Video with Hybrid (PVH)

Progressive Video with Hybrid transform (PVH) [[McCa96](#)] was used in the initial demonstration of multi-session transmission. PVH was the initial driver for adding text on layered multicast to the Real-Time Transport Protocol (RTP) [[RFC3550](#)]. Data Alignment was done using packets' RTP timestamps.

[4.2.](#) H.264 Scalable Video Coding (SVC)

H.264 Scalable Video Coding (SVC) [[I-D.ietf-avt-rtp-svc](#)] extends the H.264 [[RFC3984](#)] video standard to provide spatial, temporal, and quality (signal-to-noise) enhancements. The base layer of SVC is backward-compatible with existing H.264 decoders. A base layer sent separately using the H.264 [[RFC3984](#)] payload format can be received and processed by existing devices. The Payload Format for SVC uses the multi-session transmission approach. Currently two basic modes are defined in the SVC Payload Format for decoding order recovery of media data received from multiple sessions:

Data Alignment based on NTP timestamps: This method is used in the NI-T and NI-TC mode defined in [[I-D.ietf-avt-rtp-svc](#)]. These modes currently rely on exact NTP timestamp alignment in order to recover the decoding order.

Cross-Session Decoding Order Number (CS-DON): This method is used in the NI-C, NI-TC and I-C modes defined in [\[I-D.ietf-avt-rtp-svc\]](#). These modes rely on a number (CS-DON) which is associated to packets indicating the decoding order across sessions.

[4.3.](#) H.264 Multi-View Coding (MVC)

H.264 Multi View Coding (MVC) [\[I-D.wang-avt-rtp-mvc\]](#) extends the H.264 [\[RFC3984\]](#) video standard to provide multiple views of a video stream, for multi view and 3D applications. MVC is similarly to SVC an extension of H.264 and has a backward compatible base view, which can be also decoded by existing H.264 receivers. Thus it is possible to provide the base view of a multi sessions transmission in a compatible way using the H.264 [\[RFC3984\]](#) as Payload Format. Since the new coding approach is mainly based on exploiting temporal references to other frames of the same view or different views, there is not always the need to receive the base view in order to decode a desired view. The payload format will rely on the same approaches as defined in the RTP Payload Format for SVC video [\[I-D.ietf-avt-rtp-svc\]](#) for decoding order recovery when receiving data from multiple sessions.

[4.4.](#) G.718: Embedded Variable Bit-Rate (EV-VBR) Speech/Audio Codec

G.718, the Embedded Variable Bit-Rate (EV-VBR) speech/audio codec [\[I-D.lakaniemi-avt-rtp-evbr\]](#) provides an embedded speech-rate encoder. This codec also allows for multi-session transmission. The current draft mandates RTP SR for Data Alignment in multi-session transmission.

[4.5.](#) MPEG Surround

MPEG Surround (Spatial Audio Coding, SAC) [\[I-D.ietf-avt-rtp-mps\]](#) enhances MPEG two-channel audio with multi-channel surround sound while maintaining backward compatibility with two-channel receivers. The payload relies on NTP timestamp alignment for multi-session transmission. The audio codec typically has different sampling rates for base and enhancements.

[4.6.](#) RTP Forward Error Correction

RTP Generic Forward Error Correction [[RFC5109](#)] allows a supplemental stream to provide additional data for recovery from packet loss using a separate session for transmitting the FEC stream. The repair stream is typically sent as a separate RTP session. A special case is when the FEC stream is being sent as a secondary codec in the redundant encoding format. In this case the FEC stream is sent as a separate source in the same session as the redundant codec. Data Alignment is achieved using sequence numbers of the FEC protected packets.

FEC Grouping Issues in Session Description Protocol

[[I-D.begen-mmusic-fec-grouping-issues](#)] describes a grouping framework for FEC and media streams based on the Grouping of Media Lines in the Session Description Protocol (SDP) [[RFC3388](#)] framework. The framework relies on transmitting the FEC streams in separate sessions. Data Alignment is achieved by the FEC Framework and relies on the used FEC scheme, i.e. there is a specific solution for associating data of the protected and the protecting packet stream.

[4.7.](#) RTP Retransmission

RTP Retransmission [[RFC4588](#)] allows senders to retransmit RTP packets indicated by the receiver as lost. The re-sent packets are transported in a separate stream and may be transmitted within a separate RTP session or may be transmitted as a separate source in the same session as the media stream.

If multi-source (i.e., single-session) transmission is being used, retransmitted packets are sent with a different SSRC. Source association in this case done by sources' CNAMEs, with the further requirement that a receiver MUST NOT have two outstanding requests for the same packet sequence number in two different original streams before the association is resolved.

[5.](#) Topology Overview

A number of different RTP Topologies [[RFC5117](#)] are relevant for consideration for multi-source and multi-session transmission.

[Ed. TBD: more text on the relation between the approaches presented in the memo and the mentioned topologies.]

- o Point-to-point - Two endpoints communicating using unicast.
- o Point-to-multipoint via multicast - Using a multicast transport mechanisms to send packets of one participant to all the other participants in the multicast group.
- o Point-to-multipoint via RTP translator - Using [[RFC3550](#)] translators to send packets of one participant to other participants of a group. Packets of one or more participants may be forwarded to the group.
- o Point-to-multipoint via RTP mixer - Using [[RFC3550](#)] mixers to send packets of one participant to other participants of a group. Packets of one or more participants may be forwarded to the group.
- o Point-to-multipoint via Video Switching MCUs - Allows for sending packets from one participant to the other participants in a group. But typically only one participant's video data is forwarded at a time to the other participants.
- o Point-to-multipoint via RTCP-terminating MCUs - Each participant is running a point-to-point session with the MCU. Typically, only one participant's video data is forwarded at a time to the other participants.
- o Point-to-multipoint without a feedback channel - These channels typically provide IP multicast over a broadcast transmission medium, which naturally do not provide a bi-directional channel. This is the case, e.g. for DVB channels using IP over MPE over MPEG-2 Transport Stream as for DVB-H or the emerging DVB-SH.

[6.](#) Requirements for multi-session transmission

[6.1.](#) Requirements on Data Alignment

Synchronization of media streams received from multiple sessions is typically used for lip-synchronization of audio and video data. For this case, RTP provides a strong tool, which is the presence of (RTP) timestamps for each media frame, generated from individual clocks for each session. Additionally, RTCP Sender Report packets are sent periodically in each session containing (NTP) timestamps from a wallclock common across all of the sessions, plus a reference to the corresponding (RTP) timestamp that would be generated for a media frame with the signaled wallclock time. The interval between transmission of RTCP SRs is typically in the range of multiple

seconds. For a more detailed review of RTP synchronization techniques, see [Section 7.1](#).

For the reception of layered media, either on multiple sessions or as multiple sources, it is absolutely essential to allow for immediate Data Alignment. That is, the Data Alignment must be applied before the decoding process of the layered media. If Data Alignment is not applied before decoding, the decoder may not be able to decode the media at all, or may only be able to produce a media representation at reduced quality.

[6.2](#). Requirements on Source Correlation

For the reception of layered media, whether on multiple sessions or as multiple sources, it is absolutely essential to find out prior to decoding which sessions and sources are correlated. That is, the receiver needs to know, prior to Data Alignment and decoding, the inter-session and the inter-source dependency. Notably, for cases in which multiple independent media sources are transmitted as layered media in the same session or set of sessions, miscorrelation of sources could lead to a decoder attempting to use one source's base layer with another source's enhancement layer.

[7](#). Review of techniques for Data Alignment

[7.1](#). NTP Timestamp Alignment using RTCP Sender Report (SR) Packets

The inter-media synchronization mechanism defined in [\[RFC3550\]](#) uses RTP timestamps in the RTP packets and a combination of RTP timestamp and NTP wallclock carried in the RTCP Sender Report (SR) packets. The RTCP SR packet contains a RTP timestamp in the media timescale and as reference to an absolute wallclock time the NTP timestamp. The definitions for timestamp generation and synchronization in [section 5.1](#) and 6.4.1 of [\[RFC3550\]](#) are summarized in the following list:

- o The timestamp reflects the sampling instant of the first octet in the RTP data packet.
- o The sampling instant MUST be derived from a clock that increments monotonically and linearly in time to allow synchronization and jitter calculations (see [Section 6.4.1](#)).
- o The resolution of the clock MUST be sufficient for the desired synchronization accuracy and for measuring packet arrival jitter (one tick per video frame is typically not sufficient).
- o If RTP packets are generated periodically, the nominal sampling

instant as determined from the sampling clock is to be used, not a reading of the system clock.

- o RTP timestamps from different media streams may advance at different rates and usually have independent, random offsets. Therefore, although these timestamps are sufficient to reconstruct the timing of a single stream, directly comparing RTP timestamps from different media is not effective for synchronization. Instead, for each medium the RTP timestamp is related to the sampling instant by pairing it with a timestamp from a reference clock (wallclock) that represents the time when the data corresponding to the RTP timestamp was sampled..
- o Receivers should expect that the measurement accuracy of the timestamp may be limited to far less than the resolution of the NTP timestamp.
- o On a system that has no notion of wallclock time but does have some system-specific clock such as "system uptime", a sender MAY use that clock as a reference to calculate relative NTP timestamps.
- o It is important to choose a commonly used clock so that if separate implementations are used to produce the individual streams of a multimedia session, all implementations will use the same clock.
- o [Ed. : The RTP timestamp in the SR] corresponds to the same time as the NTP timestamp (above), but in the same units and with the same random offset as the RTP timestamps in data packets.
- o This correspondence may be used for intra- and inter-media synchronization for sources whose NTP timestamps are synchronized, and may be used by media-independent receivers to estimate the nominal RTP clock frequency.
- o Rather, it MUST be calculated from the corresponding NTP timestamp using the relationship between the RTP timestamp counter and real time as maintained by periodically checking the wallclock time at a sampling instant.

To summarize, the definitions in [[RFC3550](#)]: the RTCP SR is used for deriving the media timestamp using the RTP timestamp and the NTP wallclock. If this synchronization mechanism is correctly implemented and there is no clock jitter in neither the media clock nor in the clock thus it can be always guaranteed, that a RTP timestamp and its NTP wallclock timestamp are perfectly aligned, the RTP approach should work fine for Data Alignment. [Ed. : need more

text for summary / review of text above]

[7.1.1.](#) Identified problems

[7.1.1.1.](#) Synchronization Delay

Since [[RFC3550](#)] mandates RTCP SRs to be sent in intervals of multiple seconds, Data Alignment based on this information may introduce a delay to this process, which may lead to delayed tune-in for the

Schierl & Lennox

Expires April 30, 2009

[Page 10]

Internet-Draft

RTP Multi-Session Transmission

October 2008

decoding process. This is typically not the case for decoding media transferred in exactly one session and source, since synchronization is not required for decoding, but only for playout. A delay for playout or lip synchronization does not usually pose a fundamental problem.

[7.1.1.2.](#) Losing synchronization information

The loss of RTCP SR packets may introduce additional delay to the Data Alignment process, thus a more robust mechanism would be desirable.

[7.1.1.3.](#) Clock Skew

Clock skew between the NTP/system clock and the media clock will affect the NTP media timestamp generation derived from RTCP SRs and RTP timestamps. That typically results in different NTP timestamps for packets of the same media frame transmitted in the different sessions or transferred as different sources, and leads to misalignment for the Data Alignment. As far as we know, there is no way to always guarantee the presence of perfect clocks for media and NTP/system clock. From the standardization point of view this may seem to be an implementation issue. However, if this implementation issue puts a burden on the senders like the presence of a perfect clocks for generating timestamps, this issue needs to be solved in an easy and general way.

Following the RTP philosophy, clock skew can be estimated by observing several RTCP SRs. The receiver may use the observation to compensate for the clock skew. However, this is only possible if there is no requirement for immediate synchronization of the sort which is essential for Data Alignment of layered codecs.

The case of clock skew between in media and NTP/system clocks may be overcome by using the same clock instance, e.g. the system clock, for RTP as well as NTP timestamp generation. However, this is not compliant with RTP, since [[RFC3550](#)] mandates the use of a media clock which is different from the system clock (see definitions in RTP as cited above in [Section 7.1](#)). Indeed, for many codecs, notably audio, correct decoding requires that the timestamp difference between subsequent frames exactly correspond to the amount of data sent in each frame.

[7.1.1.4](#). Accuracy of clocks

Assuming that we have clocks without skew, there is still the question of accuracy of the clock used for generating the timestamps. Notably, the Windows system clock is only updated on each system

clock tick, typically every 10 or 15 milliseconds on Windows XP and Vista. RTP says that a receiver should not make any assumption on this, but an implementation which may have to cope with rounding done in the low-order microsecond cannot simply compare two NTP timestamps for being identical. An application may have to compare "ranges" of timestamps in order to get rid of rounding problems. However, in some cases the ranges of NTP timestamps required may indeed be greater than the time interval between consecutive media frames.

[7.1.1.5](#). Existing RTCP SR implementations

As far as we know, existing RTCP SR implementations show a wide range of alignment problems for generating exact NTP media timestamps for Data Alignment. NTP alignment issues can be modeled for existing RTCP senders by capturing an NTP and RTP timestamps in consecutive SR packets, projecting the NTP timestamp in one SR packet based on the RTP timestamp in that SR packet, the NTP and RTP timestamps in the previous SR packet, and the codec's nominal clock rate. Initial experiments have shown NTP timestamp alignment problems on the order of 40-50 milliseconds for several implementations.

[7.2](#). Review of other potential techniques for Data Alignment

[7.2.1](#). RTP Timestamp Alignment

The idea here is to signal the same RTP timestamp for packets containing data of the same media time instance in the different sessions. That is the same clock would have to be used for the multiple sessions and the same RTP random offset would have to be used. This method is backward compatible with using NTP timestamps for inter-media synchronization as well as for jitter calculation. Furthermore, this is the only alternative used up to our knowledge (see [Section 4.1](#)) for layered transmission of media.

[7.2.1.1](#). Identified problems

Using the same RTP timestamp random offset may lead to getting weak initialization vectors for the encryption method defined in [\[RFC3550\]](#) if keys are shared across the sessions or streams. Additionally, that it may be unnatural for some codecs to use the same clockrate for the multiple sessions, for example an audio wideband enhancement layer enhancing a narrow-band base layer.

[7.2.2](#). Initial RTP Timestamp or RTP Timestamp Offset Signaling

Signaling the initial RTP timestamp or the initial offsets as an media or source level attribute in SDP associated with each stream. This could be done, e.g., using

[\[I-D.ietf-mmusic-sdp-source-attributes\]](#).

[7.2.2.1](#). Identified problems

This may have an implication for implementations, since one needs to know packet stream related information as initial RTP timestamp, or offset between RTP timestamps during while offering a session. This may be a problem for sessions where multiple senders are present: it may not always be possible for an SDP creator to include all initial offsets / timestamps for all participants for sessions with multiple sending parties.

[7.2.3](#). CCM message - need NTP update

In this case, a receiver would request for immediate synchronization information. This method may reduce the initial delay, but just work for topologies with bi-directional channels.

[7.2.3.1.](#) Identified problems

This method is only feasible for topologies with bidirectional and reasonably rapid communication channels, i.e. unicast or small-group multicast. This method also assumes that the NTP timestamp alignment always works.

[7.2.4.](#) Multiple early RTCP SRs

In this case, the sender would generate more RTCP SRs than typically required and send them at an early point in the session. This method does also work for topologies with uni-directional communication channels.

[7.2.4.1.](#) Identified problems

This method may overflow the RTCP bandwidth. Enhancing the RTCP sender bandwidth may be achieved using SDP bandwidth parameters. This method may require an adjustment of the RTCP bandwidth of the session depending on the number of participants and senders. Further, this approach does not solve the problem for receivers tuning in to the session after it begins ("random entry"). This method also assumes that the NTP timestamp alignment always works.

[7.2.5.](#) Codec-Specific Mechanisms

This mechanism exploits signaling contained within the payload's data sections in order to allow the Data Alignment. Example is the Cross Session Decoding Order Number (CS-DON) as defined in [\[I-D.ietf-avt-rtp-svc\]](#) or as proposed in

[\[I-D.hannuksela-avt-rtp-svc\]](#), where a timestamp or a timestamp delta of the RTP packet to be aligned is carried by payload specific means.

[7.2.5.1.](#) Identified problems

A payload independent solution for the basic functionality of Data Alignment is desirable.

[7.2.6.](#) RTP header extension

The RTP header extension may be used to add generic signaling about

Data Alignment to RTP packets.

[7.2.6.1.](#) Identified problems

RTP header extensions are required to be ancillary information which can safely be discarded by receivers which do not understand them. Data alignment mechanisms do not satisfy this requirement.

[8.](#) Review of techniques for Source Correlation

[8.1.](#) Source Correlation using CNAME in SDP

In RTP, associated multimedia streams (e.g., audio and video sources from a single participant) have different SSRCs, and are associated using SDP CNAME fields. While in principle the same technique can be used to associate streams for multi-session or multi-source transmission, several issues arise.

Startup latency: while slow lipsync convergence of multimedia streams is often tolerable, layered sources have to be associated from the start in order to be decodable, particularly for codec types such as video with inter-frame decoding dependencies.

If multiple sources are sent from the same participant on the same session or family of sessions, e.g. multiple video cameras, they will have the same CNAME, because they are synchronized with each other and with any other sources for the session. This makes it impossible to definitively associate base and enhancement sources, as there may be more than one of each with the same CNAME. This potential for confusion is the reason for RTP retransmission's restriction on multiple outstanding RTP NACKs before stream association has completed, as described in [Section 4.7](#).

[8.2.](#) Review of other potential techniques for Source Correlation

[8.2.1.](#) Single SSRC Space

Motivated by the problems with CNAME association, RTP [[RFC3550](#)] specifies instead a single SSRC space for layered multicast (multiple-session transmission). Furthermore, as described in [Section 9.2](#), it specifies that SSRC collision detection is performed only in the base layer.

Applying SSRC collision detection in just the base layer in case of using multi-session transmission seems to work for current codec implementations.

By definition one of the multiple views possible in MVC media [Section 4.3](#) is the base view and this view is backward compatible to H.264. Decoding a view other than the base view may not require the presence of the base view. Although MVC is by its nature a layered codec, it may not always be reasonable to require the reception of the base layer for collision detection, even when it is not required for decoding.

Currently, we do not see major relevance for the MVC codec format, due to its lack in coding efficiency, thus we tend not to take MVC as the killer application for new Source Correlation functionalities. This means without taking MVC into account, the current solution of using the base layer for SSRC collision detection seems to be still appropriate.

If needed, collision detection could instead be performed across all, or a subset of, the sessions used for multi-session transmission. However, it is not entirely clear how this would work for senders or receivers that are only participating in a subset of the sessions, and this would require further study.

[8.2.2](#). SSRC Groups

The Internet-Draft [[I-D.ietf-mmusic-sdp-source-attributes](#)] specifies a mechanism by which related sources can be described as grouped in SDP. For multi-source (single-session) transmission, this can provide an alternative way to provide source association.

Clearly, this will only be effective in topologies and signaling architectures in which the SDP author can know about every source in the session that will be used for multi-source transmission, and the SDP can be updated on the addition of new sources or SSRCs collisions.

[8.2.3.](#) CNAME in Source Attributes

The draft [[I-D.ietf-mmusic-sdp-source-attributes](#)] also provides a mechanism for sources' SSRCs to be associated to their CNAMEs in SDP. This can eliminate the startup latency of stream association for the mechanism described in [Section 8.1](#), though it does not solve the problem of multiple sources for a session. It also has the same architectural limitations as [Section 8.2.2](#) in terms of using SDP.

[8.2.4.](#) Application-specific Inference of Association

As described in [Section 4.7](#), it is in some cases possible to use mechanisms specific to a particular codec or mechanism to determine stream associations. For retransmission, for instance, a NACK of a packet with sequence N with SSRC A, followed by a retransmission of a packet with sequence N on SSRC B, indicates that SSRC B is the retransmission stream for SSRC A. Such techniques are mechanism-specific and cannot easily be generalized.

[9.](#) Summary of RTP solution for Data Alignment and Source Correlation

[9.1.](#) Data Alignment in RTP

The text on layered multicast in [[RFC3550](#)] does not discuss Data Alignment among the media data carried in the different RTP sessions. We assume that the intention of the RTP specification was to use NTP timestamp alignment. However, Vic, the demonstration code for layered multicast using PVH, used RTP timestamp alignment for this purpose.

[9.2.](#) Source Correlation in RTP

The text in [section 8.3 of \[RFC3550\]](#) mandates a single SSRC to be used for multiple sessions containing data of the same layered media source. Further, the text mandates the detection of SSRC collisions using the CNAME item in SDES packets carried in the base layer:

For layered encodings transmitted on separate RTP sessions (see [Section 2.4](#)), a single SSRC identifier space SHOULD be used across the sessions of all layers and the core (base) layer SHOULD be used for SSRC identifier allocation and collision resolution. When a source discovers that it has collided, it transmits an RTCP BYE packet on only the base layer but changes the SSRC identifier to the new value in all layers. ...

[9.3.](#) Dependency signaling

For signaling the dependency of data transmitted using layered multicast, SDP [[RFC4566](#)] contains rudimentary support, in that it allows for signaling a range of transport addresses in a certain media description. By definition, a higher transport address identifies a higher layer in the one-dimensional hierarchy. A receiver needs only to decode data conveyed over this transport address and lower transport addresses to decode this Operation Point.

When the media data of one source is transmitted in multiple RTP sessions, the mechanism defined in Signaling media decoding dependency in Session Description Protocol (SDP)

[[I-D.ietf-mmusic-decoding-dependency](#)] can also be used to indicate the relationship between the multiple sessions of the same media type. Currently, this mechanism is inherited by the new Payload Formats allowing multi-session transmission: [[I-D.ietf-avt-rtp-svc](#)], [[I-D.wang-avt-rtp-mvc](#)], [[I-D.ietf-avt-rtp-mps](#)], and [[I-D.lakaniemi-avt-rtp-evbr](#)]. By definition the base layer is signaled as the RTP session which does not depend on any other session.

Since [[RFC3550](#)] mandates the correlation of one layered media with the same source, there is no mechanism to indicate dependencies of multiple sources.

[10.](#) Recommendations

We recommend for Data Alignment of media data from the same source, that the same RTP timestamp is used for packets of the same time instance as defined in

[[I-D.lennox-avt-rtp-layered-encoding-timestamps](#)]. This method comes for free and can be implemented in a backward compatible way, since NTP timing for synchronizing different types of media is not affected. This further requires the use of the same timescale of the sessions of an multi-session or multi-source transmission, which is anyway the case if the layered media is identified as a unique source. Mandating the same timescale for each of the sessions in a multi-session transmission may need to be discussed with respect to

the audio codec described in [Section 4.5](#).

For Source Correlation, we suggest to keep the mechanism defined in [\[RFC3550\]](#), i.e. all layers of a layered media source have the same SSRC and the base layer is used for SSRC collision detection. Further, it may be useful to have a signaling mechanism, which indicates the RTP session to be used for SSRC collision detection.

[11](#). Other transport related issues for multi-session transmission

[11.1](#). Inter-session Jitter

The transport of media of the same source in different sessions may introduce different jitter behaviors in the different sessions. We call this issue inter-session jitter. Inter-session jitter may be caused by sessions taking different network paths or by any other packet reordering within the network outside the control of the user. RTP implementations typically use buffers for de-jittering each of the sessions separately. In a simple A/V transmission scenario, de-jittering the audio and the video input queue separately is not problematic, since the synchronization is achieved after the decoder during playout. Using multi-session transmission, de-jittering and synchronization (Data Alignment) is required before decoding instead of synchronizing the data after decoding at playout time. And the Data Alignment via NTP timestamp must be 100% exact on a micro second base, otherwise the synchronization fails. This is definitely different from doing synchronization for lip synchronized playout of audio and video.

[11.2](#). Inter-session Interleaving

Using multi-session transmission allows for data interleaving, while the data transmitted within one session can still be sent in decoding order. Inter-session interleaving may be also realizable using Data Alignment via timestamps.

[12](#). Security Considerations

[Ed. TBD]

[13.](#) IANA Considerations

No action by IANA is required.

[14.](#) References

[14.1.](#) Normative References

- [RFC3550] Schulzrinne, H., Casner, S., Frederick, R., and V. Jacobson, "RTP: A Transport Protocol for Real-Time Applications", STD 64, [RFC 3550](#), July 2003.

Schierl & Lennox

Expires April 30, 2009

[Page 18]

Internet-Draft

RTP Multi-Session Transmission

October 2008

[14.2.](#) Informative References

- [I-D.begen-mmusic-fec-grouping-issues]
Begen, A., "FEC Grouping Issues in Session Description Protocol", [draft-begen-mmusic-fec-grouping-issues-00](#) (work in progress), February 2008.
- [I-D.hannuksela-avt-rtp-svc]
Hannuksela, M. and Y. Wang, "Session Multiplexing for SVC Video", [draft-hannuksela-avt-rtp-svc-01](#) (work in progress), July 2008.
- [I-D.ietf-avt-rtp-mps]
Bont, F., Doehla, S., Schmidt, M., and R. Sperschneider, "RTP Payload Format for Elementary Streams with MPEG Surround multi-channel audio", [draft-ietf-avt-rtp-mps-01](#) (work in progress), October 2008.
- [I-D.ietf-avt-rtp-svc]
Wenger, S., Wang, Y., Schierl, T., and A. Eleftheriadis, "RTP Payload Format for SVC Video", [draft-ietf-avt-rtp-svc-14](#) (work in progress), September 2008.
- [I-D.ietf-mmusic-decoding-dependency]
Schierl, T. and S. Wenger, "Signaling media decoding

dependency in Session Description Protocol (SDP)", [draft-ietf-mmusic-decoding-dependency-04](#) (work in progress), October 2008.

[I-D.ietf-mmusic-sdp-source-attributes]

Lennox, J., Ott, J., and T. Schierl, "Source-Specific Media Attributes in the Session Description Protocol (SDP)", [draft-ietf-mmusic-sdp-source-attributes-01](#) (work in progress), February 2008.

[I-D.lakaniemi-avt-rtp-evbr]

Lakaniemi, A. and Y. Wang, "RTP payload format for G.718 speech/audio", [draft-lakaniemi-avt-rtp-evbr-04](#) (work in progress), October 2008.

[I-D.lennox-avt-rtp-layered-encoding-timestamps]

Lennox, J., Schierl, T., and S. Ganesan, "Real-Time Transport Protocol (RTP) Timestamps for Layered Encodings", [draft-lennox-avt-rtp-layered-encoding-timestamps-00](#) (work in progress), June 2008.

[I-D.wang-avt-rtp-mvc]

Wang, Y. and T. Schierl, "RTP Payload Format for MVC Video", [draft-wang-avt-rtp-mvc-02](#) (work in progress), August 2008.

[McCa96] McCanne, S., "Scalable Compression and Transmission of Internet Multicast Video", Report No. UCB/CSD-96-928, December 1996.

Ph.D. Dissertation, University of California Berkeley.

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

[RFC3388] Camarillo, G., Eriksson, G., Holler, J., and H. Schulzrinne, "Grouping of Media Lines in the Session Description Protocol (SDP)", [RFC 3388](#), December 2002.

[RFC3984] Wenger, S., Hannuksela, M., Stockhammer, T., Westerlund,

M., and D. Singer, "RTP Payload Format for H.264 Video", [RFC 3984](#), February 2005.

- [RFC4566] Handley, M., Jacobson, V., and C. Perkins, "SDP: Session Description Protocol", [RFC 4566](#), July 2006.
- [RFC4588] Rey, J., Leon, D., Miyazaki, A., Varsa, V., and R. Hakenberg, "RTP Retransmission Payload Format", [RFC 4588](#), July 2006.
- [RFC5109] Li, A., "RTP Payload Format for Generic Forward Error Correction", [RFC 5109](#), December 2007.
- [RFC5117] Westerlund, M. and S. Wenger, "RTP Topologies", [RFC 5117](#), January 2008.

[Appendix A](#). Acknowledgements

Funding for the RFC Editor function is provided by the IETF Administrative Support Activity (IASA). Further, the author Thomas Schierl of Fraunhofer HHI is sponsored by the European Commission under the contract number FP7-ICT-214063, project SEA. The authors want to thank Colin Perkins, Ye-Kui Wang, Randell Jesup, Ingemar Johansson, Gerard Babonneau, Alex Eleftheriadis, Stefan Doehla, and Roni Even for their valuable comments on the mailing list.

Authors' Addresses

Thomas Schierl
Fraunhofer HHI
Einsteinufer 37
D-10587 Berlin
Germany

Phone: +49-30-31002-227

Email: mail@thomas-schierl.de

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

Email: jonathan@vidyo.com

contained in [BCP 78](#), and except as set forth therein, the authors retain all their rights.

This document and the information contained herein are provided on an "AS IS" basis and THE CONTRIBUTOR, THE ORGANIZATION HE/SHE REPRESENTS OR IS SPONSORED BY (IF ANY), THE INTERNET SOCIETY, THE IETF TRUST AND THE INTERNET ENGINEERING TASK FORCE DISCLAIM ALL WARRANTIES, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO ANY WARRANTY THAT THE USE OF THE INFORMATION HEREIN WILL NOT INFRINGE ANY RIGHTS OR ANY IMPLIED WARRANTIES OF MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE.

Intellectual Property

The IETF takes no position regarding the validity or scope of any Intellectual Property Rights or other rights that might be claimed to pertain to the implementation or use of the technology described in this document or the extent to which any license under such rights might or might not be available; nor does it represent that it has made any independent effort to identify any such rights. Information on the procedures with respect to rights in RFC documents can be found in [BCP 78](#) and [BCP 79](#).

Copies of IPR disclosures made to the IETF Secretariat and any assurances of licenses to be made available, or the result of an attempt made to obtain a general license or permission for the use of such proprietary rights by implementers or users of this specification can be obtained from the IETF on-line IPR repository at <http://www.ietf.org/ipr>.

The IETF invites any interested party to bring to its attention any copyrights, patents or patent applications, or other proprietary rights that may cover technology that may be required to implement this standard. Please address the information to the IETF at ietf-ipr@ietf.org.