R. van Brandenburg
H. Stokking
O. van Deventer
TNO
F. Boronat
M. Montagud
Universitat Politecnica de
Valencia
K. Gross
AVA Networks
January 17, 2013

Inter-destination Media Synchronization using the RTP Control Protocol (RTCP)
draft-ietf-avtcore-idms-08

Abstract

This document defines a new RTP Control Protocol (RTCP) Packet Type and RTCP Extended Report (XR) Block Type to be used for achieving Inter-Destination Media Synchronization (IDMS). IDMS is the process of synchronizing playout across multiple geographically distributed media receivers. Using the RTCP XR IDMS Reporting Block defined in this document, media playout information from participants in a synchronization group can be collected. Based on the collected information, an RTCP IDMS Settings Packet can then be send to distribute a common target playout point to which all the distributed receivers, sharing a media experience, can synchronize.

Typical use cases in which IDMS is usefull are social TV, shared service control (i.e. applications where two or more geographically separated users are watching a media stream together), distance learning, networked video walls, networked loudspeakers, etc.

Status of this Memo

material or to cite them other than as "work in progress."

This Internet-Draft will expire on July 21, 2013.

Table of Contents

1. Introduction

Inter-Destination Media Synchronization (IDMS) refers to the playout of media streams at two or more geographically distributed locations in a time synchronized manner. It can be applied to both unicast and multicast media streams and can be applied to any type and/or combination of streaming media, such as audio, video and text (subtitles).[Ishibashi2006] and [Boronat2009] provide an overview of technologies and algorithms for IDMS.

IDMS requires the exchange of information on media receipt and playout times among participants in an IDMS session. It may also require signaling for the initiation and maintenance of IDMS sessions and groups of receivers.

The presented RTCP specification for IDMS is independent of the used synchronization algorithm, which is out-of-scope of this document.

2. Rationale

2.1. Applicability of RTCP to IDMS

Currently, a large share of real-time applications make use of RTP and RTCP [RFC3550]. RTP provides end-to-end network transport functions suitable for applications requiring real-time data transport, such as audio, video or data, over multicast or unicast network services. The timestamps, sequence numbers, and payload (content) type identification mechanisms provided by RTP packets are very useful for reconstructing the original media timing, and for reordering and detecting packet loss at the client side.

The data transport is augmented by a control protocol (RTCP) to allow monitoring of the data delivery in a manner that is scalable to large groups, and to provide minimal control and identification functionality. RTP receivers and senders provide reception quality feedback by sending out RTCP Receiver Report (RR) and Sender Report (SR) packets [RFC3550], respectively, which may be augmented by eXtended Reports (XR) [RFC3611]. Both RTP and RTCP are intended to be tailored through modifications in order to include profile-specific information required by particular applications, and the guidelines on doing so are specified in [RFC5868].

IDMS involves the collection, summarizing and distribution of RTP packet arrival and playout times. As information on RTP packet arrival times and playout times can be considered reception quality feedback information, RTCP is well suited for carrying out IDMS, which may facilitate the implementation and deployment in typical

multimedia applications.

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] and indicate requirement levels for compliant implementations.

4. Inter-Destination Media Synchronization use cases

There are a large number of use cases in which IDMS might be useful. This section will highlight some of them. It should be noted that this section is in no way meant to be exhaustive.

A first usage scenario for IDMS is Social TV. Social TV is the combination of media content consumption by two or more users at different devices and locations combined with the real-time communication between those users. An example of Social TV is when two or more users are watching the same television broadcast at different devices and locations, while communicating with each other using text, audio and/or video. A skew in their media playout processes can have adverse effects on their experience. A well-known use case here is one friend experiencing a goal in a football match well before or after other friend(s).

Another potential use case for IDMS is a networked video wall. A video wall consists of multiple computer monitors, video projectors, or television sets tiled together contiguously or overlapped in order to form one large screen. Each of the screens reproduces a portion of the larger picture. In some implementations, each screen may be individually connected to the network and receive its portion of the overall image from a network-connected video server or video scaler. Screens are refreshed at 60 hertz (every 16-2/3 milliseconds) or potentially faster. If the refresh is not synchronized, the effect of multiple screens acting as one is broken.

A third usage scenario is that of the networked loudspeakers, in which two or more speakers are connected to the network individually. Such situations can for example be found in large conference rooms, legislative chambers, classrooms (especially those supporting distance learning) and other large-scale environments such as stadiums. Since humans are more susceptible to differences in audio delay, this use case needs even more accuracy than the video wall use case. Depending on the exact application, the need for accuracy can then be in the range of microseconds.

5. Overview of IDMS operation

This section provides a brief example of how the RTCP functionality is used for achieving IDMS. The section is tutorial in nature and does not contain any normative statements.

```
          Alice's  . . . . . . .tv:abc.com . . . . . . . . . Bob's
          TV (Sync Client)    (Sync Server)          Laptop (Sync Client)
            |                       |                          |
            |      Media Session    |                          |
            |<=====================>|                          |
            |            Invite(URL,Sync-group ID)             |
            |------------------------------------------------->|
            |                       |   Media Session Set-up   |
            |                       |<========================>|
            |                       |                          |
            |                 Call set-up                      |
            |<================================================>|
            |                       |                          |
            |       RTP Packet      |        RTP Packet        |
            |<----------------------|------------------------->|
            |  RR + XR IDMS Report  |                          |
            |---------------------->|    RR + XR IDMS Report   |
            |                       |<-------------------------|
            |   RTCP IDMS Settings  |    RTCP IDMS Settings    |
            |<----------------------|------------------------->|
            |                       |                          |
```

Figure 1: Example of a typical IDMS session

Alice is watching TV in her living room. At some point she sees that a football game of Bob's favorite team is on. She sends him an invite to watch the program together. Embedded in the invitation is the link to the media server and a unique sync-group identifier.

Bob, who is also at home, receives the invite on his laptop. He accepts Alice's invitation and the RTP client on his laptop sets up a session to the media server. A VoIP connection to Alice's TV is also set up, so that Alice and Bob can talk while watching the game together.

As is common with RTP, both the RTP client in Alice's TV as well as the one in Bob's laptop send periodic RTCP Receiver Reports (RR) to the media server. However, in order to make sure Alice and Bob see the events in the football game at (approximately) the same time, their clients also periodically send an RTCP XR IDMS Report Block to the sync server function of the media server. Included in the XR blocks are timestamps on when both Alice and Bob received (and,

optionally, when they played out) a particular RTP packet.

The sync server function in the media server calculates a reference client from the received IDMS Report Blocks (e.g. by selecting whichever client received the packet the latest as the reference client). It then sends an RTCP IDMS Settings packet containing the playout information of this reference client to the sync clients of both Alice and Bob.

In this case Bob's connection has the longest delay and the reference client therefore includes a delay similar to the one experienced by Bob. Upon reception of this information, Alice's RTP client can choose what to do with this information. In this case it decreases its playout rate temporarily until the playout time matches with the reference client playout (and thus matches Bob's playout). Another option for Alice's TV would be to simply pause playback until it catches up. The exact implementation of the synchronization algorithm is up to the client.

Upon reception of the reference client RTCP IDMS Settings packet, Bob's client does not have to do anything since it is already synchronized to the reference client (since it is based on Bob's delay). Note that other synchronization algorithms may introduce even more delay than the one experienced by the most delayed client, e.g. to account for delay variations, for new clients joining an existing synchronization group, etc.

For this functionality to work correctly, it is nessecary that the wall clocks of the receivers are synchronized with each other. Alice and Bob both report when they receive, and optionally when they play out, certain RTP packets. In order to correlate their reports to each other, it is necessary that their wallclocks are synchronized.

6. Architecture for Inter-Destination Media Synchronization

The architecture for IDMS, which is based on a sync-maestro architecture [Boronat2009], is sketched below. The Synchronization Client (SC) and Media Synchronization Application Server (MSAS) entities are shown as additional functionality for the RTP receiver and sender respectively.

It should be noted that a master/slave type of architecture is also supported by having one of the SC devices also act as an MSAS. In this case the MSAS functionality is thus embedded in an RTP receiver instead of an RTP sender.

```
+-----------------------+       +-----------------------+
|                       |  SR + |                       |
|      RTP Receiver     |  RTCP |       RTP Sender      |
|                       |  IDMS |                       |
|  +-----------------+  | <-----|  +-----------------+  |
|  |                 |  |       |  |                 |  |
|  | Synchronization |  |       |  |      Media      |  |
|  |     Client      |  |       |  | Synchronization |  |
|  |      (SC)       |  |       |  |   Application   |  |
|  |                 |  |       |  |      Server     |  |
|  |                 |  | RR+XR |  |      (MSAS)     |  |
|  |                 |  | ----->|  |                 |  |
|  +-----------------+  |       |  +-----------------+  |
|                       |       |                       |
+-----------------------+       +-----------------------+
```

IDMS Architecture Diagram

6.1. Media Synchronization Application Server (MSAS)

An MSAS collects RTP packet arrival times and playout times from one or more SC(s) in a synchronization group. The MSAS summarizes and distributes this information to the SCs in the synchronization group as synchronization settings, e.g. by determining the SC with the most lagged playout and using its reported RTP packet arrival time and playout time as a summary.

6.2. Synchronization Client (SC)

An SC reports on RTP packet arrival times and playout times of a media stream. It can receive summaries of such information, and use that to adjust its playout buffer.

6.3. Communication between MSAS and SCs

Two different message types are used for the communication between MSAS and SCs. For the SC->MSAS message containing the playout information of a particular client, an RTCP XR IDMS Report Block used (see Section 7). For the MSAS->SC message containing the synchronization settings instructions, a new RTCP IDMS Settings Packet Type is defined (see Section 8).

7. RTCP XR Block for IDMS (IDMS Report Block)

This section specifies a new RTCP XR Block Type, the RTCP XR IDMS Report Block, for reporting IDMS information to an MSAS. In particular it is used to provide feedback information on receipt

times and presentation times of RTP packets. Its definition is based on [RFC3550] and [RFC3611].

In most cases, a single RTP receiver will only be part of single IDMS session, i.e. it will report on receipt and presentation times of RTP packets from a single RTP stream in a certain synchronization group. In some cases however, an RTP receiver may be a member of multiple synchronization groups for the same RTP stream, e.g. watching a single television program simultaneously with different groups. In even further cases, a receiver may wish to synchronize different RTP streams at the same time, either as part of the same synchronization group or as part of multiple synchronization groups. These are all valid scenario's for IDMS, and will require multiple reports by an SC.

SCs SHOULD report on a recently received RTP packets. This document does not define new rules on when to sent RTCP reports, but uses the existing rules specified in [RFC3550] for sending RTCP reports. When the RTCP reporting timer allows an SC to send an IDMS report, the SC SHOULD report on an RTP packet received during the period since the last RTCP XR IDMS Report Block was sent. For more details on which packet to report on, see below under 'Packet Received RTP timestamp'.

```
 0                   1                   2                   3
 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|V=2|P| Resrv   |   PT=XR=207   |             length            |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                     SSRC of packet sender                     |
+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+
|     BT=12     | SPST  |Resrv|P|         block length=7        |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|     PT      |                 Resrv                           |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|              Media Stream Correlation Identifier              |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                     SSRC of media source                      |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|      Packet Received NTP timestamp, most significant word     |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|      Packet Received NTP timestamp, least significant word    |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|              Packet Received RTP timestamp                    |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|              Packet Presented NTP timestamp                   |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```

The first 64 bits form the header of the RTCP XR, as defined in

[RFC3611]. The SSRC of packet sender identifies the sender of the specific RTCP packet.

The IDMS report block consists of 8 32-bit words, with the following fields:

Block Type (BT): 8 bits. It identifies the block format. Its value SHALL be set to 12.

Synchronization Packet Sender Type (SPST): 4 bits. This field identifies the role of the packet sender for this specific eXtended Report. It can have the following values:

SPST=0 Reserved for future use.

SPST=1 The packet sender is an SC. It uses this XR to report synchronization status information. Timestamps relate to the SC input.

SPST=2-4 Values defined by ETSI TISPAN (see [TS183063]).

SPST=5-15 Reserved for future use.

Reserved bits (Resrv): 3 bits. These bits are reserved for future definition. In the absence of such a definition, the bits in this field MUST be set to zero and MUST be ignored by the receiver.

Packet Presented NTP timestamp flag (P): 1 bit. Bit set to 1 if the Packet Presented NTP timestamp field contains a value, 0 if it is empty. If this flag is set to zero, then the Packet Presented NTP timestamp SHALL be ignored.

Block Length: 16 bits. This field indicates the length of the block in 32 bit words minus one and SHALL be set to 7, as this RTCP Block Type has a fixed length.

Payload Type (PT): 7 bits. This field identifies the format of the media payload, according to [RFC3551]. This is the payload type of the RTP packet reported upon. The media payload is associated with an RTP timestamp clock rate. This clock rate provides the time base for the RTP timestamp counter.This clock rate is nessecary for the MSAS to relate reports from different SCs on different RTP timestamp values.

Reserved bits (Resrv): 25 bits. These bits are reserved for future use and SHALL be set to 0.

Media Stream Correlation Identifier: 32 bits. This identifier is

used to correlate synchronized media streams. The value 0 (all bits are set "0") indicates that this field is empty. The value 2^32-1 (all bits are set "1") is reserved for future use. If the RTCP Packet Sender is an SC (SPST=1), then the Media Stream Correlation Identifier field contains the Synchronization Group Identifier (SyncGroupId) to which the report applies.

SSRC: 32 bits. The SSRC of the media source SHALL be set to the value of the SSRC identifier carried in the RTP header [RFC3550] of the RTP packet to which the XR relates.

Packet Received NTP timestamp: 64 bits. This timestamp reflects the wall clock time at the moment of arrival of the first octet of the RTP packet to which the XR relates. It is formatted based on the NTP timestamp format as specified in [RFC5905]. See Section 9 for more information on how this field is used.

Packet Received RTP timestamp: 32 bits. This timestamp has the value of the RTP timestamp carried in the RTP header [RFC3550] of the RTP packet to which the XR relates. Several consecutive RTP packets will have equal timestamps if they are (logically) generated at once, e.g., belong to the same video frame. It may well be the case that one receiver reports on the first RTP packet having a certain RTP timestamp and a second receiver reports on the last RTP packet having that same RTP timestamp. This would lead to an error in the synchronization algorithm due to the faulty interpretation of considering both reports to be on the same RTP packet. When reporting on an RTP packet which is one of several consecutive RTP packets having equal timestamps, an SC SHOULD report on the RTP packet it received with the lowest sequence number. Note that with 'lowest sequence number' here is meant the first in the sequence of RTP packets just received, not from an earlier time before the last wrap-around of RTP timestamps (unless this wrap-around occurs during the sequence with equal RTP timestamps).

Packet Presented NTP timestamp: 32 bits. This timestamp reflects the wall clock time at the moment the rendered frame contained in the first byte of the associated RTP packet is presented to the user. It is based on the time format used by NTP and consists of the least significant 16 bits of the NTP seconds part and the most significant 16 bits of the NTP fractional second part. If this field is empty, then it SHALL be set to 0 and the Packet Presented NTP timestamp flag (P) SHALL be set to 0. Presented here means the moment the data is played out to the user of the system, i.e. sound played out through speakers, video images being displayed on some display, etc. The accuracy resulting from the synchronization algorithm will only be as good as the accuracy with which the receivers can determine the delay between receiving packets and presenting them to the end-user.

8. RTCP Packet Type for IDMS (IDMS Settings)

This section specifies the RTCP Packet Type for indicating synchronization settings instructions to the receivers of the RTP media stream. Its definition is based on [RFC3550].

```
  0                   1                   2                   3
  0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1
 +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
 |V=2|P| Resrv   |     PT=TBD    |             length            |
 +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
 |                      SSRC of packet sender                    |
 +=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+
 |                      SSRC of media source                     |
 +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
 |               Media Stream Correlation Identifier             |
 +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
 |       Packet Received NTP timestamp, most significant word    |
 +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
 |       Packet Received NTP timestamp, least significant word   |
 +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
 |               Packet Received RTP timestamp                   |
 +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
 |      Packet Presented NTP timestamp, most significant word    |
 +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
 |      Packet Presented NTP timestamp, least significant word   |
 +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```

The first 64 bits form the header of the RTCP Packet Type, as defined in [RFC3550]. The SSRC of packet sender identifies the sender of the specific RTCP packet.

The RTCP IDMS packet consists of 7 32-bit words, with the following fields:

PT: To be determined upon registration with IANA, inserted by the RFC Editor upon registration with IANA.

SSRC: 32 bits. The SSRC of the media source SHALL be set to the value of the SSRC identifier of the media source carried in the RTP header [RFC3550] of the RTP packet to which the RTCP IDMS packet relates.

Media Stream Correlation Identifier: 32 bits. This identifier is used to correlate synchronized media streams. The value 0 (all bits are set "0") indicates that this field is empty. The value 2^32-1 (all bits are set "1") is reserved for future use. The Media Stream Correlation Identifier contains the SyncGroupId of the group to which

this packet is sent.

Packet Received NTP timestamp: 64 bits. This timestamp reflects the wall clock time at the reference client at the moment it received the first octet of the RTP packet to which this packet relates. It can be used by the synchronization algorithm on the receiving SC to adjust its playout timing in order to achieve synchronization, e.g. to set the required playout delay. The timestamp is formatted based on the NTP timestamp format as specified in [RFC5905]. See Section 9 for more information on how this field is used. Because RTP timestamps do wrap around, the sender of this packet SHOULD use recent values, i.e. choose NTP timestamps that reflect current time and not too far in the future or in the past.

Packet Received RTP timestamp: 32 bits. This timestamp has the value of the RTP timestamp carried in the RTP header [RFC3550] of the RTP packet to which the XR relates. This SHOULD relate to the first arriving RTP packet containing this particular RTP timestamp, in case multiple RTP packets contain the same RTP timestamp.

Packet Presented NTP timestamp: 64 bits. This timestamp reflects the wall clock time at the reference client at the moment it presented the rendered frame contained in the first octet of the associated RTP packet to the user. The timestamp is formatted based on the NTP timestamp format as specified in [RFC5905]. If this field is empty, then it SHALL be set to 0. This field MAY be left empty if none or only one of the receivers reported on presentation timestamps. Presented here means the moment the data is played out to the user of the system.

In some use cases (e.g. phased array transducers), the level of control an MSAS might need to have over the exact moment of playout is so precise that a 32bit Presented Timestamp will not suffice. For this reason, this RTCP Packet Type for IDMS includes a 64bit Presented Timestamp field. Since an MSAS will in practice always add some extra delay to the delay reported by the most lagged receiver (to account for packet jitter), it suffices for the IDMS XR Block Type with which the SCs report on their playout to have a 32bit Presented Timestamp field.

9. Timing and NTP Considerations

To achieve IDMS, the different receivers involved need synchronized (wall) clocks as a common timeline for synchronization. This synchronized clock is used for reporting the Packet Received NTP Timestamps and the Packet Presented NTP Timestamp, and for interpretation of these fields in received IDMS reports. Depending

on the synchronization accuracy required, different clock synchronization methods can be used. For social TV, synchronization accuracy should be achieved on the order of hundreds of milliseconds. In that case, correct use of NTP on receivers will in most situations achieve the required accuracy. As a guideline, to deal with clock drift of receivers, receivers should synchronize their clocks at the beginning of a synchronized session. In case of high required accuracy, the synchronized clocks of different receivers should not drift beyond the accuracy required for the synchronization mechanism. In practice, this can mean that receivers need to synchronize their clocks repeatedly during a synchronization session.

Because of the stringent synchronization requirements for achieving good audio in some use cases, a high accuracy will be needed. In this case, use of the global NTP system may not be sufficient. For improved accuracy, a local NTP server could be set up, or some other more accurate clock synchronization mechanism can be used, such as GPS time or the Precision Time Protocol [IEEE-1588].

[I-D.draft-ietf-avtcore-clksrc] defines a set of SDP parameters for signaling the clock synchronization source or sources available to and used by the individual receivers. SCs MAY use this draft to indicate their clock synchronization source or sourced in use and available. Using these paramenters, an SC can indicate which synchronization source is being used at the moment, the last time the SC synchronized with this source and the synchronization frequency. An SC can also indicate any other synchronization sources available to it. This allows multiple SCs in an IDMS session to use the same or a similar clock source for their session.

Applications performing IDMS may or may not be able to choose a synchronization method for the system clock, because this may be a system-wide setting which the application cannot change. How applications deal with this is up to the implementation. The application might control the system clock, or it might use a separate application clock or even a separate IDMS session clock. It might also report on the system clock and the synchronization method used, without being able to change it.

[I-D.draft-ietf-leap-seconds] presents some guidelines on how RTP senders and receivers should deal with leap seconds. When relying on NTP for clock synchronization, IDMS is particularly sensitive to leap second induced timing discrepancies. It is RECOMMENDED to take the guideline specified in [I-D.draft-ietf-leap-seconds] into account when implementing IDMS.

10. On the use of presentation timestamps

A receiver can report on different timing events, i.e. on packet arrival times and on playout times. A receiver SHALL report on arrival times and a receiver MAY report on playout times. RTP packet arrival times are relatively easy to report on. Normally, the processing and playout of the same media stream by different receivers will take roughly the same amount of time. Synchronizing on packet arrival times, may lead to some accuracy loss, but it will be adequate for many applications, such as social TV.

Also, if the receivers are in some way controlled, e.g. having the same buffer settings and decoding times, high accuracy can be achieved. However, if all receivers in a synchronization session have the ability to report on, and thus synchronize on, actual playout times, or packet presentation times, this may be more accurate. It is up to applications and implementations of this RTCP extension whether to implement and use this.

11. SDP Signalling for RTCP IDMS Packet Type

The SDP attribute rtcp-idms is used to signal the use of the RTCP IDMS Packet Type for IDMS and the associated RTCP XR Block for IDMS. It is also used to carry an identifier of the synchronization group to which clients belong or will belong. The SDP attribute is used as a media-level attribute during session setup. This means that in case of multiple related streams, IDMS is performed on one of them. The other streams will be synchronized to this first stream using existing inter-stream synchronization (i.e. lip-sync) solutions, i.e. using Sender Reports based on a common clock source. Basic guidelines for choosing the media stream for IDMS is to choose audio above video, as humans are more sensitive to degradation in audio quality then in video quality. When using mutli-description or multi-view codecs, the IDMS control should be performed on the base layer.

This SDP attribute is defined as follows, using Augmented Backus-Naur Form [RFC5234].

```
rtcp-idms = "a=" "rtcp-idms" ":" [sync-grp] CRLF

sync-grp = "sync-group=" SyncGroupId

SyncGroupId = 1*10DIGIT ; Numerical value from 0 through 4294967294

DIGIT = %x30-39
```

SyncGroupId is a 32-bit unsigned integer and represented in decimal. SyncGroupId identifies a group of SCs for IDMS. The value SyncGroupId=0 represents an empty SyncGroupId. The value 4294967294 (2^32-1) is reserved for future use. For a description on the value of SyncGroupId to include, see Section 12.

The following is an example of the SDP attribute for IDMS.

```
a=rtcp-idms:sync-group=42
```

12. SDP rules

12.1. Offer/Answer rules

The SDP usage for IDMS follows the rules defined in RFC3611 in section 5 on SDP signalling, with the exception of what is stated here. The IDMS usage of RTCP is a (loosely coupled) collaborative attribute, in the sense that receivers sent their status information and in response the MSAS (asynchronously) sends synchronization instructions. The rtcp-idms attribute thus indicates the ability to send and receive indicated RTCP messages. This section defines how this SDP attribute should be used with regards to offer/answer.

It is expected that in most cases, the rtcp-idms attribute will be used in an offer/answer context where receivers will have pre-determined, through some means outside the scope of this document, a SyncGroupId before the media session is setup. However, it is also supported that the MSAS assigns such a SyncGroupId, for example if the MSAS contains group management functionality. Thus, both the MSAS and the SC can insert the attribute and the SyncGroupId. Furthermore, it is allowed to insert the attribute for more than one media stream, allowing an SC to become part of multiple synchronization groups simultaneously. This effectively couples two (or more) synchronization groups to each other. If the rtcp-idms attribute is inserted more than once for a particular media session, each SyncGroupId SHALL only be inserted once.

In order to join an IDMS session, the receiver (the SC) inserts the rtcp-idms attribute as a media level attribute in the SDP offer. This SDP offer can be an initial offer, if the media session is starting as a synchronized session. The SDP offer can also be an update to an existing media session, converting the session to an IMDS session. If the receiver has a pre-determined SyncGroupId value, it SHOULD use this value for setting the SyncGroupId parameter in the rtcp-idms attribute. If the receiver does not know the SyncGroupId to be used, it MAY leave the SyncGroupId parameter empty by setting its value to 0.

The MSAS SHALL include the rtcp-idms attribute in its answer. If the value of the SyncGroupId parameter in the offer was not empty (not equal to 0), the MSAS SHOULD NOT change the SyncGroupId in its answer. If the SyncGroupId was empty, the MSAS SHALL include the proper SyncGroupId in its answer. If the MSAS receives an offer with the value of the SyncGroupId parameter set to 0, and cannot determine the proper SyncGroupId, it SHALL remove the attribute from its answer.

An MSAS receiving an SDP offer without the rtcp-idms attribute can also decide that IDMS is applicable to that media session. In such a case, the MSAS MAY insert the rtcp-idms attribute, including a non-empty SyncGroupId, as part of its answer.

A receiver receiving an rtcp-idms attribute as part of the SDP answer from an MSAS, SHALL start sending IDMS XR reports (following all the normal RTCP rules for sending RTCP XR blocks) and SHALL be ready to start receiving IDMS Settings. As usual, if a receiver does not support the attribute (e.g. in case of an MSAS-inserted IDMS attribute), it SHALL ignore the attribute.

Different updates are applicable to such an IDMS session. Updates can be sent ommitting the rtcp-idms attribute, thereby ending the (involvement in) the synchronization session. Updates can also be sent including the rtcp-idms attribute, but with a different SyncGroupId. This indicates a switch in synchronization group. Updates can also be sent including another rtcp-idms attribute, indicating a membership of another synchronization group, effectively merging the current group(s) with the new one.

12.2. Declarative cases

In certain situations, there is no offer/answer context, but only a declarative modus. In this case, the MSAS just inserts the rtcp-idms attribute and a valid SyncGroupId. Any receiver receiving the rtcp-idms attribute in such a declarative case, SHALL start sending IDMS XR Report Blocks and SHALL be ready to start receiving RTCP IDMS Settings packets.

13. Security Considerations

The security considerations described in [RFC3611] apply to this document as well.

The RTCP XR IDMS Report Block defined in this document is used to collect, summarize and distribute information on packet reception- and playout-times of streaming media. The information may be used to

orchestrate the media playout at multiple devices.

Errors in the information, either accidental or malicious, may lead to undesired behavior. For example, if one device erroneously or maliciously reports a two-hour delayed playout, then another device in the same synchronization group could decide to delay its playout by two hours as well, in order to keep its playout synchronized. A user would likely interpret this two hour delay as a malfunctioning service.

Therefore, the application logic of both Synchronization Clients and Media Synchronization Application Servers should check for inconsistent information. Differences in playout time exceeding configured limits (e.g. more than ten seconds) could be an indication of such inconsistent information.

No new mechanisms are introduced in this document to ensure confidentiality. Encryption procedures, such as those being suggested for a Secure RTP (SRTP) at the time that this document was written, can be used when confidentiality is a concern to end hosts.

14. IANA Considerations

This document defines a new RTCP packet type, the RTCP IDMS Packet (IDMS Settings), within the existing Internet Assigned Numbers Authority (IANA) registry of RTCP Control Packet Types. This document also defines a new RTCP XR Block Type, the IDMS XR Report Block, within the existing IANA registry of RTCP Extended Reports (RTCP XR) Block Types.

Further, this document defines a new SDP attribute "rtcp-idms" within the existing IANA registry of SDP Parameters, part of the "att-field (media level only)"

14.1. RTCP IDMS Packet Type

This document assigns the packet type value TBD in the IANA 'RTCP Control Packet types (PT) Registry' to the RTCP IDMS Packet Type.

[Note to RFC Editor: please replace TBD with the IANA-provided RTCP Packet Type value for this packet type]

14.2. RTCP XR IDMS Report Block

This document assigns the block type value 12 in the IANA "RTCP XR Block Type Registry" to the RTCP XR IDMS Report Block.

[Note to RFC Editor: this block type value is currently assigned to [TS183063]. This document replaces [TS183063] as the normative specification of the RTCP XR IDMS Report Block. Upon publication of this document as RFC, [TS183063] will be changed to reflect this.

14.3. RTCP-IDMS SDP Attribute

The SDP attribute "rtcp-idms" defined by this document is registered with the IANA registry of SDP Parameters as follows:

SDP Attribute ("att-field"):

Attribute name: rtcp-idms

Long form: RTCP IDMS Parameters

Type of name: att-field

Type of attribute: media level

Subject to charset: no

Purpose: see Section 11 of this document

Reference: this document

Values: see this document

14.4. Contact Information for Registrations

The contact information for the registrations is:

Ray van Brandenburg (ray.vanbrandenburg@tno.nl)

Brassersplein 2

2612CT, Delft, The Netherlands

15. Contributors

The following people have participated as co-authors or provided substantial contributions to this document: Omar Niamut, Fabian Walraven, Ishan Vaishnavi and Rufael Mekuria. In addition the authors would like to thank Aidan Williams, Colin Perkins, Magnus Westerlund, Roni Even, Peter Musgrave, Ali Begen, Qin Wu and Rob

Koenen for their review comments and contributions to the text.

16. References

16.1. Normative References

[I-D.draft-ietf-avtcore-clksrc]
Williams, A., van Brandenburg, R., Stokking, H., and K. Gross, "RTP Clock Source Signalling, draft-ietf-avtcore-clksrc-01", October 2012.

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

[RFC3550] Schulzrinne, H., Casner, S., Frederick, R., and V. Jacobson, "RTP: A Transport Protocol for Real-Time Applications, RFC3550", July 2003.

[RFC3551] Schulzrinne, H. and S. Casner, "RTP Profile for Audio and Video conferences with Minimal Control, RFC3551", July 2003.

[RFC3611] Friedman, T., Ed., Caceres, R., Ed., and A. Clark, Ed., "RTP Control Protocol Extended Reports (RTCP XR), RFC3611", November 2003.

[RFC4566] Handley, M., Jacobson, V., and C. Perkins, "SDP: Session Description Protocol, RFC4566", July 2006.

[RFC5234] Crocker, D., Ed. and P. Overell, "Augmented BNF for Syntax Specifications, RFC5234", January 2008.

[RFC5760] Ott, J., Chesterfield, J., and E. Schooler, "RTP Control Protocol (RTCP) Extensions for Single-Source Multicast Sessions with Unicast Feedback, RFC5760", February 2010.

[RFC5905] Mills, D., Martin, J., Ed., Burbank, J., and W. Kasch, "Network Time Protocol Version 4: Protocol and Algorithms Specifications, RFC5905", February 2010.

[TS183063]
"IMS-based IPTV stage 3 specification, TS 183 063 v3.4.1", June 2010.

16.2. Informative References

[Boronat2009]
Boronat, F., Lloret, J., and M. Garcia, "Multimedia group and inter-stream synchronization techniques: a comparative study, Elsevier Information Systems 34 (2009), pp. 108-131".

[I-D.draft-ietf-leap-seconds]
Gross, K. and R. Brandenburg, van, "RTP and Leap Seconds, draft-ietf-leap-seconds-01", October 2012.

[IEEE-1588]
"1588-2008 - IEEE Standard for a Precision Clock Synchronization Protocol for Networked Measurement and Control Systems", 2008.

[Ishibashi2006]
Ishibashi, Y., Nagasaka, M., and N. Fujiyoshi, "Subjective Assessment of Fairness among users in multipoint communications, Proceedings of the 2006 ACM SIGCHI internation conference on Advances in computer entertainment technology, 2006".

[RFC5868] Ott, J. and C. Perkins, "Guidelines for Extending the RTP Control Protocol (RTCP), RFC5968", September 2010.

Authors' Addresses

Ray van Brandenburg
TNO
Brassersplein 2
Delft 2612CT
the Netherlands

Phone: +31-88-866-7000
Email: ray.vanbrandenburg@tno.nl

Hans Stokking
TNO
Brassersplein 2
Delft 2612CT
the Netherlands

Phone: +31-88-866-7000
Email: hans.stokking@tno.nl

M. Oskar van Deventer
TNO
Brassersplein 2
Delft 2612CT
the Netherlands

Phone: +31-88-866-7000
Email: oskar.vandeventer@tno.nl

Fernando Boronat
Universitat Politecnica de Valencia
IGIC Institute, Universitat Politecnica de Valencia-Campus de Gandia (UPV),
C/ Paraninfo, 1, Grao de Gandia, C/ Paraninfo, 1, Grao de Gandia
Valencia 46730
Spain

Phone: +34 962 849 341
Email: fboronat@dcom.upv.es

Mario Montagud
Universitat Politecnica de Valencia
IGIC Institute, Universitat Politecnica de Valencia-Campus de Gandia (UPV),
C/ Paraninfo, 1, Grao de Gandia, C/ Paraninfo, 1, Grao de Gandia
Valencia 46730
Spain

Phone: +34 962 849 341
Email: mamontor@posgrado.upv.es

Kevin Gross
AVA Networks

Phone: +1-303-447-0517
Email: Kevin.Gross@AVAnw.com