

INTERNET-DRAFT
[draft-ietf-avt-rtcp-feedback-00.txt](#)

Jörg Ott/Universität Bremen TZI
Stephan Wenger/TU Berlin
Shigeru Fukunaga/OKI
Noriyuki Sato/OKI
Koichi Yano/Fast Forward Networks
Akihiro Miyazaki/Matsushita
Koichi Hata/Matsushita
Rolf Hakenberg/Matsushita
Carsten Burmeister/Matsushita

13 July, 2001
Expires January 2002

Extended RTP Profile for RTCP-based Feedback

Status of this Memo

This document is an Internet-Draft and is in full conformance with all provisions of [Section 10 of RFC 2026](#). 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/1id-abstracts.txt>

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

Abstract

Real-time media streams are not resilient against packet losses. RTP [1] provides all the necessary mechanisms to restore ordering and timing to properly reproduce a media stream at the recipient. RTP also provides continuous feedback about the overall reception quality from all receivers -- thereby allowing the sender(s) in the mid-term (in the order of several seconds to minutes) to adapt their coding scheme and transmission behavior to the observed network QoS. However, except for a few payload specific mechanisms [10], RTP makes no provision for timely feedback that would allow a sender to repair the media stream immediately: through retransmissions, retro-active FEC, or media-specific mechanisms such as reference picture selection.

Generally, real-time transport of media streams across IP networks follows RTP[1] in conjunction with the RTP Profile for Audio and Video Conferences with Minimal Control [2]. This document modifies the profile defined in [2] in two ways:

- . by providing additional RTCP messages that enable a receiver to convey more precise feedback to a sender and
- . by adapting the timing algorithm for scheduling RTCP packets in order to allow for occasional timely feedback about events observed by a receiver (such as lost packets).

The result is an RTP Profile for Audio and Video Conferences with Minimal Control that allows for more explicit and more immediate receiver feedback but shares all other properties (including all other message types and formats, all code points for codecs, payload formats, scaling capabilities, etc. of [2]). Therefore, this document only specifies the additions and modifications to [2] rather than the repeating the entire specification.

1. Introduction

Real-time media streams are not resilient against packet losses. RTP [1] provides all the necessary mechanisms to restore ordering and timing present at the sender to properly reproduce a media stream at a recipient. RTP also provides continuous feedback about the overall reception quality from all receivers -- thereby allowing the sender(s) in the mid-term (in the order of several seconds to minutes) to adapt their coding scheme and transmission behavior to the observed network QoS. However, except for a few payload specific mechanisms [10], RTP makes no provision for timely feedback that would allow a sender to repair the media stream immediately: through retransmissions, retro-active FEC, or media-specific mechanisms such as reference picture selection.

Current mechanisms available with RTP to improve error resilience include audio redundancy coding [7], video redundancy coding [11], RTP-level FEC [5], and general considerations on more robust media streams transmission [6]. Particularly in small groups, however, virtually all kinds of real-time media streams could benefit from a mechanism that would enable a sender to perform media stream repair - including but not limited to audio, video, DTMF, and text chat streams. In some cases of networks with acceptable round-trip times but scarce bandwidth, occasional retransmissions may be much preferred over continuous transmission of redundant information.

For example, predictive video coding is not loss resilient. Any loss of coded data leads to annoying artifacts not only in the reproduced

picture in which the loss occurred, but also in subsequent pictures. Error resilience can be achieved by allocating bits to convey redundant information using source coding based mechanisms or transport based mechanisms. This can be done without the use of any feedback between the decoder(s) and the encoder. Similar consideration apply to protecting e.g. DTMF (and other tones) carried in an RTP stream [\[9\]](#).

Alternatively, where applicable, receivers can inform the sender through a feedback channel about a loss situation, and the sender can react accordingly. This approach provides better media quality and is more efficient with respect to the bandwidth used by the sender to achieve a given media quality. However, using feedback mechanisms is limited to certain application scenarios identified by encoder characteristics, delay constraints, and/or the number of recipients.

This document specifies a modified RTP Profile for Audio and Video conferences with minimal control based upon [1] and [2] by means two modifications/additions:

To achieve timely feedback the concepts of Immediate Feedback messages and Early RTCP messages as well as algorithms allowing for low delay feedback in small multicast groups (and preventing feedback implosion in large ones) are introduced. Special consideration is given to point-to-point scenarios.

In addition, various types of general-purpose feedback messages as well as a format for codec and application-specific feedback information are defined as specific RTCP payloads.

1.1 Definitions

The definitions from [1] and [2] apply. In addition, the following definitions are used in this document:

Early RTCP mode:

The mode of operation in which a receiver of a media stream is, statistically, often (but not always) capable of reporting events of interest back to the sender close to their occurrence. In Early RTCP mode, RTCP feedback messages are transmitted according to the timing rules defined in this document.

Early RTCP packet:

An Early RTCP packet is a packet which is transmitted earlier than would be allowed following the scheduling algorithm of [1], the reason being that an event observed by a receiver. Early RTCP packets may be sent in Immediate feedback and in Early RTCP mode.

Event:

An observation made by the receiver of a media stream that is (potentially) of interest to the sender -- such as a packet loss or packet reception, frame loss, etc. -- and thus to be reported back to the sender by means of a Feedback message.

Feedback (FB) message:

An RTCP message as defined in this document used to convey events observed at a receiver -- in addition to long term receiver status information which is carried in RTCP RRs -- back to the sender of the media stream.

Feedback (FB) threshold:

The FB threshold indicates the "borderline" between Immediate Feedback and Early RTCP mode. For a multicast scenario, the FB threshold indicates the maximum group size at which, on average, each receiver is able to report each event back to the sender(s) immediately, i.e. without having to wait for its regularly scheduled RTCP interval. This threshold is highly dependent on network QoS (e.g. packet loss probability and distribution), codec and packetization in use, and application requirements. Hence, no formal definition is presented in this document.

Immediate Feedback mode:

Mode of operation in which each receiver of a media is, statistically, capable of reporting each event of interest immediately back to the media stream sender. In Immediate Feedback mode, RTCP feedback messages are transmitted according to the timing rules defined in this document.

Regular RTCP mode:

Mode of operation in which no preferred transmission of feedback messages is allowed. Instead, RTCP messages are sent following the rules of [1] and may contain feedback messages information as defined in this document.

Regularly Scheduled RTCP packet:

An RTCP packet that is not sent as an Early RTCP packet.

1.2 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 [RFC 2119](#) [8]

2. RTP and RTCP Packet Formats and Protocol Behavior

The rules defined in [2] also apply to this profile except for those rules mentioned in the following:

RTCP packet types:

Three additional RTCP packet types to convey feedback information are defined in [section 4](#).

RTCP report intervals:

This memo describes three modes of operation which influence the RTCP report intervals (see [section 3.2](#)). In regular RTCP mode, all rules from [1] apply. In both Immediate

Feedback and Early RTCP modes the minimal interval of 5 seconds between 2 RTCP reports is dropped and the rules specified in [section 3](#) apply if RTCP packets containing feedback messages (defined in [section 4](#)) are to be transmitted.

The rules set forth in [1] may be overridden by session descriptions specifying different parameters (e.g. for the bandwidth share assigned to RTCP for senders and receivers, respectively. For sessions defined using the Session Description Protocol (SDP) [3], the rules of [4] apply.

Congestion control:

The same basic rules as detailed in [2] apply. Beyond this, in [section 5](#), further consideration is given to the impact of feedback and a sender's reaction to feedback messages.

[3. Rules for RTCP Feedback](#)

3.1 Compound RTCP Feedback Packets

Two components constitute RTCP-based feedback as described in this memo:

- . Status reports are contained in SR/RR messages and are transmitted at regular intervals as part of compound RTCP packets (which also include SDES and possibly other messages); these status reports provide an overall indication for the recent reception quality of a media stream.
- . Feedback messages as defined in this document that indicate loss or reception of particular pieces of a media stream (or provide some other form of rather immediate feedback on the data received). Rules for the transmission of feedback messages are newly introduced in this memo.

RTCP Feedback (FB) messages are just another RTCP packet type (see [section 4](#)). Therefore, multiple FB messages MAY be combined in a single compound RTCP packet and they MAY also be sent combined with other RTCP packets.

RTCP packets containing Feedback packets as defined in this document MUST contain RTCP packets in the order as defined in [1]:

- . OPTIONAL encryption prefix that MUST be present if the RTCP message is to be encrypted.
- . MANDATORY SR or RR.
- . MANDATORY SDES which MUST contain the CNAME item; all other SDES items are OPTIONAL.
- . One or more FB messages.

The FB MUST be placed in the compound packet after all RTCP packets defined in [1]. The ordering with respect to other RTCP extensions is not defined.

Two types of compound RTCP packets carrying feedback packets are used in this document:

a) Minimal compound RTCP feedback packet

A minimal compound RTCP feedback packet MUST contain only the mandatory information as listed above: encryption prefix if necessary, exactly one RR or SR, exactly one SDES with only the CNAME item present, and the feedback message(s). This is to minimize the size of the RTCP packet transmitted to convey feedback and thus to maximize the frequency at which feedback can be provided while still adhering to the RTCP bandwidth limitations.

This packet format SHOULD be used whenever an RTCP feedback message is sent as part of an Early RTCP packet.

b) (Full) compound RTCP feedback packet

A (full) compound RTCP feedback packet MAY contain any additional number of RTCP packets (additional RRs, further SDES items, etc.).

This packet format MUST be used whenever an RTCP feedback message is sent as part of a regularly scheduled RTCP packet or in Regular RTCP mode. This packet format MAY also be used to send RTCP feedback messages in Immediate Feedback or Early RTCP mode.

RTCP packets that do not contain FB messages are referred to as non-FB RTCP packets.

3.2 Algorithm Outline

FB messages are part of the RTCP control streams and are thus subject to the same bandwidth constraints as other RTCP traffic. This means in particular that it may not be possible to report an event observed at a receiver immediately back to the sender. However, the value of feedback given to a sender typically decreases over time -- in terms of the media quality as perceived by the user at the receiving end and/or the cost required to achieve media stream repair.

RTP [1] and the commonly used RTP profile [2] specify rules when compound RTCP packets should be sent. This document modifies those rules in order to allow applications to timely report media loss or reception events to accommodate algorithms that use FB messages and are sensitive to the feedback timing.

The modified algorithm can be outlined as follows: Normally, when no FB messages have to be conveyed, compound RTCP packets are sent following the rules of RTP [1] -- except that the 5s minimum interval between RTCP reports is not enforced. If a receiver detects the need

for an FB message, the receiver waits for a short, random dithering interval (in case of multicast) and then checks whether it has already seen a corresponding FB message from any other receiver (which it can do with all FB messages that are transmitted via multicast; for unicast sessions, there is no such delay). If this is the case then the receiver refrains from sending the FB message and continues to follow the regular RTCP sending schedule. If the

receiver has not yet seen a similar FB message from any other receiver, it checks whether it has recently exceeded its RTCP bit rate budget to transmit another FB message (without waiting for its regularly scheduled RTCP transmission time). Only if this is not the case, it sends the FB message as part of a (minimal) compound RTCP packet.

FB messages may also be sent as part of full compound RTCP packets which are interspersed as per [1] in regular intervals.

3.3 Modes of Operation

RTCP-based feedback may operate in one of three modes (figure 1):

- a) Immediate feedback mode: the group size is below the FB threshold which gives each receiving party sufficient bandwidth to transmit the feedback traffic for the intended purpose. This means, for each receiver there is enough bandwidth to report each event it is supposed/expected to by means of a virtually "immediate" RTCP feedback packet.

The group size threshold is a function of a number of parameters including (but not necessarily limited to) the type of feedback used (e.g. ACK vs. NACK), bandwidth, packet rate, packet loss probability and distribution, media type, codec, and -- again depending on the type of FB used -- the (worst case or observed) frequency of events to report (e.g. frame received, packet lost).

A special case of this is the ACK mode (where positive acknowledgements are used to confirm reception of data) which is restricted to point-to-point communications.

- b) Early RTCP mode: In this mode, the group size and other parameters no longer allow each receiver to react to each event that would be worth (or needed) to report. But feedback can still be given sufficiently often so that it allows the sender to adapt the media stream transmission accordingly and thereby increase the overall reproduced media quality.
- c) From some group size upwards, it is no longer useful to provide feedback from individual receivers at all -- because of the time scale in which the feedback could be provided and/or because in large groups the sender(s) have no chance to react to individual feedback anymore.

As the feedback algorithm described in this memo scales smoothly, there is no need for an agreement on the precise values of the respective "thresholds" within the group. Hence the borders between all these modes are allowed to be fluent.


```

    ACK
feedback
V
: <- - - - NACK feedback - - - -> //
:
:   Immediate   ||
: Feedback mode || Early RTCP mode   Regular RTCP mode
: <=====> || <=====> // <=====>
:
:               ||
+-----+ || ----- // -----> group size
2               ||
Application-specific FB Threshold
    = f(data rate, packet loss, codec, ...)

```

Figure 1: Modes of operation

The respective thresholds depend on a number of technical parameters (of the codec, the transport, the feedback used, etc.) but also on the respective application scenarios. [Section 3.5](#) provides some useful hints (but no complete precise calculations) on estimating these thresholds.

3.4 Definitions

The following pieces of state information need to be maintained (largely taken from [\[1\]](#)):

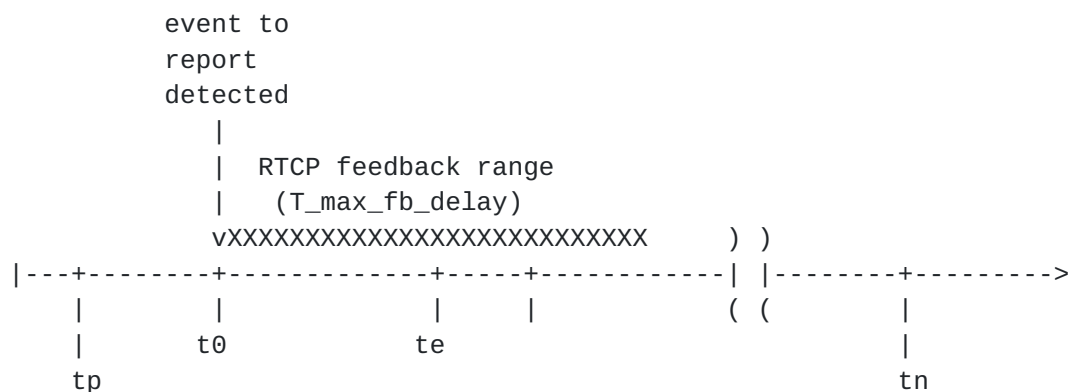
- a) Let senders be the number of active senders in the RTP session.
- b) Let members be the current estimate of the number of receivers in the RTP session.
- c) Let T_{rtt} be the maximum round trip time as measured by RTCP (if available to the receiver). Note that this may be asymmetric.
- d) Let t_n and t_p be the time for the next (last) scheduled RTCP RR transmission calculated prior to reconsideration.
- e) Let T_{rr} be the interval after which, having just sent a regularly scheduled RTCP packet, a receiver would schedule the transmission of its next RTCP packet following the rules of [\[1\]](#): $T_{rr} = t_n \cup t_p$. Note that the 5s minimum interval between two report as defined in [\[1\]](#) SHOULD NOT be enforced.
- f) Let t_0 be the time at which an event is detected by a receiver.
- g) Let T_{dither_max} be the maximum interval for which an RTCP feedback packet may be additionally delayed (to prevent

implosions).

h) Let $T_{\text{max_fb_delay}}$ be the upper bound within which feedback to an event needs to be reported back to the sender to be useful at all.

i) Let t_e be the time for which a feedback packet is scheduled.

- If a compound RTCP feedback packet is scheduled, the time slot for the next scheduled compound RTCP packet is updated accordingly to a new t_n .



_____ /
 V/
T_dither_max

Figure 2: Event report and parameters for Early RTCP scheduling

3.5 Early RTCP Algorithm

Assume an active sender S_0 (out of S senders) and a number N of receivers with R being one of these receivers.

Assume further that R has verified that using feedback mechanisms is reasonable at the current constellation (which is highly application specific and hence not specified in this memo).

Then, receiver R MUST use the following rules for transmitting a Feedback messages as minimal or full compound RTCP packet:

Initially, R MUST set `allow_early := TRUE`.

R has transmitted the last RTCP RR packet at t_p and has scheduled the next transmission (prior to reconsideration) for t_n .

At time t_0 , R detects the need to transmit a feedback message (e.g. because a media "unit" needs to be ACKed or NACKed) and finds that sending the feedback message is useful for the sender.

R first checks whether there is still an RTCP feedback packet waiting for transmission. If so, the new feedback message MUST be appended to the packet; the schedule for the waiting RTCP feedback packet MUST remain unchanged.

If no RTCP feedback message is already awaiting transmission (as part of an Early RTCP packet), a new (minimal) compound RTCP feedback packet MUST be created and the interval `T_dither_max` MUST be chosen as follows:

i) If the session is a unicast session (group size = 2) then
`T_dither_max := 0`.

ii) If the receiver has an RTT estimate to the originator of the media unit to provide feedback about, then

`T_dither_max := k * T_rtt / 2 * members`

with `k=1`.

iii) If the receiver does not have an RTT estimate to the originator, then

`T_dither_max := l * T_rr`

with `l=0.5`.

(Application-specific feedback considerations may make it worthwhile to increase `T_dither_max` beyond this value. This is up to the discretion of the implementer.)

Then, R MUST check whether its next regularly scheduled RTCP packet is within the time bounds for the RTCP FB ($t_0 + T_dither_max > t_n$). If so, an Early RTCP packet MUST NOT be scheduled; instead the FB

message MUST be stored to be appended to the regular RTCP packet scheduled for t_n .

Otherwise, R MUST check whether it is allowed to transmit an Early RTCP packet (`allow_early == TRUE`).

If so, R MUST schedule an Early RTCP packet for $t_e := t_0 + \text{RND} * T_{\text{dither_max}}$ with the RND function evenly distributed between 0 and 1.

If, while waiting for t_e , R receives an RTCP feedback packet R MUST act as follows:

1. If R understands the received feedback message's semantics and the message contents is a superset of the feedback R wanted to send then R MUST discard its own feedback message and MUST re-schedule the next regular RTCP message transmission for t_n (as calculated before).
2. If R understands the received feedback message's semantics and the message contents is not a superset of the feedback R wanted to send then R SHOULD transmit its own feedback message as scheduled.
3. If R does not understand the received feedback message's semantics then R MAY send its own feedback message as or Early RTCP packet. Alternatively, R MAY re-schedule the next regular RTCP message transmission for t_n (as calculated before) and MAY append the feedback message the now regularly scheduled RTCP message.

Refer to [section 4](#) on the comparison of feedback messages and for which feedback messages must be understood by a receiver.

Otherwise, when t_e is reached, R MUST transmit the RTCP packet containing the FB message. R then MUST set `allow_early := FALSE` and MUST recalculate $t_n := t_p + 2 * T_{\text{rr}}$. As soon as R sends its next regularly scheduled RTCP RR (at the new t_n), it MUST set `allow_early := TRUE` again.

If `allow_early == FALSE` then R MUST check the time for the next scheduled RR:

1. If $t_n \hat{=} t_0 < T_{\text{max_fb_delay}}$ (i.e. if, despite late reception, the feedback could still be useful for the sender) then R MAY create an RTCP FB message for transmission along with the RTCP packet at t_n .
2. Otherwise, R MUST discard the RTCP feedback message.

In regular RTCP intervals as specified by [\[1\]](#) (except for the five second minimum), a full compound RTCP packet is sent (which may also contain a feedback message if one has been created according to the above rules and scheduled for transmission along the full compound RTCP message).

The E bit in the message header is used upon reception to detect whether this RTCP feedback message was sent as Early RTCP or not. Hence, a feedback message that is sent as an Immediate or Early RTCP packet MUST set the E bit in the message header to "1". Feedback messages piggy-backed on regularly scheduled RTCP packets MUST set the E bit to "0". If a receiver R receives an Early RTCP packet (E=1), then it MAY set allow_early := TRUE.

Whenever an RTCP packet is sent or received -- minimal or full compound, early or regularly scheduled -- the avg_rtcp_size variable is updated accordingly (see [1]) and the tn is calculated using the new avg_rtcp_size.

3.6 Considerations on the Group Size

This section provides guidelines to the group sizes at which the various feedback modes may be used.

3.6.1 ACK mode

The group size MUST be exactly two participants, i.e. point-to-point communications. Unicast addresses SHOULD be used in the session description.

For unidirectional as well as bi-directional communication between two parties, 2.5% of the RTP session bandwidth are available for RTCP traffic from the receivers including feedback. , Assuming that out of ten RTCP packets, nine are sent as minimal compound RTCP packets and one as full compound RTCP packet, at 64kbit/s unidirectional communication scenario, a receiver can report 1.5 events per second back to the sender, at 256kbit/s 6 events and so forth.

From 1 Mbit/s upwards, a receiver would be able to acknowledge each individual frame (not packet!) in a 25 fps video stream.

ACK strategies should be defined accordingly to work properly with these bandwidth limitations.

3.6.2 NACK mode

Negative acknowledgements (or similar types of feedback) MUST be used for all groups larger than two. Of course, NACKs MAY be used for point-to-point communications as well.

Whether or not the use of Immediate or Early RTCP packets should be considered depends upon a number of parameters including session bandwidth, codec, special type of feedback, number of senders and

receivers, among many others.

The crucial parameters -- to which virtually all of the above can be reduced -- is the allowed minimal interval between two RTCP reports and the (average) number of events that presumably need reporting per time interval (plus their distribution over time, of course). The minimum interval is derived from the available RTCP bandwidth and the

expected average size of an RTCP packet. The number events to report e.g. per second may be derived from the packet loss rate and sender's rate of transmitting packets. From these two values, the allowable group size for the Immediate feedback mode can be calculated.

The upper bound for the Early RTCP mode then solely depends on the acceptable quality degradation, i.e. how many events per time interval may go unreported.

Example: If a 256kbit/s video with 30 fps is transmitted through a network with an MTU size of some 1500 bytes, then, in most cases, each frame would fit in its own packet leading to a packet rate of 30 packets per second. If 5% packet loss occurs in the network (equally distributed, no inter-dependence between receivers), then each receiver will have to report 3 packets lost each two seconds. Assuming a single sender and more than three receivers, this yields 3.75% of the RTCP bandwidth allocated to the receivers and thus 9.6kbit/s. Assuming further a size of 120 bytes for the average compound RTCP packet allows 10 RTCP packets to be sent per second or 20 in two seconds. If every receiver needs to report three packets, this yields a maximum group size of 6-7 receivers if all loss events shall be reported. The rules for transmission of immediate RTCP packets should provide sufficient flexibility for most of this reporting to occur in a timely fashion.

Extending this example to determine the upper bound for Early RTCP mode leads to the following considerations: assume that the underlying coding scheme and the application (as well as the tolerant users) allow in the order of one loss without repair per two seconds. Thus the number of packets to be reported by each receiver decreases to two per two seconds and increases the group size to 10. Assuming further that some number of packet losses are correlated, feedback traffic is further reduced and group sizes of some 12 to 16 (maybe even 20) can be reasonably well supported using Early RTCP mode.

3.7 Summary of decision steps

3.7.1 General Hints

Before even considering whether or not to send RTCP feedback information an application has to determine whether this mechanism is applicable:

- 1) An application has to decide whether -- for the current ratio of packet rate with the associated (application-specific) maximum feedback delay and the currently observed round-trip time (if available) -- feedback mechanisms can be applied at all.

This decision may obviously be based upon (and dynamically revised following) regular RTCP reception statistics.

- 2) The application has to decide whether -- for a certain observed error rate, assigned bandwidth, frame rate, and group size -- (and which) feedback mechanisms can be applied.

Regular RTCP provides valuable input to this step, too.

- 3) If these tests pass, the application has to follow the rules for transmitting Early RTCP packets or regularly scheduled RTCP packets with piggybacked feedback.

3.7.2 Session Description Attributes

A number of additional SDP parameters MAY be used to describe a session. These are defined as media level attributes.

3.7.2.1 Profile identification

The AV profile defined in [4] is referred to as "AVP" in the context of e.g. the Session Description Protocol (SDP) [3]. The profile specified in this document is referred to as "AVPF".

Feedback information following the modified timing rules as specified in this document MUST NOT be sent for a particular media session unless the profile for this session indicates the use of the "AVPF" profile.

Feedback information as part of regularly scheduled compound RTCP packets following the timing rules of [1] and [2] MAY be sent for media sessions for which the "AVP" profile is specified. In this case, however, the receiver providing feedback MUST NOT rely on the sender reacting to the feedback at all.

3.7.2.2 RTCP Feedback Capability Attribute

A new payload format-specific SDP attribute (for use with "a=fmtp:") is defined to indicate the capability of using RTCP feedback as specified in this document: "rtcp-fb". The "rtcp-fb" attribute MAY only be used as an SDP media attribute and MUST NOT be provided at the session level. The rtcp-fb attribute MUST only be used in media sessions for which the "AVPF" is specified.

The rtcp-fb attribute is used to indicate which RTCP feedback messages MAY be used in this media session for the indicated payload type. If several types of feedback are supported, several a=rtcp-fb: lines MUST be used.

If no rtcp-fb attribute is specified the RTP receivers SHOULD assume that the RTP senders only support generic NACKs. In addition, the RTP receivers MAY send feedback using other suitable RTCP feedback packets as defined for the respective media type. The RTP receivers

MUST NOT rely on the RTP senders reacting to any of the feedback messages.

If one or more rtcp-fb attributes are present in a media session description, the RTP receivers for the media session(s) containing the "rtcp-fb"

- . MUST ignore all rtcp-fb attributes of which they do not fully understand the semantics (i.e. understand the meaning of all values in the a=fmtp:rtcp-fb line);
- . SHOULD provide feedback information as specified in this document using any of the RTCP feedback packets as specified in one of the rtcp-fb attributes for this media session; and
- . MUST NOT use other feedback messages than those listed in one of the rtcp-fb attribute lines.

RTP senders MUST be prepared to receive any kind of RTCP feedback messages and MUST silently discard all those RTCP feedback messages that they do not understand.

The syntax of the rtcp-fb attribute is as follows (the feedback types and optional parameters are all case sensitive):

```
rtcp-fb-syntax      = "a=fmtp:" <format> WS "rtcp-fb" WS rtcp-fb-value
rtcp-fb-value       = "ack" rtcp-fb-param
                    | "nack" rtcp-fb-nack-param
                    | rtcp-fb-id rtcp-fb-param
rtcp-fb-id          = 1*(alpha-numeric | "-" | "_")
rtcp-fb-param       = "app"
                    | byte-string
                    | ; empty
rtcp-fb-nack-param  = "pli"
                    | "sli"
                    | "rpsi"
                    | "app"
                    | byte-string
                    | ; empty
```

The literals of the above grammar have the following semantics:

Feedback type "ack":

This feedback type indicates that positive acknowledgements for feedback are supported.

The feedback type "ack" MUST only be used if the media session is allowed to operate in ACK mode as defined in 3.6.1.2.

Parameters may be provided to further distinguish different

types of positive acknowledgement feedback. If no parameters are present, the Generic ACK as specified in [section 4.1.2](#) is implied.

If the parameter "app" is specified, this indicates the use of application layer feedback. In this case, additional parameters following "app" MAY be used to further differentiate various

types of application layer feedback. This document does not define any parameters specific to "app".

Further parameters for "ack" MAY be defined in other documents.

Feedback type "nack":

This feedback type indicates that negative acknowledgements for feedback are supported.

The feedback type "nack", without parameters, indicates use of the General NACK feedback format as defined in [section 4.2.1](#).

The following three parameters are defined in this document for use with "nack" in conjunction with the media type "video":

- . "pli" indicates the use of Picture Loss Indication feedback as defined in [section 4.3.1](#).
- . "sli" indicates the use of Slice Loss Indication feedback as defined in [section 4.3.2](#).
- . "rpsi" indicates the use of Reference Picture Selection Indication feedback as defined in [section 4.3.3](#).
- . "app" indicates the use of application layer feedback. Additional parameters after "app" MAY be provided to differentiate different types of application layer feedback. No parameters specific to "app" are defined in this document.

Further parameters for "nack" MAY be defined in other documents.

Other feedback types <rtcp-fb-id>:

Other documents MAY define additional types of feedback; to keep the grammar extensible for those cases, the rtcp-fb-id is introduced as a placeholder. A new feedback scheme name needs to be unique (and thus has to be registered with IANA). Along with a new name, its semantics, packet formats (if necessary), and rules for its operation need to be specified.

Note that it is assumed that more specific information about application layer feedback (as defined in [section 4.2.3](#)) will be conveyed as feedback types and parameters defined elsewhere. Hence, no further provision for any types and parameters is made in this document.

Further types of feedback as well as further parameters may be defined in other documents.

It is up to the recipients whether or not they send feedback information and up to the sender(s) to make use of feedback provided.

3.7.2.3 Unicasting

If an m= line in the SDP describing a session indicates unicast addresses for a particular media type (and does not operate in multi-unicast mode with all recipients listed explicitly but still

addressed via unicast), the RTCP feedback MAY operate in ACK feedback mode.

3.7.2.4 RTCP Bandwidth Modifiers

The standard RTCP bandwidth assignments as defined in [1] and [2] may be overridden by bandwidth modifiers as specified in [4]: b=RS:<bw> and b=RR:<bw> MAY be used to assign a different bandwidth (measured in bits per second) to RTP senders and receivers, respectively. The precedence rules of [4] apply to determine the actual bandwidth to be used by senders and receivers.

Applications operating knowingly over highly asymmetric links (such as satellite links) SHOULD use this mechanism to reduce the feedback rate for high bandwidth streams to prevent deterministic congestion of the feedback path(s).

3.7.2.5 Examples

Example 1: The following session description indicates a session made up from an audio and a DTMF for point-to-point communication in which the DTMF stream uses Generic ACKs. This session description could be contained in a SIP INVITE, 200 OK, or ACK message to indicate that its sender is capable of and willing to receive feedback for the DTMF stream it transmits.

```
v=0
o=alice 3203093520 3203093520 IN IP4 host.example.com
s=Media with feedback
t=0 0
c=IN IP4 host.example.com
m=audio 49170 RTP/AVPF 0 96
a=rtpmap:0 PCMU/8000
a=rtpmap:96 telephone-event/8000
a=fmtp:96 0-16
a=fmtp:96 rtcp-fb ack
```

Example 2: The following session description indicates a multicast video-only session (using H.263+) with the video source accepting Generic NACKs and Reference Picture Selection. Such a description may have been conveyed using the Session Announcement Protocol (SAP).

```
v=0
o=alice 3203093520 3203093520 IN IP4 host.example.com
s=Multicast video with feedback
t=3203130148 3203137348
m=audio 49170 RTP/AVP 0
```

c=IN IP4 224.2.1.183
a=rtpmap:0 PCMU/8000
m=video 51372 RTP/AVP 98
c=IN IP4 224.2.1.184
a=rtpmap:98 H263-1998/90000
a=fmtp:98 rtcp-fb nack
a=fmtp:98 rtcp-fb nack rpsi

4. Format of RTCP Feedback messages

This section defines the format of the low delay RTCP feedback messages. These messages classified into three categories as follows:

- Transport layer feedback messages
- Payload-specific feedback messages
- Application layer feedback messages

Transport layer feedback messages are intended to transmit general purpose feedback information, i.e. information independent of the particular codec or the application in use. The information is expected to be generated and processed at the transport/RTP layer. Currently, only a general positive acknowledgement (ACK) and negative acknowledgement (NACK) message are defined.

Payload-specific feedback messages transport information that is specific to a certain payload and will be generated and acted upon at the codec "layer". This document defines a common header to be used in conjunction with all payload-specific feedback messages. The definition of specific messages is left to either RTP Payload Format specifications or to additional feedback format documents.

Application layer feedback messages provide a means to transparently convey feedback from the receiver's to the sender's application. The information contained in such a message is not expected to be acted upon at the transport/RTP or the codec layer. The data to be exchanged between two application instances is usually defined in the application protocol's specification and thus can be identified by the application so that there is no need for additional external information. Hence, this document defines only a common header to be used along with all application layer feedback messages. From a protocol point of view, an application layer feedback message is treated as a special case of a payload-specific feedback message.

This document defines two transport layer feedback and three (video) payload-specific feedback messages as well as a container for application layer feedback messages. Additional transport layer and payload specific feedback messages may be defined in other documents and are registered through IANA (see section IANA considerations).

The general syntax and semantics for the above RTCP feedback message types is described in the following subsections.

4.1 Common Packet Format for Feedback Message

All feedback message share a common packet format that is depicted in figure 3:

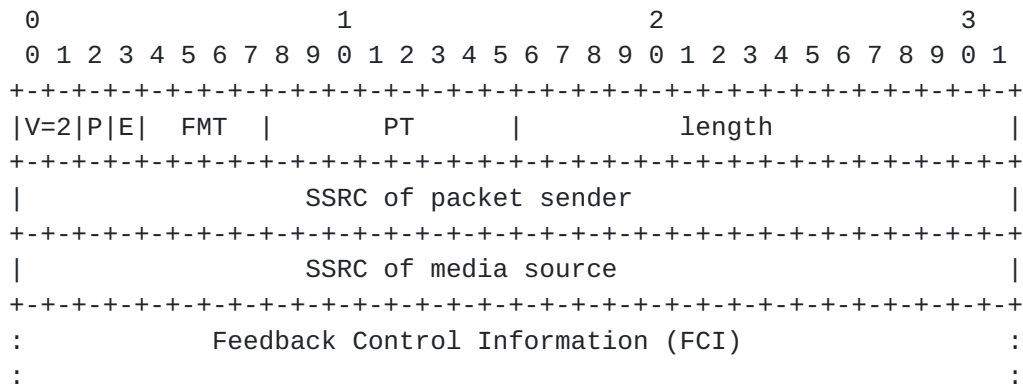


Figure 3: Common Packet Format for Feedback Messages

The various fields V, P, SSRC and length are defined in the RTP specification [2], the respective meaning being summarized below:

version (V): 2 bits

This field identifies the RTP version. The current version is 2.

padding (P): 1 bit

If set, the padding bit indicates that the packet contains additional padding octets at the end which are not part of the control information but are included in the length field.

Early RTCP (E): 1 bit

This bit MUST be set if the packet is sent as an Immediate Feedback or as an Early RTCP packet.

Feedback message type (FMT): 4 bits

This field identifies the type of the feedback message and is interpreted relative to the RTCP message type (transport, payload-specific, or application feedback). The values for each of the three feedback types are defined in the respective sections below.

Payload type (PT): 8 bits

This is the RTCP packet type which identifies the packet as being an RTCP Feedback Message. Two values are defined (TBA. By IANA):

Name	Value	Brief Description
-----+-----+-----		
RTPFB	2xx	Transport layer feedback message
PSFB	2xy	Payload-specific feedback message

Length: 16 bits

The length of this packet in 32-bit words minus one, including the header and any padding. This is in line with the definition of the length field used in RTCP sender and receiver reports [3].

SSRC of packet sender: 32 bits

The synchronization source identifier for the originator of this packet.

SSRC of media source: 32 bits

The synchronization source identifier of the media source that this piece of feedback information is related to.

Feedback Control Information (FCI): variable length

The following three sections define which additional information is included in the feedback message for each type of feedback.

4.2 Transport Layer Feedback Messages

Transport Layer Feedback messages are identified by the value RTPFB as RTCP message type.

Two general purpose transport layer feedback messages are defined so far: General ACK and General NACK. They are identified by means of the FMT parameter as follows:

```

0:    forbidden
1:    General NACK
2:    General ACK
3-15: reserved

```

The following two subsections define the packet formats for these messages.

4.2.1 Generic NACK

The Generic NACK message is identified by PT=RTPFB and FMT=1.

The Generic NACK packet is used to indicate the loss of one or more RTP packets. The lost packet(s) are identified by the means of a packet identifier and a bit mask.

The Feedback control information (FCI) field has the following Syntax (figure 4):

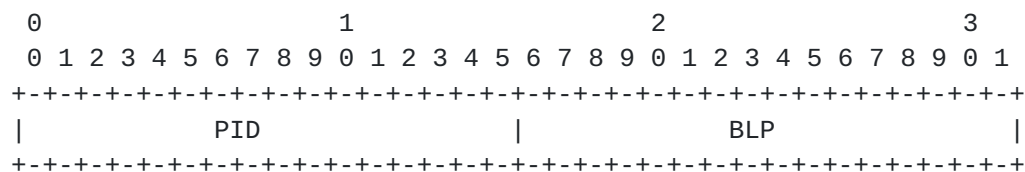


Figure 4: Syntax for the Generic NACK message

Packet ID (PID): 16 bits

The PID field is used to specify a lost packet. Typically, the RTP sequence number is used for PID as the default format, but

RTP Payload Formats may decide to identify a packet differently.

bitmask of following lost packets (BLP): 16 bits

The BLP allows for reporting losses of any of the 16 RTP packets immediately following the RTP packet indicated by the PID. The BLP's definition is identical to that given in [\[10\]](#). Denoting the BLP's least significant bit as bit 1, and its most

significant bit as bit 16, then bit *i* of the bit mask is set to 1 if the sender has not received RTP packet number PID+*i* (modulo 2¹⁶) and the receiver decides this packet is lost; bit *i* is set to 0 otherwise. Note that the sender **MUST NOT** assume that a receiver has received a packet because its bit mask was set to 0. For example, the least significant bit of the BLP would be set to 1 if the packet corresponding to the PID and the following packet have been lost. However, the sender cannot infer that packets PID+2 through PID+16 have been received simply because bits 2 through 15 of the BLP are 0; all the sender knows is that the receiver has not reported them as lost at this time.

4.2.2 Generic ACK

The Generic ACK message is identified by PT=RTPFB and FMT=2.

The Generic ACK packet is used to indicate that one or several RTP packets were received correctly. The received packet(s) are identified by the means of a packet identifier and a bit mask. ACKing of a range of consecutive packets is also possible.

The Feedback control information (FCI) field has the following syntax:

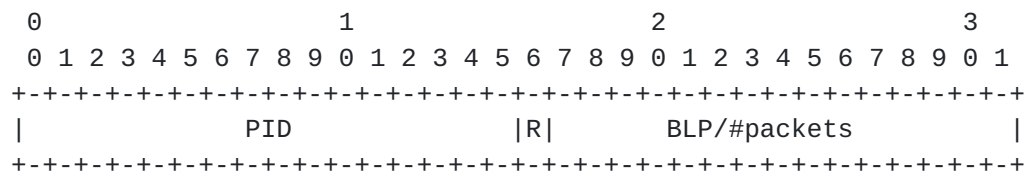


Figure 5: Syntax for the Generic ACK message

Packet ID (1st PID): 16 bits

This PID field is used to specify a correctly received packet. Typically, the RTP sequence number is used for PID as the default format, but RTP Payload Formats may decide to identify a packet differently.

Range of ACKs (R): 1 bit

The R-bit indicates that a range of consecutive packets are received correctly. If R=1 then the PID field specifies the first packet of that range and the next field (BLP/#packets) will carry the number of packets being acknowledged. If R=0 then PID specifies the first packet to be acknowledged and BLP/#packets provides a bit mask to selectively indicate individual packets that are acknowledged.

Bit mask of lost packets (BLP)/#packets (PID): 15 bits

The semantics of this field depends on the value of the R-bit.

If R=1, this field is used to identify the number of additional packets of to be acknowledged:

$\#packets = \langle \text{highest seq\# to be ACKed} \rangle - \langle \text{PID} \rangle$

That is, #packets MUST indicate the number of packet to be ACKed minus one. In particular, if only a single packet is to be ACKed and R=1 then #packets MUST be set to 0x0000.

Example: If all packets between and including PIDx=380 and PIDy = 422 have been received, the Generic ACK would contain PID = PIDx = 380 and #packets = PIDy - PID = 42. In case the PID wraps around, modulo arithmetic is used to calculate the number of packets.

If R=0, this field carries a bit mask. The BLP allows for reporting reception of any of the 15 RTP packets immediately following the RTP packet indicated by the PID. The BLP's definition is identical to that given in [\[10\]](#) except that, here, BLP is only 15 bits wide. Denoting the BLP's least significant bit as bit 1, and its most significant bit as bit 15, then bit *i* of the bitmask is set to 1 if the sender has received RTP packet number PID+i (modulo 2¹⁶) and the receiver decides to ACK this packet; bit *i* is set to 0 otherwise. If only the packet indicated by PID is to be ACKed and R=0 then BLP MUST be set to 0x0000.

4.3 Payload Specific Feedback Messages

Payload-Specific Feedback Messages are identified by the value PSFB as RTCP message type.

Three payload-specific feedback messages are defined so far. They are identified by means of the FMT parameter as follows:

- 0: forbidden
- 1: Picture Loss Indication (PLI)
- 2: Slice Lost Indication (SLI)
- 3: Reference Picture Selection Indication (RPSI)
- 4-14: reserved
- 15: Application layer feedback message

The following subsections define the packet formats for these messages.

4.3.1 Picture Loss Indication (PLI)

The PLI feedback message is identified by PT=PSFB and FMT=1.

4.3.1.1 Semantics

With the Picture Loss Indication message a decoder informs the

encoder about the loss of one or more full pictures.

4.3.1.2 Message Format

PLI does not require parameters. Therefore, the length field MUST be 2, and there MUST NOT be any Feedback Control Information.

4.3.1.3 Timing Rules

The timing follows the rules outlined in [section 3](#). In systems that employ both PLI and other types of feedback it may be advisable to follow the regular RTCP RR timing rules for PLI, since PLI is not as delay critical as other FB types.

4.3.1.4 Remarks

PLI messages typically trigger the sending of full Intra pictures. Intra Pictures are several times larger than predicted (Inter) pictures. Their size is independent of the time they are generated. In most environments, especially when employing bandwidth-limited links, the use of an Intra picture implies an allowed delay that is a significant multitude of the typical frame duration. An example: If the sending frame rate is 10 fps, and an Intra picture is assumed to be 10 times as big as an Inter picture (not an unrealistic assumption, see [\[14\]](#) for details), then a full second of latency has to be accepted. In such an environment there is no need for a particular short delay in sending the feedback message. Hence waiting for the next possible time slot allowed by RTCP timing rules as per [\[2\]](#) does not have a negative impact on the system performance.

4.3.2 Slice Lost Indication (SLI)

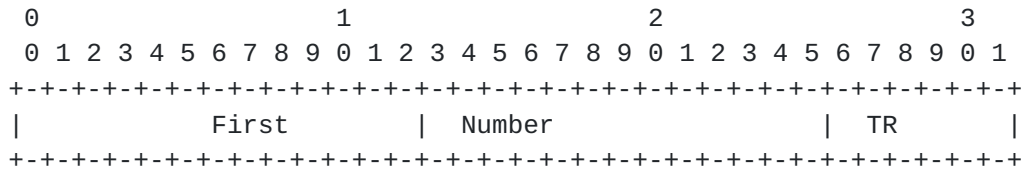
The SLI feedback message is identified by PT=PSFB and FMT=2.

4.3.2.1 Semantics

With the Slice Lost Indication a decoder can inform an encoder that it was unable to decode one, or several consecutive, macroblocks. The encoder can take appropriate action in order to re-synchronize encoder and decoder by means of its choice, typically by sending the lost macroblocks in Intra mode. This feedback message SHALL NOT be used for video codecs with non-uniform, dynamically changeable macroblock sizes such as H.263 with enabled Annex Q. In such a case, an encoder cannot always identify the corrupted spatial region.

4.3.2.2 Format

When FBT indicates a Slice Lost Indication, then there is one additional PCI field the content of which is depicted in figure 6. The length of the feedback message MUST be set to 3.



macroblocks to all those areas, regardless of the timing of the FB (see H.263 (2000) [Appendix I \[13\]](#)] and [\[15\]](#)). While, when those algorithms are used, the timing of the FB is less critical than without, it has to be observed that those algorithms correct large parts of the picture and, therefore, have to transmit many more bits in case of delayed FBs.

4.3.3 Reference Picture Selection Indication (RPSI)

The RPSI feedback message is identified by PT=PSFB and FMT=3.

4.3.3.1 Semantics

Modern video coding standards such as MPEG-4 visual version 2 [[12](#)] or H.263 version 2 [[13](#)] allow the use of older reference pictures than the most recent one. Typically, a first-in-first-out queue of reference pictures is maintained. If an encoder has learned about a loss of encoder-decoder synchronicity, a known-as-correct reference picture can be used. As this reference picture is temporally further away than usual, the resulting predictively coded picture will use more bits.

Both MPEG-4 and H.263 define a binary format for the payload of an RPSI message that includes information such as the temporal ID of the damaged picture and the size of the damaged region. This bit string is typically small - a couple of dozen bits -, of variable length, and self-contained, i.e. contains all information that is necessary to perform reference picture selection.

Note that both MPEG-4 and H.263 allow the use of RPSI with positive feedback information as well. That is, all corrected pictures are reported. Any form of positive feedback **MUST NOT** be used when in a multicast environment (reporting positive feedback about individual reference pictures at RTCP intervals is not expected to be of much use anyway). For point-to-point communication, positive feedback **MAY** be used but, again, the bit rate budget of RTCP feedback will prevent the use in most scenarios anyway.

4.3.3.2 Format

When FB indicates an RPSI, then the length field is set to the number of bits of the following bit string that contains the RPS information. This bit string follows byte aligned in the UCI field. Bit padding is used to achieve 32-bit word alignment of the UCI message (and the whole packet).

4.3.3.3 Timing Rules

RPS is even more critical to delay than algorithms using SLI. This is due to the fact that the older the RPS message is, the more bits the encoder has to spend to achieve encoder-decoder synchronicity. See [[14](#)] and [[15](#)] for some information about the overhead of RPS for certain bit rate/frame rate/loss rate scenarios.

Therefore, RPS messages should typically be sent as soon as possible, employing the algorithm of [section 3](#).

4.4 Application Layer Feedback Messages

Payload-Specific Feedback Messages are a special case of payload-specific messages and identified by PT=PSFB and FMT=15.

These messages are used to transport application defined data directly from the receiver's to the sender's application. The data that is transported is not identified by the feedback message. Therefore the application must be able to identify the messages payload.

Usually applications define their own set of messages, e.g. NEWPRED messages in MPEG-4 or feedback messages in H.263/Annex N,U. These messages do not need any additional information from the RTCP message. Thus the application message is simply placed into the FCI field as follows and the length field is set accordingly.

Application Message (FCI): variable length

This field contains the original application message that should be transported from the receiver to the source. The format is application dependent. The length of this field is variable. If the application data is not four-byte aligned, padding must be added.

As there is no need for additional identification at the RTCP level, the FMT field is unused and MUST be set to zero:

5. Early Feedback and Congestion Control

In the previous sections, the feedback messages were defined as well as the timing rules according to which to send these messages. The way to react to the feedback received depends on the application using the feedback mechanisms and hence is beyond the scope of this document.

However, across all applications, there is a common requirement for (TCP-friendly) congestion control on the media stream as defined in [1] and [2] when operating in a best-effort network environment.

Low delay feedback supports the use of congestion control algorithms in two ways:

- . The potentially more frequent RTCP messages allow the sender to monitor the network state more closely than with regular RTCP and therefore enable reacting to upcoming congestion in a more

timely fashion.

- . The feedback messages themselves may convey additional information as input to congestion control algorithms and thus improve reaction over conventional RTCP. (For example, ACK-based feedback may even allow to construct closed loop algorithms and

NACK-based systems may provide further information on the packet loss distribution.)

A congestion control algorithm that shares the available bandwidth fair with competing TCP connections, e.g. TFRC [16], SHOULD be used to determine the data rate for the media stream (if the low delay RTP session is transmitted in a best effort environment).

RTCP feedback messages or RTCP SR/RR packets that indicate recent packet loss MUST NOT lead to a (mid-term) increase in the transmission data rate and SHOULD lead to a (short-term) decrease of the transmission data rate. Such messages SHOULD cause the sender to adjust the transmission data rate to the order of the throughput TCP would achieve under similar conditions (e.g. using TFRC).

RTCP feedback messages or RTCP SR/RR packets that indicate no recent packet loss MAY cause the sender to increase the transmission data rate to roughly the throughput TCP would achieve under similar conditions (e.g. using TFRC).

6. Security Considerations

RTP packets transporting information with the proposed payload format are subject to the security considerations discussed in the RTP specification [1]. This implies that confidentiality of the media streams is achieved by encryption.

If the entire stream (extension data and AU data) is to be secured and all the participants are expected to have the keys to decode the entire stream, then the encryption is performed in the usual manner, and there is no conflict between the two operations (encapsulation and encryption).

The need for a portion of stream (e.g. extension data) to be encrypted with a different key, or not to be encrypted, would require application level signaling protocols to be aware of the usage of the XT field, and to exchange keys and negotiate their usage on the media and extension data separately.

7. IANA Considerations

The feedback profile as an extension to the profile for audio-visual conferences with minimal control needs to be registered: "AVPF".

For the Session Description Protocol, the following "fmp:" attribute needs to be registered: "rtcp-fb".

Along with "rtcp-fb", the feedback types "ack" and "nack" need to be registered.

Along with "nack", the feedback type parameters "sli", "pli", and "rpsi" need to be registered.

Two RTCP Control Packet Types: for the class of transport layer feedback messages ("RTPFB") and for the class of payload-specific feedback messages ("PSFB").

Within the RTPFB range, three format (FMT) values need to be registered:

- 0: forbidden
- 1: General NACK
- 2: General ACK

Within the PSFB range, five format (FMT) values need to be registered:

- 0: forbidden
- 1: Picture Loss Indication (PLI)
- 2: Slice Loss Indication (SLI)
- 3: Reference Picture Selection Indication (SLI)
- 15: Application layer feedback (AFB)

8. Acknowledgements

This document is a product of the Audio-Visual Transport (AVT) Working Group of the IETF. The authors would like to thank Steve Casner and Colin Perkins for their comments and suggestions as well as for their responsiveness to numerous questions.

9. Full Copyright Statement

Copyright (C) The Internet Society (2001). All Rights Reserved.

This document and translations of it may be copied and furnished to others, and derivative works that comment on or otherwise explain it or assist in its implementation may be prepared, copied, published and distributed, in whole or in part, without restriction of any kind, provided that the above copyright notice and this paragraph are included on all such copies and derivative works.

However, this document itself may not be modified in any way, such as by removing the copyright notice or references to the Internet Society or other Internet organizations, except as needed for the purpose of developing Internet standards in which case the procedures for copyrights defined in the Internet Standards process must be followed, or as required to translate it into languages other than English.

The limited permissions granted above are perpetual and will not be revoked by the Internet Society or its successors or assigns.

This document and the information contained herein is provided on an "AS IS" basis and THE INTERNET SOCIETY AND THE INTERNET ENGINEERING TASK FORCE DISCLAIMS 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."

10. Authors' Addresses

J÷rg Ott {sip,mailto}:jo@tzi.uni-bremen.de
Universit t Bremen TZI
MZH 5180
Bibliothekstr. 1
D-28359 Bremen
Germany

Stephan Wenger stewe@cs.tu-berlin.de
TU Berlin
Skr. FR 6-3
Franklinstr. 28-29
D-10587 Berlin
Germany

Shigeru Fukunaga
Oki Electric Industry Co., Ltd.
1-2-27 Shiromi, Chuo-ku, Osaka 540-6025 Japan
Tel. +81 6 6949 5101
Fax. +81 6 6949 5108
Mail fukunaga444@oki.co.jp

Noriyuki Sato
Oki Electric Industry Co., Ltd.
1-2-27 Shiromi, Chuo-ku, Osaka 540-6025 Japan
Tel. +81 6 6949 5101
Fax. +81 6 6949 5108
Mail sato652@oki.co.jp

Koichi Yano
FastForward Networks,
75 Hawthorne St. #601
San Francisco, CA 94105
Tel. +1.415.430.2500

Akihiro Miyazaki
Matsushita Electric Industrial Co., Ltd
1006, Kadoma, Kadoma City, Osaka, Japan
Tel. +81-6-6900-9192
Fax. +81-6-6900-9193
Mail akihiro@isl.mei.co.jp

Koichi Hata
Matsushita Electric Industrial Co., Ltd
1006, Kadoma, Kadoma City, Osaka, Japan
Tel. +81-6-6900-9192
Fax. +81-6-6900-9193

Mail hata@isl.mei.co.jp

Ott et al.

Expires January 2002

[Page 29]

Rolf Hakenberg
Panasonic European Laboratories GmbH
Monzastr. 4c, 63225 Langen, Germany
Tel. +49-(0)6103-766-162
Fax. +49-(0)6103-766-166
Mail hakenberg@panasonic.de

Carsten Burmeister
Panasonic European Laboratories GmbH
Monzastr. 4c, 63225 Langen, Germany
Tel. +49-(0)6103-766-263
Fax. +49-(0)6103-766-166
Mail burmeister@panasonic.de

11. Bibliography

- [1] H. Schulzrinne, S. Casner, R. Frederick, and V. Jacobson, "RTP - A Transport Protocol for Real-time Applications," Internet Draft, [draft-ietf-avt-rtp-new-09.txt](#), Work in Progress, March 2001.
- [2] H. Schulzrinne and S. Casner, "RTP Profile for Audio and Video Conferences with Minimal Control," Internet Draft [draft-ietf-avt-profile-new-10.txt](#), March 2001.
- [3] M. Handley and V. Jacobson, "SDP: Session Description Protocol", [RFC 2327](#), April 1998.
- [4] S. Casner, "SDP Bandwidth Modifiers for RTCP Bandwidth", Internet Draft [draft-ietf-avt-rtcp-bw-03.txt](#), March 2001.
- [5] C. Perkins and O. Hodson, "2354 Options for Repair of Streaming Media," [RFC 2354](#), June 1998.
- [6] J. Rosenberg and H. Schulzrinne, "An RTP Payload Format for Generic Forward Error Correction," [RFC 2733](#), December 1999.
- [7] C. Perkins, I. Kouvelas, O. Hodson, V. Hardman, M. Handley, J.C. Bolot, A. Vega-Garcia, and S. Fosse-Parisis, "RTP Payload for Redundant Audio Data," [RFC 2198](#), September 1997.
- [8] S. Bradner, "Key words for use in RFCs to Indicate Requirement Levels," [RFC 2119](#), March 1997.
- [9] H. Schulzrinne and S. Petrack, "RTP Payload for DTMF Digits, Telephony Tones and Telephony Signals," [RFC 2833](#), May 2000.
- [10] T. Turletti and C. Huitema, "RTP Payload Format for H.261 Video Streams," [RFC 2032](#), October 1996.

- [11] C. Bormann, L. Cline, G. Deisher, T. Gardos, C. Maciocco, D. Newell, J. Ott, G. Sullivan, S. Wenger, and C. Zhu, "RTP Payload Format for the 1998 Version of ITU-T Rec. H.263 Video (H.263+)," [RFC 2429](#), October 1998.

- [12] ISO/IEC 14496-2:1999/Amd.1:2000, "Information technology - Coding of audio-visual objects - Part2: Visual", July 2000.
- [13] ITU-T Recommendation H.263, "Video Coding for Low Bit Rate Communication," November 2000.
- [14] S. Wenger, "Media-aware Protocols -- transport aware Media Coding," Habilitation thesis, in preparation, 2001.
- [15] B. Girod, N. Faerber, "Feedback-based error control for mobile video transmission," Proceedings IEEE, Vol. 87, No. 10, pp. 1707-1723, October, 1999.
- [16] M. Handley, J. Padhye, S. Floyd, J. Widmer, "TCP friendly Rate Control (TFRC): Protocol Specification," Internet Draft, [draft-ietf-tsvwg-02.txt](#), Work in Progress, May 2001.

[Appendix A](#). Some Background and Motivation (Informative)

A.1 Example: Predictive Video Coding

A.1.1 Video Encoder-decoder synchronicity

Most current video coding schemes for compressed video, such as the ITU-T H.261 and H.263 and ISO/IEC MPEG[124] employ a mechanism known as Inter Picture Prediction. Each picture is divided into macroblocks of uniform size. For each macroblock, one or more motion vectors may be identified and transmitted. The residual signal after motion compensation is DCT-transformed, quantized, entropy coded, and transmitted as well. The encoder reconstructs, based on this information, a so-called reference picture, which is used to perform the motion compensation and residual signal coding steps for the subsequent picture. Since the reference picture is generated using only such information that is also available at the decoder, the reference picture is identical to the reconstructed picture at the decoder. Having identical reference pictures at the encoder and decoder is referred to as encoder-decoder-synchronicity.

Whenever data is damaged or lost on the way between the encoder and the decoder, the reconstructed picture at the decoder is no more identical with the encoder's reference picture -- the encoder-decoder synchronicity is lost.

Any loss of the encoder-decoder synchronicity results in annoying artifacts at the decoder. Because the prediction of subsequent pictures in the decoder is based on a damaged reference picture, the annoying artifacts are present not only in the picture in which the loss occurred; they propagate to all subsequent pictures, until, through source coding based mechanisms, the encoder-decoder synchronicity is restored. Therefore, the goal of systems employing predictive video coding in a lossy environment must be to keep the encoder-decoder synchronicity, or, if this is not possible, to regain that synchronicity as quickly as possible.

A.1.2. Non-feedback based mechanisms

Avoiding the loss of the encoder-decoder synchronicity corresponds to avoiding the loss of coded picture data. Such a task can be performed on the transport layer. In RTP environments, the use of packet-based FEC is a good example for such a technique. (The use of TCP or reliable multicast as the transport for media streams would be an even better one but is inappropriate for low-delay (interactive) real-time systems.) FEC schemes, interleaving, and other means for repairing real-time media streams may also add additional delay and significant bit rate overhead without being able to guarantee

compensation of virtually all packet losses.

Once the encoder-decoder synchronicity is lost, only source coding oriented mechanisms can help to regain it. One common way is to send a non-predictively coded picture (known as Intra picture). Intra pictures have the disadvantage of being several times bigger than predictively coded pictures (Inter pictures). Therefore, sending Intra pictures has negative implications both on the bandwidth and

(in bandwidth limited environments) delay. Another way is to use Intra macroblock refresh. Here, certain parts of the picture (those affected by a packet loss) are coded non-predictively in order to resynchronize the encoder and decoder over time. Intra macroblock refresh has better delay characteristics than full Intra pictures because the picture size can be kept constant, but is less efficient in terms of bit rate/distortion than full Intra pictures. More sophisticated means such as Reference Picture Selection (RPS) are also available in modern video coding standards.

Systems not employing feedback channels may use any combination of the mechanisms described above to add error resilience -- at the cost of added bit rate and, sometimes, added delay. The number of additional bits spent for error resilience can be adapted using the long-term packet loss rate information in the RTCP receiver reports. But, even when using such adaptive means, it is still likely that systems spend many more bits than theoretically necessary to achieve error resilience in order to be on the safe side. Plus, as regular RTCP feedback is aimed at longer terms, reactivity to sudden losses is limited. In all practical applications today this means that fewer bits are available for non redundant picture data, and hence the overall picture quality suffers.

A.1.3 Feedback based systems

Feedback-based systems try to avoid spending too many bits for redundant information by informing the encoder about a loss situation at the decoder(s). The encoder can then react accordingly and spend redundant bits only when needed possibly only for the part of the picture that was effected by the loss -- thereby reducing the number of redundant bits and leaving more bits for useful information. As a result, a higher reproduced picture quality can generally be expected when feedback channels are available.

Similar to the observations of [section 2.1.2](#), transport and source coding based mechanisms can be distinguished that react on loss situations reported by feedback.

Transport based systems employing feedback react media unaware, by re-transmitting lost packets. TCP is a good example for a protocol following such a scheme. Transport-based feedback in real-time and/or multicast environments is a complex matter and subject of a lot of engineering and research in and outside of the IETF. This specification is not concerned with pure transport-based feedback.

Source coding based mechanisms may react upon the arrival of a feedback message indicating a loss situation by adding bits that restore, or at least make an effort to restore, the encoder-decoder

synchronicity. This process has to be performed by a real-time encoder. However, schemes were reported, that allow the use of feedback also for non-real-time encoders by storing multiple representations of the same data (e.g. Inter and Intra coded), and dynamically switching between those representations.

Several types of feedback messages, called Feedback Messages or FB messages, can be defined for such a case. An FB message can be as

simple as a Boolean condition, indicating for example the loss of a full picture (and, therefore, the need of a full Intra picture transmission). Other feedback messages may contain more complex information such as information about the damage of a spatial region of the picture. A special form consists of a message the format and semantics of which are not known at the transport level, because they are defined in the video codec standards.

A.2 Feedback Messages

Most FB messages contain negative acknowledge information, indicating an erroneous situation at the decoder. In others, the nature of the acknowledge (positive, negative, or both) is part of the feedback message itself. When used in multicast environments, positive acknowledge must not be used.

This document assumes that feedback messages are transmitted using RTCP packets. RTCP messages from the receivers to the sender cannot be sent at any possible time, in order to prevent traffic explosion in case of large multicast groups. Instead, the bit rate for all RTCP messages of all receivers together has to obey a maximum fraction of the total RTP session bit rate, yielding a very limited bit rate budget for a single receiver when having a large multicast group. This, in turn, leads to an increased average delay when the size of the receiving multicast group grows. (see section 6 of [1] for details)

This specification defines an algorithm that adheres to the bit rate limitations for the feedback channel on the long term, but allows short-term overdrafting for any receiver (but not all of them simultaneously). Thus, the algorithm allows for better real-time performance than the one specified in [1]. Traffic explosion in such cases in which many receivers identify a picture damage simultaneously is prevented by dithering.

As this specification assumes a sender that has full control over its transmission bit rate (e.g. a real-time encoder), there is no scaling problem on the forward channel. Any reaction to negative feedback generates additional bits, which have to be conveyed but this is taken from the sender's total bit rate budget. The encoder can take this into account by, for example, changing the encoding mode, packet size, and so forth. The sender is also free to simply ignore feedback messages. Adjusting the tradeoff between the reproduced media quality of all receivers of a multicast group and the amount of additional repair traffic is a media-dependent, very complex task and is not covered in this specification.

Finally, frequent RTCP-based feedback messages may provide additional

input to the sender(s)'s congestion control algorithms and thus improve its reactivity towards network congestion.

Feedback messages as well as sender and receiver behavior are to be specified in separate documents (such as [\[7\]](#)). Such specifications need to consider that, frequently, packet loss is an indication of network congestion and thus define mechanisms for media-specific

congestion control in the presence of feedback as defined in this memo.

A.3. Applications and Relationships to other Standards

This specification is based on RTCP, which implies its use in an RTP environment. RTP itself is used in a variety of systems such as in SIP- or H.323-based multimedia conferencing/telephony, SAP-announced Mbone conferences, and RTSP-based media streaming.

As for the video codecs, there is currently a small set of standards that are, for the purpose of this discussion, roughly comparable. Many mechanisms for regaining encoder-decoder synchronicity are applicable to all video codecs. Others require certain tools (such as Reference Picture Selection, aka NEWPRED) that are available only in certain versions of the standards, and/or optional tools whose use must be negotiated prior to being used.

A few RTP payload specifications such as [RFC 2032](#) [10] already define a feedback mechanism for some of the coding algorithms considered in this specification. An application capable of performing both schemes MUST use the feedback mechanism defined in this specification, although, for backward compatibility reasons, it MUST also be capable to conform to the feedback scheme defined in the respective RTP payload format, if this is required by that payload format.

Also, audio, DTMF, and text streams could benefit from more immediate feedback even though the redundancy payload formats work well for these media.

All kinds of non-interactive media streams (such as RTSP-controlled media streaming applications) could benefit significantly as without interactivity there is more time available for media repair.

A.4 Remarks on the size of the multicast group

This specification prevents traffic explosion on the feedback channel in a very similar way as RTP does, with the exception of allowing individual receivers to overdraft their bit rate budget from time to time. This is necessary in order to allow for low delay, which is needed by the algorithms reacting to Feedback messages.

This scaling, however, limits the usefulness of this mechanism in multicast groups from a certain size upwards (where the size threshold depends on a number of parameters including loss rate, frame rate, number of packets per frame, and session bandwidth). The maximum size of the multicast group is soft and also depends on

application requirements and is therefore not specified here.
Considerations on the multicast group sizes are presented in [section 3.5](#).