

Payload Working Group  
Internet-Draft  
Intended status: Standards Track  
Expires: April 21, 2016

J. Lennox  
D. Hong  
Vidyo  
J. Uberti  
S. Holmer  
M. Flodman  
Google  
October 19, 2015

The Layer Refresh Request (LRR) RTCP Feedback Message  
draft-ietf-avtext-lrr-01

Abstract

This memo describes the RTCP Payload-Specific Feedback Message "Layer Refresh Request" (LRR), which can be used to request a state refresh of one or more substreams of a layered media stream. It also defines its use with several scalable media formats.

Status of This Memo

This Internet-Draft is submitted in full conformance with the provisions of BCP 78 and BCP 79.

Internet-Drafts are working documents of the Internet Engineering Task Force (IETF). Note that other groups may also distribute working documents as Internet-Drafts. The list of current Internet-Drafts is at <http://datatracker.ietf.org/drafts/current/>.

Internet-Drafts are draft documents valid for a maximum of six months and may be updated, replaced, or obsoleted by other documents at any time. It is inappropriate to use Internet-Drafts as reference material or to cite them other than as "work in progress."

This Internet-Draft will expire on April 21, 2016.

Copyright Notice

Copyright (c) 2015 IETF Trust and the persons identified as the document authors. All rights reserved.

This document is subject to BCP 78 and the IETF Trust's Legal Provisions Relating to IETF Documents (<http://trustee.ietf.org/license-info>) in effect on the date of publication of this document. Please review these documents carefully, as they describe your rights and restrictions with respect to this document. Code Components extracted from this document must

include Simplified BSD License text as described in Section 4.e of the Trust Legal Provisions and are provided without warranty as described in the Simplified BSD License.

## Table of Contents

1. Introduction . . . . .	2
2. Conventions, Definitions and Acronyms . . . . .	2
2.1. Terminology . . . . .	3
3. Layer Refresh Request . . . . .	5
3.1. Message Format . . . . .	5
4. Usage with specific codecs . . . . .	6
4.1. H264 SVC . . . . .	7
4.2. VP8 . . . . .	8
4.3. H265 . . . . .	8
4.4. VP9 . . . . .	10
5. Usage with different scalability transmission mechanisms . .	10
6. Security Considerations . . . . .	11
7. IANA Considerations . . . . .	11
8. References . . . . .	11
8.1. Normative References . . . . .	11
8.2. Informative References . . . . .	12
Authors' Addresses . . . . .	12

## 1. Introduction

This memo describes an RTCP [RFC3550] Payload-Specific Feedback Message [RFC4585] "Layer Refresh Request" (LRR). It is designed to allow a receiver of a layered media stream to request that one or more of its substreams be refreshed, such that it can then be decoded by an endpoint which previously was not receiving those layers, without requiring that the entire stream be refreshed (as it would be if the receiver sent a Full Intra Request (FIR) [RFC5104]).

The message is designed to be applicable both to temporally and spatially scaled streams, and to both single-stream and multi-stream scalability modes.

## 2. Conventions, Definitions and Acronyms

The key words "MUST", "MUST NOT", "REQUIRED", "SHALL", "SHALL NOT", "SHOULD", "SHOULD NOT", "RECOMMENDED", "MAY", and "OPTIONAL" in this document are to be interpreted as described in [RFC2119].

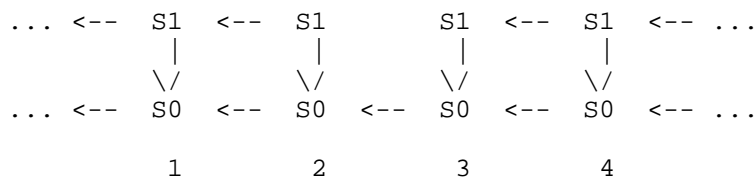
## 2.1. Terminology

A "Layer Refresh Point" is a point in a scalable stream after which a decoder, which previously had been able to decode only some (possibly none) of the available layers of stream, is able to decode a greater number of the layers.

For spatial (or quality) layers, layer refresh typically requires that a spatial layer be encoded in a way that references only lower-layer subpictures of the current picture, not any earlier pictures of that spatial layer. Additionally, the encoder must promise that no earlier pictures of that spatial layer will be used as reference in the future.

In a layer refresh, however, other layers than the ones requested for refresh may still maintain dependency on earlier content of the stream. This is the difference between a layer refresh and a Full Intra Request [RFC5104]. This minimizes the coding overhead of refresh to only those parts of the stream that actually need to be refreshed at any given time.

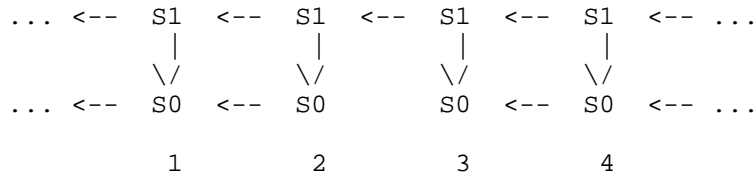
An illustration of spatial layer refresh of an enhancement layer is shown below.



In this illustration, frame 3 is a layer refresh point for spatial layer S1; a decoder which had previously only been decoding spatial layer S0 would be able to decode layer S1 starting at frame 3.

Figure 1

An illustration of spatial layer refresh of a base layer is shown below.

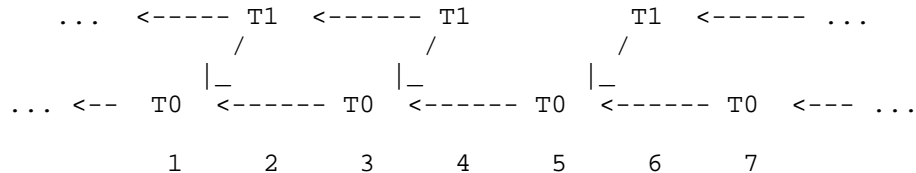


In this illustration, frame 3 is a layer refresh point for spatial layer S0; a decoder which had previously not been decoding the stream at all could decode layer S0 starting at frame 3.

Figure 2

For temporal layers, layer refresh requires that the layer be "temporally nested", i.e. use as reference only earlier frames of a lower temporal layer, not any earlier frames of this temporal layer, and also promise that no future frames of this temporal layer will reference frames of this temporal layer before the refresh point. In many cases, the temporal structure of the stream will mean that all frames are temporally nested, in which case decoders will have no need to send LRR messages for the stream.

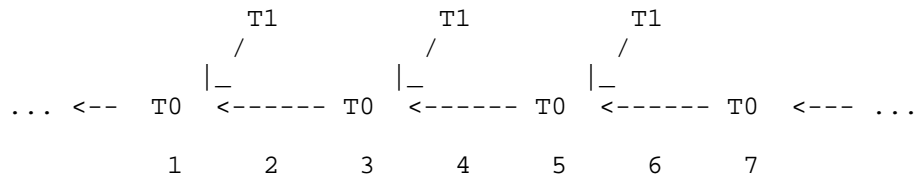
An illustration of temporal layer refresh is shown below.



In this illustration, frame 6 is a layer refresh point for temporal layer T1; a decoder which had previously only been decoding temporal layer T0 would be able to decode layer T1 starting at frame 6.

Figure 3

An illustration of an inherently temporally nested stream is shown below.



In this illustration, the stream is temporally nested in its ordinary structure; a decoder receiving layer T0 can begin decoding layer T1 at any point.

Figure 4

### 3. Layer Refresh Request

A layer refresh frame can be requested by sending a Layer Refresh Request (LRR), which is an RTCP payload-specific feedback message [RFC4585] asking the encoder to encode a frame which makes it possible to upgrade to a higher layer. The LRR contains one or two tuples, indicating the layer the decoder wants to upgrade to, and (optionally) the currently highest layer the decoder can decode.

The specific format of the tuples, and the mechanism by which a receiver recognizes a refresh frame, is codec-dependent. Usage for several codecs is discussed in Section 4.

LRR follows the model of the Full Intra Request (FIR) [RFC5104](Section 3.5.1) for its retransmission, reliability, and use in multipoint conferences. TODO: expand these here.

The LRR message is identified by RTCP packet type value PT=PSFB and FMT=TBD. The FCI field MUST contain one or more FIR entries. Each entry applies to a different media sender, identified by its SSRC.

#### 3.1. Message Format

The Feedback Control Information (FCI) for the Layer Refresh Request consists of one or more FCI entries, the content of which is depicted in Figure 5. The length of the LRR feedback message MUST be set to  $2+3*N$ , where N is the number of FCI entries.

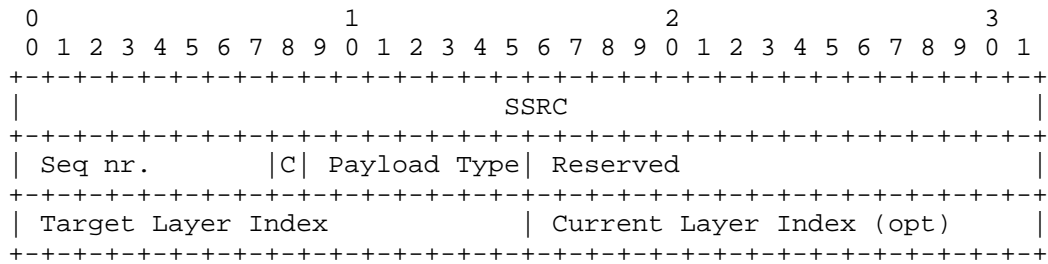


Figure 5

SSRC (32 bits) The SSRC value of the media sender that is requested to send a layer refresh point.

Seq nr. (8 bits) Command sequence number. The sequence number space is unique for each pairing of the SSRC of command source and the SSRC of the command target. The sequence number SHALL be increased by 1 modulo 256 for each new command. A repetition SHALL NOT increase the sequence number. The initial value is arbitrary.

C (1 bit) A flag bit indicating whether the "Current Layer Index" field is present in the FCI. If this bit is false, the sender of the LRR message is requesting refresh of all layers up to and including the target layer.

Payload Type (7 bits) The RTP payload type for which the LRR is being requested. This gives the context in which the target layer index is to be interpreted.

Reserved (16 bits) All bits SHALL be set to 0 by the sender and SHALL be ignored on reception.

Target Layer Index (16 bits) The target layer for which the receiver wishes a refresh point. Its format is dependent on the payload type field.

Current Layer Index (16 bits) If C is 1, the current layer being decoded by the receiver. This message is not requesting refresh of layers at or below this layer. If C is 0, this field SHALL be set to 0 by the sender and SHALL be ignored on reception.

#### 4. Usage with specific codecs

#### 4.1. H264 SVC

H.264 SVC [RFC6190] defines temporal, dependency (spatial), and quality scalability modes.

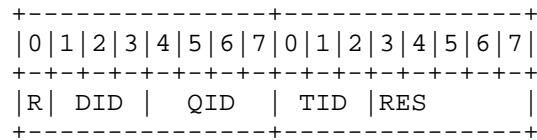


Figure 6

Figure 6 shows the format of the layer index field for H.264 SVC streams. This is designed to follow the same layout as the third and fourth bytes of the H.264 SVC NAL unit extension, which carry the stream's layer information. The "R" and "RES" fields MUST be set to 0 on transmission and ignored on reception. See [RFC6190] Section 1.1.3 for details on the DID, QID, and TID fields.

A dependency or quality layer refresh of a given layer in H.264 SVC can be identified by the "I" bit (idr\_flag) in the extended NAL unit header, present in NAL unit types 14 (prefix NAL unit) and 20 (coded scalable slice). Layer refresh of the base layer can also be identified by its NAL unit type of its coded slices, which is "5" rather than "1". A dependency or quality layer refresh is complete once this bit has been seen on all the appropriate layers (in decoding order) above the current layer index (if any, or beginning from the base layer if not) through the target layer index.

Note that as the "I" bit in a PACSI header is set if the corresponding bit is set in any of the aggregated NAL units it describes; thus, it is not sufficient to identify layer refresh when NAL units of multiple dependency or quality layers are aggregated.

In H.264 SVC, temporal layer refresh information can be determined from various Supplemental Encoding Information (SEI) messages in the bitstream.

Whether an H.264 SVC stream is scalably nested can be determined from the Scalability Information SEI message's temporal\_id\_nesting flag. If this flag is set in a stream's currently applicable Scalability Information SEI, receivers SHOULD NOT send temporal LRR messages for that stream, as every frame is implicitly a temporal layer refresh point. (The Scalability Information SEI message may also be available in the signaling negotiation of H.264 SVC, as the sprop-scalability-info parameter.)

If a stream's `temporal_id_nesting` flag is not set, the Temporal Level Switching Point SEI message identifies temporal layer switching points. A temporal layer refresh is satisfied when this SEI message is present in a frame with the target layer index, if the message's `delta_frame_num` refer to a frame with the requested current layer index. (Alternately, temporal layer refresh can also be satisfied by a complete state refresh, such as an IDR.) Senders which support receiving LRR for non-scalably-nested streams MUST insert Temporal Level Switching Point SEI messages as appropriate.

#### 4.2. VP8

The VP8 RTP payload format [I-D.ietf-payload-vp8] defines temporal scalability modes. It does not support spatial scalability.

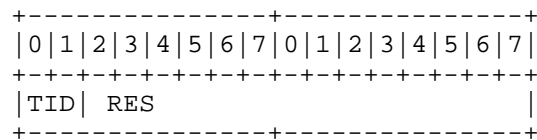


Figure 7

Figure 7 shows the format of the layer index field for VP8 streams. The "RES" fields MUST be set to 0 on transmission and ignored on reception. See [I-D.ietf-payload-vp8] Section 4.2 for details on the TID field.

A VP8 layer refresh point can be identified by the presence of the "Y" bit in the VP8 payload header. When this bit is set, this and all subsequent frames depend only on the current base temporal layer. On receipt of an LRR for a VP8 stream, A sender which supports LRR MUST encode the stream so it can set the Y bit in a packet whose temporal layer is at or below the target layer index.

Note that in VP8, not every layer switch point can be identified by the Y bit, since the Y bit implies layer switch of all layers, not just the layer in which it is sent. Thus the use of LRR with VP8 can result in some inefficiency in transmission. However, this is not expected to be a major issue for temporal structures in normal use.

#### 4.3. H265

The initial version of the H.265 payload format [I-D.ietf-payload-rtp-h265] defines temporal scalability, with protocol elements reserved for spatial or other scalability modes (which are expected to be defined in a future version of the specification).



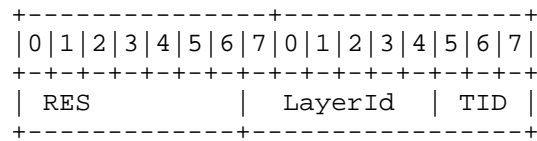


Figure 8

Figure 8 shows the format of the layer index field for H.265 streams. This is designed to follow the same layout as the first and second bytes of the H.265 NAL unit header, which carry the stream's layer information. The "RES" field MUST be set to 0 on transmission and ignored on reception. See [I-D.ietf-payload-rtp-h265] Section 1.1.4 for details on the LayerId and TID fields.

H.265 streams signal whether they are temporally nested, using the `vps_temporal_id_nesting_flag` in the Video Parameter Set (VPS), and the `sps_temporal_id_nesting_flag` in the Sequence Parameter Set (SPS). If this flag is set in a stream's currently applicable VPS or SPS, receivers SHOULD NOT send temporal LRR messages for that stream, as every frame is implicitly a temporal layer refresh point.

If a stream's `sps_temporal_id_nesting_flag` is not set, the NAL unit types 2 to 5 inclusively identify temporal layer switching points. A layer refresh to any higher target temporal layer is satisfied when a NAL unit type of 4 or 5 with TID equal to 1 more than current TID is seen. Alternatively, layer refresh to a target temporal layer can be incrementally satisfied with NAL unit type of 2 or 3. In this case, given current TID = T0 and target TID = TN, layer refresh to TN is satisfied when NAL unit type of 2 or 3 is seen for TID = T1, then TID = T2, all the way up to TID = TN. During this incremental process, layer refresh to TN can be completely satisfied as soon as a NAL unit type of 2 or 3 is seen.

Of course, temporal layer refresh can also be satisfied whenever any Intra Random Access Point (IRAP) NAL unit type (with values 16-23, inclusively) is seen. An IRAP picture is similar to an IDR picture in H.264 (NAL unit type of 5 in H.264) where decoding of the picture can start without any older pictures.

In the (future) H.265 payloads that support spatial scalability, a spatial layer refresh of a specific layer can be identified by NAL units with the requested layer ID and NAL unit types between 16 and 21 inclusive. A dependency or quality layer refresh is complete once NAL units of this type have been seen on all the appropriate layers (in decoding order) above the current layer index (if any, or beginning from the base layer if not) through the target layer index.

#### 4.4. VP9

The RTP payload format for VP9 [I-D.uberti-payload-vp9] defines how it can be used for spatial and temporal scalability.

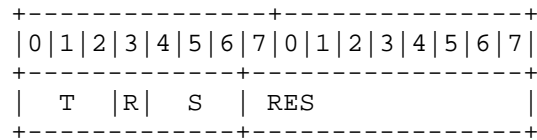


Figure 9

Figure 9 shows the format of the layer index field for VP9 streams. This is designed to follow the same layout as the "L" byte of the VP9 payload header, which carries the stream's layer information. The "R" and "RES" fields MUST be set to 0 on transmission and ignored on reception. See [I-D.uberti-payload-vp9] for details on the T and S fields.

Identification of a layer refresh frame can be derived from the reference IDs of each frame by backtracking the dependency chain until reaching a point where only decodable frames are being referenced. Therefore it's recommended for both the flexible and the non-flexible mode that, when upgrade frames are being encoded in response to a LRR, those packets should contain layer indices and the reference fields so that the decoder or an MCU can make this derivation.

Example:

LRR {1,0}, {2,1} is sent by an MCU when it is currently relaying {1,0} to a receiver and which wants to upgrade to {2,1}. In response the encoder should encode the next frames in layers {1,1} and {2,1} by only referring to frames in {1,0}, or {0,0}.

In the non-flexible mode, periodic upgrade frames can be defined by the layer structure of the SS, thus periodic upgrade frames can be automatically identified by the picture ID.

#### 5. Usage with different scalability transmission mechanisms

Several different mechanisms are defined for how scalable streams can be transmitted in RTP. The RTP Taxonomy [I-D.ietf-avtext-rtp-grouping-taxonomy] Section 3.7 defines three mechanisms: Single RTP Stream on a Single Media Transport (SRST), Multiple RTP Streams on a Single Media Transport (MRST), and Multiple RTP Streams on Multiple Media Transports (MRMT).

The LRR message is applicable to all these mechanisms. For MRST and MRMT mechanisms, the "media source" field of the LRR FCI is set to the SSRC of the RTP stream containing the layer indicated by the Current Layer Index (if "C" is 1), or the stream containing the base encoded stream (if "C" is 0). For MRMT, it is sent on the RTP session on which this stream is sent. On receipt, the sender MUST refresh all the layers requested in the stream, simultaneously in decode order.

Note: arguably, for the MRST and MRMT mechanisms, FIR feedback messages could instead be used to refresh specific individual layers. However, the usage of FIR for MRSR/MRMT is not explicitly specified anywhere, and if FIR is interpreted as refreshing layers, there is no way to request an actual full, synchronized refresh of all the layers of an MRST/MRMT layered source. Thus, the authors feel that interpreting FIR as refreshing the entire source, and using LRR for the individual layers, would be more useful.

## 6. Security Considerations

All the security considerations of FIR feedback packets [RFC5104] apply to LRR feedback packets as well. Additionally, media senders receiving LRR feedback packets MUST validate that the payload types and layer indices they are receiving are valid for the stream they are currently sending, and discard the requests if not.

## 7. IANA Considerations

The IANA is requested to register the following values:

- TODO: PSFB value for LRR

## 8. References

### 8.1. Normative References

[I-D.ietf-payload-rtp-h265]

Wang, Y., Sanchez, Y., Schierl, T., Wenger, S., and M. Hannuksela, "RTP Payload Format for H.265/HEVC Video", draft-ietf-payload-rtp-h265-14 (work in progress), August 2015.

[I-D.ietf-payload-vp8]

Westin, P., Lundin, H., Glover, M., Uberti, J., and F. Galligan, "RTP Payload Format for VP8 Video", draft-ietf-payload-vp8-17 (work in progress), September 2015.

- [I-D.uberti-payload-vp9]  
Uberti, J., Holmer, S., Flodman, M., Lennox, J., and D. Hong, "RTP Payload Format for VP9 Video", draft-uberti-payload-vp9-01 (work in progress), March 2015.
- [RFC2119] Bradner, S., "Key words for use in RFCs to Indicate Requirement Levels", BCP 14, RFC 2119, DOI 10.17487/RFC2119, March 1997, <<http://www.rfc-editor.org/info/rfc2119>>.
- [RFC3550] Schulzrinne, H., Casner, S., Frederick, R., and V. Jacobson, "RTP: A Transport Protocol for Real-Time Applications", STD 64, RFC 3550, DOI 10.17487/RFC3550, July 2003, <<http://www.rfc-editor.org/info/rfc3550>>.
- [RFC4585] Ott, J., Wenger, S., Sato, N., Burmeister, C., and J. Rey, "Extended RTP Profile for Real-time Transport Control Protocol (RTCP)-Based Feedback (RTP/AVPF)", RFC 4585, DOI 10.17487/RFC4585, July 2006, <<http://www.rfc-editor.org/info/rfc4585>>.
- [RFC6190] Wenger, S., Wang, Y., Schierl, T., and A. Eleftheriadis, "RTP Payload Format for Scalable Video Coding", RFC 6190, DOI 10.17487/RFC6190, May 2011, <<http://www.rfc-editor.org/info/rfc6190>>.

## 8.2. Informative References

- [I-D.ietf-avtext-rtp-grouping-taxonomy]  
Lennox, J., Gross, K., Nandakumar, S., Salgueiro, G., and B. Burman, "A Taxonomy of Semantics and Mechanisms for Real-Time Transport Protocol (RTP) Sources", draft-ietf-avtext-rtp-grouping-taxonomy-08 (work in progress), July 2015.
- [RFC5104] Wenger, S., Chandra, U., Westerlund, M., and B. Burman, "Codec Control Messages in the RTP Audio-Visual Profile with Feedback (AVPF)", RFC 5104, DOI 10.17487/RFC5104, February 2008, <<http://www.rfc-editor.org/info/rfc5104>>.

## Authors' Addresses

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

Email: jonathan@vidyo.com

Danny Hong  
Vidyo, Inc.  
433 Hackensack Avenue  
Seventh Floor  
Hackensack, NJ 07601  
US

Email: danny@vidyo.com

Justin Uberti  
Google, Inc.  
747 6th Street South  
Kirkland, WA 98033  
USA

Email: justin@uberti.name

Stefan Holmer  
Google, Inc.  
Kungsbron 2  
Stockholm 111 22  
Sweden

Email: holmer@google.com

Magnus Flodman  
Google, Inc.  
Kungsbron 2  
Stockholm 111 22  
Sweden

Email: mflodman@google.com