

Internet Engineering Task Force (IETF)
Request for Comments: 6185
Category: Standards Track
ISSN: 2070-1721

T. Kristensen
P. Luthi
TANDBERG
May 2011

RTP Payload Format for
H.264 Reduced-Complexity Decoding Operation (RCDO) Video

Abstract

This document describes an RTP payload format for the Reduced-Complexity Decoding Operation (RCDO) for H.264 Baseline profile bitstreams, as specified in ITU-T Recommendation H.241. RCDO reduces the decoding cost and resource consumption of the video processing. The RCDO RTP payload format is based on the H.264 RTP payload format.

Status of This Memo

This is an Internet Standards Track document.

This document is a product of the Internet Engineering Task Force (IETF). It represents the consensus of the IETF community. It has received public review and has been approved for publication by the Internet Engineering Steering Group (IESG). Further information on Internet Standards is available in Section 2 of RFC 5741.

Information about the current status of this document, any errata, and how to provide feedback on it may be obtained at <http://www.rfc-editor.org/info/rfc6185>.

Copyright Notice

Copyright (c) 2011 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.

This document may contain material from IETF Documents or IETF Contributions published or made publicly available before November 10, 2008. The person(s) controlling the copyright in some of this material may not have granted the IETF Trust the right to allow modifications of such material outside the IETF Standards Process. Without obtaining an adequate license from the person(s) controlling the copyright in such materials, this document may not be modified outside the IETF Standards Process, and derivative works of it may not be created outside the IETF Standards Process, except to format it for publication as an RFC or to translate it into languages other than English.

Table of Contents

1. Introduction	2
2. Conventions Used in This Document	3
3. Media Format Background	3
4. Payload Format	3
5. Congestion Control Considerations	3
6. Payload Format Parameters	3
6.1. Media Type Definition	4
7. Mapping to SDP	19
7.1. Offer/Answer Considerations	20
7.2. Declarative SDP Considerations	20
8. IANA Considerations	20
9. Security Considerations	20
10. Acknowledgements	21
11. References	21
11.1. Normative References	21
11.2. Informative References	21

1. Introduction

ITU-T Recommendation H.241 [3] specifies a Reduced-Complexity Decoding Operation (RCDO) for use with H.264 [2] Baseline profile bitstreams. It also specifies a bitstream constraint associated with RCDO and a mechanism for signaling RCDO within the bitstream. The RCDO signaling indicates that the bitstream conforms to the bitstream constraint and that the decoder shall apply the RCDO decoding process to the bitstream.

RCDO for H.264 offers a solution to support higher resolutions at the same high frame rates used in current implementations. This is achieved by reducing the processing requirements and thus reducing the decoding cost/resource consumption of the video processing.

This document defines media type parameters and allows use in systems based on the Session Description Protocol (SDP) [8] for signaling.

2. Conventions Used in This Document

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 [4].

3. Media Format Background

The Reduced-Complexity Decoding Operation (RCDO) for H.264 Baseline profile bitstreams is specified in Annex B of H.241 [3]. RCDO is specified as a separate H.264 mode and is distinct from any profile defined in H.264. An RCDO bitstream obeys all the constraints of the Baseline profile.

The media format is based on the H.264 RTP payload format as specified in RFC 6184 [1]. Therefore, RFC 6184 constitutes the basis for this document and is referred to several times.

In order to signal H.264 additional modes, Table 8-13 of H.241 [3] specifies an `AdditionalModesSupported` parameter. Currently, the only additional mode defined is RCDO.

Informative note: Other additional modes may be defined in the future. H.264 additional modes may or may not be distinct from the profiles in H.264.

A separate media subtype, named H264-RCDO, is defined to ensure backward compatibility with deployed implementations of H.264.

4. Payload Format

The payload format defined in Section 5 of RFC 6184 [1] SHALL be used. This includes the RTP header usage and the payload format in RFC 6184. Examples of typical RTP packets can be found in RFC 6184.

5. Congestion Control Considerations

Congestion control for RTP SHALL be used in accordance with RFC 3550 [6] and with any applicable RTP profile, e.g., RFC 3551 [7]. If best-effort service is being used, users of this payload format SHALL monitor packet loss to ensure that the packet loss rate is within acceptable parameters.

6. Payload Format Parameters

This RTP payload format is identified using the H264-RCDO media subtype, which is registered in accordance with RFC 4855 [10], and using the template of RFC 4288 [13].

6.1. Media Type Definition

Informative note: The media subtype definition for H264-RCDO is based on the definition of the H264 media subtype as specified in Section 8.1 of RFC 6184 [1]. Except for the profile-level-id parameter, for which new semantics are specified below, the optional parameters are copied from RFC 6184 [1] in order to provide a complete, self-contained media subtype registration to IANA. The references are updated to match the numbering used in this document.

The media subtype for RCDO for H.264 has been allocated from the IETF tree.

Type name: video

Subtype name: H264-RCDO

Required parameters:

rate: Indicates the RTP timestamp clock rate. The rate value MUST be 90000.

Optional parameters:

profile-level-id: A base16 RFC 4648 [9] (hexadecimal) representation of the following three bytes in the sequence parameter set NAL unit is specified in H.264 [2]: 1) profile_idc, 2) a byte herein referred to as profile-iop, composed of the values of constraint_set0_flag, constraint_set1_flag, constraint_set2_flag, constraint_set3_flag, constraint_set4_flag, constraint_set5_flag, and reserved_zero_2bits in bit-significance order, starting from the most-significant bit, and 3) level_idc. Note that reserved_zero_2bits is required to be equal to 0 in H.264 [2], but other values for it may be specified in the future by ITU-T or ISO/IEC.

The profile-level-id parameter indicates the default sub-profile (i.e., the subset of coding tools that may have been used to generate the stream or that the receiver supports) and the default level of the stream or the receiver supports.

RCDO is distinct from any profile; this implies that the profile value 0 (no profile) and the profile_idc byte of the profile-level-id parameter are equal to 0. An RCDO bitstream MUST obey all the constraints of the Baseline profile. Therefore, only constraint_set0_flag is equal to 1 in the profile-iop part of the profile-level-id parameter; the remaining bits are set to 0.

If the profile-level-id parameter is used to indicate properties of a NAL unit stream, it indicates that, to decode the stream, the minimum subset of coding tools a decoder has to support is the default sub-profile, and the lowest level the decoder has to support is the default level.

If the profile-level-id parameter is used for capability exchange or session setup, it indicates the subset of coding tools, which is equal to the default sub-profile, that the codec supports for both receiving and sending. If max-recv-level is not present, the default level from profile-level-id indicates the highest level the codec wishes to support. If max-recv-level is present, it indicates the highest level the codec supports for receiving. For either receiving or sending, all levels that are lower than the highest level supported MUST also be supported.

For example, if a codec supports level 1.3, the profile-level-id becomes 00800d, in which 00 indicates the "no profile" value, 80 indicates the constraints of the Baseline profile, and 0d indicates level 1.3. When level 2.1 is supported, the profile-level-id becomes 008015.

If no profile-level-id is present, level 1 (i.e., equivalent to profile-level-id 00800a) MUST be implied.

Informative note: The definitions of the remaining optional parameters below are copied verbatim from Section 8.1 of RFC 6184 [1]. Only the references are updated to match the numbering used in this document.

max-recv-level: This parameter MAY be used to indicate the highest level a receiver supports when the highest level is higher than the default level (the level indicated by profile-level-id). The value of max-recv-level is a base16 (hexadecimal) representation of the two bytes after the syntax element profile_idc in the sequence parameter set NAL unit specified in H.264 [2]: profile_iop (as defined above) and level_idc. If the level_idc byte of max-recv-level is equal to 11 and bit 4 of the profile_iop byte of max-recv-level is equal to 1 or if the level_idc byte of max-recv-level is equal to 9 and bit 4 of the profile_iop byte of max-recv-level is equal to 0, the highest level the receiver supports is Level 1b. Otherwise, the highest level the receiver supports is equal to the level_idc byte of max-recv-level divided by 10.

max-recv-level MUST NOT be present if the highest level the receiver supports is not higher than the default level.

max-mbps, max-smbps, max-fs, max-cpb, max-dpb, and max-br: These parameters MAY be used to signal the capabilities of a receiver implementation. These parameters MUST NOT be used for any other purpose. The highest level conveyed in the value of the profile-level-id parameter or the max-recv-level parameter MUST be such that the receiver is fully capable of supporting. max-mbps, max-smbps, max-fs, max-cpb, max-dpb, and max-br MAY be used to indicate capabilities of the receiver that extend the required capabilities of the signaled highest level, as specified below.

When more than one parameter from the set (max-mbps, max-smbps, max-fs, max-cpb, max-dpb, max-br) is present, the receiver MUST support all signaled capabilities simultaneously. For example, if both max-mbps and max-br are present, the signaled highest level with the extension of both the frame rate and bitrate is supported. That is, the receiver is able to decode NAL unit streams in which the macroblock processing rate is up to max-mbps (inclusive), the bitrate is up to max-br (inclusive), the coded picture buffer size is derived as specified in the semantics of the max-br parameter below, and the other properties comply with the highest level specified in the value of the profile-level-id parameter or the max-recv-level parameter.

If a receiver can support all the properties of Level A, the highest level specified in the value of the profile-level-id parameter or the max-recv-level parameter MUST be Level A (i.e., MUST NOT be lower than Level A). In other words, a receiver MUST NOT signal values of max-mbps, max-fs, max-cpb, max-dpb, and max-br that taken together meet the requirements of a higher level compared to the highest level specified in the value of the profile-level-id parameter or the max-recv-level parameter.

Informative note: When the OPTIONAL media type parameters are used to signal the properties of a NAL unit stream, max-mbps, max-smbps, max-fs, max-cpb, max-dpb, and max-br are not present, and the value of profile-level-id must always be such that the NAL unit stream complies fully with the specified profile and level.

max-mbps: The value of max-mbps is an integer indicating the maximum macroblock processing rate in units of macroblocks per second. The max-mbps parameter signals that the receiver is capable of decoding video at a higher rate than is required by the signaled highest level conveyed in the value of the profile-level-id parameter or the max-recv-level parameter. When max-mbps is signaled, the receiver MUST be able to decode NAL unit streams that conform to the signaled highest level, with the exception that the MaxMBPS value in Table A-1 of H.264 [2] for the signaled

highest level is replaced with the value of max-mbps. The value of max-mbps MUST be greater than or equal to the value of MaxMBPS given in Table A-1 of H.264 [2] for the highest level. Senders MAY use this knowledge to send pictures of a given size at a higher picture rate than is indicated in the signaled highest level.

max-smbps: The value of max-smbps is an integer indicating the maximum static macroblock processing rate in units of static macroblocks per second, under the hypothetical assumption that all macroblocks are static macroblocks. When max-smbps is signaled, the MaxMBPS value in Table A-1 of H.264 [2] should be replaced with the result of the following computation:

- o If the parameter max-mbps is signaled, set a variable MaxMacroblocksPerSecond to the value of max-mbps. Otherwise, set MaxMacroblocksPerSecond equal to the value of MaxMBPS in Table A-1 of H.264 [2] for the signaled highest level conveyed in the value of the profile-level-id parameter or the max-recv-level parameter.
- o Set a variable P_non-static to the proportion of non-static macroblocks in picture n.
- o Set a variable P_static to the proportion of static macroblocks in picture n.
- o The value of MaxMBPS in Table A-1 of H.264 [2] should be considered by the encoder to be equal to:

$$\text{MaxMacroblocksPerSecond} * \text{max-smbps} / (\text{P_non-static} * \text{max-smbps} + \text{P_static} * \text{MaxMacroblocksPerSecond})$$

The encoder should recompute this value for each picture. The value of max-smbps MUST be greater than or equal to the value of MaxMBPS given explicitly as the value of the max-mbps parameter or implicitly in Table A-1 of H.264 [2] for the signaled highest level. Senders MAY use this knowledge to send pictures of a given size at a higher picture rate than is indicated in the signaled highest level.

max-fs: The value of max-fs is an integer indicating the maximum frame size in units of macroblocks. The max-fs parameter signals that the receiver is capable of decoding larger picture sizes than are required by the signaled highest level conveyed in the value of the profile-level-id parameter or the max-recv-level parameter. When max-fs is signaled, the receiver MUST be able to decode NAL unit streams that conform to the signaled highest level, with the

exception that the MaxFS value in Table A-1 of H.264 [2] for the signaled highest level is replaced with the value of max-fs. The value of max-fs MUST be greater than or equal to the value of MaxFS given in Table A-1 of H.264 [2] for the highest level. Senders MAY use this knowledge to send larger pictures at a proportionally lower frame rate than is indicated in the signaled highest level.

max-cpb: The value of max-cpb is an integer indicating the maximum coded picture buffer size in units of 1000 bits for the VCL HRD parameters and in units of 1200 bits for the NAL HRD parameters. Note that this parameter does not use units of cpbBrVclFactor and cpbBrNALFactor (see Table A-1 of H.264 [2]). The max-cpb parameter signals that the receiver has more memory than the minimum amount of coded picture buffer memory required by the signaled highest level conveyed in the value of the profile-level-id parameter or the max-recv-level parameter. When max-cpb is signaled, the receiver MUST be able to decode NAL unit streams that conform to the signaled highest level, with the exception that the MaxCPB value in Table A-1 of H.264 [2] for the signaled highest level is replaced with the value of max-cpb (after taking cpbBrVclFactor and cpbBrNALFactor into consideration when needed). The value of max-cpb (after taking cpbBrVclFactor and cpbBrNALFactor into consideration when needed) MUST be greater than or equal to the value of MaxCPB given in Table A-1 of H.264 [2] for the highest level. Senders MAY use this knowledge to construct coded video streams with greater variation of bitrate than can be achieved with the MaxCPB value in Table A-1 of H.264 [2].

Informative note: The coded picture buffer is used in the hypothetical reference decoder (Annex C of H.264). The use of the hypothetical reference decoder is recommended in H.264 encoders to verify that the produced bitstream conforms to the standard and to control the output bitrate. Thus, the coded picture buffer is conceptually independent of any other potential buffers in the receiver, including de-interleaving and de-jitter buffers. The coded picture buffer need not be implemented in decoders as specified in Annex C of H.264, but rather standard-compliant decoders can have any buffering arrangements provided that they can decode standard-compliant bitstreams. Thus, in practice, the input buffer for a video decoder can be integrated with de-interleaving and de-jitter buffers of the receiver.

max-dpb: The value of max-dpb is an integer indicating the maximum decoded picture buffer size in units of 8/3 macroblocks. The max-dpb parameter signals that the receiver has more memory than the minimum amount of decoded picture buffer memory required by the signaled highest level conveyed in the value of the profile-level-id parameter or the max-recv-level parameter. When max-dpb is signaled, the receiver MUST be able to decode NAL unit streams that conform to the signaled highest level, with the exception that the MaxDpbMbs value in Table A-1 of H.264 [2] for the signaled highest level is replaced with the value of $\text{max-dpb} * 3 / 8$. Consequently, a receiver that signals max-dpb MUST be capable of storing the following number of decoded frames, complementary field pairs, and non-paired fields in its decoded picture buffer:

$$\text{Min}(\text{max-dpb} * 3 / 8 / (\text{PicWidthInMbs} * \text{FrameHeightInMbs}), 16)$$

Wherein PicWidthInMbs and FrameHeightInMbs are defined in H.264 [2].

The value of max-dpb MUST be greater than or equal to the value of $\text{MaxDpbMbs} * 3 / 8$, wherein the value of MaxDpbMbs is given in Table A-1 of H.264 [2] for the highest level. Senders MAY use this knowledge to construct coded video streams with improved compression.

Informative note: This parameter was added primarily to complement a similar codepoint in the ITU-T Recommendation H.245, so as to facilitate signaling gateway designs. The decoded picture buffer stores reconstructed samples. There is no relationship between the size of the decoded picture buffer and the buffers used in RTP, especially de-interleaving and de-jitter buffers.

Informative note: In RFC 3984, which is obsoleted by RFC 6184, the unit of this parameter was 1024 bytes. The unit has been changed to 8/3 macroblocks in this document. The reason for this change was due to the changes from the 2003 version of the H.264 specification referenced by RFC 3984 to the 2010 version of the H.264 specification referenced by this document, particularly the changes to Table A-1 in the H.264 specification due to addition of color formats and bit depths not supported earlier. The changed semantics of this parameter keeps backward compatibility to RFC 3984 and supports all profiles defined in the 2010 version of the H.264 specification.

max-br: The value of max-br is an integer indicating the maximum video bitrate in units of 1000 bits per second for the VCL HRD parameters and in units of 1200 bits per second for the NAL HRD parameters. Note that this parameter does not use units of cpbBrVclFactor and cpbBrNALFactor (see Table A-1 of H.264 [2]).

The max-br parameter signals that the video decoder of the receiver is capable of decoding video at a higher bitrate than is required by the signaled highest level conveyed in the value of the profile-level-id parameter or the max-recv-level parameter.

When max-br is signaled, the video codec of the receiver MUST be able to decode NAL unit streams that conform to the signaled highest level, with the following exceptions in the limits specified by the highest level:

- o The value of max-br (after taking cpbBrVclFactor and cpbBrNALFactor into consideration when needed) replaces the MaxBR value in Table A-1 of H.264 [2] for the highest level.
- o When the max-cpb parameter is not present, the result of the following formula replaces the value of MaxCPB in Table A-1 of H.264 [2]: $(\text{MaxCPB of the signaled level}) * \text{max-br} / (\text{MaxBR of the signaled highest level})$.

For example, if a receiver signals capability for Main profile Level 1.2 with max-br equal to 1550, this indicates a maximum video bitrate of 1550 kbits/sec for VCL HRD parameters, a maximum video bitrate of 1860 kbits/sec for NAL HRD parameters, and a CPB size of 4036458 bits $(1550000 / 384000 * 1000 * 1000)$.

The value of max-br (after taking cpbBrVclFactor and cpbBrNALFactor into consideration when needed) MUST be greater than or equal to the value MaxBR given in Table A-1 of H.264 [2] for the signaled highest level.

Senders MAY use this knowledge to send higher bitrate video as allowed in the level definition of Annex A of H.264 to achieve improved video quality.

Informative note: This parameter was added primarily to complement a similar codepoint in the ITU-T Recommendation H.245, so as to facilitate signaling gateway designs. The assumption that the network is capable of handling such bitrates at any given time cannot be made from the value of this parameter. In particular, no conclusion can be drawn that the signaled bitrate is possible under congestion control constraints.

redundant-pic-cap: This parameter signals the capabilities of a receiver implementation. When equal to 0, the parameter indicates that the receiver makes no attempt to use redundant coded pictures to correct incorrectly decoded primary coded pictures. When equal to 1, the receiver is capable of using redundant slices; therefore, a sender **SHOULD** avoid sending redundant slices to save bandwidth. When equal to 1, the receiver is capable of decoding any such redundant slice that covers a corrupted area in a primary decoded picture (at least partly), and therefore a sender **MAY** send redundant slices. When the parameter is not present, a value of 0 **MUST** be used for **redundant-pic-cap**. When present, the value of **redundant-pic-cap** **MUST** be either 0 or 1.

When the **profile-level-id** parameter is present in the same signaling as the **redundant-pic-cap** parameter and the profile indicated in **profile-level-id** is such that it disallows the use of redundant coded pictures (e.g., Main profile), the value of **redundant-pic-cap** **MUST** be equal to 0. When a receiver indicates **redundant-pic-cap** equal to 0, the received stream **SHOULD NOT** contain redundant coded pictures.

Informative note: Even if **redundant-pic-cap** is equal to 0, the decoder is able to ignore redundant codec pictures provided that the decoder supports a profile (Baseline, Extended) in which redundant coded pictures are allowed.

Informative note: Even if **redundant-pic-cap** is equal to 1, the receiver may also choose other error concealment strategies to replace or complement decoding of redundant slices.

sprop-parameter-sets: This parameter **MAY** be used to convey any sequence and picture parameter set NAL units (herein referred to as the initial parameter set NAL units) that can be placed in the NAL unit stream to precede any other NAL units in decoding order. The parameter **MUST NOT** be used to indicate codec capability in any capability exchange procedure. The value of the parameter is a comma-separated (',') list of base64 RFC 4648 [9] representations of parameter set NAL units as specified in Sections 7.3.2.1 and 7.3.2.2 of H.264 [2]. Note that the number of bytes in a parameter set NAL unit is typically less than 10, but a picture parameter set NAL unit can contain several hundred bytes.

Informative note: When several payload types are offered in the SDP Offer/Answer model, each with its own **sprop-parameter-sets** parameter, the receiver cannot assume that those parameter sets do not use conflicting storage locations (i.e., identical values of parameter set identifiers). Therefore, a receiver

should buffer all sprop-parameter-sets and make them available to the decoder instance that decodes a certain payload type.

The sprop-parameter-sets parameter MUST only contain parameter sets that are conforming to the profile-level-id, i.e., the subset of coding tools indicated by any of the parameter sets MUST be equal to the default sub-profile, and the level indicated by any of the parameter sets MUST be equal to the default level.

sprop-level-parameter-sets: This parameter MAY be used to convey any sequence and picture parameter set NAL units (herein referred to as the initial parameter set NAL units) that can be placed in the NAL unit stream to precede any other NAL units in decoding order and that are associated with one or more levels different than the default level. The parameter MUST NOT be used to indicate codec capability in any capability exchange procedure.

The sprop-level-parameter-sets parameter contains parameter sets for one or more levels that are different than the default level. All parameter sets associated with one level are clustered and prefixed with a three-byte field that has the same syntax as profile-level-id. This enables the receiver to install the parameter sets for one level and discard the rest. The three-byte field is named PLId, and all parameter sets associated with one level are named PSL, which has the same syntax as sprop-parameter-sets. Parameter sets for each level are represented in the form of PLId:PSL, i.e., PLId followed by a colon (':') and the base64 RFC 4648 [9] representation of the initial parameter set NAL units for the level. Each pair of PLId:PSLs is also separated by a colon. Note that a PSL can contain multiple parameter sets for that level, separated with commas (',').

The subset of coding tools indicated by each PLId field MUST be equal to the default sub-profile, and the level indicated by each PLId field MUST be different than the default level. All sequence parameter sets contained in each PSL MUST have the three bytes from profile_idc to level_idc, inclusive, equal to the preceding PLId.

Informative note: This parameter allows for efficient level downgrade or upgrade in SDP Offer/Answer and out-of-band transport of parameter sets simultaneously.

use-level-src-parameter-sets: This parameter MAY be used to indicate a receiver capability. The value MAY be equal to either 0 or 1. When the parameter is not present, the value MUST be inferred to be equal to 0. The value 0 indicates that the receiver does not understand the sprop-level-parameter-sets parameter, does not

understand the "fmtp" source attribute as specified in Section 6.3 of RFC 5576 [14], will ignore sprop-level-parameter-sets when present, and will ignore sprop-parameter-sets when conveyed using the "fmtp" source attribute. The value 1 indicates that the receiver understands the sprop-level-parameter-sets parameter, understands the "fmtp" source attribute as specified in Section 6.3 of RFC 5576 [14], and is capable of using parameter sets contained in the sprop-level-parameter-sets or contained in the sprop-parameter-sets that is conveyed using the "fmtp" source attribute.

Informative note: An RFC 3984 receiver does not understand sprop-level-parameter-sets, use-level-src-parameter-sets, or the "fmtp" source attribute as specified in Section 6.3 of RFC 5576 [14]. Therefore, during SDP Offer/Answer, an RFC 3984 receiver as the answerer will simply ignore sprop-level-parameter-sets when present in an offer and sprop-parameter-sets conveyed using the "fmtp" source attribute, as specified in Section 6.3 of RFC 5576 [14]. Assume that the offered payload type was accepted at a level lower than the default level. If the offered payload type included sprop-level-parameter-sets or included sprop-parameter-sets conveyed using the "fmtp" source attribute and if the offerer sees that the answerer has not included use-level-src-parameter-sets equal to 1 in the answer, the offerer knows that in-band transport of parameter sets is needed.

in-band-parameter-sets: This parameter MAY be used to indicate a receiver capability. The value MAY be equal to either 0 or 1. The value 1 indicates that the receiver discards out-of-band parameter sets in sprop-parameter-sets and sprop-level-parameter-sets; therefore, the sender MUST transmit all parameter sets in-band. The value 0 indicates that the receiver utilizes out-of-band parameter sets included in sprop-parameter-sets and/or sprop-level-parameter-sets. However, in this case, the sender MAY still choose to send parameter sets in-band. When in-band-parameter-sets is equal to 1, use-level-src-parameter-sets MUST NOT be present or MUST be equal to 0. When the parameter is not present, this receiver capability is not specified, and therefore the sender MAY send out-of-band parameter sets only, it MAY send in-band-parameter-sets only, or it MAY send both.

level-asymmetry-allowed: This parameter MAY be used in SDP Offer/Answer to indicate whether level asymmetry, i.e., sending media encoded at a different level in the offerer-to-answerer direction than the level in the answerer-to-offerer direction, is allowed. The value MAY be equal to either 0 or 1. When the parameter is not present, the value MUST be inferred to be equal to 0. The

value 1 in both the offer and the answer indicates that level asymmetry is allowed. The value of 0 in either the offer or the answer indicates that level asymmetry is not allowed.

If level-asymmetry-allowed is equal to 0 (or not present) in either the offer or the answer, level asymmetry is not allowed. In this case, the level to use in the direction from the offerer to the answerer MUST be the same as the level to use in the opposite direction.

packetization-mode: This parameter signals the properties of an RTP payload type or the capabilities of a receiver implementation. Only a single configuration point can be indicated; thus, when capabilities to support more than one packetization-mode are declared, multiple configuration points (RTP payload types) must be used.

When the value of packetization-mode is equal to 0 or packetization-mode is not present, the single NAL mode MUST be used. This mode is in use in standards using ITU-T Recommendation H.241 [3] (see Section 12.1). When the value of packetization-mode is equal to 1, the non-interleaved mode MUST be used. When the value of packetization-mode is equal to 2, the interleaved mode MUST be used. The value of packetization-mode MUST be an integer in the range of 0 to 2, inclusive.

sprop-interleaving-depth: This parameter MUST NOT be present when packetization-mode is not present or the value of packetization-mode is equal to 0 or 1. This parameter MUST be present when the value of packetization-mode is equal to 2.

This parameter signals the properties of an RTP packet stream. It specifies the maximum number of VCL NAL units that precede any VCL NAL unit in the RTP packet stream in transmission order and that follow the VCL NAL unit in decoding order. Consequently, it is guaranteed that receivers can reconstruct NAL unit decoding order when the buffer size for NAL unit decoding order recovery is at least the value of sprop-interleaving-depth + 1 in terms of VCL NAL units.

The value of sprop-interleaving-depth MUST be an integer in the range of 0 to 32767, inclusive.

sprop-deint-buf-req: This parameter MUST NOT be present when packetization-mode is not present or the value of packetization-mode is equal to 0 or 1. It MUST be present when the value of packetization-mode is equal to 2.

sprop-deint-buf-req signals the required size of the de-interleaving buffer for the RTP packet stream. The value of the parameter MUST be greater than or equal to the maximum buffer occupancy (in units of bytes) required in such a de-interleaving buffer that is specified in Section 7.2 of RFC 6184 [1]. It is guaranteed that receivers can perform the de-interleaving of interleaved NAL units into NAL unit decoding order, when the de-interleaving buffer size is at least the value of sprop-deint-buf-req in terms of bytes.

The value of sprop-deint-buf-req MUST be an integer in the range of 0 to 4294967295, inclusive.

Informative note: sprop-deint-buf-req indicates the required size of the de-interleaving buffer only. When network jitter can occur, an appropriately sized jitter buffer has to be provisioned for as well.

deint-buf-cap: This parameter signals the capabilities of a receiver implementation and indicates the amount of de-interleaving buffer space in units of bytes that the receiver has available for reconstructing the NAL unit decoding order. A receiver is able to handle any stream for which the value of the sprop-deint-buf-req parameter is smaller than or equal to this parameter.

If the parameter is not present, then a value of 0 MUST be used for deint-buf-cap. The value of deint-buf-cap MUST be an integer in the range of 0 to 4294967295, inclusive.

Informative note: deint-buf-cap indicates the maximum possible size of the de-interleaving buffer of the receiver only. When network jitter can occur, an appropriately sized jitter buffer has to be provisioned for as well.

sprop-init-buf-time: This parameter MAY be used to signal the properties of an RTP packet stream. The parameter MUST NOT be present if the value of packetization-mode is equal to 0 or 1.

The parameter signals the initial buffering time that a receiver MUST wait before starting decoding to recover the NAL unit decoding order from the transmission order. The parameter is the maximum value of (decoding time of the NAL unit - transmission time of a NAL unit), assuming reliable and instantaneous transmission, the same timeline for transmission and decoding, and commencement of decoding when the first packet arrives.

An example of specifying the value of sprop-init-buf-time follows. A NAL unit stream is sent in the following interleaved order, in

which the value corresponds to the decoding time and the transmission order is from left to right:

0 2 1 3 5 4 6 8 7 ...

Assuming a steady transmission rate of NAL units, the transmission times are:

0 1 2 3 4 5 6 7 8 ...

Subtracting the decoding time from the transmission time column-wise results in the following series:

0 -1 1 0 -1 1 0 -1 1 ...

Thus, in terms of intervals of NAL unit transmission times, the value of `sprop-init-buf-time` in this example is 1. The parameter is coded as a non-negative base10 integer representation in clock ticks of a 90-kHz clock. If the parameter is not present, then no initial buffering time value is defined. Otherwise, the value of `sprop-init-buf-time` MUST be an integer in the range of 0 to 4294967295, inclusive.

In addition to the signaled `sprop-init-buf-time`, receivers SHOULD take into account the transmission delay jitter buffering, including buffering for the delay jitter caused by mixers, translators, gateways, proxies, traffic-shapers, and other network elements.

`sprop-max-don-diff`: This parameter MAY be used to signal the properties of an RTP packet stream. It MUST NOT be used to signal transmitter, receiver, or codec capabilities. The parameter MUST NOT be present if the value of `packetization-mode` is equal to 0 or 1. `sprop-max-don-diff` is an integer in the range of 0 to 32767, inclusive. If `sprop-max-don-diff` is not present, the value of the parameter is unspecified. `sprop-max-don-diff` is calculated as follows:

$$\text{sprop-max-don-diff} = \max\{\text{AbsDON}(i) - \text{AbsDON}(j)\}, \text{ for any } i \text{ and any } j > i,$$

where *i* and *j* indicate the index of the NAL unit in the transmission order and `AbsDON` denotes a decoding order number of the NAL unit that does not wrap around to 0 after 65535. In other words, `AbsDON` is calculated as follows: let *m* and *n* be consecutive NAL units in transmission order. For the very first NAL unit in transmission order (whose index is 0), `AbsDON(0) = DON(0)`. For other NAL units, `AbsDON` is calculated as follows:


```
If DON(m) == DON(n), AbsDON(n) = AbsDON(m)

If (DON(m) < DON(n) and DON(n) - DON(m) < 32768),
AbsDON(n) = AbsDON(m) + DON(n) - DON(m)

If (DON(m) > DON(n) and DON(m) - DON(n) >= 32768),
AbsDON(n) = AbsDON(m) + 65536 - DON(m) + DON(n)

If (DON(m) < DON(n) and DON(n) - DON(m) >= 32768),
AbsDON(n) = AbsDON(m) - (DON(m) + 65536 - DON(n))

If (DON(m) > DON(n) and DON(m) - DON(n) < 32768),
AbsDON(n) = AbsDON(m) - (DON(m) - DON(n))
```

where DON(i) is the decoding order number of the NAL unit having index i in the transmission order. The decoding order number is specified in Section 5.5 of RFC 6184 [1].

Informative note: Receivers may use sprop-max-don-diff to trigger which NAL units in the receiver buffer can be passed to the decoder.

max-rcmd-nalu-size: This parameter MAY be used to signal the capabilities of a receiver. The parameter MUST NOT be used for any other purposes. The value of the parameter indicates the largest NALU size in bytes that the receiver can handle efficiently. The parameter value is a recommendation, not a strict upper boundary. The sender MAY create larger NALUs but must be aware that the handling of these may come at a higher cost than NALUs conforming to the limitation.

The value of max-rcmd-nalu-size MUST be an integer in the range of 0 to 4294967295, inclusive. If this parameter is not specified, no known limitation to the NALU size exists. Senders still have to consider the MTU size available between the sender and the receiver and SHOULD run MTU discovery for this purpose.

This parameter is motivated by, for example, an IP to H.223 video telephony gateway, where NALUs smaller than the H.223 transport data unit will be more efficient. A gateway may terminate IP; thus, MTU discovery will normally not work beyond the gateway.

Informative note: Setting this parameter to a lower than necessary value may have a negative impact.

sar-understood: This parameter MAY be used to indicate a receiver capability and nothing else. The parameter indicates the maximum value of aspect_ratio_idc (specified in H.264 [2]) smaller than 255 that the receiver understands. Table E-1 of H.264 [2] specifies aspect_ratio_idc equal to 0 as "unspecified"; 1 to 16, inclusive, as specific Sample Aspect Ratios (SARs); 17 to 254, inclusive, as "reserved"; and 255 as the Extended SAR, for which SAR width and SAR height are explicitly signaled. Therefore, a receiver with a decoder according to H.264 [2] understands aspect_ratio_idc in the range of 1 to 16, inclusive, and aspect_ratio_idc equal to 255, in the sense that the receiver knows exactly what the SAR is. For such a receiver, the value of sar-understood is 16. In the future, if Table E-1 of H.264 [2] is extended, e.g., such that the SAR for aspect_ratio_idc equal to 17 is specified, then for a receiver with a decoder that understands the extension, the value of sar-understood is 17. For a receiver with a decoder according to the 2003 version of H.264 [2], the value of sar-understood is 13, as the minimum reserved aspect_ratio_idc therein is 14.

When sar-understood is not present, the value MUST be inferred to be equal to 13.

sar-supported: This parameter MAY be used to indicate a receiver capability and nothing else. The value of this parameter is an integer in the range of 1 to sar-understood, inclusive, equal to 255. The value of sar-supported equal to N smaller than 255 indicates that the receiver supports all the SARs corresponding to H.264 aspect_ratio_idc values (see Table E-1 of H.264 [2]) in the range from 1 to N, inclusive, without geometric distortion. The value of sar-supported equal to 255 indicates that the receiver supports all sample aspect ratios that are expressible using two 16-bit integer values as the numerator and denominator, i.e., those that are expressible using the H.264 aspect_ratio_idc value of 255 (Extended_SAR, see Table E-1 of H.264 [2]), without geometric distortion.

H.264-compliant encoders SHOULD NOT send an aspect_ratio_idc equal to 0 or an aspect_ratio_idc larger than sar-understood and smaller than 255. H.264-compliant encoders SHOULD send an aspect_ratio_idc that the receiver is able to display without geometrical distortion. However, H.264-compliant encoders MAY choose to send pictures using any SAR.

Note that the actual sample aspect ratio or extended sample aspect ratio, when present, of the stream is conveyed in the Video Usability Information (VUI) part of the sequence parameter set.

Encoding considerations: This type is only defined for transfer via RTP (RFC 3550) and is framed and binary (see Section 4.8 in RFC 4288).

Security considerations: See Section 9 of RFC 6185.

Interoperability considerations: None

Published specification: RFC 6185 and its reference section

Applications that use this media type: Video streaming and conferencing applications

Additional information: None

Magic number(s):

File extension(s):

Macintosh file type code(s):

Person & email address to contact for further information:

Tom Kristensen <tom.kristensen@tandberg.com>, <tomkri@ifi.uio.no>

Intended usage: COMMON

Restrictions on usage: This type depends on RTP framing; hence, it is only defined for transfer via RTP (see RFC 3550). Transport within other framing protocols is not defined at this time.

Author: Tom Kristensen

Change controller: IETF Audio/Video Transport Working Group
delegated from the IESG

7. Mapping to SDP

The mapping of the above defined payload format media subtype and its parameters SHALL be done according to Section 3 of RFC 4855 [10].

An example of the "fmp" attribute in the media representation of a level 2.2 bitstream is as follows:

a=fmp:97 profile-level-id=008016

7.1. Offer/Answer Considerations

When H264-RCDO is offered over RTP using SDP in an Offer/Answer model [5] for unicast and multicast usage, the limitations and rules described in Section 8.2.2 of RFC 6184 [1] apply. Note that the `profile_idc` byte of the H264-RCDO `profile-level-id` parameter can only take the value of 0 (no profile).

For interoperability with systems not supporting H264-RCDO, it is RECOMMENDED to offer the H264 media subtype as well. As specified in RFC 3264 [5], listing the payload number for H264-RCDO before H264 in the format list on the "m=" line signals that H264-RCDO is preferred over H264. Following is an example where this scheme is applied:

```
m=video 5555 RTP/AVP 97 98

a=rtpmap:97 H264-RCDO/90000

a=fmtp:97 profile-level-id=008016;max-mbps=42000;max-smbps=323500

a=rtpmap:98 H264/90000

a=fmtp:98 profile-level-id=428016;max-mbps=35000;max-smbps=323500
```

7.2. Declarative SDP Considerations

When H264-RCDO over RTP is offered with SDP in a declarative style, as in the Real Time Streaming Protocol (RTSP) [11] or the Session Announcement Protocol (SAP) [12], the considerations in Section 8.2.3 of RFC 6184 [1] apply. Note that the `profile_idc` byte of the H264-RCDO `profile-level-id` parameter can only take the value of 0 (no profile).

8. IANA Considerations

IANA has registered H264-RCDO as specified in Section 6.1. The media subtype has also been added to the IANA registry for "RTP Payload Format MIME types" (<http://www.iana.org>).

9. Security Considerations

RTP packets using the payload format defined in this specification are subject to the security considerations discussed in the RTP specification [6] and in any applicable RTP profile. Refer also to the security considerations of the RTP Payload Format for H.264 Video specification in RFC 6184 [1]. No additional security considerations are introduced by this specification.

10. Acknowledgements

The authors would like to acknowledge Gisle Bjoentegaard and Arild Fuldseth for their technical contribution to the specification. In the final phases, Roni Even did a helpful review.

11. References

11.1. Normative References

- [1] Wang, Y., Even, R., Kristensen, T., and R. Jesup, "RTP Payload Format for H.264 Video", RFC 6184, May 2011.
- [2] International Telecommunications Union, "Advanced video coding for generic audiovisual services", ITU-T Recommendation H.264, March 2010.
- [3] International Telecommunications Union, "Extended video procedures and control signals for H.300-series terminals", ITU-T Recommendation H.241, May 2006.
- [4] Bradner, S., "Key words for use in RFCs to Indicate Requirement Levels", BCP 14, RFC 2119, March 1997.
- [5] Rosenberg, J. and H. Schulzrinne, "An Offer/Answer Model with Session Description Protocol (SDP)", RFC 3264, June 2002.
- [6] Schulzrinne, H., Casner, S., Frederick, R., and V. Jacobson, "RTP: A Transport Protocol for Real-Time Applications", STD 64, RFC 3550, July 2003.
- [7] Schulzrinne, H. and S. Casner, "RTP Profile for Audio and Video Conferences with Minimal Control", STD 65, RFC 3551, July 2003.
- [8] Handley, M., Jacobson, V., and C. Perkins, "SDP: Session Description Protocol", RFC 4566, July 2006.
- [9] Josefsson, S., "The Base16, Base32, and Base64 Data Encodings", RFC 4648, October 2006.
- [10] Casner, S., "Media Type Registration of RTP Payload Formats", RFC 4855, February 2007.

11.2. Informative References

- [11] Schulzrinne, H., Rao, A., and R. Lanphier, "Real Time Streaming Protocol (RTSP)", RFC 2326, April 1998.

- [12] Handley, M., Perkins, C., and E. Whelan, "Session Announcement Protocol", RFC 2974, October 2000.
- [13] Freed, N. and J. Klensin, "Media Type Specifications and Registration Procedures", BCP 13, RFC 4288, December 2005.
- [14] Lennox, J., Ott, J., and T. Schierl, "Source-Specific Media Attributes in the Session Description Protocol (SDP)", RFC 5576, June 2009.

Authors' Addresses

Tom Kristensen

TANDBERG

Philip Pedersens vei 22

N-1366 Lysaker

Norway

Phone: +47 67125125

EMail: tom.kristensen@tandberg.com, tomkri@ifi.uio.no

URI: <http://www.tandberg.com>

Patrick Luthi

TANDBERG

Philip Pedersens vei 22

N-1366 Lysaker

Norway

EMail: patrick.luthi@tandberg.com

URI: <http://www.tandberg.com>

