Network Working Group
Internet Draft
Intended status: Standards track
Expires: May 2016

Y.-K. Wang
Qualcomm
Y. Sanchez
T. Schierl
Fraunhofer HHI
S. Wenger
Vidyo
M. M. Hannuksela
Nokia

November 5, 2015

RTP Payload Format for H.265/HEVC Video
draft-ietf-payload-rtp-h265-15.txt

Abstract

This memo describes an RTP payload format for the video coding standard ITU-T Recommendation H.265 and ISO/IEC International Standard 23008-2, both also known as High Efficiency Video Coding (HEVC) and developed by the Joint Collaborative Team on Video Coding (JCT-VC). The RTP payload format allows for packetization of one or more Network Abstraction Layer (NAL) units in each RTP packet payload, as well as fragmentation of a NAL unit into multiple RTP packets. Furthermore, it supports transmission of an HEVC bitstream over a single as well as multiple RTP streams. When multiple RTP streams are used, a single or multiple transports may be utilized. The payload format has wide applicability in videoconferencing, Internet video streaming, and high bit-rate entertainment-quality video, among others.

Status of this Memo

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

This Internet-Draft will expire on May 5, 2016.

Copyright and License Notice

Table of Contents

1 Introduction

The High Efficiency Video Coding [HEVC], formally known as ITU-T Recommendation H.265 and ISO/IEC International Standard 23008-2 was ratified by ITU-T in April 2013 and reportedly provides significant coding efficiency gains over H.264 [H.264].

This memo describes an RTP payload format for HEVC. It shares its basic design with the RTP payload formats of [RFC6184] and [RFC6190]. With respect to design philosophy, security, congestion control, and overall implementation complexity, it has similar properties to those earlier payload format specifications. This is a conscious choice, as at least RFC6184 is widely deployed and generally known in the relevant implementer communities. Mechanisms from RFC6190 were incorporated as HEVC version 1 supports temporal scalability.

In order to help the overlapping implementer community, frequently only the differences between RFC6184/RFC6190 and the HEVC payload format are highlighted in non-normative, explanatory parts of this memo. Basic familiarity with both specifications is assumed for those parts. However, the normative parts of this memo do not require study of RFC6184 or RFC6190.

1.1 Overview of the HEVC Codec

H.264 and HEVC share a similar hybrid video codec design. In this memo, we provide a very brief overview of those features of HEVC that are in some form addressed by the payload format specified herein. Implementers have to read and understand, and apply the ITU-T/ISO/IEC specifications pertaining to HEVC to arrive at interoperable, well-performing implementations. Implementers should consider testing their design (including the interworking between the payload format implementation and the core video codec) using the tools provided by ITU-T/ISO/IEC; for example, conformance bitstreams as specified in [add confermance spec). Not doing so has historically led to badly performing and unsecure systems.

Conceptually, both H.264 and HEVC include a video coding layer (VCL), which is often used to refer to the coding-tool features,

and a network abstraction layer (NAL), which is often used to refer to the systems and transport interface aspects of the codecs.

1.1.1 Coding-Tool Features

Similarly to earlier hybrid-video-coding-based standards, including H.264, the following basic video coding design is employed by HEVC. A prediction signal is first formed either by intra or motion compensated prediction, and the residual (the difference between the original and the prediction) is then coded. The gains in coding efficiency are achieved by redesigning and improving almost all parts of the codec over earlier designs. In addition, HEVC includes several tools to make the implementation on parallel architectures easier. Below is a summary of HEVC coding-tool features.

Quad-tree block and transform structure

One of the major tools that contribute significantly to the coding efficiency of HEVC is the usage of flexible coding blocks and transforms, which are defined in a hierarchical quad-tree manner. Unlike H.264, where the basic coding block is a macroblock of fixed size 16x16, HEVC defines a Coding Tree Unit (CTU) of a maximum size of 64x64. Each CTU can be divided into smaller units in a hierarchical quad-tree manner and can represent smaller blocks down to size 4x4. Similarly, the transforms used in HEVC can have different sizes, starting from 4x4 and going up to 32x32. Utilizing large blocks and transforms contribute to the major gain of HEVC, especially at high resolutions.

Entropy coding

HEVC uses a single entropy coding engine, which is based on Context Adaptive Binary Arithmetic Coding (CABAC) [CABAC], whereas H.264 uses two distinct entropy coding engines. CABAC in HEVC shares many similarities with CABAC of H.264, but contains several improvements. Those include improvements in coding efficiency and lowered implementation complexity, especially for parallel architectures.

In-loop filtering

H.264 includes an in-loop adaptive deblocking filter, where the blocking artifacts around the transform edges in the reconstructed picture are smoothed to improve the picture quality and compression efficiency. In HEVC, a similar deblocking filter is employed but with somewhat lower complexity. In addition, pictures undergo a subsequent filtering operation called Sample Adaptive Offset (SAO), which is a new design element in HEVC. SAO basically adds a pixel-level offset in an adaptive manner and usually acts as a de-ringing filter. It is observed that SAO improves the picture quality, especially around sharp edges contributing substantially to visual quality improvements of HEVC.

Motion prediction and coding

There have been a number of improvements in this area that are summarized as follows. The first category is motion merge and advanced motion vector prediction (AMVP) modes. The motion information of a prediction block can be inferred from the spatially or temporally neighboring blocks. This is similar to the DIRECT mode in H.264 but includes new aspects to incorporate the flexible quad-tree structure and methods to improve the parallel implementations. In addition, the motion vector predictor can be signaled for improved efficiency. The second category is high-precision interpolation. The interpolation filter length is increased to 8-tap from 6-tap, which improves the coding efficiency but also comes with increased complexity. In addition, the interpolation filter is defined with higher precision without any intermediate rounding operations to further improve the coding efficiency.

Intra prediction and intra coding

Compared to 8 intra prediction modes in H.264, HEVC supports angular intra prediction with 33 directions. This increased flexibility improves both objective coding efficiency and visual quality as the edges can be better predicted and ringing artifacts around the edges can be reduced. In addition, the reference samples are adaptively smoothed based on the prediction

direction. To avoid contouring artifacts a new interpolative prediction generation is included to improve the visual quality. Furthermore, discrete sine transform (DST) is utilized instead of traditional discrete cosine transform (DCT) for 4x4 intra transform blocks.

Other coding-tool features

HEVC includes some tools for lossless coding and efficient screen content coding, such as skipping the transform for certain blocks. These tools are particularly useful for example when streaming the user-interface of a mobile device to a large display.

1.1.2 Systems and Transport Interfaces

HEVC inherited the basic systems and transport interfaces designs, such as the NAL-unit-based syntax structure, the hierarchical syntax and data unit structure from sequence-level parameter sets, multi-picture-level or picture-level parameter sets, slice-level header parameters, lower-level parameters, the supplemental enhancement information (SEI) message mechanism, the hypothetical reference decoder (HRD) based video buffering model, and so on. In the following, a list of differences in these aspects compared to H.264 is summarized.

Video parameter set

A new type of parameter set, called video parameter set (VPS), was introduced. For the first (2013) version of [HEVC], the video parameter set NAL unit is required to be available prior to its activation, while the information contained in the video parameter set is not necessary for operation of the decoding process. For future HEVC extensions, such as the 3D or scalable extensions, the video parameter set is expected to include information necessary for operation of the decoding process, e.g. decoding dependency or information for reference picture set construction of enhancement layers. The VPS provides a "big picture" of a bitstream, including what types of operation points are provided, the profile, tier, and level of the operation points, and some other high-level properties of the bitstream

that can be used as the basis for session negotiation and content selection, etc. (see Section 7.1).

Profile, tier and level

The profile, tier and level syntax structure that can be included in both VPS and sequence parameter set (SPS) includes 12 bytes of data to describe the entire bitstream (including all temporally scalable layers, which are referred to as sub-layers in the HEVC specification), and can optionally include more profile, tier and level information pertaining to individual temporally scalable layers. The profile indicator indicates the "best viewed as" profile when the bitstream conforms to multiple profiles, similar to the major brand concept in the ISO base media file format (ISOBMFF) [ISOBMFF] and file formats derived based on ISOBMFF, such as the 3GPP file format [3GPPFF]. The profile, tier and level syntax structure also includes indications such as 1) whether the bitstream is free of frame-packed content, 2) whether the bitstream is free of interlaced source content, and 3) whether the bitstream is free of field pictures. When the answer is yes for both 2) and 3), the bitstream contains only frame pictures of progressive source. Based on these indications, clients/players without support of post-processing functionalities for handling of frame-packed, interlaced source content or field pictures can reject those bitstreams that contain such pictures.

Bitstream and elementary stream

HEVC includes a definition of an elementary stream, which is new compared to H.264. An elementary stream consists of a sequence of one or more bitstreams. An elementary stream that consists of two or more bitstreams has typically been formed by splicing together two or more bitstreams (or parts thereof). When an elementary stream contains more than one bitstream, the last NAL unit of the last access unit of a bitstream (except the last bitstream in the elementary stream) must contain an end of bitstream NAL unit and the first access unit of the subsequent bitstream must be an intra random access point (IRAP) access unit. This IRAP access unit may be a clean random access (CRA),

broken link access (BLA), or instantaneous decoding refresh (IDR) access unit.

Random access support

HEVC includes signaling in the NAL unit header, through NAL unit types, of IRAP pictures beyond IDR pictures. Three types of IRAP pictures, namely IDR, CRA and BLA pictures are supported, wherein IDR pictures are conventionally referred to as closed group-of-pictures (closed-GOP) random access points, and CRA and BLA pictures are those conventionally referred to as open-GOP random access points. BLA pictures usually originate from splicing of two bitstreams or part thereof at a CRA picture, e.g. during stream switching. To enable better systems usage of IRAP pictures, altogether six different NAL units are defined to signal the properties of the IRAP pictures, which can be used to better match the stream access point (SAP) types as defined in the ISOBMFF [ISOBMFF], which are utilized for random access support in both 3GP-DASH [3GPDASH] and MPEG DASH [MPEGDASH]. Pictures following an IRAP picture in decoding order and preceding the IRAP picture in output order are referred to as leading pictures associated with the IRAP picture. There are two types of leading pictures, namely random access decodable leading (RADL) pictures and random access skipped leading (RASL) pictures. RADL pictures are decodable when the decoding started at the associated IRAP picture, and RASL pictures are not decodable when the decoding started at the associated IRAP picture and are usually discarded. HEVC provides mechanisms to enable the specification of conformance of bitstreams with RASL pictures being discarded, thus to provide a standard-compliant way to enable systems components to discard RASL pictures when needed.

Temporal scalability support

HEVC includes an improved support of temporal scalability, by inclusion of the signaling of TemporalId in the NAL unit header, the restriction that pictures of a particular temporal sub-layer cannot be used for inter prediction reference by pictures of a lower temporal sub-layer, the sub-bitstream extraction process, and the requirement that each sub-bitstream extraction output be

a conforming bitstream. Media-aware network elements (MANEs) can utilize the TemporalId in the NAL unit header for stream adaptation purposes based on temporal scalability.

Temporal sub-layer switching support

HEVC specifies, through NAL unit types present in the NAL unit header, the signaling of temporal sub-layer access (TSA) and stepwise temporal sub-layer access (STSA). A TSA picture and pictures following the TSA picture in decoding order do not use pictures prior to the TSA picture in decoding order with TemporalId greater than or equal to that of the TSA picture for inter prediction reference. A TSA picture enables up-switching, at the TSA picture, to the sub-layer containing the TSA picture or any higher sub-layer, from the immediately lower sub-layer. An STSA picture does not use pictures with the same TemporalId as the STSA picture for inter prediction reference. Pictures following an STSA picture in decoding order with the same TemporalId as the STSA picture do not use pictures prior to the STSA picture in decoding order with the same TemporalId as the STSA picture for inter prediction reference. An STSA picture enables up-switching, at the STSA picture, to the sub-layer containing the STSA picture, from the immediately lower sub-layer.

Sub-layer reference or non-reference pictures

The concept and signaling of reference/non-reference pictures in HEVC are different from H.264. In H.264, if a picture may be used by any other picture for inter prediction reference, it is a reference picture; otherwise it is a non-reference picture, and this is signaled by two bits in the NAL unit header. In HEVC, a picture is called a reference picture only when it is marked as "used for reference". In addition, the concept of sub-layer reference picture was introduced. If a picture may be used by another other picture with the same TemporalId for inter prediction reference, it is a sub-layer reference picture; otherwise it is a sub-layer non-reference picture. Whether a picture is a sub-layer reference picture or sub-layer non-reference picture is signaled through NAL unit type values.

Extensibility

Besides the TemporalId in the NAL unit header, HEVC also includes the signaling of a six-bit layer ID in the NAL unit header, which must be equal to 0 for a single-layer bitstream. Extension mechanisms have been included in VPS, SPS, PPS, SEI NAL unit, slice headers, and so on. All these extension mechanisms enable future extensions in a backward compatible manner, such that bitstreams encoded according to potential future HEVC extensions can be fed to then-legacy decoders (e.g. HEVC version 1 decoders) and the then-legacy decoders can decode and output the base layer bitstream.

Bitstream extraction

HEVC includes a bitstream extraction process as an integral part of the overall decoding process, as well as specification of the use of the bitstream extraction process in description of bitstream conformance tests as part of the hypothetical reference decoder (HRD) specification.

Reference picture management

The reference picture management of HEVC, including reference picture marking and removal from the decoded picture buffer (DPB) as well as reference picture list construction (RPLC), differs from that of H.264. Instead of the sliding window plus adaptive memory management control operation (MMCO) based reference picture marking mechanism in H.264, HEVC specifies a reference picture set (RPS) based reference picture management and marking mechanism, and the RPLC is consequently based on the RPS mechanism. A reference picture set consists of a set of reference pictures associated with a picture, consisting of all reference pictures that are prior to the associated picture in decoding order, that may be used for inter prediction of the associated picture or any picture following the associated picture in decoding order. The reference picture set consists of five lists of reference pictures; RefPicSetStCurrBefore, RefPicSetStCurrAfter, RefPicSetStFoll, RefPicSetLtCurr and RefPicSetLtFoll. RefPicSetStCurrBefore, RefPicSetStCurrAfter and RefPicSetLtCurr contain all reference pictures that may be used

in inter prediction of the current picture and that may be used in inter prediction of one or more of the pictures following the current picture in decoding order. RefPicSetStFoll and RefPicSetLtFoll consist of all reference pictures that are not used in inter prediction of the current picture but may be used in inter prediction of one or more of the pictures following the current picture in decoding order. RPS provides an "intra-coded" signaling of the DPB status, instead of an "inter-coded" signaling, mainly for improved error resilience. The RPLC process in HEVC is based on the RPS, by signaling an index to an RPS subset for each reference index; this process is simpler than the RPLC process in H.264.

Ultra low delay support

HEVC specifies a sub-picture-level HRD operation, for support of the so-called ultra-low delay. The mechanism specifies a standard-compliant way to enable delay reduction below one picture interval. Sub-picture-level coded picture buffer (CPB) and DPB parameters may be signaled, and utilization of these information for the derivation of CPB timing (wherein the CPB removal time corresponds to decoding time) and DPB output timing (display time) is specified. Decoders are allowed to operate the HRD at the conventional access-unit-level, even when the sub-picture-level HRD parameters are present.

New SEI messages

HEVC inherits many H.264 SEI messages with changes in syntax and/or semantics making them applicable to HEVC. Additionally, there are a few new SEI messages reviewed briefly in the following paragraphs.

The display orientation SEI message informs the decoder of a transformation that is recommended to be applied to the cropped decoded picture prior to display, such that the pictures can be properly displayed, e.g. in an upside-up manner.

The structure of pictures SEI message provides information on the NAL unit types, picture order count values, and prediction dependencies of a sequence of pictures. The SEI message can be

used for example for concluding what impact a lost picture has on other pictures.

The decoded picture hash SEI message provides a checksum derived from the sample values of a decoded picture. It can be used for detecting whether a picture was correctly received and decoded.

The active parameter sets SEI message includes the IDs of the active video parameter set and the active sequence parameter set and can be used to activate VPSs and SPSs. In addition, the SEI message includes the following indications: 1) An indication of whether "full random accessibility" is supported (when supported, all parameter sets needed for decoding of the remaining of the bitstream when random accessing from the beginning of the current CVS by completely discarding all access units earlier in decoding order are present in the remaining bitstream and all coded pictures in the remaining bitstream can be correctly decoded); 2) An indication of whether there is no parameter set within the current CVS that updates another parameter set of the same type preceding in decoding order. An update of a parameter set refers to the use of the same parameter set ID but with some other parameters changed. If this property is true for all CVSs in the bitstream, then all parameter sets can be sent out-of-band before session start.

The decoding unit information SEI message provides coded picture buffer removal delay information for a decoding unit. The message can be used in very-low-delay buffering operations.

The region refresh information SEI message can be used together with the recovery point SEI message (present in both H.264 and HEVC) for improved support of gradual decoding refresh. This supports random access from inter-coded pictures, wherein complete pictures can be correctly decoded or recovered after an indicated number of pictures in output/display order.

1.1.3 Parallel Processing Support

The reportedly significantly higher encoding computational demand of HEVC over H.264, in conjunction with the ever increasing video resolution (both spatially and temporally) required by the

market, led to the adoption of VCL coding tools specifically targeted to allow for parallelization on the sub-picture level. That is, parallelization occurs, at the minimum, at the granularity of an integer number of CTUs. The targets for this type of high-level parallelization are multicore CPUs and DSPs as well as multiprocessor systems. In a system design, to be useful, these tools require signaling support, which is provided in Section 7 of this memo. This section provides a brief overview of the tools available in [HEVC].

Many of the tools incorporated in HEVC were designed keeping in mind the potential parallel implementations in multi-core/multi-processor architectures. Specifically, for parallelization, four picture partition strategies, as described below, are available.

Slices are segments of the bitstream that can be reconstructed independently from other slices within the same picture (though there may still be interdependencies through loop filtering operations). Slices are the only tool that can be used for parallelization that is also available, in virtually identical form, in H.264. Slices based parallelization does not require much inter-processor or inter-core communication (except for inter-processor or inter-core data sharing for motion compensation when decoding a predictively coded picture, which is typically much heavier than inter-processor or inter-core data sharing due to in-picture prediction), as slices are designed to be independently decodable. However, for the same reason, slices can require some coding overhead. Further, slices (in contrast to some of the other tools mentioned below) also serve as the key mechanism for bitstream partitioning to match Maximum Transfer Unit (MTU) size requirements, due to the in-picture independence of slices and the fact that each regular slice is encapsulated in its own NAL unit. In many cases, the goal of parallelization and the goal of MTU size matching can place contradicting demands to the slice layout in a picture. The realization of this situation led to the development of the more advanced tools mentioned below.

Dependent slice segments allow for fragmentation of a coded slice into fragments at CTU boundaries without breaking any in-picture prediction mechanism. They are complementary to the

fragmentation mechanism described in this memo in that they need the cooperation of the encoder. As a dependent slice segment necessarily contains an integer number of CTUs, a decoder using multiple cores operating on CTUs can process a dependent slice segment without communicating parts of the slice segment's bitstream to other cores. Fragmentation, as specified in this memo, in contrast, does not guarantee that a fragment contains an integer number of CTUs.

In wavefront parallel processing (WPP), the picture is partitioned into rows of CTUs. Entropy decoding and prediction are allowed to use data from CTUs in other partitions. Parallel processing is possible through parallel decoding of CTU rows, where the start of the decoding of a row is delayed by two CTUs, so to ensure that data related to a CTU above and to the right of the subject CTU is available before the subject CTU is being decoded. Using this staggered start (which appears like a wavefront when represented graphically), parallelization is possible with up to as many processors/cores as the picture contains CTU rows.

Because in-picture prediction between neighboring CTU rows within a picture is allowed, the required inter-processor/inter-core communication to enable in-picture prediction can be substantial. The WPP partitioning does not result in the creation of more NAL units compared to when it is not applied, thus WPP cannot be used for MTU size matching, though slices can be used in combination for that purpose.

Tiles define horizontal and vertical boundaries that partition a picture into tile columns and rows. The scan order of CTUs is changed to be local within a tile (in the order of a CTU raster scan of a tile), before decoding the top-left CTU of the next tile in the order of tile raster scan of a picture. Similar to slices, tiles break in-picture prediction dependencies (including entropy decoding dependencies). However, they do not need to be included into individual NAL units (same as WPP in this regard), hence tiles cannot be used for MTU size matching, though slices can be used in combination for that purpose. Each tile can be processed by one processor/core, and the inter-processor/inter-core communication required for in-picture prediction between

processing units decoding neighboring tiles is limited to conveying the shared slice header in cases a slice is spanning more than one tile, and loop filtering related sharing of reconstructed samples and metadata. Insofar, tiles are less demanding in terms of inter-processor communication bandwidth compared to WPP due to the in-picture independence between two neighboring partitions.

1.1.4 NAL Unit Header

HEVC maintains the NAL unit concept of H.264 with modifications. HEVC uses a two-byte NAL unit header, as shown in Figure 1. The payload of a NAL unit refers to the NAL unit excluding the NAL unit header.

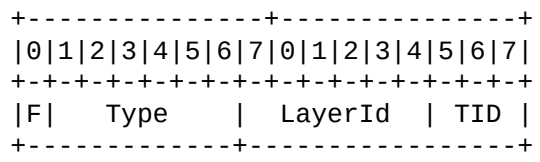

Figure 1 The structure of HEVC NAL unit header

The semantics of the fields in the NAL unit header are as specified in [HEVC] and described briefly below for convenience. In addition to the name and size of each field, the corresponding syntax element name in [HEVC] is also provided.

F: 1 bit
forbidden_zero_bit. Required to be zero in [HEVC]. Note that the inclusion of this bit in the NAL unit header was to enable transport of HEVC video over MPEG-2 transport systems (avoidance of start code emulations) [MPEG2S]. In the context of this memo, the value 1 may be used to indicate a syntax violation, e.g. for a NAL unit resulted from aggregating a number of fragmented units of a NAL unit but missing the last fragment, as described in Section 4.4.3.

Type: 6 bits

nal_unit_type. This field specifies the NAL unit type as defined in Table 7-1 of [HEVC]. If the most significant bit of this field of a NAL unit is equal to 0 (i.e. the value of this field is less than 32), the NAL unit is a VCL NAL unit. Otherwise, the NAL unit is a non-VCL NAL unit. For a reference of all currently defined NAL unit types and their semantics, please refer to Section 7.4.1 in [HEVC].

LayerId: 6 bits
nuh_layer_id. Required to be equal to zero in [HEVC]. It is anticipated that in future scalable or 3D video coding extensions of this specification, this syntax element will be used to identify additional layers that may be present in the CVS, wherein a layer may be, e.g. a spatial scalable layer, a quality scalable layer, a texture view, or a depth view.

TID: 3 bits
nuh_temporal_id_plus1. This field specifies the temporal identifier of the NAL unit plus 1. The value of TemporalId is equal to TID minus 1. A TID value of 0 is illegal to ensure that there is at least one bit in the NAL unit header equal to 1, so to enable independent considerations of start code emulations in the NAL unit header and in the NAL unit payload data.

1.2 Overview of the Payload Format

This payload format defines the following processes required for transport of HEVC coded data over RTP [RFC3550]:

- Usage of RTP header with this payload format
- Packetization of HEVC coded NAL units into RTP packets using three types of payload structures, namely single NAL unit packet, aggregation packet, and fragment unit
- Transmission of HEVC NAL units of the same bitstream within a single RTP stream or multiple RTP streams (within one or more RTP sessions), where within an RTP stream transmission of NAL units may be either non-interleaved (i.e. the transmission

order of NAL units is the same as their decoding order) or interleaved (i.e. the transmission order of NAL units is different from their decoding order)

- Media type parameters to be used with the Session Description Protocol (SDP) [RFC4566]

- A payload header extension mechanism and data structures for enhanced support of temporal scalability based on that extension mechanism.

2 Conventions

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 BCP 14, RFC 2119 [RFC2119].

In this document, these key words will appear with that interpretation only when in ALL CAPS. Lower case uses of these words are not to be interpreted as carrying the RFC 2119 significance.

This specification uses the notion of setting and clearing a bit when bit fields are handled. Setting a bit is the same as assigning that bit the value of 1 (On). Clearing a bit is the same as assigning that bit the value of 0 (Off).

3 Definitions and Abbreviations

3.1 Definitions

This document uses the terms and definitions of [HEVC]. Section 3.1.1 lists relevant definitions copied from [HEVC] (the April 2013 version of the H.265 specification) for convenience. Section 3.1.2 provides definitions specific to this memo.

3.1.1 Definitions from the HEVC Specification

access unit: A set of NAL units that are associated with each other according to a specified classification rule, are

consecutive in decoding order, and contain exactly one coded picture.

BLA access unit: An access unit in which the coded picture is a BLA picture.

BLA picture: An IRAP picture for which each VCL NAL unit has nal_unit_type equal to BLA_W_LP, BLA_W_RADL, or BLA_N_LP.

coded video sequence (CVS): A sequence of access units that consists, in decoding order, of an IRAP access unit with NoRaslOutputFlag equal to 1, followed by zero or more access units that are not IRAP access units with NoRaslOutputFlag equal to 1, including all subsequent access units up to but not including any subsequent access unit that is an IRAP access unit with NoRaslOutputFlag equal to 1.

Informative note: An IRAP access unit may be an IDR access unit, a BLA access unit, or a CRA access unit. The value of NoRaslOutputFlag is equal to 1 for each IDR access unit, each BLA access unit, and each CRA access unit that is the first access unit in the bitstream in decoding order, is the first access unit that follows an end of sequence NAL unit in decoding order, or has HandleCraAsBlaFlag equal to 1.

CRA access unit: An access unit in which the coded picture is a CRA picture.

CRA picture: A RAP picture for which each VCL NAL unit has nal_unit_type equal to CRA_NUT.

IDR access unit: An access unit in which the coded picture is an IDR picture.

IDR picture: A RAP picture for which each VCL NAL unit has nal_unit_type equal to IDR_W_RADL or IDR_N_LP.

IRAP access unit: An access unit in which the coded picture is an IRAP picture.

IRAP picture: A coded picture for which each VCL NAL unit has nal_unit_type in the range of BLA_W_LP (16) to RSV_IRAP_VCL23 (23), inclusive.

layer: A set of VCL NAL units that all have a particular value of nuh_layer_id and the associated non-VCL NAL units, or one of a set of syntactical structures having a hierarchical relationship.

operation point: bitstream created from another bitstream by operation of the sub-bitstream extraction process with the another bitstream, a target highest TemporalId, and a target layer identifier list as inputs.

random access: The act of starting the decoding process for a bitstream at a point other than the beginning of the bitstream.

sub-layer: A temporal scalable layer of a temporal scalable bitstream consisting of VCL NAL units with a particular value of the TemporalId variable, and the associated non-VCL NAL units.

sub-layer representation: A subset of the bitstream consisting of NAL units of a particular sub-layer and the lower sub-layers.

tile: A rectangular region of coding tree blocks within a particular tile column and a particular tile row in a picture.

tile column: A rectangular region of coding tree blocks having a height equal to the height of the picture and a width specified by syntax elements in the picture parameter set.

tile row: A rectangular region of coding tree blocks having a height specified by syntax elements in the picture parameter set and a width equal to the width of the picture.

3.1.2 Definitions Specific to This Memo

dependee RTP stream: An RTP stream on which another RTP stream depends. All RTP streams in an MRST or MRMT except for the highest RTP stream are dependee RTP streams.

highest RTP stream: The RTP stream on which no other RTP stream depends. The RTP stream in an SRST is the highest RTP stream.

media aware network element (MANE): A network element, such as a middlebox, selective forwarding unit, or application layer gateway that is capable of parsing certain aspects of the RTP payload headers or the RTP payload and reacting to their contents.

Informative note: The concept of a MANE goes beyond normal routers or gateways in that a MANE has to be aware of the signaling (e.g. to learn about the payload type mappings of the media streams), and in that it has to be trusted when working with SRTP. The advantage of using MANEs is that they allow packets to be dropped according to the needs of the media coding. For example, if a MANE has to drop packets due to congestion on a certain link, it can identify and remove those packets whose elimination produces the least adverse effect on the user experience. After dropping packets, MANEs must rewrite RTCP packets to match the changes to the RTP stream as specified in Section 7 of [RFC3550].

Media Transport: As used in the MRST, MRMT, and SRST definitions below, Media Transport denotes the transport of packets over a transport association identified by a 5-tuple (source address, source port, destination address, destination port, transport protocol). See also Section 2.1.13 of [I-D.ietf-avtext-rtp-grouping-taxonomy].

Informative note: The term "bitstream" in this document is equivalent to the term "encoded stream" in [I-D.ietf-avtext-rtp-grouping-taxonomy].

Multiple RTP streams on a Single Transport (MRST): Multiple RTP streams carrying a single HEVC bitstream on a Single Transport. See also Section 3.5 of [I-D.ietf-avtext-rtp-grouping-taxonomy].

Multiple RTP streams on Multiple Transports (MRMT): Multiple RTP streams carrying a single HEVC bitstream on Multiple Transports. See also Section 3.5 of [I-D.ietf-avtext-rtp-grouping-taxonomy].

NAL unit decoding order: A NAL unit order that conforms to the constraints on NAL unit order given in Section 7.4.2.4 in [HEVC].

NAL unit output order: A NAL unit order in which NAL units of different access units are in the output order of the decoded pictures corresponding to the access units, as specified in [HEVC], and in which NAL units within an access unit are in their decoding order.

NAL-unit-like structure: A data structure that is similar to NAL units in the sense that it also has a NAL unit header and a payload, with a difference that the payload does not follow the start code emulation prevention mechanism required for the NAL unit syntax as specified in Section 7.3.1.1 of [HEVC]. Examples NAL-unit-like structures defined in this memo are packet payloads of AP, PACI, and FU packets.

NALU-time: The value that the RTP timestamp would have if the NAL unit would be transported in its own RTP packet.

RTP stream: See [I-D.ietf-avtext-rtp-grouping-taxonomy]. Within the scope of this memo, one RTP stream is utilized to transport one or more temporal sub-layers.

Single RTP stream on a Single Transport (SRST): Single RTP stream carrying a single HEVC bitstream on a Single (Media) Transport. See also Section 3.5 of [I-D.ietf-avtext-rtp-grouping-taxonomy].

transmission order: The order of packets in ascending RTP sequence number order (in modulo arithmetic). Within an aggregation packet, the NAL unit transmission order is the same as the order of appearance of NAL units in the packet.

3.2 Abbreviations

AP Aggregation Packet

BLA Broken Link Access

CRA Clean Random Access

CTB Coding Tree Block

CTU Coding Tree Unit

CVS	Coded Video Sequence
DPH	Decoded Picture Hash
FU	Fragmentation Unit
HRD	Hypothetical Reference Decoder
IDR	Instantaneous Decoding Refresh
IRAP	Intra Random Access Point
MANE	Media Aware Network Element
MRMT	Multiple RTP streams on Multiple Transports
MRST	Multiple RTP streams on a Single Transport
MTU	Maximum Transfer Unit
NAL	Network Abstraction Layer
NALU	Network Abstraction Layer Unit
PACI	PAyload Content Information
PHES	Payload Header Extension Structure
PPS	Picture Parameter Set
RADL	Random Access Decodable Leading (Picture)
RASL	Random Access Skipped Leading (Picture)
RPS	Reference Picture Set
SEI	Supplemental Enhancement Information
SPS	Sequence Parameter Set
SRST	Single RTP stream on a Single Transport
STSA	Step-wise Temporal Sub-layer Access

TSA Temporal Sub-layer Access

TSCI Temporal Scalability Control Information

VCL Video Coding Layer

VPS Video Parameter Set

4 RTP Payload Format

4.1 RTP Header Usage

The format of the RTP header is specified in [RFC3550] and reprinted in Figure 2 for convenience. This payload format uses the fields of the header in a manner consistent with that specification.

The RTP payload (and the settings for some RTP header bits) for aggregation packets and fragmentation units are specified in Sections 4.4.2 and 4.4.3, respectively.

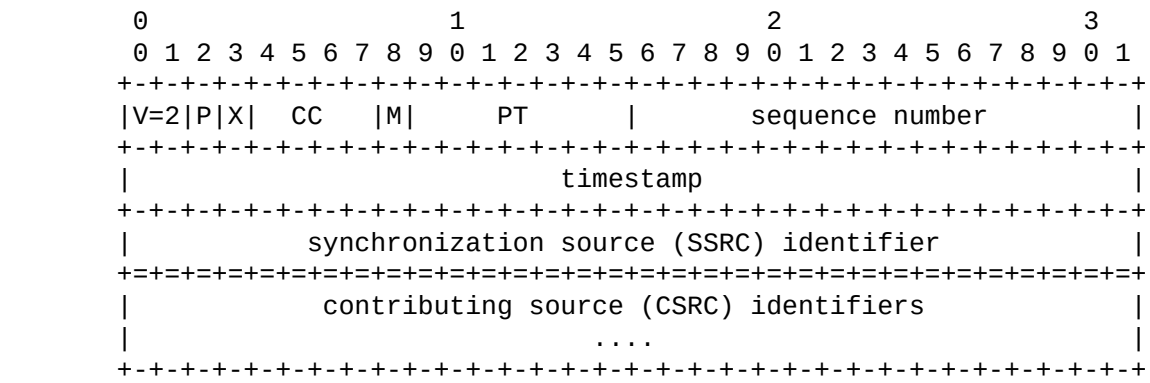

Figure 2 RTP header according to [RFC3550]

The RTP header information to be set according to this RTP payload format is set as follows:

Marker bit (M): 1 bit

Set for the last packet of the access unit, carried in the current RTP stream. This is in line with the normal use of the M bit in video formats to allow an efficient playout buffer handling. When MRST or MRMT is in use, if an access unit appears in multiple RTP streams, the marker bit is set on each RTP stream's last packet of the access unit.

Informative note: The content of a NAL unit does not tell whether or not the NAL unit is the last NAL unit, in decoding order, of an access unit. An RTP sender implementation may obtain these information from the video encoder. If, however, the implementation cannot obtain these information directly from the encoder, e.g. when the bitstream was pre-encoded, and also there is no timestamp allocated for each NAL unit, then the sender implementation can inspect subsequent NAL units in decoding order to determine whether or not the NAL unit is the last NAL unit of an access unit as follows. A NAL unit is determined to be the last NAL unit of an access unit if it is the last NAL unit of the bitstream. A NAL unit naluX is also determined to be the last NAL unit of an access unit if both the following conditions are true: 1) the next VCL NAL unit naluY in decoding order has the high-order bit of the first byte after its NAL unit header equal to 1, and 2) all NAL units between naluX and naluY, when present, have nal_unit_type in the range of 32 to 35, inclusive, equal to 39, or in the ranges of 41 to 44, inclusive, or 48 to 55, inclusive.

Payload type (PT): 7 bits

The assignment of an RTP payload type for this new packet format is outside the scope of this document and will not be specified here. The assignment of a payload type has to be performed either through the profile used or in a dynamic way.

Informative note: It is not required to use different payload type values for different RTP streams in MRST or MRMT.

Sequence number (SN): 16 bits

Set and used in accordance with RFC 3550 [RFC3550].

Timestamp: 32 bits

The RTP timestamp is set to the sampling timestamp of the content. A 90 kHz clock rate MUST be used.

If the NAL unit has no timing properties of its own (e.g. parameter set and SEI NAL units), the RTP timestamp MUST be set to the RTP timestamp of the coded picture of the access unit in which the NAL unit (according to Section 7.4.2.4.4 of [HEVC]) is included.

Receivers MUST use the RTP timestamp for the display process, even when the bitstream contains picture timing SEI messages or decoding unit information SEI messages as specified in [HEVC]. However, this does not mean that picture timing SEI messages in the bitstream should be discarded, as picture timing SEI messages may contain frame-field information that is important in appropriately rendering interlaced video.

Synchronization source (SSRC): 32-bits

Used to identify the source of the RTP packets. When using SRST, by definition a single SSRC is used for all parts of a single bitstream. In MRST or MRMT, different SSRCs are used for each RTP stream containing a subset of the sub-layers of the single (temporally scalable) bitstream. A receiver is required to correctly associate the set of SSRCs that are included parts of the same bitstream.

4.2 Payload Header Usage

The first two bytes of the payload of an RTP packet are referred to as the payload header. The payload header consists of the same fields (F, Type, LayerId, and TID) as the NAL unit header as shown in Section 1.1.4, irrespective of the type of the payload structure.

The TID value indicates (among other things) the relative importance of an RTP packet, for example because NAL units belonging to higher temporal sub-layers are not used for the decoding of lower temporal sub-layers. A lower value of TID indicates a higher importance. More important NAL units MAY be better protected against transmission losses than less important NAL units.

4.3 Transmission Modes

This memo enables transmission of an HEVC bitstream over

- a single RTP stream on a single Media Transport (SRST),
- multiple RTP streams over a single Media Transport (MRST), or
- multiple RTP streams over multiple Media Transports (MRMT).

Informative Note: While this specification enables the use of MRST within the H.265 RTP payload, the signaling of MRST within SDP Offer/Answer is not fully specified at the time of this writing. See [RFC5576] and [RFC5583] for what is supported today as well as [I-D.ietf-avtcore-rtp-multi-stream] and [I-D.ietf-mmusic-sdp-bundle-negotiation] for future directions.

When in MRMT, the dependency of one RTP stream on another RTP stream is typically indicated as specified in [RFC5583]. [RFC5583] can also be utilized to specify dependencies within MRST, but only if the RTP streams utilize distinct payload types.

SRST or MRST SHOULD be used for point-to-point unicast scenarios, while MRMT SHOULD be used for point-to-multipoint multicast scenarios where different receivers require different operation points of the same HEVC bitstream, to improve bandwidth utilizing efficiency.

Informative note: A multicast may degrade to a unicast after all but one receivers have left (this is a justification of the first "SHOULD" instead of "MUST"), and there might be scenarios where MRMT is desirable but not possible e.g. when IP multicast is not deployed in certain network (this is a justification of the second "SHOULD" instead of "MUST").

The transmission mode is indicated by the tx-mode media parameter (see Section 7.1). If tx-mode is equal to "SRST", SRST MUST be used. Otherwise, if tx-mode is equal to "MRST", MRST MUST be used. Otherwise (tx-mode is equal to "MRMT"), MRMT MUST be used.

Informative note: When an RTP stream does not depend on other RTP streams, any of SRST, MRST and MRMT may be in use for the RTP stream.

Receivers MUST support all of SRST, MRST, and MRMT.

Informative note: The required support of MRMT by receivers does not imply that multicast must be supported by receivers.

4.4 Payload Structures

Four different types of RTP packet payload structures are specified. A receiver can identify the type of an RTP packet payload through the Type field in the payload header.

The four different payload structures are as follows:

- Single NAL unit packet: Contains a single NAL unit in the payload, and the NAL unit header of the NAL unit also serves as the payload header. This payload structure is specified in Section 4.4.1.

- Aggregation packet (AP): Contains more than one NAL unit within one access unit. This payload structure is specified in Section 4.4.2.

- Fragmentation unit (FU): Contains a subset of a single NAL unit. This payload structure is specified in Section 4.4.3.

- PACI carrying RTP packet: Contains a payload header (that differs from other payload headers for efficiency), a Payload Header Extension Structure (PHES), and a PACI payload. This payload structure is specified in Section 4.4.4.

4.4.1 Single NAL Unit Packets

A single NAL unit packet contains exactly one NAL unit, and consists of a payload header (denoted as PayloadHdr), a conditional 16-bit DONL field (in network byte order), and the NAL unit payload data (the NAL unit excluding its NAL unit header) of the contained NAL unit, as shown in Figure 3.

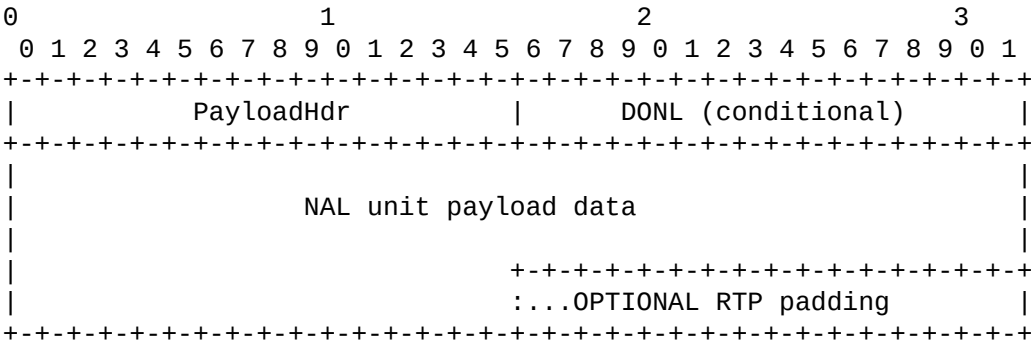

Figure 3 The structure a single NAL unit packet

The payload header SHOULD be an exact copy of the NAL unit header of the contained NAL unit. However, the Type (i.e. nal_unit_type) field MAY be changed, e.g. when it is desirable to handle a CRA picture to be a BLA picture [JCTVC-J0107].

The DONL field, when present, specifies the value of the 16 least significant bits of the decoding order number of the contained NAL unit. If sprop-max-don-diff is greater than 0 for any of the RTP streams, the DONL field MUST be present, and the variable DON for the contained NAL unit is derived as equal to the value of the DONL field. Otherwise (sprop-max-don-diff is equal to 0 for all the RTP streams), the DONL field MUST NOT be present.

4.4.2 Aggregation Packets (APs)

Aggregation packets (APs) are introduced to enable the reduction of packetization overhead for small NAL units, such as most of the non-VCL NAL units, which are often only a few octets in size.

An AP aggregates NAL units within one access unit. Each NAL unit to be carried in an AP is encapsulated in an aggregation unit. NAL units aggregated in one AP are in NAL unit decoding order.

An AP consists of a payload header (denoted as PayloadHdr) followed by two or more aggregation units, as shown in Figure 4.

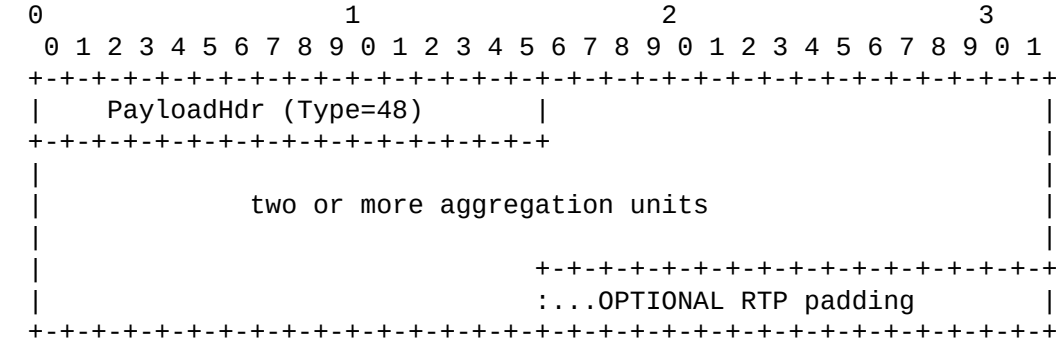

Figure 4 The structure of an aggregation packet

The fields in the payload header are set as follows. The F bit MUST be equal to 0 if the F bit of each aggregated NAL unit is equal to zero; otherwise, it MUST be equal to 1. The Type field MUST be equal to 48. The value of LayerId MUST be equal to the lowest value of LayerId of all the aggregated NAL units. The value of TID MUST be the lowest value of TID of all the aggregated NAL units.

> Informative Note: All VCL NAL units in an AP have the same TID value since they belong to the same access unit. However, an AP may contain non-VCL NAL units for which the TID value in the NAL unit header may be different than the TID value of the VCL NAL units in the same AP.

An AP MUST carry at least two aggregation units and can carry as many aggregation units as necessary; however, the total amount of data in an AP obviously MUST fit into an IP packet, and the size SHOULD be chosen so that the resulting IP packet is smaller than the MTU size so to avoid IP layer fragmentation. An AP MUST NOT

contain Fragmentation Units (FUs) specified in Section 4.4.3. APs MUST NOT be nested; i.e. an AP must not contain another AP.

The first aggregation unit in an AP consists of a conditional 16-bit DONL field (in network byte order) followed by a 16-bit unsigned size information (in network byte order) that indicates the size of the NAL unit in bytes (excluding these two octets, but including the NAL unit header), followed by the NAL unit itself, including its NAL unit header, as shown in Figure 5.

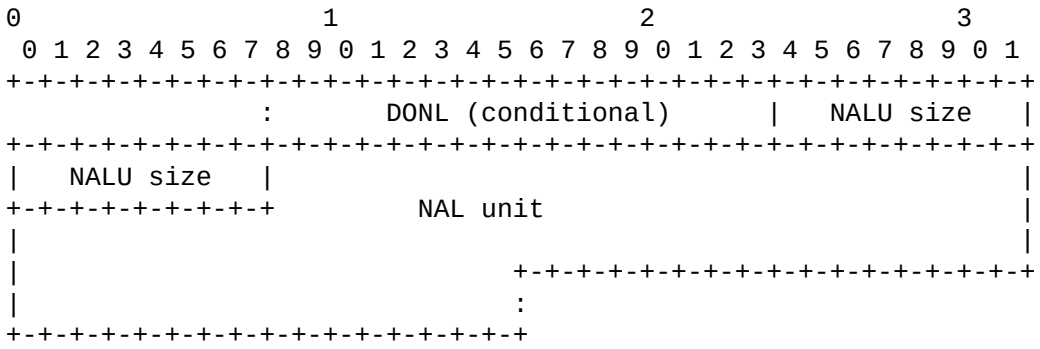

Figure 5 The structure of the first aggregation unit in an AP

The DONL field, when present, specifies the value of the 16 least significant bits of the decoding order number of the aggregated NAL unit.

If sprop-max-don-diff is greater than 0 for any of the RTP streams, the DONL field MUST be present in an aggregation unit that is the first aggregation unit in an AP, and the variable DON for the aggregated NAL unit is derived as equal to the value of the DONL field. Otherwise (sprop-max-don-diff is equal to 0 for all the RTP streams), the DONL field MUST NOT be present in an aggregation unit that is the first aggregation unit in an AP.

An aggregation unit that is not the first aggregation unit in an AP consists of a conditional 8-bit DOND field followed by a 16-bit unsigned size information (in network byte order) that indicates the size of the NAL unit in bytes (excluding these two

octets, but including the NAL unit header), followed by the NAL unit itself, including its NAL unit header, as shown in Figure 6.

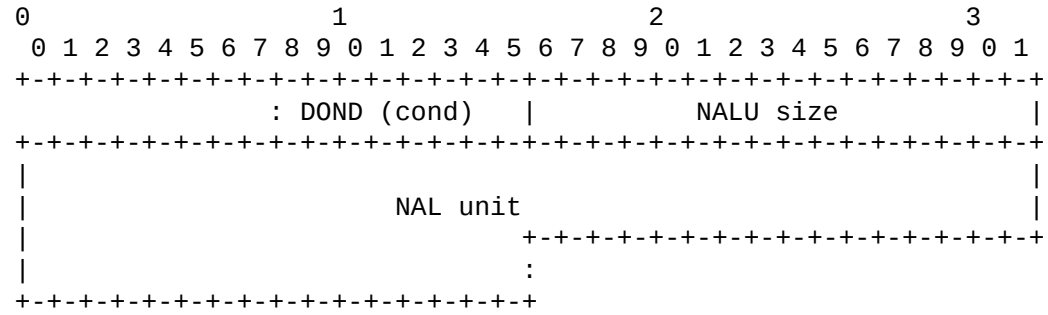

Figure 6 The structure of an aggregation unit that is not the first aggregation unit in an AP

When present, the DOND field plus 1 specifies the difference between the decoding order number values of the current aggregated NAL unit and the preceding aggregated NAL unit in the same AP.

If sprop-max-don-diff is greater than 0 for any of the RTP streams, the DOND field MUST be present in an aggregation unit that is not the first aggregation unit in an AP, and the variable DON for the aggregated NAL unit is derived as equal to the DON of the preceding aggregated NAL unit in the same AP plus the value of the DOND field plus 1 modulo 65536. Otherwise (sprop-max-don-diff is equal to 0 for all the RTP streams), the DOND field MUST NOT be present in an aggregation unit that is not the first aggregation unit in an AP, and in this case the transmission order and decoding order of NAL units carried in the AP are the same as the order the NAL units appear in the AP.

Figure 7 presents an example of an AP that contains two aggregation units, labeled as 1 and 2 in the figure, without the DONL and DOND fields being present.

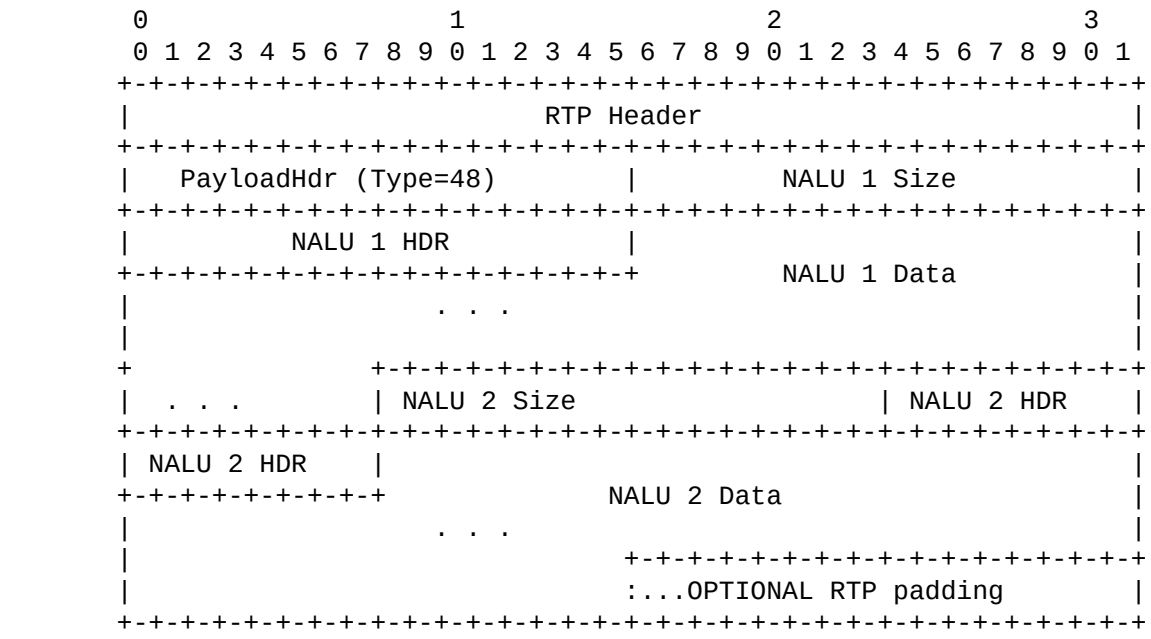

Figure 7 An example of an AP packet containing two aggregation units without the DONL and DOND fields

Figure 8 presents an example of an AP that contains two aggregation units, labeled as 1 and 2 in the figure, with the DONL and DOND fields being present.

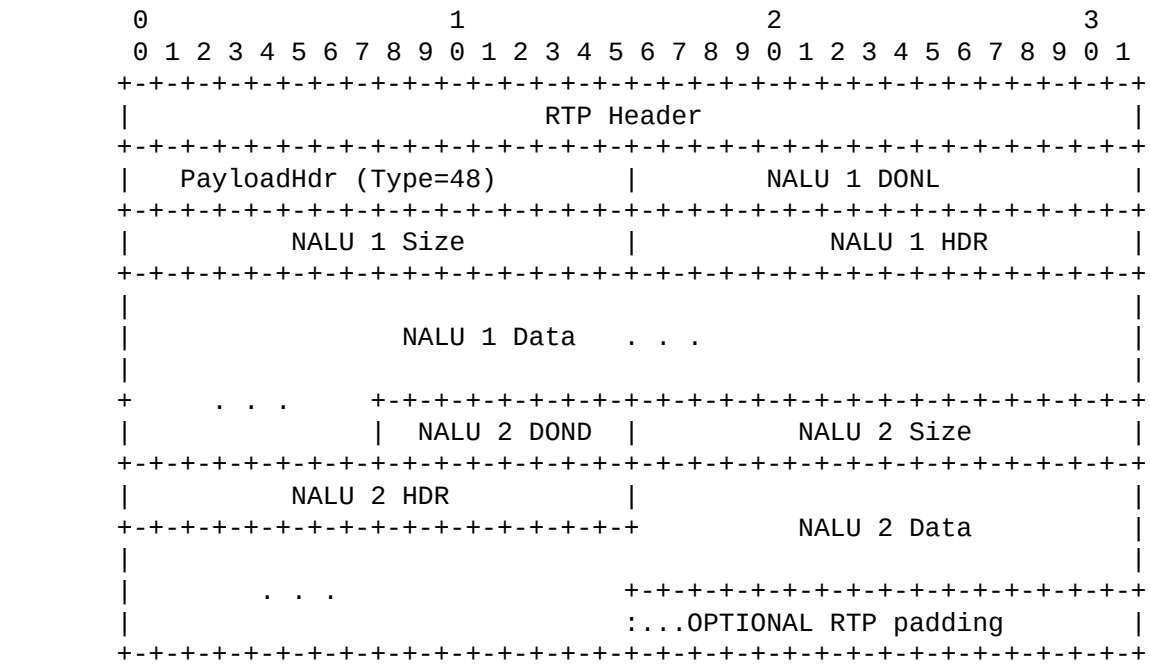

Figure 8 An example of an AP containing two aggregation units with the DONL and DOND fields

4.4.3 Fragmentation Units (FUs)

Fragmentation units (FUs) are introduced to enable fragmenting a single NAL unit into multiple RTP packets, possibly without cooperation or knowledge of the HEVC encoder. A fragment of a NAL unit consists of an integer number of consecutive octets of that NAL unit. Fragments of the same NAL unit MUST be sent in consecutive order with ascending RTP sequence numbers (with no other RTP packets within the same RTP stream being sent between the first and last fragment).

When a NAL unit is fragmented and conveyed within FUs, it is referred to as a fragmented NAL unit. APs MUST NOT be fragmented. FUs MUST NOT be nested; i.e. an FU must not contain a subset of another FU.

The RTP timestamp of an RTP packet carrying an FU is set to the NALU-time of the fragmented NAL unit.

An FU consists of a payload header (denoted as PayloadHdr), an FU header of one octet, a conditional 16-bit DONL field (in network byte order), and an FU payload, as shown in Figure 9.

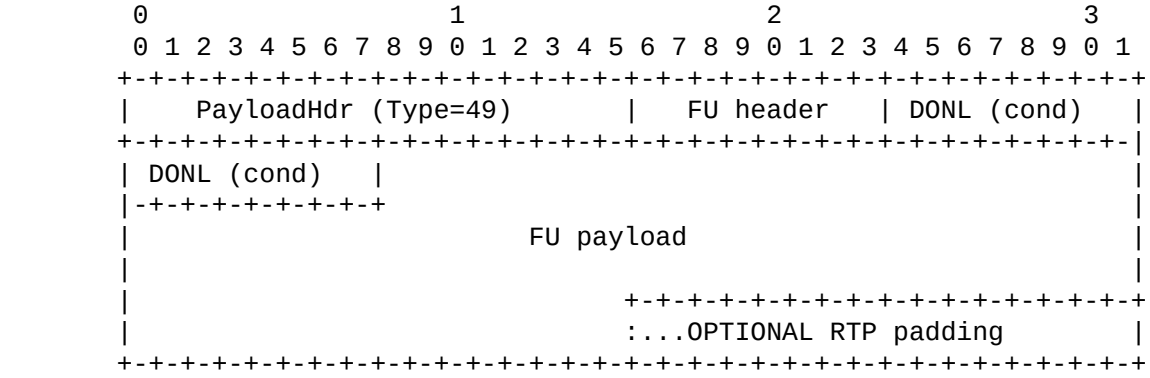

Figure 9 The structure of an FU

The fields in the payload header are set as follows. The Type field MUST be equal to 49. The fields F, LayerId, and TID MUST be equal to the fields F, LayerId, and TID, respectively, of the fragmented NAL unit.

The FU header consists of an S bit, an E bit, and a 6-bit FuType field, as shown in Figure 10.

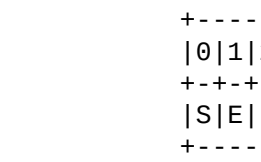

Figure 10 The structure of FU header

The semantics of the FU header fields are as follows:

S: 1 bit
When set to one, the S bit indicates the start of a fragmented NAL unit i.e. the first byte of the FU payload is also the first byte of the payload of the fragmented NAL unit. When the FU payload is not the start of the fragmented NAL unit payload, the S bit MUST be set to zero.

E: 1 bit
When set to one, the E bit indicates the end of a fragmented NAL unit, i.e. the last byte of the payload is also the last byte of the fragmented NAL unit. When the FU payload is not the last fragment of a fragmented NAL unit, the E bit MUST be set to zero.

FuType: 6 bits
The field FuType MUST be equal to the field Type of the fragmented NAL unit.

The DONL field, when present, specifies the value of the 16 least significant bits of the decoding order number of the fragmented NAL unit.

If sprop-max-don-diff is greater than 0 for any of the RTP streams, and the S bit is equal to 1, the DONL field MUST be present in the FU, and the variable DON for the fragmented NAL unit is derived as equal to the value of the DONL field. Otherwise (sprop-max-don-diff is equal to 0 for all the RTP streams, or the S bit is equal to 0), the DONL field MUST NOT be present in the FU.

A non-fragmented NAL unit MUST NOT be transmitted in one FU; i.e. the Start bit and End bit must not both be set to one in the same FU header.

The FU payload consists of fragments of the payload of the fragmented NAL unit so that if the FU payloads of consecutive FUs, starting with an FU with the S bit equal to 1 and ending with an FU with the E bit equal to 1, are sequentially

concatenated, the payload of the fragmented NAL unit can be reconstructed. The NAL unit header of the fragmented NAL unit is not included as such in the FU payload, but rather the information of the NAL unit header of the fragmented NAL unit is conveyed in F, LayerId, and TID fields of the FU payload headers of the FUs and the FuType field of the FU header of the FUs. An FU payload MUST NOT be empty.

If an FU is lost, the receiver SHOULD discard all following fragmentation units in transmission order corresponding to the same fragmented NAL unit, unless the decoder in the receiver is known to be prepared to gracefully handle incomplete NAL units.

A receiver in an endpoint or in a MANE MAY aggregate the first n-1 fragments of a NAL unit to an (incomplete) NAL unit, even if fragment n of that NAL unit is not received. In this case, the forbidden_zero_bit of the NAL unit MUST be set to one to indicate a syntax violation.

4.4.4 PACI packets

This section specifies the PACI packet structure. The basic payload header specified in this memo is intentionally limited to the 16 bits of the NAL unit header so to keep the packetization overhead to a minimum. However, cases have been identified where it is advisable to include control information in an easily accessible position in the packet header, despite the additional overhead. One such control information is the Temporal Scalability Control Information as specified in Section 4.5 below. PACI packets carry this and future, similar structures.

The PACI packet structure is based on a payload header extension mechanism that is generic and extensible to carry payload header extensions. In this section, the focus lies on the use within this specification. Section 4.4.4.2 below provides guidance for the specification designers in how to employ the extension mechanism in future specifications.

A PACI packet consists of a payload header (denoted as PayloadHdr), for which the structure follows what is described in

Section 4.2 above. The payload header is followed by the fields A, cType, PHSsize, F[0..2] and Y.

Figure 11 shows a PACI packet in compliance with this memo; that is, without any extensions.

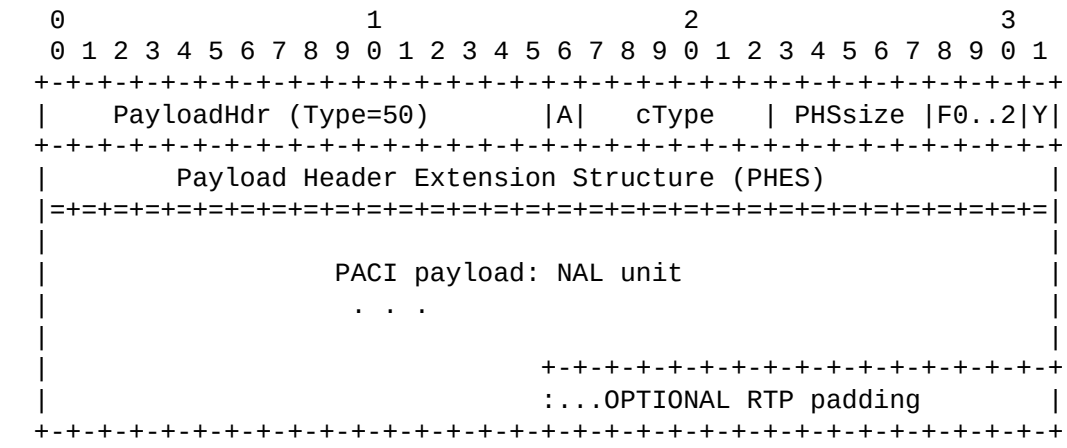

Figure 11 The structure of a PACI

The fields in the payload header are set as follows. The F bit MUST be equal to 0. The Type field MUST be equal to 50. The value of LayerId MUST be a copy of the LayerId field of the PACI payload NAL unit or NAL-unit-like structure. The value of TID MUST be a copy of the TID field of the PACI payload NAL unit or NAL-unit-like structure.

The semantics of other fields are as follows:

A: 1 bit
Copy of the F bit of the PACI payload NAL unit or NAL-unit-like structure.

cType: 6 bits
Copy of the Type field of the PACI payload NAL unit or NAL-unit-like structure.

PHSsize: 5 bits
Indicates the length of the PHES field. The value is limited to be less than or equal to 32 octets, to simplify encoder design for MTU size matching.

F0
This field equal to 1 specifies the presence of a temporal scalability support extension in the PHES.

F1, F2
MUST be 0, available for future extensions, see Section 4.4.4.2. Receivers compliant with this version of the HEVC payload format MUST ignore F1=1 and/or F2=1, and also ignore any information in the PHES indicated as present by F1=1 and/or F2=1.

Informative note: The receiver can do that by first decoding information associated with F0=1, and then skipping over any remaining bytes of the PHES based on the value of PHSsize.

Y: 1 bit
MUST be 0, available for future extensions, see Section 4.4.4.2. Receivers compliant with this version of the HEVC payload format MUST ignore Y=1, and also ignore any information in the PHES indicated as present by Y.

PHES: variable number of octets
A variable number of octets as indicated by the value of PHSsize.

PACI Payload
The single NAL unit packet or NAL-unit-like structure (such as: FU or AP) to be carried, not including the first two octets.

Informative note: The first two octets of the NAL unit or NAL-unit-like structure carried in the PACI payload are not

included in the PACI payload. Rather, the respective values are copied in locations of the PayloadHdr of the RTP packet. This design offers two advantages: first, the overall structure of the payload header is preserved, i.e. there is no special case of payload header structure that needs to be implemented for PACI. Second, no additional overhead is introduced.

A PACI payload MAY be a single NAL unit, an FU, or an AP. PACIs MUST NOT be fragmented or aggregated. The following subsection documents the reasons for these design choices.

4.4.4.1 Reasons for the PACI rules (informative)

A PACI cannot be fragmented. If a PACI could be fragmented, and a fragment other than the first fragment would get lost, access to the information in the PACI would not be possible. Therefore, a PACI must not be fragmented. In other words, an FU must not carry (fragments of) a PACI.

A PACI cannot be aggregated. Aggregation of PACIs is inadvisable from a compression viewpoint, as, in many cases, several to be aggregated NAL units would share identical PACI fields and values which would be carried redundantly for no reason. Most, if not all the practical effects of PACI aggregation can be achieved by aggregating NAL units and bundling them with a PACI (see below). Therefore, a PACI must not be aggregated. In other words, an AP must not contain a PACI.

The payload of a PACI can be a fragment. Both middleboxes and sending systems with inflexible (often hardware-based) encoders occasionally find themselves in situations where a PACI and its headers, combined, are larger than the MTU size. In such a scenario, the middlebox or sender can fragment the NAL unit and encapsulate the fragment in a PACI. Doing so preserves the payload header extension information for all fragments, allowing downstream middleboxes and the receiver to take advantage of that information. Therefore, a sender may place a fragment into a PACI, and a receiver must be able to handle such a PACI.

The payload of a PACI can be an aggregation NAL unit. HEVC bitstreams can contain unevenly sized and/or small (when compared to the MTU size) NAL units. In order to efficiently packetize such small NAL units, AP were introduced. The benefits of APs are independent from the need for a payload header extension. Therefore, a sender may place an AP into a PACI, and a receiver must be able to handle such a PACI.

4.4.4.2 PACI extensions (Informative)

This section includes recommendations for future specification designers on how to extent the PACI syntax to accommodate future extensions. Obviously, designers are free to specify whatever appears to be appropriate to them at the time of their design. However, a lot of thought has been invested into the extension mechanism described below, and we suggest that deviations from it warrant a good explanation.

This memo defines only a single payload header extension (Temporal Scalability Control Information, described below in Section 4.5), and, therefore, only the F0 bit carries semantics. F1 and F2 are already named (and not just marked as reserved, as a typical video spec designer would do). They are intended to signal two additional extensions. The Y bit allows to, recursively, add further F and Y bits to extend the mechanism beyond 3 possible payload header extensions. It is suggested to define a new packet type (using a different value for Type) when assigning the F1, F2, or Y bits different semantics than what is suggested below.

When a Y bit is set, an 8 bit flag-extension is inserted after the Y bit. A flag-extension consists of 7 flags F[n..n+6], and another Y bit.

The basic PACI header already includes F0, F1, and F2. Therefore, the Fx bits in the first flag-extensions are numbered F3, F4, ..., F9, the F bits in the second flag-extension are numbered F10, F11, ..., F16, and so forth. As a result, at least 3 Fx bits are always in the PACI, but the number of Fx bits (and associated types of extensions), can be increased by setting the next Y bit and adding an octet of flag-extensions, carrying 7

flags and another Y bit. The size of this list of flags is subject to the limits specified in Section 4.4.4 (32 octets for all flag-extensions and the PHES information combined).

Each of the F bits can indicate either the presence of information in the Payload Header Extension Structure (PHES), described below, or a given F bit can indicate a certain condition, without including additional information in the PHES.

When a spec developer devises a new syntax that takes advantage of the PACI extension mechanism, he/she must follow the constraints listed below; otherwise the extension mechanism may break.

1) The fields added for a particular Fx bit MUST be fixed in length and not depend on what other Fx bits are set (no parsing dependency).
2) The Fx bits must be assigned in order.
3) An implementation that supports the n-th Fn bit for any value of n must understand the syntax (though not necessarily the semantics) of the fields Fk (with k < n), so to be able to either use those bits when present, or at least be able to skip over them.

4.5 Temporal Scalability Control Information

This section describes the single payload header extension defined in this specification, known as Temporal Scalability Control Information (TSCI). If, in the future, additional payload header extensions become necessary, they could be specified in this section of an updated version of this document, or in their own documents.

When F0 is set to 1 in a PACI, this specifies that the PHES field includes the TSCI fields TL0PICIDX, IrapPicID, S, and E as follows:

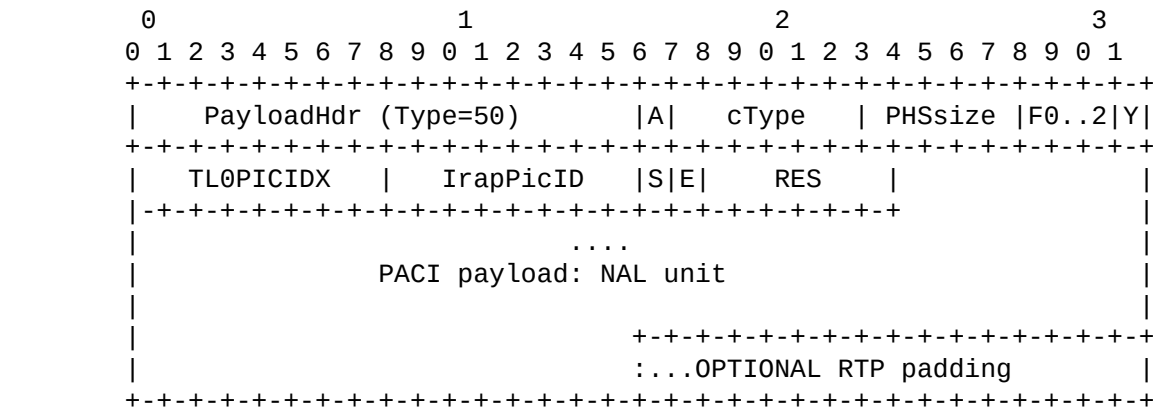

Figure 12 The structure of a PACI with a PHES containing a TSCI

TL0PICIDX (8 bits)
When present, the TL0PICIDX field MUST be set to equal to temporal_sub_layer_zero_idx as specified in Section D.3.22 of [H.265] for the access unit containing the NAL unit in the PACI.

IrapPicID (8 bits)
When present, the IrapPicID field MUST be set to equal to irap_pic_id as specified in Section D.3.22 of [H.265] for the access unit containing the NAL unit in the PACI.

S (1 bit)
The S bit MUST be set to 1 if any of the following conditions is true and MUST be set to 0 otherwise:
- The NAL unit in the payload of the PACI is the first VCL NAL unit, in decoding order, of a picture.
- The NAL unit in the payload of the PACI is an AP and the NAL unit in the first contained aggregation unit is the first VCL NAL unit, in decoding order, of a picture.
- The NAL unit in the payload of the PACI is an FU with its S bit equal to 1 and the FU payload containing a fragment of the first VCL NAL unit, in decoding order of a picture.

E (1 bit)
The E bit MUST be set to 1 if any of the following conditions is true and MUST be set to 0 otherwise:
- The NAL unit in the payload of the PACI is the last VCL NAL unit, in decoding order, of a picture.
- The NAL unit in the payload of the PACI is an AP and the NAL unit in the last contained aggregation unit is the last VCL NAL unit, in decoding order, of a picture.
- The NAL unit in the payload of the PACI is an FU with its E bit equal to 1 and the FU payload containing a fragment of the last VCL NAL unit, in decoding order of a picture.

RES (6 bits)
MUST be equal to 0. Reserved for future extensions.

The value of PHSsize MUST be set to 3. Receivers MUST allow other values of the fields F0, F1, F2, Y, and PHSsize, and MUST ignore any additional fields, when present, than specified above in the PHES.

4.6 Decoding Order Number

For each NAL unit, the variable AbsDon is derived, representing the decoding order number that is indicative of the NAL unit decoding order.

Let NAL unit n be the n-th NAL unit in transmission order within an RTP stream.

If sprop-max-don-diff is equal to 0 for all the RTP streams carrying the HEVC bitstream, AbsDon[n], the value of AbsDon for NAL unit n, is derived as equal to n.

Otherwise (sprop-max-don-diff is greater than 0 for any of the RTP streams), AbsDon[n] is derived as follows, where DON[n] is the value of the variable DON for NAL unit n:

- If n is equal to 0 (i.e. NAL unit n is the very first NAL unit in transmission order), AbsDon[0] is set equal to DON[0].

o Otherwise (n is greater than 0), the following applies for derivation of AbsDon[n]:

```
If DON[n] == DON[n-1],
    AbsDon[n] = AbsDon[n-1]

If (DON[n] > DON[n-1] and DON[n] - DON[n-1] < 32768),
    AbsDon[n] = AbsDon[n-1] + DON[n] - DON[n-1]

If (DON[n] < DON[n-1] and DON[n-1] - DON[n] >= 32768),
    AbsDon[n] = AbsDon[n-1] + 65536 - DON[n-1] + DON[n]

If (DON[n] > DON[n-1] and DON[n] - DON[n-1] >= 32768),
    AbsDon[n] = AbsDon[n-1] - (DON[n-1] + 65536 -
DON[n])

If (DON[n] < DON[n-1] and DON[n-1] - DON[n] < 32768),
    AbsDon[n] = AbsDon[n-1] - (DON[n-1] - DON[n])
```

For any two NAL units m and n, the following applies:

o AbsDon[n] greater than AbsDon[m] indicates that NAL unit n follows NAL unit m in NAL unit decoding order.

o When AbsDon[n] is equal to AbsDon[m], the NAL unit decoding order of the two NAL units can be in either order.

o AbsDon[n] less than AbsDon[m] indicates that NAL unit n precedes NAL unit m in decoding order.

Informative note: When two consecutive NAL units in the NAL unit decoding order have different values of AbsDon, the absolute difference between the two AbsDon values may be greater than or equal to 1.

Informative note: There are multiple reasons to allow for the absolute difference of the values of AbsDon for two consecutive NAL units in the NAL unit decoding order to be greater than one. An increment by one is not required, as at the time of associating values of AbsDon to NAL units, it may not be known whether all NAL units are to be delivered to the receiver. For example, a gateway may not forward VCL NAL

units of higher sub-layers or some SEI NAL units when there is congestion in the network. In another example, the first intra-coded picture of a pre-encoded clip is transmitted in advance to ensure that it is readily available in the receiver, and when transmitting the first intra-coded picture, the originator does not exactly know how many NAL units will be encoded before the first intra-coded picture of the pre-encoded clip follows in decoding order. Thus, the values of AbsDon for the NAL units of the first intra-coded picture of the pre-encoded clip have to be estimated when they are transmitted, and gaps in values of AbsDon may occur. Another example is MRST or MRMT with sprop-max-don-diff greater than 0, where the AbsDon values must indicate cross-layer decoding order for NAL units conveyed in all the RTP streams.

5 Packetization Rules

The following packetization rules apply:

- If sprop-max-don-diff is greater than 0 for any of the RTP streams, the transmission order of NAL units carried in the RTP stream MAY be different than the NAL unit decoding order and the NAL unit output order. Otherwise (sprop-max-don-diff is equal to 0 for all the RTP streams), the transmission order of NAL units carried in the RTP stream MUST be the same as the NAL unit decoding order, and, when tx-mode is equal to "MRST" or "MRMT", MUST also be the same as the NAL unit output order.

- A NAL unit of a small size SHOULD be encapsulated in an aggregation packet together with one or more other NAL units in order to avoid the unnecessary packetization overhead for small NAL units. For example, non-VCL NAL units such as access unit delimiters, parameter sets, or SEI NAL units are typically small and can often be aggregated with VCL NAL units without violating MTU size constraints.

- Each non-VCL NAL unit SHOULD, when possible from an MTU size match viewpoint, be encapsulated in an aggregation packet together with its associated VCL NAL unit, as typically a non-VCL NAL unit would be meaningless without the associated VCL NAL unit being available.

- For carrying exactly one NAL unit in an RTP packet, a single NAL unit packet MUST be used.

6 De-packetization Process

The general concept behind de-packetization is to get the NAL units out of the RTP packets in an RTP stream and all RTP streams the RTP stream depends on, if any, and pass them to the decoder in the NAL unit decoding order.

The de-packetization process is implementation dependent. Therefore, the following description should be seen as an example of a suitable implementation. Other schemes may be used as well as long as the output for the same input is the same as the process described below. The output is the same when the set of output NAL units and their order are both identical. Optimizations relative to the described algorithms are possible.

All normal RTP mechanisms related to buffer management apply. In particular, duplicated or outdated RTP packets (as indicated by the RTP sequences number and the RTP timestamp) are removed. To determine the exact time for decoding, factors such as a possible intentional delay to allow for proper inter-stream synchronization must be factored in.

NAL units with NAL unit type values in the range of 0 to 47, inclusive may be passed to the decoder. NAL-unit-like structures with NAL unit type values in the range of 48 to 63, inclusive, MUST NOT be passed to the decoder.

The receiver includes a receiver buffer, which is used to compensate for transmission delay jitter within individual RTP streams and across RTP streams, to reorder NAL units from transmission order to the NAL unit decoding order, and to recover the NAL unit decoding order in MRST or MRMT, when applicable. In this section, the receiver operation is described under the assumption that there is no transmission delay jitter within an RTP stream and across RTP streams. To make a difference from a practical receiver buffer that is also used for compensation of transmission delay jitter, the receiver buffer is here after called the de-packetization buffer in this section. Receivers

should also prepare for transmission delay jitter; i.e. either reserve separate buffers for transmission delay jitter buffering and de-packetization buffering or use a receiver buffer for both transmission delay jitter and de-packetization. Moreover, receivers should take transmission delay jitter into account in the buffering operation; e.g. by additional initial buffering before starting of decoding and playback.

When sprop-max-don-diff is equal to 0 for all the received RTP streams, the de-packetization buffer size is zero bytes and the process described in the remainder of this paragraph applies. When there is only one RTP stream received, the NAL units carried in the single RTP stream are directly passed to the decoder in their transmission order, which is identical to their decoding order. When there is more than one RTP stream received, the NAL units carried in the multiple RTP streams are passed to the decoder in their NTP timestamp order. When there are several NAL units of different RTP streams with the same NTP timestamp, the order to pass them to the decoder is their dependency order, where NAL units of a dependee RTP stream are passed to the decoder prior to the NAL units of the dependent RTP stream. When there are several NAL units of the same RTP stream with the same NTP timestamp, the order to pass them to the decoder is their transmission order.

> Informative note: The mapping between RTP and NTP timestamps is conveyed in RTCP SR packets. In addition, the mechanisms for faster media timestamp synchronization discussed in [RFC6051] may be used to speed up the acquisition of the RTP-to-wall-clock mapping.

When sprop-max-don-diff is greater than 0 for any the received RTP streams, the process described in the remainder of this section applies.

There are two buffering states in the receiver: initial buffering and buffering while playing. Initial buffering starts when the reception is initialized. After initial buffering, decoding and playback are started, and the buffering-while-playing mode is used.

Regardless of the buffering state, the receiver stores incoming NAL units, in reception order, into the de-packetization buffer. NAL units carried in RTP packets are stored in the de-packetization buffer individually, and the value of AbsDon is calculated and stored for each NAL unit. When MRST or MRMT is in use, NAL units of all RTP streams of a bitstream are stored in the same de-packetization buffer. When NAL units carried in any two RTP streams are available to be placed into the de-packetization buffer, those NAL units carried in the RTP stream that is lower in the dependency tree are placed into the buffer first. For example, if RTP stream A depends on RTP stream B, then NAL units carried in RTP stream B are placed into the buffer first.

Initial buffering lasts until condition A (the difference between the greatest and smallest AbsDon values of the NAL units in the de-packetization buffer is greater than or equal to the value of sprop-max-don-diff of the highest RTP stream) or condition B (the number of NAL units in the de-packetization buffer is greater than the value of sprop-depack-buf-nalus) is true.

After initial buffering, whenever condition A or condition B is true, the following operation is repeatedly applied until both condition A and condition B become false:

- The NAL unit in the de-packetization buffer with the smallest value of AbsDon is removed from the de-packetization buffer and passed to the decoder.

When no more NAL units are flowing into the de-packetization buffer, all NAL units remaining in the de-packetization buffer are removed from the buffer and passed to the decoder in the order of increasing AbsDon values.

7 Payload Format Parameters

This section specifies the parameters that MAY be used to select optional features of the payload format and certain features or properties of the bitstream or the RTP stream. The parameters are specified here as part of the media type registration for the HEVC codec. A mapping of the parameters into the Session

Description Protocol (SDP) [RFC4566] is also provided for applications that use SDP. Equivalent parameters could be defined elsewhere for use with control protocols that do not use SDP.

7.1 Media Type Registration

The media subtype for the HEVC codec is allocated from the IETF tree.

The receiver MUST ignore any unrecognized parameter.

Media Type name: video

Media subtype name: H265

Required parameters: none

OPTIONAL parameters:

profile-space, tier-flag, profile-id, profile-compatibility-indicator, interop-constraints, and level-id:

These parameters indicate the profile, tier, default level, and some constraints of the bitstream carried by the RTP stream and all RTP streams the RTP stream depends on, or a specific set of the profile, tier, default level, and some constraints the receiver supports.

The profile and some constraints are indicated collectively by profile-space, profile-id, profile-compatibility-indicator, and interop-constraints. The profile specifies the subset of coding tools that may have been used to generate the bitstream or that the receiver supports.

Informative note: There are 32 values of profile-id, and there are 32 flags in profile-compatibility-indicator, each flag corresponding to one value of profile-id. According to HEVC version 1 in [HEVC], when more than one of the 32 flags is set for a bitstream, the bitstream would comply with all the profiles corresponding to the set flags. However, in a draft of

HEVC version 2 in [HEVC draft v2], subclause A.3.5, 19 Format Range Extensions profiles have been specified, all using the same value of profile-id (4), differentiated by some of the 48 bits in interop-constraints - this (rather unexpected way of profile signalling) means that one of the 32 flags may correspond to multiple profiles. To be able to support whatever HEVC extension profile that might be specified and indicated using profile-space, profile-id, profile-compatibility-indicator, and interop-constraints in the future, it would be safe to require symmetric use of these parameters in SDP offer/answer unless recv-sub-layer-id is included in the SDP answer for choosing one of the sub-layers offered.

The tier is indicated by tier-flag. The default level is indicated by level-id. The tier and the default level specify the limits on values of syntax elements or arithmetic combinations of values of syntax elements that are followed when generating the bitstream or that the receiver supports.

A set of profile-space, tier-flag, profile-id, profile-compatibility-indicator, interop-constraints, and level-id parameters ptlA is said to be consistent with another set of these parameters ptlB if any decoder that conforms to the profile, tier, level, and constraints indicated by ptlB can decode any bitstream that conforms to the profile, tier, level, and constraints indicated by ptlA.

In SDP offer/answer, when the SDP answer does not include the recv-sub-layer-id parameter that is less than the sprop-sub-layer-id parameter in the SDP offer, the following applies:

- The profile-space, tier-flag, profile-id, profile-compatibility-indicator, and interop-constraints parameters MUST be used symmetrically, i.e. the value of each of these parameters in the offer MUST be the same as that in the answer, either explicitly signalled or implicitly inferred.

- The level-id parameter is changeable as long as the highest level indicated by the answer is either equal to or lower than that in the offer. Note that the highest level is indicated by level-id and max-recv-level-id together.

In SDP offer/answer, when the SDP answer does include the recv-sub-layer-id parameter that is less than the sprop-sub-layer-id parameter in the SDP offer, the set of profile-space, tier-flag, profile-id, profile-compatibility-indicator, interop-constraints, and level-id parameters included in the answer MUST be consistent with that for the chosen sub-layer representation as indicated in the SDP offer, with the exception that the level-id parameter in the SDP answer is changable as long as the highest level indicated by the answer is either lower than or equal to that in the offer.

More specifications of these parameters, including how they relate to the values of the profile, tier, and level syntax elements specified in [HEVC] are provided below.

profile-space, profile-id:

The value of profile-space MUST be in the range of 0 to 3, inclusive. The value of profile-id MUST be in the range of 0 to 31, inclusive.

When profile-space is not present, a value of 0 MUST be inferred. When profile-id is not present, a value of 1 (i.e. the Main profile) MUST be inferred.

When used to indicate properties of a bitstream, profile-space and profile-id are derived from the profile, tier, and level syntax elements in SPS or VPS NAL units as follows, where general_profile_space, general_profile_idc, sub_layer_profile_space[j], and sub_layer_profile_idc[j] are specified in [HEVC]:

If the RTP stream is the highest RTP stream, the following applies:

- o profile_space = general_profile_space
- o profile_id = general_profile_idc

Otherwise (the RTP stream is a dependee RTP stream), the following applies, with j being the value of the sprop-sub-layer-id parameter:

- o profile_space = sub_layer_profile_space[j]
- o profile_id = sub_layer_profile_idc[j]

tier-flag, level-id:

The value of tier-flag MUST be in the range of 0 to 1, inclusive. The value of level-id MUST be in the range of 0 to 255, inclusive.

If the tier-flag and level-id parameters are used to indicate properties of a bitstream, they indicate the tier and the highest level the bitstream complies with.

If the tier-flag and level-id parameters are used for capability exchange, the following applies. If max-recv-level-id is not present, the default level defined by level-id indicates the highest level the codec wishes to support. Otherwise, max-recv-level-id 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.

If no tier-flag is present, a value of 0 MUST be inferred and if no level-id is present, a value of 93 (i.e. level 3.1) MUST be inferred.

When used to indicate properties of a bitstream, the tier-flag and level-id parameters are derived from the profile, tier, and level syntax elements in SPS or VPS NAL units as follows, where general_tier_flag, general_level_idc, sub_layer_tier_flag[j], and sub_layer_level_idc[j] are specified in [HEVC]:

If the RTP stream is the highest RTP stream, the following applies:

- o tier-flag = general_tier_flag
- o level-id = general_level_idc

Otherwise (the RTP stream is a dependee RTP stream), the following applies, with j being the value of the sprop-sub-layer-id parameter:

- o tier-flag = sub_layer_tier_flag[j]
- o level-id = sub_layer_level_idc[j]

interop-constraints:

A base16 [RFC4648] (hexadecimal) representation of six bytes of data, consisting of progressive_source_flag, interlaced_source_flag, non_packed_constraint_flag, frame_only_constraint_flag, and reserved_zero_44bits.

If the interop-constraints parameter is not present, the following MUST be inferred:

- o progressive_source_flag = 1
- o interlaced_source_flag = 0
- o non_packed_constraint_flag = 1
- o frame_only_constraint_flag = 1
- o reserved_zero_44bits = 0

When the interop-constraints parameter is used to indicate properties of a bitstream, the following applies, where general_progressive_source_flag, general_interlaced_source_flag, general_non_packed_constraint_flag, general_non_packed_constraint_flag, general_frame_only_constraint_flag, general_reserved_zero_44bits, sub_layer_progressive_source_flag[j], sub_layer_interlaced_source_flag[j], sub_layer_non_packed_constraint_flag[j],

sub_layer_frame_only_constraint_flag[j], and
sub_layer_reserved_zero_44bits[j] are specified in [HEVC]:

If the RTP stream is the highest RTP stream, the following applies:

o progressive_source_flag = general_progressive_source_flag
o interlaced_source_flag = general_interlaced_source_flag
o non_packed_constraint_flag = general_non_packed_constraint_flag
o frame_only_constraint_flag = general_frame_only_constraint_flag
o reserved_zero_44bits = general_reserved_zero_44bits

Otherwise (the RTP stream is a dependee RTP stream), the following applies, with j being the value of the sprop-sub-layer-id parameter:

o progressive_source_flag = sub_layer_progressive_source_flag[j]
o interlaced_source_flag = sub_layer_interlaced_source_flag[j]
o non_packed_constraint_flag = sub_layer_non_packed_constraint_flag[j]
o frame_only_constraint_flag = sub_layer_frame_only_constraint_flag[j]
o reserved_zero_44bits = sub_layer_reserved_zero_44bits[j]

Using interop-constraints for capability exchange results in a requirement on any bitstream to be compliant with the interop-constraints.

profile-compatibility-indicator:

A base16 [RFC4648] representation of four bytes of data.

When profile-compatibility-indicator is used to indicate properties of a bitstream, the following applies, where general_profile_compatibility_flag[j] and sub_layer_profile_compatibility_flag[i][j] are specified in [HEVC]:

> The profile-compatibility-indicator in this case indicates additional profiles to the profile defined by profile_space, profile_id, and interop-constraints the bitstream conforms to. A decoder that conforms to any of all the profiles the bitstream conforms to would be capable of decoding the bitstream. These additional profiles are defined by profile-space, each set bit of profile-compatibility-indicator, and interop-constraints.
>
> If the RTP stream is the highest RTP stream, the following applies for each value of j in the range of 0 to 31, inclusive:
>
> o bit j of profile-compatibility-indicator = general_profile_compatibility_flag[j]
>
> Otherwise (the RTP stream is a dependee RTP stream), the following applies for i equal to sprop-sub-layer-id and for each value of j in the range of 0 to 31, inclusive:
>
> o bit j of profile-compatibility-indicator = sub_layer_profile_compatibility_flag[i][j]

Using profile-compatibility-indicator for capability exchange results in a requirement on any bitstream to be compliant with the profile-compatibility-indicator. This is intended to handle cases where any future HEVC profile is defined as an intersection of two or more profiles.

If this parameter is not present, this parameter defaults to the following: bit j, with j equal to profile-id, of profile-compatibility-indicator is inferred to be equal to 1, and all other bits are inferred to be equal to 0.

sprop-sub-layer-id:

This parameter MAY be used to indicate the highest allowed value of TID in the bitstream. When not present, the value of sprop-sub-layer-id is inferred to be equal to 6.

The value of sprop-sub-layer-id MUST be in the range of 0 to 6, inclusive.

recv-sub-layer-id:

This parameter MAY be used to signal a receiver’s choice of the offered or declared sub-layer representations in the sprop-vps. The value of recv-sub-layer-id indicates the TID of the highest sub-layer of the bitstream that a receiver supports. When not present, the value of recv-sub-layer-id is inferred to be equal to the value of the sprop-sub-layer-id parameter in the SDP offer.

The value of recv-sub-layer-id MUST be in the range of 0 to 6, inclusive.

max-recv-level-id:

This parameter MAY be used to indicate the highest level a receiver supports. The highest level the receiver supports is equal to the value of max-recv-level-id divided by 30.

The value of max-recv-level-id MUST be in the range of 0 to 255, inclusive.

When max-recv-level-id is not present, the value is inferred to be equal to level-id.

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

tx-mode:

This parameter indicates whether the transmission mode is SRST, MRST, or MRMT.

The value of tx-mode MUST be equal to "SRST", "MRST" or "MRMT". When not present, the value of tx-mode is inferred to be equal to "SRST".

If the value is equal to "MRST", MRST MUST be in use. Otherwise, if the value is equal to "MRMT", MRMT MUST be in use. Otherwise (the value is equal to "SRST"), SRST MUST be in use.

The value of tx-mode MUST be equal to "MRST" for all RTP streams in an MRST.

The value of tx-mode MUST be equal to "MRMT" for all RTP streams in an MRMT.

sprop-vps:

This parameter MAY be used to convey any video parameter set NAL unit of the bitstream for out-of-band transmission of video parameter sets. The parameter MAY also be used for capability exchange and to indicate sub-stream characteristics (i.e. properties of sub-layer representations as defined in [HEVC]). The value of the parameter is a comma-separated (',') list of base64 [RFC4648] representations of the video parameter set NAL units as specified in Section 7.3.2.1 of [HEVC].

The sprop-vps parameter MAY contain one or more than one video parameter set NAL unit. However, all other video parameter sets contained in the sprop-vps parameter MUST be consistent with the first video parameter set in the sprop-vps parameter. A video parameter set vpsB is said to be consistent with another video parameter set vpsA if any decoder that conforms to the profile, tier, level, and constraints indicated by the 12 bytes of data starting from the syntax element general_profile_space to the syntax element general_level_id, inclusive, in the first profile_tier_level( ) syntax structure in vpsA can decode any bitstream that conforms to the profile, tier, level, and constraints indicated by the 12 bytes of data starting from the syntax element general_profile_space to the syntax

element general_level_id, inclusive, in the first profile_tier_level( ) syntax structure in vpsB.

sprop-sps:

This parameter MAY be used to convey sequence parameter set NAL units of the bitstream for out-of-band transmission of sequence parameter sets. The value of the parameter is a comma-separated (',') list of base64 [RFC4648] representations of the sequence parameter set NAL units as specified in Section 7.3.2.2 of [HEVC].

sprop-pps:

This parameter MAY be used to convey picture parameter set NAL units of the bitstream for out-of-band transmission of picture parameter sets. The value of the parameter is a comma-separated (',') list of base64 [RFC4648] representations of the picture parameter set NAL units as specified in Section 7.3.2.3 of [HEVC].

sprop-sei:

This parameter MAY be used to convey one or more SEI messages that describe bitstream characteristics. When present, a decoder can rely on the bitstream characteristics that are described in the SEI messages for the entire duration of the session, independently from the persistence scopes of the SEI messages as specified in [HEVC].

The value of the parameter is a comma-separated (',') list of base64 [RFC4648] representations of SEI NAL units as specified in Section 7.3.2.4 of [HEVC].

Informative note: Intentionally, no list of applicable or inapplicable SEI messages is specified here. Conveying certain SEI messages in sprop-sei may be sensible in some application scenarios and meaningless in others. However, a few examples are described below:

1) In an environment where the bitstream was created from film-based source material, and no splicing is going to occur during the lifetime of the session, the film grain characteristics SEI message or the tone mapping information SEI message are likely meaningful, and sending them in sprop-sei rather than in the bitstream at each entry point may help saving bits and allows to configure the renderer only once, avoiding unwanted artifacts.
2) The structure of pictures information SEI message in sprop-sei can be used to inform a decoder of information on the NAL unit types, picture order count values, and prediction dependencies of a sequence of pictures. Having such knowledge can be helpful for error recovery.
3) Examples for SEI messages that would be meaningless to be conveyed in sprop-sei include the decoded picture hash SEI message (it is close to impossible that all decoded pictures have the same hash-tag), the display orientation SEI message when the device is a handheld device (as the display orientation may change when the handheld device is turned around), or the filler payload SEI message (as there is no point in just having more bits in SDP).

max-lsr, max-lps, max-cpb, max-dpb, max-br, max-tr, max-tc:

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 (specified by max-recv-level-id) MUST be the highest that the receiver is fully capable of supporting. max-lsr, max-lps, max-cpb, max-dpb, max-br, max-tr, and max-tc MAY be used to indicate capabilities of the receiver that extend the required capabilities of the highest level, as specified below.

When more than one parameter from the set (max-lsr, max-lps, max-cpb, max-dpb, max-br, max-tr, max-tc) is present, the receiver MUST support all signaled capabilities simultaneously. For example, if both max-lsr and max-br are present, the highest level with the extension of both

the picture rate and bitrate is supported. That is, the receiver is able to decode bitstreams in which the luma sample rate is up to max-lsr (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 by max-recv-level-id.

Informative note: When the OPTIONAL media type parameters are used to signal the properties of a bitstream, and max-lsr, max-lps, max-cpb, max-dpb, max-br, max-tr, and max-tc are not present, the values of profile-space, tier-flag, profile-id, profile-compatibility-indicator, interop-constraints, and level-id must always be such that the bitstream complies fully with the specified profile, tier, and level.

max-lsr:
The value of max-lsr is an integer indicating the maximum processing rate in units of luma samples per second. The max-lsr parameter signals that the receiver is capable of decoding video at a higher rate than is required by the highest level.

When max-lsr is signaled, the receiver MUST be able to decode bitstreams that conform to the highest level, with the exception that the MaxLumaSR value in Table A-2 of [HEVC] for the highest level is replaced with the value of max-lsr. Senders MAY use this knowledge to send pictures of a given size at a higher picture rate than is indicated in the highest level.

When not present, the value of max-lsr is inferred to be equal to the value of MaxLumaSR given in Table A-2 of [HEVC] for the highest level.

The value of max-lsr MUST be in the range of MaxLumaSR to 16 * MaxLumaSR, inclusive, where MaxLumaSR is given in Table A-2 of [HEVC] for the highest level.

max-lps:
The value of max-lps is an integer indicating the maximum picture size in units of luma samples. The max-lps parameter signals that the receiver is capable of decoding larger picture sizes than are required by the highest level. When max-lps is signaled, the receiver MUST be able to decode bitstreams that conform to the highest level, with the exception that the MaxLumaPS value in Table A-1 of [HEVC] for the highest level is replaced with the value of max-lps. Senders MAY use this knowledge to send larger pictures at a proportionally lower picture rate than is indicated in the highest level.

When not present, the value of max-lps is inferred to be equal to the value of MaxLumaPS given in Table A-1 of [HEVC] for the highest level.

The value of max-lps MUST be in the range of MaxLumaPS to 16 * MaxLumaPS, inclusive, where MaxLumaPS is given in Table A-1 of [HEVC] for the highest level.

max-cpb:
The value of max-cpb is an integer indicating the maximum coded picture buffer size in units of CpbBrVclFactor bits for the VCL HRD parameters and in units of CpbBrNalFactor bits for the NAL HRD parameters, where CpbBrVclFactor and CpbBrNalFactor are defined in Section A.4 of [HEVC]. The max-cpb parameter signals that the receiver has more memory than the minimum amount of coded picture buffer memory required by the highest level. When max-cpb is signaled, the receiver MUST be able to decode bitstreams that conform to the highest level, with the exception that the MaxCPB value in Table A-1 of [HEVC] for the highest level is replaced with the value of max-cpb. Senders MAY use this knowledge to construct coded bitstreams with greater variation of bitrate than can be achieved with the MaxCPB value in Table A-1 of [HEVC].

When not present, the value of max-cpb is inferred to be equal to the value of MaxCPB given in Table A-1 of [HEVC] for the highest level.

The value of max-cpb MUST be in the range of MaxCPB to 16 * MaxCPB, inclusive, where MaxLumaCPB is given in Table A-1 of [HEVC] for the highest level.

Informative note: The coded picture buffer is used in the hypothetical reference decoder (Annex C of HEVC). The use of the hypothetical reference decoder is recommended in HEVC 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-packetization and de-jitter buffers. The coded picture buffer need not be implemented in decoders as specified in Annex C of HEVC, 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-packetization 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 decoded pictures at the MaxLumaPS for the highest level, i.e. the number of decoded pictures at the maximum picture size defined by the highest level. The value of max-dpb MUST be in the range of 1 to 16, respectively. The max-dpb parameter signals that the receiver has more memory than the minimum amount of decoded picture buffer memory required by default, which is MaxDpbPicBuf as defined in [HEVC] (equal to 6). When max-dpb is signaled, the receiver MUST be able to decode bitstreams that conform to the highest level, with the exception that the MaxDpbPicBuff value defined in [HEVC] as 6 is replaced with the value of max-dpb. Consequently, a receiver that signals max-dpb MUST be capable of storing the following number of decoded pictures (MaxDpbSize) in its decoded picture buffer:

```
if( PicSizeInSamplesY <= ( MaxLumaPS >> 2 ) )
   MaxDpbSize = Min( 4 * max-dpb, 16 )
else if ( PicSizeInSamplesY <= ( MaxLumaPS >> 1 ) )
```

```
            MaxDpbSize = Min( 2 * max-dpb, 16 )
          else if ( PicSizeInSamplesY <= ( ( 3 * MaxLumaPS ) >> 2
        ) )
            MaxDpbSize = Min( (4 * max-dpb) / 3, 16 )
          else
            MaxDpbSize = max-dpb
```

Wherein MaxLumaPS given in Table A-1 of [HEVC] for the highest level and PicSizeInSamplesY is the current size of each decoded picture in units of luma samples as defined in [HEVC].

The value of max-dpb MUST be greater than or equal to the value of MaxDpbPicBuf (i.e. 6) as defined in [HEVC]. Senders MAY use this knowledge to construct coded bitstreams with improved compression.

When not present, the value of max-dpb is inferred to be equal to the value of MaxDpbPicBuf (i.e. 6) as defined in [HEVC].

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-packetization and de-jitter buffers.

max-br:

The value of max-br is an integer indicating the maximum video bitrate in units of CpbBrVclFactor bits per second for the VCL HRD parameters and in units of CpbBrNalFactor bits per second for the NAL HRD parameters, where CpbBrVclFactor and CpbBrNalFactor are defined in Section A.4 of [HEVC].

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 highest level.

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

- The value of max-br replaces the MaxBR value in Table A-2 of [HEVC] for the highest level.
- When the max-cpb parameter is not present, the result of the following formula replaces the value of MaxCPB in Table A-1 of [HEVC]:

  (MaxCPB of the highest level) * max-br / (MaxBR of the highest level)

For example, if a receiver signals capability for Main profile Level 2 with max-br equal to 2000, this indicates a maximum video bitrate of 2000 kbits/sec for VCL HRD parameters, a maximum video bitrate of 2200 kbits/sec for NAL HRD parameters, and a CPB size of 2000000 bits (2000000 / 1500000 * 1500000).

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

When not present, the value of max-br is inferred to be equal to the value of MaxBR given in Table A-2 of [HEVC] for the highest level.

The value of max-br MUST be in the range of MaxBR to 16 * MaxBR, inclusive, where MaxBR is given in Table A-2 of [HEVC] for the highest level.

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.

max-tr:
The value of max-tr is an integer indication the maximum number of tile rows. The max-tr parameter signals that the receiver is capable of decoding video with a larger number of tile rows than the value allowed by the highest level.

When max-tr is signaled, the receiver MUST be able to decode bitstreams that conform to the highest level, with the exception that the MaxTileRows value in Table A-1 of [HEVC] for the highest level is replaced with the value of max-tr.

Senders MAY use this knowledge to send pictures utilizing a larger number of tile rows than the value allowed by the highest level.

When not present, the value of max-tr is inferred to be equal to the value of MaxTileRows given in Table A-1 of [HEVC] for the highest level.

The value of max-tr MUST be in the range of MaxTileRows to 16 * MaxTileRows, inclusive, where MaxTileRows is given in Table A-1 of [HEVC] for the highest level.

max-tc:
The value of max-tc is an integer indication the maximum number of tile columns. The max-tc parameter signals that the receiver is capable of decoding video with a larger number of tile columns than the value allowed by the highest level.

When max-tc is signaled, the receiver MUST be able to decode bitstreams that conform to the highest level, with the exception that the MaxTileCols value in Table A-1 of [HEVC] for the highest level is replaced with the value of max-tc.

Senders MAY use this knowledge to send pictures utilizing a larger number of tile columns than the value allowed by the highest level.

When not present, the value of max-tc is inferred to be equal to the value of MaxTileCols given in Table A-1 of [HEVC] for the highest level.

The value of max-tc MUST be in the range of MaxTileCols to 16 * MaxTileCols, inclusive, where MaxTileCols is given in Table A-1 of [HEVC] for the highest level.

max-fps:

The value of max-fps is an integer indicating the maximum picture rate in units of pictures per 100 seconds that can be effectively processed by the receiver. The max-fps parameter MAY be used to signal that the receiver has a constraint in that it is not capable of processing video effectively at the full picture rate that is implied by the highest level and, when present, one or more of the parameters max-lsr, max-lps, and max-br.

The value of max-fps is not necessarily the picture rate at which the maximum picture size can be sent, it constitutes a constraint on maximum picture rate for all resolutions.

Informative note: The max-fps parameter is semantically different from max-lsr, max-lps, max-cpb, max-dpb, max-br, max-tr, and max-tc in that max-fps is used to signal a constraint, lowering the maximum picture rate from what is implied by other parameters.

The encoder MUST use a picture rate equal to or less than this value. In cases where the max-fps parameter is absent the encoder is free to choose any picture rate according to the highest level and any signaled optional parameters.

The value of max-fps MUST be smaller than or equal to the full picture rate that is implied by the highest level and,

when present, one or more of the parameters max-lsr, max-lps, and max-br.

sprop-max-don-diff:

If tx-mode is equal to "SRST" and there is no NAL unit naluA that is followed in transmission order by any NAL unit preceding naluA in decoding order (i.e. the transmission order of the NAL units is the same as the decoding order), the value of this parameter MUST be equal to 0.

Otherwise, if tx-mode is equal to "MRST" or "MRMT", the decoding order of the NAL units of all the RTP streams is the same as the NAL unit transmission order and the NAL unit output order, the value of this parameter MUST be equal to either 0 or 1.

Otherwise, if tx-mode is equal to "MRST" or "MRMT" and the decoding order of the NAL units of all the RTP streams is the same as the NAL unit transmission order but not the same as the NAL unit output order, the value of this parameter MUST be equal to 1.

Otherwise, this parameter specifies the maximum absolute difference between the decoding order number (i.e., AbsDon) values of any two NAL units naluA and naluB, where naluA follows naluB in decoding order and precedes naluB in transmission order.

The value of sprop-max-don-diff MUST be an integer in the range of 0 to 32767, inclusive.

When not present, the value of sprop-max-don-diff is inferred to be equal to 0.

sprop-depack-buf-nalus:

This parameter specifies the maximum number of NAL units that precede a NAL unit in transmission order and follow the NAL unit in decoding order.

The value of sprop-depack-buf-nalus MUST be an integer in the range of 0 to 32767, inclusive.

When not present, the value of sprop-depack-buf-nalus is inferred to be equal to 0.

When sprop-max-don-diff is present and greater than 0, this parameter MUST be present and the value MUST be greater than 0.

sprop-depack-buf-bytes:

This parameter signals the required size of the de-packetization buffer in units of bytes. The value of the parameter MUST be greater than or equal to the maximum buffer occupancy (in units of bytes) of the de-packetization buffer as specified in Section 6.

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

When sprop-max-don-diff is present and greater than 0, this parameter MUST be present and the value MUST be greater than 0. When not present, the value of sprop-depack-buf-bytes is inferred to be equal to 0.

Informative note: The value of sprop-depack-buf-bytes indicates the required size of the de-packetization buffer only. When network jitter can occur, an appropriately sized jitter buffer has to be available as well.

depack-buf-cap:

This parameter signals the capabilities of a receiver implementation and indicates the amount of de-packetization buffer space in units of bytes that the receiver has available for reconstructing the NAL unit decoding order from NAL units carried in one or more RTP streams. A receiver is able to handle any RTP stream, and all RTP streams the RTP stream depends on, when present, for which

the value of the sprop-depack-buf-bytes parameter is smaller than or equal to this parameter.

When not present, the value of depack-buf-cap is inferred to be equal to 4294967295. The value of depack-buf-cap MUST be an integer in the range of 1 to 4294967295, inclusive.

Informative note: depack-buf-cap indicates the maximum possible size of the de-packetization buffer of the receiver only, without allowing for network jitter.

sprop-segmentation-id:

This parameter MAY be used to signal the segmentation tools present in the bitstream and that can be used for parallelization. The value of sprop-segmentation-id MUST be an integer in the range of 0 to 3, inclusive. When not present, the value of sprop-segmentation-id is inferred to be equal to 0.

When sprop-segmentation-id is equal to 0, no information about the segmentation tools is provided. When sprop-segmentation-id is equal to 1, it indicates that slices are present in the bitstream. When sprop-segmentation-id is equal to 2, it indicates that tiles are present in the bitstream. When sprop-segmentation-id is equal to 3, it indicates that WPP is used in the bitstream.

sprop-spatial-segmentation-idc:

A base16 [RFC4648] representation of the syntax element min_spatial_segmentation_idc as specified in [HEVC]. This parameter MAY be used to describe parallelization capabilities of the bitstream.

dec-parallel-cap:

This parameter MAY be used to indicate the decoder’s additional decoding capabilities given the presence of tools enabling parallel decoding, such as slices, tiles,

and WPP, in the bitstream. The decoding capability of the decoder may vary with the setting of the parallel decoding tools present in the bitstream, e.g. the size of the tiles that are present in a bitstream. Therefore, multiple capability points may be provided, each indicating the minimum required decoding capability that is associated with a parallelism requirement, which is a requirement on the bitstream that enables parallel decoding.

Each capability point is defined as a combination of 1) a parallelism requirement, 2) a profile (determined by profile-space and profile-id), 3) a highest level, and 4) a maximum processing rate, a maximum picture size, and a maximum video bitrate that may be equal to or greater than that determined by the highest level. The parameter's syntax in ABNF [RFC5234] is as follows:

```
dec-parallel-cap = "dec-parallel-cap={" cap-point *(","
                   cap-point) "}"

cap-point = ("w" / "t") ":" spatial-seg-idc 1*(";"
             cap-parameter)

spatial-seg-idc = 1*4DIGIT ; (1-4095)

cap-parameter = tier-flag / level-id / max-lsr
                / max-lps / max-br

tier-flag = "tier-flag" EQ ("0" / "1")

level-id  = "level-id" EQ 1*3DIGIT ; (0-255)

max-lsr   = "max-lsr" EQ  1*20DIGIT ; (0-
18,446,744,073,709,551,615)

max-lps   = "max-lps" EQ 1*10DIGIT ; (0-4,294,967,295)

max-br    = "max-br"  EQ 1*20DIGIT ; (0-
18,446,744,073,709,551,615)

EQ = "="
```

The set of capability points expressed by the dec-parallel-cap parameter is enclosed in a pair of curly braces ("{}"). Each set of two consecutive capability points is separated by a comma (’,’). Within each capability point, each set of two consecutive parameters, and when present, their values, is separated by a semicolon (’;’).

The profile of all capability points is determined by profile-space and profile-id that are outside the dec-parallel-cap parameter.

Each capability point starts with an indication of the parallelism requirement, which consists of a parallel tool type, which may be equal to ’w’ or ’t’, and a decimal value of the spatial-seg-idc parameter. When the type is ’w’, the capability point is valid only for H.265 bitstreams with WPP in use, i.e. entropy_coding_sync_enabled_flag equal to 1. When the type is ’t’, the capability point is valid only for H.265 bitstreams with WPP not in use (i.e. entropy_coding_sync_enabled_flag equal to 0). The capability-point is valid only for H.265 bitstreams with min_spatial_segmentation_idc equal to or greater than spatial-seg-idc.

After the parallelism requirement indication, each capability point continues with one or more pairs of parameter and value in any order for any of the following parameters:

- tier-flag
- level-id
- max-lsr
- max-lps
- max-br

At most one occurrence of each of the above five parameters is allowed within each capability point.

The values of dec-parallel-cap.tier-flag and dec-parallel-cap.level-id for a capability point indicate the highest level of the capability point. The values of dec-parallel-

cap.max-lsr, dec-parallel-cap.max-lps, and dec-parallel-cap.max-br for a capability point indicate the maximum processing rate in units of luma samples per second, the maximum picture size in units of luma samples, and the maximum video bitrate (in units of CpbBrVclFactor bits per second for the VCL HRD parameters and in units of CpbBrNalFactor bits per second for the NAL HRD parameters where CpbBrVclFactor and CpbBrNalFactor are defined in Section A.4 of [HEVC]).

When not present, the value of dec-parallel-cap.tier-flag is inferred to be equal to the value of tier-flag outside the dec-parallel-cap parameter. When not present, the value of dec-parallel-cap.level-id is inferred to be equal to the value of max-recv-level-id outside the dec-parallel-cap parameter. When not present, the value of dec-parallel-cap.max-lsr, dec-parallel-cap.max-lps, or dec-parallel-cap.max-br is inferred to be equal to the value of max-lsr, max-lps, or max-br, respectively, outside the dec-parallel-cap parameter.

The general decoding capability, expressed by the set of parameters outside of dec-parallel-cap, is defined as the capability point that is determined by the following combination of parameters: 1) the parallelism requirement corresponding to the value of sprop-segmentation-id equal to 0 for a bitstream, 2) the profile determined by profile-space, profile-id, profile-compatibility-indicator, and interop-constraints, 3) the tier and the highest level determined by tier-flag and max-recv-level-id, and 4) the maximum processing rate, the maximum picture size, and the maximum video bitrate determined by the highest level. The general decoding capability MUST NOT be included as one of the set of capability points in the dec-parallel-cap parameter.

For example, the following parameters express the general decoding capability of 720p30 (Level 3.1) plus an additional decoding capability of 1080p30 (Level 4) given that the spatially largest tile or slice used in the bitstream is equal to or less than 1/3 of the picture size:

```
a=fmtp:98 level-id=93;dec-parallel-cap={t:8;level-
id=120}
```

For another example, the following parameters express an additional decoding capability of 1080p30, using dec-parallel-cap.max-lsr and dec-parallel-cap.max-lps, given that WPP is used in the bitstream:

```
a=fmtp:98 level-id=93;dec-parallel-cap={w:8;
            max-lsr=62668800;max-lps=2088960}
```

Informative note: When min_spatial_segmentation_idc is present in a bitstream and WPP is not used, [HEVC] specifies that there is no slice or no tile in the bitstream containing more than 4 * PicSizeInSamplesY / (min_spatial_segmentation_idc + 4) luma samples.

include-dph:

This parameter is used to indicate the capability and preference to utilize or include decoded picture hash (DPH) SEI messages (See Section D.3.19 of [HEVC]) in the bitstream. DPH SEI messages can be used to detect picture corruption so the receiver can request picture repair, see Section 8. The value is a comma separated list of hash types that is supported or requested to be used, each hash type provided as an unsigned integer value (0-255), with the hash types listed from most preferred to the least preferred. Example: "include-dph=0,2", which indicates the capability for MD5 (most preferred) and Checksum (less preferred). If the parameter is not included or the value contains no hash types, then no capability to utilize DPH SEI messages is assumed. Note that DPH SEI messages MAY still be included in the bitstream even when there is no declaration of capability to use them, as in general SEI messages do not affect the normative decoding process and decoders are allowed to ignore SEI messages.

Encoding considerations:

This type is only defined for transfer via RTP (RFC 3550).

Security considerations:

See Section 9 of RFC XXXX.

Public specification:

Please refer to Section 13 of RFC XXXX.

Additional information: None

File extensions: none

Macintosh file type code: none

Object identifier or OID: none

Person & email address to contact for further information:

Ye-Kui Wang (yekuiw@qti.qualcomm.com).

Intended usage: COMMON

Author: See Section 14 of RFC XXXX.

Change controller:

IETF Audio/Video Transport Payloads working group delegated from the IESG.

7.2 SDP Parameters

The receiver MUST ignore any parameter unspecified in this memo.

7.2.1 Mapping of Payload Type Parameters to SDP

The media type video/H265 string is mapped to fields in the Session Description Protocol (SDP) [RFC4566] as follows:

- The media name in the "m=" line of SDP MUST be video.
- The encoding name in the "a=rtpmap" line of SDP MUST be H265 (the media subtype).

- The clock rate in the "a=rtpmap" line MUST be 90000.

- The OPTIONAL parameters "profile-space", "profile-id", "tier-flag", "level-id", "interop-constraints", "profile-compatibility-indicator", "sprop-sub-layer-id", "recv-sub-layer-id", "max-recv-level-id", "tx-mode", "max-lsr", "max-lps", "max-cpb", "max-dpb", "max-br", "max-tr", "max-tc", "max-fps", "sprop-max-don-diff", "sprop-depack-buf-nalus", "sprop-depack-buf-bytes", "depack-buf-cap", "sprop-segmentation-id", "sprop-spatial-segmentation-idc", "dec-parallel-cap", and "include-dph", when present, MUST be included in the "a=fmtp" line of SDP. This parameter is expressed as a media type string, in the form of a semicolon separated list of parameter=value pairs.

- The OPTIONAL parameters "sprop-vps", "sprop-sps", and "sprop-pps", when present, MUST be included in the "a=fmtp" line of SDP or conveyed using the "fmtp" source attribute as specified in Section 6.3 of [RFC5576]. For a particular media format (i.e. RTP payload type), "sprop-vps" "sprop-sps", or "sprop-pps" MUST NOT be both included in the "a=fmtp" line of SDP and conveyed using the "fmtp" source attribute. When included in the "a=fmtp" line of SDP, these parameters are expressed as a media type string, in the form of a semicolon separated list of parameter=value pairs. When conveyed in the "a=fmtp" line of SDP for a particular payload type, the parameters "sprop-vps", "sprop-sps", and "sprop-pps" MUST be applied to each SSRC with the payload type. When conveyed using the "fmtp" source attribute, these parameters are only associated with the given source and payload type as parts of the "fmtp" source attribute.

  Informative note: Conveyance of "sprop-vps", "sprop-sps", and "sprop-pps" using the "fmtp" source attribute allows for out-of-band transport of parameter sets in topologies like Topo-Video-switch-MCU as specified in [RFC5117].

An example of media representation in SDP is as follows:

```
m=video 49170 RTP/AVP 98
a=rtpmap:98 H265/90000
```

```
a=fmtp:98 profile-id=1;
          sprop-vps=<video parameter sets data>
```

7.2.2 Usage with SDP Offer/Answer Model

When HEVC is offered over RTP using SDP in an Offer/Answer model [RFC3264] for negotiation for unicast usage, the following limitations and rules apply:

- The parameters identifying a media format configuration for HEVC are profile-space, profile-id, tier-flag, level-id, interop-constraints, profile-compatibility-indicator, and tx-mode. These media configuration parameters, except level-id, MUST be used symmetrically when the answerer does not include recv-sub-layer-id in the answer for the media format (payload type) or the included recv-sub-layer-id is equal to sprop-sub-layer-id in the offer. The answerer MUST

  1) maintain all configuration parameters with the values remaining the same as in the offer for the media format (payload type), with the exception that the value of level-id is changeable as long as the highest level indicated by the answer is not higher than that indicated by the offer;

  2) include in the answer the recv-sub-layer-id parameter, with a value less than the sprop-sub-layer-id parameter in the offer, for the media format (payload type), and maintain all configuration parameters with the values being the same as signalled in the sprop-vps for the chosen sub-layer representation, with the exception that the value of level-id is changeable as long as the highest level indicated by the answer is not higher than the level indicated by the sprop-vps in offer for the chosen sub-layer representation; or

  3) remove the media format (payload type) completely (when one or more of the parameter values are not supported).

  Informative note: The above requirement for symmetric use does not apply for level-id, and does not apply for the

other bitstream or RTP stream properties and capability parameters.

- The profile-compatibility-indicator, when offered as sendonly, describe bitstream properties. The answerer MAY accept an RTP payload type even if the decoder is not capable of handling the profile indicated by the profile-space, profile-id, and interop-constraints parameters, but capable of any of the profiles indicated by the profile-space, profile-compatibility-indicator, and interop-constraints. However, when the profile-compatibility-indicator is used in a recvonly or sendrecv media description, the bitstream using this RTP payload type is required to conform to all profiles indicated by profile-space, profile-compatibility-indicator, and interop-constraints.

- To simplify handling and matching of these configurations, the same RTP payload type number used in the offer SHOULD also be used in the answer, as specified in [RFC3264].

- The same RTP payload type number used in the offer for the media subtype H265 MUST be used in the answer when the answer includes recv-sub-layer-id. When the answer does not include recv-sub-layer-id, the answer MUST NOT contain a payload type number used in the offer for the media subtype H265 unless the configuration is exactly the same as in the offer or the configuration in the answer only differs from that in the offer with a different value of level-id. The answer MAY contain the recv-sub-layer-id parameter if an HEVC bitstream contains multiple operation points (using temporal scalability and sub-layers) and sprop-vps is included in the offer where information of sub-layers are present in the first video parameter set contained in sprop-vps. If the sprop-vps is provided in an offer, an answerer MAY select a particular operation point indicated in the first video parameter set contained in sprop-vps. When the answer includes recv-sub-layer-id that is less than sprop-sub-layer-id in the offer, all video parameter sets contained in the sprop-vps parameter in the SDP answer and all video parameter sets sent in-band for either the offerer-to-answerer direction or the answerer-to-offerer direction MUST be consistent with the first video

parameter set in the sprop-vps parameter of the offer (see the semantics of sprop-vps in Section 7.1 of this document on one video parameter set being consistent with another video parameter set), and the bitstream sent in either direction MUST conform to the profile, tier, level, and constraints of the chosen sub-layer representation as indicated by the first profile_tier_level( ) syntax structure in the first video parameter set in the sprop-vps parameter of the offer.

Informative note: When an offerer receives an answer that does not include recv-sub-layer-id, it has to compare payload types not declared in the offer based on the media type (i.e. video/H265) and the above media configuration parameters with any payload types it has already declared. This will enable it to determine whether the configuration in question is new or if it is equivalent to configuration already offered, since a different payload type number may be used in the answer. The ability to perform operation point selection enables a receiver to utilize the temporal scalable nature of an HEVC bitstream.

- The parameters sprop-max-don-diff, sprop-depack-buf-nalus, and sprop-depack-buf-bytes describe the properties of an RTP stream, and all RTP streams the RTP stream depends on, when present, that the offerer or the answerer is sending for the media format configuration. This differs from the normal usage of the Offer/Answer parameters: normally such parameters declare the properties of the bitstream or RTP stream that the offerer or the answerer is able to receive. When dealing with HEVC, the offerer assumes that the answerer will be able to receive media encoded using the configuration being offered.

  Informative note: The above parameters apply for any RTP stream and all RTP streams the RTP stream depends on, when present, sent by a declaring entity with the same configuration. In other words, the applicability of the above parameters to RTP streams depends on the source endpoint. Rather than being bound to the payload type, the values may have to be applied to another payload type when being sent, as they apply for the configuration.

- The capability parameters max-lsr, max-lps, max-cpb, max-dpb, max-br, max-tr, and max-tc MAY be used to declare further capabilities of the offerer or answerer for receiving. These parameters MUST NOT be present when the direction attribute is "sendonly".

- The capability parameter max-fps MAY be used to declare lower capabilities of the offerer or answerer for receiving. The parameters MUST NOT be present when the direction attribute is "sendonly".

- The capability parameter dec-parallel-cap MAY be used to declare additional decoding capabilities of the offerer or answerer for receiving. Upon receiving such a declaration of a receiver, a sender MAY send a bitstream to the receiver utilizing those capabilities under the assumption that the bitstream fulfills the parallelism requirement. A bitstream that is sent based on choosing a capability point with parallel tool type ’w’ from dec-parallel-cap MUST have entropy_coding_sync_enabled_flag equal to 1 and min_spatial_segmentation_idc equal to or larger than dec-parallel-cap.spatial-seg-idc of the capability point. A bitstream that is sent based on choosing a capability point with parallel tool type ’t’ from dec-parallel-cap MUST have entropy_coding_sync_enabled_flag equal to 0 and min_spatial_segmentation_idc equal to or larger than dec-parallel-cap.spatial-seg-idc of the capability point.

- An offerer has to include the size of the de-packetization buffer, sprop-depack-buf-bytes, as well as sprop-max-don-diff and sprop-depack-buf-nalus, in the offer for an interleaved HEVC bitstream or for the MRST or MRMT transmission mode when sprop-max-don-diff is greater than 0 for at least one of the RTP streams. To enable the offerer and answerer to inform each other about their capabilities for de-packetization buffering in receiving RTP streams, both parties are RECOMMENDED to include depack-buf-cap. For interleaved RTP streams or in MRST or MRMT, it is also RECOMMENDED to consider offering multiple payload types with different buffering requirements when the capabilities of the receiver are unknown.

- The capability parameter include-dph MAY be used to declare the capability to utilize decoded picture hash SEI messages and which types of hashes in any HEVC RTP streams received by the offerer or answerer.

- The sprop-vps, sprop-sps, or sprop-pps, when present (included in the "a=fmtp" line of SDP or conveyed using the "fmtp" source attribute as specified in Section 6.3 of [RFC5576]), are used for out-of-band transport of the parameter sets (VPS, SPS, or PPS respectively).

- The answerer MAY use either out-of-band or in-band transport of parameter sets for the bitstream it is sending, regardless of whether out-of-band parameter sets transport has been used in the offerer-to-answerer direction. Parameter sets included in an answer are independent of those parameter sets included in the offer, as they are used for decoding two different bitstreams, one from the answerer to the offerer and the other in the opposite direction. In case some RTP stream(s) are sent before SDP offer/answer settles down, in-band parameter sets MUST be used for those RTP stream parts sent before the SDP offer/answer.

- The following rules apply to transport of parameter set in the offerer-to-answerer direction.

  - An offer MAY include sprop-vps, sprop-sps, and/or sprop-pps. If none of these parameters is present in the offer, then only in-band transport of parameter sets is used.

  - If the level to use in the offerer-to-answerer direction is equal to the default level in the offer, the answerer MUST be prepared to use the parameter sets included in sprop-vps, sprop-sps, and sprop-pps (either included in the "a=fmtp" line of SDP or conveyed using the "fmtp" source attribute) for decoding the incoming bitstream, e.g. by passing these parameter set NAL units to the video decoder before passing any NAL units carried in the RTP streams. Otherwise, the answerer MUST ignore sprop-vps, sprop-sps, and sprop-pps (either included in the "a=fmtp"

line of SDP or conveyed using the "fmtp" source attribute) and the offerer MUST transmit parameter sets in-band.

- In MRST or MRMT, the answerer MUST be prepared to use the parameter sets out-of-band transmitted for the RTP stream and all RTP streams the RTP stream depends on, when present, for decoding the incoming bitstream, e.g. by passing these parameter set NAL units to the video decoder before passing any NAL units carried in the RTP streams.

- The following rules apply to transport of parameter set in the answerer-to-offerer direction.

  - An answer MAY include sprop-vps, sprop-sps, and/or sprop-pps. If none of these parameters is present in the answer, then only in-band transport of parameter sets is used.

  - The offerer MUST be prepared to use the parameter sets included in sprop-vps, sprop-sps, and sprop-pps (either included in the "a=fmtp" line of SDP or conveyed using the "fmtp" source attribute) for decoding the incoming bitstream, e.g. by passing these parameter set NAL units to the video decoder before passing any NAL units carried in the RTP streams.

  - In MRST or MRMT, the offerer MUST be prepared to use the parameter sets out-of-band transmitted for the RTP stream and all RTP streams the RTP stream depends on, when present, for decoding the incoming bitstream, e.g. by passing these parameter set NAL units to the video decoder before passing any NAL units carried in the RTP streams.

- When sprop-vps, sprop-sps, and/or sprop-pps are conveyed using the "fmtp" source attribute as specified in Section 6.3 of [RFC5576], the receiver of the parameters MUST store the parameter sets included in sprop-vps, sprop-sps, and/or sprop-pps and associate them with the source given as part of the "fmtp" source attribute. Parameter sets associated with one source (given as part of the "fmtp" source attribute) MUST only be used to decode NAL units conveyed in RTP packets from

the same source (given as part of the "fmtp" source attribute). When this mechanism is in use, SSRC collision detection and resolution MUST be performed as specified in [RFC5576].

For bitstreams being delivered over multicast, the following rules apply:

- The media format configuration is identified by profile-space, profile-id, tier-flag, level-id, interop-constraints, profile-compatibility-indicator, and tx-mode. These media format configuration parameters, including level-id, MUST be used symmetrically; that is, the answerer MUST either maintain all configuration parameters or remove the media format (payload type) completely. Note that this implies that the level-id for Offer/Answer in multicast is not changeable.

- To simplify the handling and matching of these configurations, the same RTP payload type number used in the offer SHOULD also be used in the answer, as specified in [RFC3264]. An answer MUST NOT contain a payload type number used in the offer unless the configuration is the same as in the offer.

- Parameter sets received MUST be associated with the originating source and MUST only be used in decoding the incoming bitstream from the same source.

- The rules for other parameters are the same as above for unicast as long as the three above rules are obeyed.

Table 1 lists the interpretation of all the parameters that MUST be used for the various combinations of offer, answer, and direction attributes. Note that the two columns wherein the recv-sub-layer-id parameter is used only apply to answers, whereas the other columns apply to both offers and answers.

Table 1. Interpretation of parameters for various combinations of offers, answers, direction attributes, with and without recv-sub-layer-id. Columns that do not indicate offer or answer apply to both.

				sendonly --+	
		answer: recvonly, recv-sub-layer-id --+		|	
		recvonly w/o recv-sub-layer-id --+		|	|
	answer: sendrecv, recv-sub-layer-id --+			|	|
	sendrecv w/o recv-sub-layer-id --+			|	|
	|	|	|	|	|
profile-space	C	D	C	D	P
profile-id	C	D	C	D	P
tier-flag	C	D	C	D	P
level-id	D	D	D	D	P
interop-constraints	C	D	C	D	P
profile-compatibility-indicator	C	D	C	D	P
tx-mode	C	C	C	C	P
max-recv-level-id	R	R	R	R	-
sprop-max-don-diff	P	P	-	-	P
sprop-depack-buf-nalus	P	P	-	-	P
sprop-depack-buf-bytes	P	P	-	-	P
depack-buf-cap	R	R	R	R	-
sprop-segmentation-id	P	P	P	P	P
sprop-spatial-segmentation-idc	P	P	P	P	P
max-br	R	R	R	R	-
max-cpb	R	R	R	R	-
max-dpb	R	R	R	R	-
max-lsr	R	R	R	R	-
max-lps	R	R	R	R	-
max-tr	R	R	R	R	-
max-tc	R	R	R	R	-
max-fps	R	R	R	R	-
sprop-vps	P	P	-	-	P
sprop-sps	P	P	-	-	P
sprop-pps	P	P	-	-	P
sprop-sub-layer-id	P	P	-	-	P
recv-sub-layer-id	X	O	X	O	-
dec-parallel-cap	R	R	R	R	-
include-dph	R	R	R	R	-

Legend:

C: configuration for sending and receiving bitstreams
D: changable configuration, same as C except possible
to answer with a different but consistent value (see the

semantics of the six parameters related to profile, tier, and level on these parameters being consistent)
P: properties of the bitstream to be sent
R: receiver capabilities
O: operation point selection
X: MUST NOT be present
-: not usable, when present MUST be ignored

Parameters used for declaring receiver capabilities are in general downgradable; i.e. they express the upper limit for a sender's possible behavior. Thus, a sender MAY select to set its encoder using only lower/lesser or equal values of these parameters.

When the answer does not include recv-sub-layer-id that is less than the sprop-sub-layer-id in the offer, parameters declaring a configuration point are not changeable, with the exception of the level-id parameter for unicast usage, and these parameters express values a receiver expects to be used and MUST be used verbatim in the answer as in the offer.

When a sender's capabilities are declared with the configuration parameters, these parameters express a configuration that is acceptable for the sender to receive bitstreams. In order to achieve high interoperability levels, it is often advisable to offer multiple alternative configurations. It is impossible to offer multiple configurations in a single payload type. Thus, when multiple configuration offers are made, each offer requires its own RTP payload type associated with the offer. However, it is possible to offer multiple operation points using one configuration in a single payload type by including sprop-vps in the offer and recv-sub-layer-id in the answer.

A receiver SHOULD understand all media type parameters, even if it only supports a subset of the payload format's functionality. This ensures that a receiver is capable of understanding when an offer to receive media can be downgraded to what is supported by the receiver of the offer.

An answerer MAY extend the offer with additional media format configurations. However, to enable their usage, in most cases a

second offer is required from the offerer to provide the bitstream property parameters that the media sender will use. This also has the effect that the offerer has to be able to receive this media format configuration, not only to send it.

7.2.3 Usage in Declarative Session Descriptions

When HEVC over RTP is offered with SDP in a declarative style, as in Real Time Streaming Protocol (RTSP) [RFC2326] or Session Announcement Protocol (SAP) [RFC2974], the following considerations are necessary.

- All parameters capable of indicating both bitstream properties and receiver capabilities are used to indicate only bitstream properties. For example, in this case, the parameter profile-tier-level-id declares the values used by the bitstream, not the capabilities for receiving bitstreams. This results in that the following interpretation of the parameters MUST be used:

  - Declaring actual configuration or bitstream properties:
    - profile-space
    - profile-id
    - tier-flag
    - level-id
    - interop-constraints
    - profile-compatibility-indicator
    - tx-mode
    - sprop-vps
    - sprop-sps
    - sprop-pps
    - sprop-max-don-diff
    - sprop-depack-buf-nalus
    - sprop-depack-buf-bytes
    - sprop-segmentation-id
    - sprop-spatial-segmentation-idc

  - Not usable (when present, they MUST be ignored):
    - max-lps
    - max-lsr
    - max-cpb

- max-dpb
- max-br
- max-tr
- max-tc
- max-fps
- max-recv-level-id
- depack-buf-cap
- sprop-sub-layer-id
- dec-parallel-cap
- include-dph

o A receiver of the SDP is required to support all parameters and values of the parameters provided; otherwise, the receiver MUST reject (RTSP) or not participate in (SAP) the session. It falls on the creator of the session to use values that are expected to be supported by the receiving application.

7.2.4 Parameter Sets Considerations

When out-of-band transport of parameter sets is used, parameter sets MAY still be additionally transported in-band unless explicitly disallowed by an application, and some of these additionally in-band transported parameter sets may update some of the out-of-band transported parameter sets. Update of a parameter set refers to sending of a parameter set of the same type using the same parameter set ID but with different values for at least one other parameter of the parameter set.

7.2.5 Dependency Signaling in Multi-Stream Mode

If MRST or MRMT is used, the rules on signaling media decoding dependency in SDP as defined in [RFC5583] apply. The rules on "hierarchical or layered encoding" with multicast in Section 5.7 of [RFC4566] do not apply. This means that the notation for Connection Data "c=" SHALL NOT be used with more than one address, i.e. the sub-field <number of addresses> in the sub-field <connection-address> of the "c=" field, described in [RFC4566], must not be present. The order of session dependency is given from the RTP stream containing the lowest temporal sub-layer to the RTP stream containing the highest temporal sub-layer.

8 Use with Feedback Messages

The following subsections define the use of the Picture Loss Indication (PLI), Slice Lost Indication (SLI), Reference Picture Selection Indication (RPSI), and Full Intra Request (FIR) feedback messages with HEVC. The PLI, SLI, and RPSI messages are defined in RFC 4585 [RFC4585], and the FIR message is defined in RFC 5104 [RFC5104].

8.1 Picture Loss Indication (PLI)

As specified in RFC 4585 Section 6.3.1, the reception of a picture loss indication by a media sender indicates "the loss of an undefined amount of coded video data belonging to one or more pictures." Without having any specific knowledge of the setup of the bitstream (such as: use and location of in-band parameter sets, non-IDR decoder refresh points, picture structures, and so forth) a reaction to the reception of an PLI by an HEVC sender SHOULD be to send an IDR picture and relevant parameter sets; potentially with sufficient redundancy so to ensure correct reception. However, sometimes information about the bitstream structure is known. For example, state could have been established outside of the mechanisms defined in this document that parameter sets are conveyed out of band only, and stay static for the duration of the session. In that case, it is obviously unnecessary to send them in-band as a result of the reception of a PLI. Other examples could be devised based on a priori knowledge of different aspects of the bitstream structure. In all cases, the timing and congestion control mechanisms of RFC 4585 MUST be observed.

8.2 Slice Loss Indication (SLI)

RFC 4585's Slice Loss Indication can be used to indicate, to a sender, the loss of a number of Coded Tree Blocks (CTBs) in CTB raster scan order of a picture. In the SLI's Feedback Control Indication (FCI) field, the subfield "First" MUST be set to the CTB address of the first lost CTB. Note that the CTB address is in CTB raster scan order of a picture. For the first CTB of a slice segment, the CTB address is the value of slice_segment_address when present; or 0 when the value of

first_slice_segement_in_pic_flag is equal to 1; both syntax elements are in the slice segment header. The subfield "Number" MUST be set to the number of consecutive lost CTBs, again in CTB raster scan order of a picture. Note that due to both the "First" and "Number" are counted in CTBs in CTB raster scan order, of a picture, not in tile scan order (which is the bitstream order of CTBs), multiple SLI messages may be needed to report the loss of one tile covering multiple CTB rows but less wide than the picture.

The subfield "PictureID" MUST be set to the 6 least significant bits of a binary representation of the value of PicOrderCntVal, as defined in [HEVC], of the picture for which the lost CTBs are indicated. Note that for IDR pictures the syntax element slice_pic_order_cnt_lsb is not present, but then the value is inferred to be equal to 0.

As described in RFC 4585, an encoder in a media sender can use these information to "clean up" the corrupted picture by sending intra information, while observing the constraints described in RFC 4585, for example with respect to congestion control. In many cases, error tracking is required to identify the corrupted region in the receiver’s state (reference pictures) because of error import in uncorrupted regions of the picture through motion compensation. Reference picture selection can also be used to "clean up" the corrupted picture, which is usually more efficient and less likely to generate congestion than sending intra information.

In contrast to the video codecs contemplated in RFC 4585 and RFC 5104 [RFC5104], in HEVC, the "macroblock size" is not fixed to 16x16 luma samples, but variable. That, however, does not create a conceptual difficulty with SLI, because the setting of the CTB size is a sequence-level functionality, and using a slice loss indication across CVS boundaries is meaningless as there is no prediction across sequence boundaries. However, a proper use of SLI messages is not as straightforward as it was with older, fixed-macroblock-sized video codecs, as the state of the sequence parameter set (where the CTB size is located) has to be taken into account when interpreting the "First" subfield in the FCI.

8.3 Reference Picture Selection Indication (RPSI)

Feedback based reference picture selection has been shown as a powerful tool to stop temporal error propagation for improved error resilience [Girod99][Wang05]. In one approach, the decoder side tracks errors in the decoded pictures and informs to the encoder side that a particular picture that has been decoded relatively earlier is correct and still present in the decoded picture buffer and requests the encoder to use that correct picture availability information when encoding the next picture, so to stop further temporal error propagation. For this approach, the decoder side should use the RPSI feedback message.

Encoders can encode some long-term reference pictures as specified in H.264 or HEVC for purposes described in the previous paragraph without the need of a huge decoded picture buffer. As shown in [Wang05], with a flexible reference picture management scheme as in H.264 and HEVC, even a decoded picture buffer size of two picture storage buffers would work for the approach described in the previous paragraph.

The field "Native RPSI bit string defined per codec" is a base16 [RFC4648] representation of the 8 bits consisting of 2 most significant bits equal to 0 and 6 bits of nuh_layer_id, as defined in [HEVC], followed by the 32 bits representing the value of the PicOrderCntVal (in network byte order), as defined in [HEVC], for the picture that is indicated by the RPSI feedback message.

The use of the RPSI feedback message as positive acknowledgement with HEVC is deprecated. In other words, the RPSI feedback message MUST only be used as a reference picture selection request, such that it can also be used in multicast.

8.4 Full Intra Request (FIR)

The purpose of the FIR message is to force an encoder to send an independent decoder refresh point as soon as possible (observing, for example, the congestion control related constraints set out in RFC 5104).

Upon reception of a FIR, a sender MUST send an IDR picture. Parameter sets MUST also be sent, except when there is a priori knowledge that the parameter sets have been correctly established. A typical example for that is an understanding between sender and receiver, established by means outside this document, that parameter sets are exclusively sent out of band.

9 Security Considerations

The scope of this Security Considerations section is limited to the payload format itself, and to one feature of HEVC that may pose a particularly serious security risk if implemented naively. The payload format, in isolation, does not form a complete system. Implementers are advised to read and understand relevant security related documents, especially those pertaining to RTP (see the security considerations section in RFC 3550 [RFC3550]), and the security of the call control stack chosen (that may make use of the media type registration of this memo). Implementers should also consider known security vulnerabilities of video coding and decoding implementations in general and avoid those.

Within this RTP payload format, and with the exception of the user data SEI message as described below, no security threats other than those common to RTP payload formats are known. In other words, neither the various media plane based mechanisms, nor the signaling part of this memo, seems to pose a security risk beyond those common to all RTP based systems.

RTP packets using the payload format defined in this specification are subject to the security considerations discussed in the RTP specification [RFC3550], and in any applicable RTP profile such as RTP/AVP [RFC3551], RTP/AVPF [RFC4585], RTP/SAVP [RFC3711], or RTP/SAVPF [RFC5124]. However, as "Securing the RTP Protocol Framework: Why RTP Does Not Mandate a Single Media Security Solution" RFC 7202 [RFC7202] discusses, it is not an RTP payload format's responsibility to discuss or mandate what solutions are used to meet the basic security goals like confidentiality, integrity and source authenticity for RTP in general. This responsibility lays on anyone using RTP in an application. They can find guidance on available security mechanisms and important considerations in Options for Securing

RTP Sessions [RFC7201]. Applications SHOULD use one or more appropriate strong security mechanisms. The rest of this security consideration section discusses the security impacting properties of the payload format itself.

Because the data compression used with this payload format is applied end-to-end, any encryption needs to be performed after compression. A potential denial-of-service threat exists for data encodings using compression techniques that have non-uniform receiver-end computational load. The attacker can inject pathological datagrams into the bitstream that are complex to decode and that cause the receiver to be overloaded. H.265 is particularly vulnerable to such attacks, as it is extremely simple to generate datagrams containing NAL units that affect the decoding process of many future NAL units. Therefore, the usage of data origin authentication and data integrity protection of at least the RTP packet is RECOMMENDED, for example, with SRTP [RFC3711].

Like [H.264], HEVC includes a user data Supplementary Enhancement Information (SEI) message. This SEI message allows inclusion of an arbitrary bitstring into the video bitstream. Such a bitstring could include JavaScript, machine code, and other active content. HEVC leaves the handling of this SEI message to the receiving system. In order to avoid harmful side effects of the user data SEI message, decoder implementations cannot naviely trust its content. For example, it would be a bad and insecure implementation practice to forward any JavaScript a decoder implementation detects to a web browser. The safest way to deal with user data SEI messages is to simply discard them, but that can have negative side effects on the quality of experience by the user.

End-to-end security with authentication, integrity, or confidentiality protection will prevent a MANE from performing media-aware operations other than discarding complete packets. In the case of confidentiality protection, it will even be prevented from discarding packets in a media-aware way. To be allowed to perform such operations, a MANE is required to be a trusted entity that is included in the security context establishment.

10 Congestion Control

Congestion control for RTP SHALL be used in accordance with RTP [RFC3550] and with any applicable RTP profile, e.g. AVP [RFC3551]. If best-effort service is being used, an additional requirement is that users of this payload format MUST monitor packet loss to ensure that the packet loss rate is within an acceptable range. Packet loss is considered acceptable if a TCP flow across the same network path, and experiencing the same network conditions, would achieve an average throughput, measured on a reasonable timescale, that is not less than all RTP streams combined is achieving. This condition can be satisfied by implementing congestion control mechanisms to adapt the transmission rate, the number of layers subscribed for a layered multicast session, or by arranging for a receiver to leave the session if the loss rate is unacceptably high.

The bitrate adaptation necessary for obeying the congestion control principle is easily achievable when real-time encoding is used, for example by adequately tuning the quantization parameter.

However, when pre-encoded content is being transmitted, bandwidth adaptation requires the pre-coded bitstream to be tailored for such adaptivity. The key mechanism available in HEVC is temporal scalability. A media sender can remove NAL units belonging to higher temporal sub-layers (i.e. those NAL units with a high value of TID) until the sending bitrate drops to an acceptable range. HEVC contains mechanisms that allow the lightweight identification of switching points in temporal enhancement layers, as discussed in Section 1.1.2 of this memo. An HEVC media sender can send packets belonging to NAL units of temporal enhancement layers starting from these switching points to probe for available bandwidth and to utilized bandwidth that has been shown to be available.

Above mechanisms generally work within a defined profile and level and, therefore, no renegotiation of the channel is required. Only when non-downgradable parameters (such as profile) are required to be changed does it become necessary to

terminate and restart the RTP stream(s). This may be accomplished by using different RTP payload types.

MANEs MAY remove certain unusable packets from the RTP stream when that RTP stream was damaged due to previous packet losses. This can help reduce the network load in certain special cases. For example, MANES can remove those FUs where the leading FUs belonging to the same NAL unit have been lost or those dependent slice segments when the leading slice segments belonging to the same slice have been lost, because the trailing FUs or dependent slice segments are meaningless to most decoders. MANES can also remove higher temporal scalable layers if the outbound transmission (from the MANE's viewpoint) experiences congestion.

11 IANA Consideration

A new media type, as specified in Section 7.1 of this memo, should be registered with IANA.

12 Acknowledgements

Muhammed Coban and Marta Karczewicz are thanked for discussions on the specification of the use with feedback messages and other aspects in this memo. Jonathan Lennox and Jill Boyce are thanked for their contributions to the PACI design included in this memo. Rickard Sjoberg, Arild Fuldseth, Bo Burman, Magnus Westerlund, and Tom Kristensen are thanked for their contributions to parallel processing related signalling. Magnus Westerlund, Jonathan Lennox, Bernard Aboba, Jonatan Samuelsson, Roni Even, Rickard Sjoberg, Sachin Deshpande, Woo Johnman, Mo Zanaty, Ross Finlayson, Danny Hong, Bo Burman, Ben Campbell, Brian Carpenter, Qin Wu, and Stephen Farrell made valuable reviewing comments that led to improvements.

This document was prepared using 2-Word-v2.0.template.dot, and the .txt file was generated using the online Word-post procesor available here: http://www.isi.edu/touch/tools/rfc-word-template.html.

13 References

13.1 Normative References

[HEVC] ITU-T Recommendation H.265, "High efficiency video coding", April 2013.

[H.264] ITU-T Recommendation H.264, "Advanced video coding for generic audiovisual services", April 2013.

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

[RFC3264] Rosenberg, J. and H. Schulzrinne, "An Offer/Answer Model with Session Description Protocol (SDP)", RFC 3264, June 2002.

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

[RFC3551] Schulzrinne, H. and Casner, S., "RTP Profile for Audio and Video Conferences with Minimal Control", STD 65, RFC 3551, July 2003.

[RFC3711] Baugher, M., McGrew, D., Naslund, M., Carrara, E., and Norrman, K., "The Secure Real-time Transport Protocol (SRTP)", RFC 3711, March 2004.

[RFC4566] Handley, M., Jacobson, V., and Perkins, C., "SDP: Session Description Protocol", RFC 4566, July 2006.

[RFC4585] Ott, J., Wenger, S., Sato, N., Burmeister, C., and Rey, J., "Extended RTP Profile for Real-time Transport Control Protocol (RTCP)-Based Feedback (RTP/AVPF)", RFC 4585, July 2006.

[RFC4648] Josefsson, S., "The Base16, Base32, and Base64 Data Encodings", RFC 4648, October 2006.

[RFC5104] Wenger, S., Chandra, U., Westerlund, M., and Burman, B., "Codec Control Messages in the RTP Audio-Visual Profile with Feedback (AVPF)", RFC 5104, February 2008.

[RFC5124] Ott, J. and Carrara, E., "Extended Secure RTP Profile for Real-time Transport Control Protocol (RTCP)-Based Feedback (RTP/SAVPF)", RFC 5124, February 2008.

[RFC5234] Crocker, D. and Overell, P., "Augmented BNF for Syntax Specifications: ABNF", RFC 5234, January 2008.

[RFC5576] Lennox, J., Ott, J., and Schierl, T., "Source-Specific Media Attributes in the Session Description Protocol", RFC 5576, June 2009.

[RFC5583] Schierl, T. and Wenger, S., "Signaling Media Decoding Dependency in the Session Description Protocol (SDP)", RFC 5583, July 2009.

13.2 Informative References

[3GPDASH] 3GPP TS 26.247, "Transparent end-to-end Packet-switched Streaming Service (PSS); Progressive Download and Dynamic Adaptive Streaming over HTTP (3GP-DASH)", v12.1.0, December 2013.

[3GPPFF] 3GPP TS 26.244, "Transparent end-to-end packet switched streaming service (PSS); 3GPP file format (3GP)", v12.20, December 2013.

[CABAC] Sole, J., Joshi, R., Nguyen, N., Ji, T., Karczewicz, M., Clare, G., Henry, F., and Duenas, A., "Transform coefficient coding in HEVC", IEEE Transactions on Circuts and Systems for Video Technology, Vol. 22, No. 12, pp. 1765-1777, December 2012.

[Girod99] Girod, B. and Faerber, F., "Feedback-based error control for mobile video transmission", Proceedings IEEE, Vol. 87, No. 10, pp. 1707-1723, October 1999.

[HEVC draft v2]
Draft version 2 of HEVC, "High Efficiency Video Coding (HEVC) Range Extensions text specification: Draft 7", JCT-VC document JCTVC-Q1005, 17th JCT-VC meeting, 27 March - 4 April 2014, Valencia, Spain.

[I-D.ietf-avtcore-rtp-multi-stream]
Lennox, J., Westerlund, M., Wu, W., and C. Perkins, "Sending Multiple Media Streams in a Single RTP Session", draft-ietf-avtcore-rtp-multi-stream-09 (work in progress), September 2015.

[I-D.ietf-mmusic-sdp-bundle-negotiation]
Holmberg, C., Alvestrand, H., and C. Jennings, "Multiplexing Negotiation Using Session Description Protocol (SDP) Port Numbers", draft-ietf-mmusic-sdp-bundle-negotiation-23 (work in progress), July 2015.

[I-D.ietf-avtext-rtp-grouping-taxonomy]
Lennox, J., Gross, K., Nandakumar, S., Salgueiro, G., and Burman, B. "A Taxonomy of Grouping Semantics and Mechanisms for Real-Time Transport", draft-ietf-avtext-rtp-grouping-taxonomy-08 (work in progress), July 2015.

[ISOBMFF] IS0/IEC 14496-12 | 15444-12: "Information technology - Coding of audio-visual objects - Part 12: ISO base media file format" | "Information technology - JPEG 2000 image coding system - Part 12: ISO base media file format", 2012.

[JCTVC-J0107]
Wang, Y.-K., Chen, Y., Joshi, R., and Ramasubramonian, K., "AHG9: On RAP pictures", JCT-VC document JCTVC-L0107, 10th JCT-VC meeting, July 2012, Stockholm, Sweden.

[MPEG2S] ISO/IEC 13818-1, "Information technology - Generic coding of moving pictures and associated audio information: Systems", 2013.

[MPEGDASH] ISO/IEC 23009-1, "Information technology - Dynamic adaptive streaming over HTTP (DASH) - Part 1: Media presentation description and segment formats", 2012.

[RFC2326] Schulzrinne, H., Rao, A., and Lanphier R., "Real Time Streaming Protocol (RTSP)", RFC 2326, April 1998.

[RFC2974] Handley, M., Perkins C., and Whelan E., "Session Announcement Protocol", RFC 2974, October 2000.

[RFC5117] Westerlund, M. and Wenger, S., "RTP Topologies", RFC 5117, January 2008.

[RFC6051] Perkins, C. and T. Schierl, "Rapid Synchronisation of RTP Flows", RFC 6051, November 2010.

[RFC6184] Wang, Y.-K., Even, R., Kristensen, T., and R. Jesup, "RTP Payload Format for H.264 Video", RFC 6184, May 2011.

[RFC6190] Wenger, S., Wang, Y.-K., Schierl, T., and A. Eleftheriadis, "RTP Payload Format for Scalable Video Coding", RFC 6190, May 2011.

[RFC7201] Westerlund, M. and Perkins, C., "Options for Securing RTP Sessions", RFC 7201, April 2014.

[RFC7202] Perkins, C. and Westerlund, M., "Securing the RTP Framework: Why RTP Does Not Mandate a Single Media Security Solution", RFC 7202, April 2014.

[Wang05] Wang, Y.-K., Zhu, C., and Li, H., "Error resilient video coding using flexible reference fames", Visual Communications and Image Processing 2005 (VCIP 2005), July 2005, Beijing, China.

14 Authors' Addresses

Ye-Kui Wang
Qualcomm Incorporated
5775 Morehouse Drive
San Diego, CA 92121, USA

Phone: +1-858-651-8345
EMail: yekui.wang@gmail.com

Yago Sanchez
Fraunhofer HHI
Einsteinufer 37
D-10587 Berlin, Germany
Phone: +49-30-31002-227
Email: yago.sanchez@hhi.fraunhofer.de

Thomas Schierl
Fraunhofer HHI
Einsteinufer 37
D-10587 Berlin, Germany
Phone: +49-30-31002-227
Email: ts@thomas-schierl.de

Stephan Wenger
Vidyo, Inc.
433 Hackensack Ave., 7th floor
Hackensack, N.J. 07601, USA
Phone: +1-415-713-5473
EMail: stewe@stewe.org

Miska M. Hannuksela
Nokia Corporation
P.O. Box 1000
33721 Tampere, Finland
Phone: +358-7180-08000
EMail: miska.hannuksela@nokia.com