Network Working Group
Internet-Draft
Intended status: Standards Track
Expires: March 26, 2020

S. Zhao
S. Wenger
Tencent

September 23, 2019

RTP Payload Format for Versatile Video Coding (VVC)
draft-zhao-avtcore-rtp-vvc-00

Abstract

This memo describes an RTP payload format for the video coding standard ITU-T Recommendation H.266 and ISO/IEC International Standard 23090-3, both also known as Versatile Video Coding (VVC) and developed by the Joint Video Experts Team (JVET). 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. The payload format has wide applicability in videoconferencing, Internet video streaming, and high-bitrate entertainment-quality video, among others.

Status of This Memo

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

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

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

This Internet-Draft will expire on March 26, 2020.

Table of Contents

1. Introduction

The VVC specification, formally published as both ITU-T Recommendation H.266 and ISO/IEC International Standard 23090-23 [ISO23090-3], is planned for ratification in mid 2020. A draft that's currently in the approval process of ISO/IEC can be found as [VVC]. H.266 is reported to provide significant coding efficiency gains over H.265 [H.265] and earlier video codec formats.

This memo describes an RTP payload format for [VVC]. It shares its basic design with the NAL unit-based RTP payload formats of [RFC7798], [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 RFC 6184 is widely deployed and generally known in the relevant implementer communities. Certain mechanisms known from RFC 6190 were incorporated as [VVC] version 1 supports all temporal, spatial, and SNR scalability.

1.1. Overview of the VVC Codec

[VVC] and H.265 share a similar hybrid video codec design. In this memo, we provide a very brief overview of those features of [VVC] that are, in some form, addressed by the payload format specified herein. Implementers have to read, understand, and apply the ITU-T/ISO/IEC specifications pertaining to [VVC] to arrive at interoperable, well-performing implementations.

Conceptually, both [VVC] 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 (informative)

Coding tool features are described below with occasional reference to the coding tool set of HEVC, which is believed to be well known in the community.

Similar to earlier hybrid-video-coding-based standards, including HEVC, the following basic video coding design is employed by [VVC]. A prediction signal is first formed by either 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, VVC includes several tools to make the implementation on parallel architectures easier.

Finally, VVC includes temporal, spatial, and SNR scalability as well as multiview coding support.

Coding blocks and transform structure

Among major coding-tool differences between HEVC and [VVC], one of the important improvements is the more flexible coding tree structure in VVC, i.e., multi-type tree. In addition to quadtree, binary and ternary trees are also supported, which contributes significant improvement in coding efficiency. Moreover, the maximum size of Coding Tree Unit (CTU) is increased from 64x64 to 128x128. To improve the coding efficiency of chroma signal, luma chroma separated trees at CTU level may be employed for intra-slices. As to transform, the square transforms in HEVC are extended to non-square transforms for rectangular blocks resulted from binary and ternary tree splits. Besides, [VVC] supports multiple transform sets (MTS), including DCT-2, DST-7, and DCT-8 as well as the non-separable secondary transform. The transforms used in [VVC] can have different sizes with support for larger transform sizes. For DCT-2, the transform sizes range from 2x2 to 64x64, and for DST-7 and DCT-8, the transform sizes range from 4x4 to 32x32. In addition, [VVC] also support sub-block transform for both intra and inter coded blocks. For intra coded blocks, intra sub-partitioning (ISP) may be used to allow sub-block based intra prediction and transform. For inter blocks, sub-block transform may be used assuming that only a part of an inter-block has non-zero transform coefficients.

Entropy coding

Similar to HEVC , [VVC] uses a single entropy-coding engine, which is based on Context Adaptive Binary Arithmetic Coding (CABAC) [CABAC], but with the support of multi-window sizes. The window sizes can be initialized differently for different context models. Due to such a design, it has more efficient adaptation speed and better coding efficiency. A joint chroma residual coding scheme is applied to further exploit the correlation between the residuals of two colour components. In [VVC], different residual coding schemes are applied for regular transform coefficients and residual samples generated using transform-skip mode.

In-loop filtering

[VVC] has more feature supports in loop filters than HEVC. The deblocking filter in [VVC] is similar to HEVC but operates at a smaller grid. After deblocking and sample adaptive offset (SAO), an adaptive loop filter (ALF) may be used. As a Wiener filter, ALF reduces distortion of decoded pictures. Besides, [VVC] introduces a new module before deblocking called luma mapping with chroma scaling

to fully utilize the dynamic range of signal so that rate-distortion performance of both SDR and HDR content is improved.

Motion prediction and coding

Compared to HEVC, [VVC] introduces several improvements in this area. First, there is the Adaptive motion vector resolution (AMVR), which can save bit cost for motion vectors by adaptively signaling motion vector resolution. Then the Affine motion compensation is included to capture complicated motion like zooming and rotation. Meanwhile, prediction refinement with the optical flow with affine mode (PROF) is further deployed to mimic affine motion at the pixel level. Thirdly the decoder side motion vector refinement (DMVR) is a method to derive MV vector at decoder side so that fewer bits may be spent on motion vectors. Bi-directional optical flow (BDOF) is a similar method to DMVR but at 4x4 sub-block level. Another difference is that DMVR is based on block matching while BDOF derives MVs with equations. Furthermore, merge with motion vector difference (MMVD) is a special mode, which further signals a limited set of motion vector differences on top of merge mode. In addition to MMVD, there are another three types of special merge modes, i.e., sub-block merge, triangle, and combined intra-/inter- prediction (CIIP). Sub-block merge list includes one candidate of sub-block temporal motion vector prediction (SbTMVP) and up to four candidates of affine motion vectors. Triangle is based on triangular block motion compensation. CIIP combines intra- and inter- predictions with weighting. Moreover, weighting in bi-prediction has more flexibility then HEVC. Adaptive weighting may be employed with a block-level tool called bi-prediction with CU based weighting (BCW).

Intra prediction and intra-coding

To capture the diversified local image texture directions with finer granularity, [VVC] supports 65 angular directions instead of 33 directions in HEVC. The intra mode coding is based on a 6 most probable mode scheme, and the 6 most probable modes are derived using the neighboring intra prediction directions. In addition, to deal with the different distributions of intra prediction angles for different block aspect ratios, a wide-angle intra prediction (WAIP) scheme is applied in [VVC] by including intra prediction angles beyond those present in HEVC. Unlike HEVC which only allows using the most adjacent line of reference samples for intra prediction, [VVC] also allows using two further reference lines, as known as multi-reference-line (MRL) intra prediction. The additional reference lines can be only used for 6 most probable intra prediction modes. To capture the strong correlation between different colour components, in [VVC], a cross-component linear mode (CCLM) is utilized which assumes a linear relationship between the luma sample

values and their associated chroma samples. For intra prediction, [VVC] also applies a position-dependent prediction combination (PDPC) for refining the prediction samples closer to the intra prediction block boundary. Matrix-based intra-prediction (MIP) modes are also used in [VVC] which generates an up to 8x8 intra prediction block using a weighted sum of downsampled neighboring reference samples, and the weightings are hardcoded constants.

Other coding-tool feature

[VVC] introduces dependent quantization (DQ) to reduce quantization error by state-based switching between two quantizers.

1.1.2. Systems and Transport Interfaces

[VVC] inherits the basic systems and transport interfaces designs from HEVC and H.264. These include the NAL-unit-based syntax structure, the hierarchical syntax and data unit structure, the Supplemental Enhancement Information (SEI) message mechanism, and the video buffering model based on the Hypothetical Reference Decoder (HRD). The scalability features of [VVC] are conceptually similar to the scalable variant of HEVC known as SHVC. The hierarchical syntax and data unit structure consists of parameter sets at various levels (decoder, sequence (including layers), sequence (per layer), picture), slice-level header parameters, and lower-level parameters.

Below described are a number of key components that influenced the Network Abstraction Layer design of VVC as well as this memo.

Decoder parameter set

The Decoder parameter set includes parameters that stay constant for the lifetime of a Video Bitstream, which in IETF terms can translate to the lifetime of a session. Decoder parameter sets can include profile, level, and sub-profile information to determine a maximum complexity interop point that is guaranteed to be never exceeded, even if splicing of video sequences occurs within a session. It further optionally includes constraint flags, which indicate that the video bitstream will be constraint of the use of certain features as indicated by the values of those flags. With this, a bitstream can be labelled as not using certain tools, which allows among other things for resource allocation in a decoder implementation. As all parameter sets, also the decoder parameter set is required to be present when first referenced, and it is necessarily referenced by the very first picture in a video sequence, implying that it has to be sent among the first NAL units in the bitstream (see section xxx below). While multiple DPSs can be in the bitstream, the value of

the syntax elements therein cannot be inconsistent when being referenced.

Video parameter set

The Video Parameter Set (VPS) includes decoding dependency or information for reference picture set construction of enhancement layers. The VPS provides a "big picture" of a scalable sequence, 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 xxx).

Sequence parameter set

The Sequence Parameter Set (SPS) contains syntax elements pertaining to a coded video sequence (CVS), which is a group of pictures, starting with a random access point, and followed by pictures that may depend on each other and the random access point picture. In MPGEG-2, the equivalent of a CVS was a Group of Pictures (GOP), which normally started with an I frame and was followed by P and B frames. While more complex in its options of random access points, [VVC] retains this basic concept. In many TV-like applications, a CVS contains a few hundred milliseconds to a few seconds of video. In video conferencing (without switching MCUs involved), a CVS can be as long in duration as the whole session.

Picture and Adaptation parameter set

The Picture Parameter Set and the Adaptation Parameter Set (PPS and APS, respectively) carry information pertaining to a single picture. The PPS contains information that is likely to stay constant from picture to picture-at least for pictures for a certain type-whereas the APS contains information, such as adaptive loop filter coefficients, that are likely to change from picture to picture.

Profile, tier, and level

The profile, tier, and level syntax structure can be included in all DPS, VPS, and SPS. Somewhat oversimplified, they can be viewed to provide information about maximum bitstream complexity in the dimensions of tools used (profile), sample count (level), and maximum bitrate (tier). Level and tier are onion shaped, in that a decoder that can decode a certain level or tier can also decode lower levels or tiers. Profiles are not necessarily onion shaped and do not necessarily form a hierarchy. Therefore, the profile_tier_level structure in the video bitstream contains a bitmask which allows an encoder to mark a bitstream to be compatible with multiple profiles.

Sub-Profiles

Within the [VVC] specification, a sub-profile is simply a 32 bit number coded according to ITU-T Rec. T.35, that does not carry a semantic. It is carried in the profile_tier_level structure and hence (potentially) present in the DPS, VPS, and SPS. External registration bodies can register a T.35 codepoint with ITU-T registration authorities and associate with their registration a description of bitstream complexity restrictions beyond the profiles defined by ITU-T and ISO/IEC. This would allow encoder manufacturers to label the bitstreams generated by their encoder as complying with such sub-profile. It is expected that upstream standardization organizations (such as: DVB and ATSC), as well as large walled-garden video services will take advantage of this labelling system. In contrast to "normal" profiles, it is expected that sub-profiles may indicate encoder choices traditionally left open in the (decoder-centric) video coding specs, such as GOP structures, minimum/maximum QP values, and the mandatory use of certain tools or SEI messages.

Constraint Flags

The profile_tier_level structure optionally carries a considerable number of constraint flags, which an encoder can use to indicate to a decoder that it will not use a certain tool or technology. They were included in reaction to a perceived market need for labelling a bitstream as not exercising a certain tool that has become commercially unviable.

Temporal scalability support

Edt. note: this section may need adjustment as JVET work on bitstream extraction is in progress.

[VVC] includes 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.

Spatial, SNR, View Scalability

[VVC] includes support for spatial, SNR, and View scalability. Scalable video coding is widely considered to have technical benefits and enrich services for various video applications. Until recently, however, the functionality has not been included in the main profiles

of video codecs and not wide deployed due to additional costs. In VVC, however, all those forms of scalability are supported natively through the signaling of the layer_id in the NAL unit header, the VPS which associates layers with given layer_ids to each other, reference picture selection, reference picture resampling for spatial scalability, and a number of other mechanisms not relevant for this memo. Scalability support can be implemented in a single decoding "loop" and is widely considered a comparatively lightweight operation.

Spatial Scalability

With the existence of Reference Picture Resampling, likely in the "main" profile of VVC, the additional burden for scalability support is just a minor modification of the high-level syntax (HLS). In technical aspects, the inter-layer prediction is employed in a scalable system to improve the coding efficiency of the enhancement layers. In addition to the spatial and temporal motion-compensated predictions that are available in a single-layer codec, the inter-layer prediction in [VVC] uses the resampled video data of the reconstructed reference picture from a reference layer to predict the current enhancement layer. Then, the resampling process for inter-layer prediction is performed at the block-level, by modifying the existing interpolation process for motion compensation. It means that no additional resampling process is needed to support scalability.

SNR Scalability>

SNR scalability is similar to Spatial Scalability except that the resampling factors are 1:1--in other words, tehre is no change in resolution, but there is inter-layer prediction.

View Scalability>

Placeholder

SEI Messages

Supplementary Enhancement Information (SEI) messages are codepoints in the bitstream that do not influence the decoding process as specified in the [VVC] spec, but address issues of representation/ rendering of the decoded bitstream, label the bitstream for certain applications, among other, similar tasks. The overall concept of SEI messages and many of the messages themselves has been inherited from the H.264 and HEVC specs. In the [VVC] environment, some of the SEI messages considered to be generally useful also in other video coding technologies have been moved out of the main specification info a

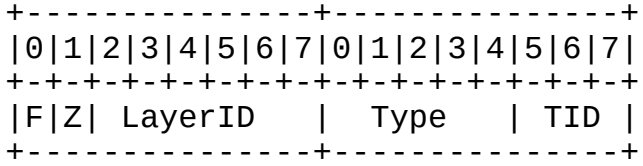
companion document (TO DO: add reference once ITU designation is known).

1.1.3. Parallel Processing Support (informative)

Compared to HEVC [RFC7798], the [VVC] design to support parallelization offers numerous improvements. Some of those improvements are still undergoing changes in JVET. Information, to the extent relevant for this memo, will be added in future versions of this memo as the standardization in JVET progresses and the technology stabilizes.

1.1.4. NAL Unit Header

[VVC] maintains the NAL unit concept of HEVC with modifications. VVC 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.



The Structure of the [VVC] NAL Unit Header.

Figure 1

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

F: 1 bit

forbidden_zero_bit. Required to be zero in [VVC]. Note that the inclusion of this bit in the NAL unit header was to enable transport of [VVC] 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 TBD.

Z: 1 bit

nuh_reserved_zero_bit. Required to be zero in [VVC], and reserved for future extensions by ITU-T and ISO/IEC. This memo does not

overload the "Z" bit for local extensions, as a) overloading the "F" bit is sufficient and b) to preserve the usefulness of this memo to possible future versions of [VVC].

LayerId: 6 bits

nuh_layer_id. Identifies the layer a NAL unit belongs to, wherein a layer may be, e.g., a spatial scalable layer, a quality scalable layer .

Type: 6 bits

nal_unit_type. This field specifies the NAL unit type as defined in Table 7-1 of [VVC]. For a reference of all currently defined NAL unit types and their semantics, please refer to Section 7.4.2.2 in [VVC].

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 [VVC] coded data over RTP [RFC3550]:

- Usage of RTP header with this payload format
- Packetization of [VVC] coded NAL units into RTP packets using three types of payload structures: a single NAL unit packet, aggregation packet, and fragment unit
- Transmission of HEVC NAL units of the same bitstream within a single RTP stream.
- Media type parameters to be used with the Session Description Protocol (SDP) [RFC4566]

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 [RFC2119]. In

this document, the above key words will convey that interpretation only when in ALL CAPS. Lowercase uses of these words are not to be interpreted as carrying the significance described in RFC 2119. 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 [VVC]. Section 3.1.1 lists relevant definitions from [VVC] for convenience. Section 3.1.2 provides definitions specific to this memo.

3.1.1. Definitions from the VVC Specification

Placeholder

3.1.2. Definitions Specific to This Memo

Placeholder

3.2. Abbreviations

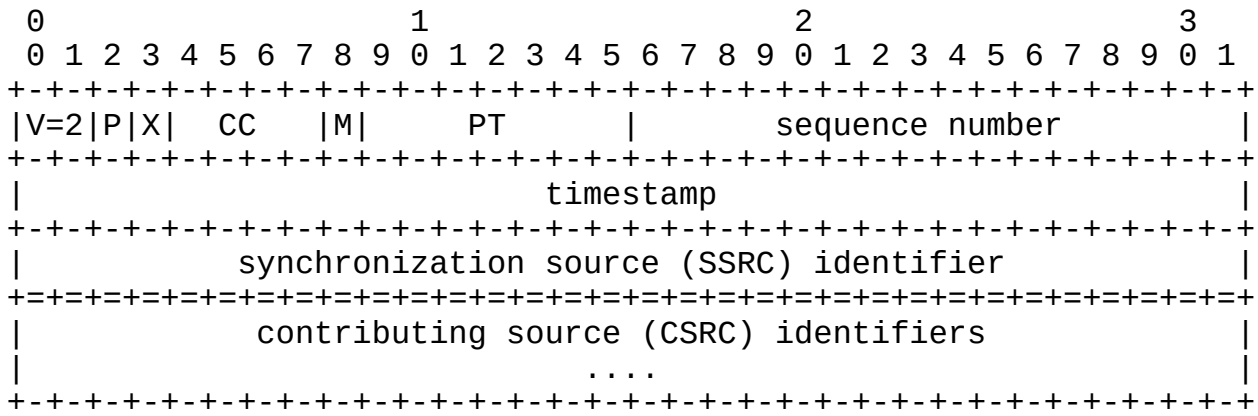
Placeholder

4. RTP Payload Format

4.1. RTP Header Usage

The format of the RTP header is specified in [RFC3550] (reprinted as 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.



RTP Header According to [RFC3550]

Figure 2

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.

The informative note below needs updating once the NAL unit type table is stable in the [VVC] spec

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 this information from the video encoder. If, however, the implementation cannot obtain this 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.

Sequence Number (SN): 16 bits

Set and used in accordance with [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 xxx of [VVC]) 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 [VVC]. 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.

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, Z, LayerId, Type, 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.

For Discussion: quite possibly something similar can be said for the Layer_id in layered coding, but perhaps not in multiview coding. (The relevant part of the spec is relatively new, therefore the soft language). However, for serious layer pruning, interpretation of the VPS is required. We can add language about the need for starteful interpretation of LayerID vis-a-vis stateless interpretation of TID later.

4.3. 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:

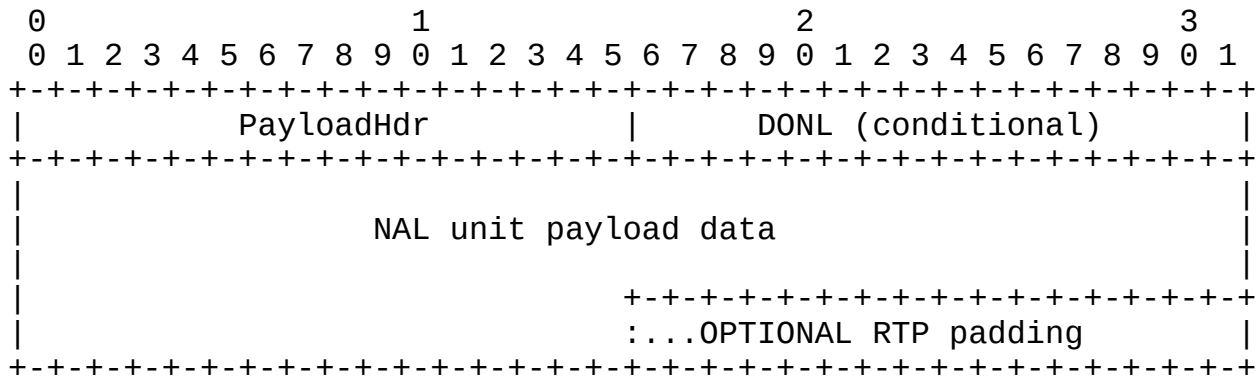
- Single NAL unit packet: Contains a single NAL unit in the payload, and the NAL unit header of the NAL unit also serves as the payload header. This payload structure is specified in Section 4.4.1.

- 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.

4.3.1. 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.



The Structure of a Single NAL Unit Packet

Figure 3

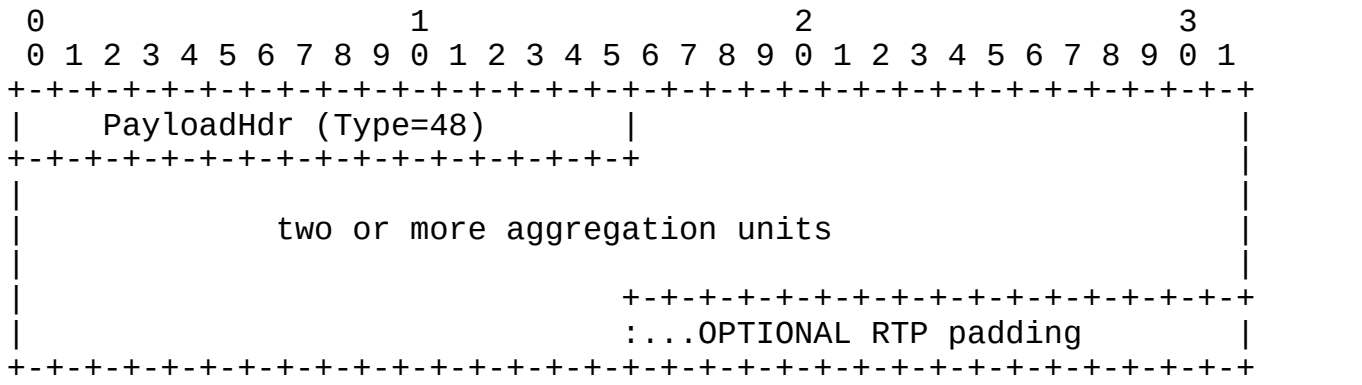
The DONL field, when present, specifies the value of the 16 least significant bits of the decoding order number of the contained NAL unit. If sprop-max-don-diff is greater than 0 for any of the RTP streams, the DONL field MUST be present, and the variable DON for the contained NAL unit is derived as 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.

4.3.2. 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.



The Structure of an Aggregation Packet

Figure 4

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.

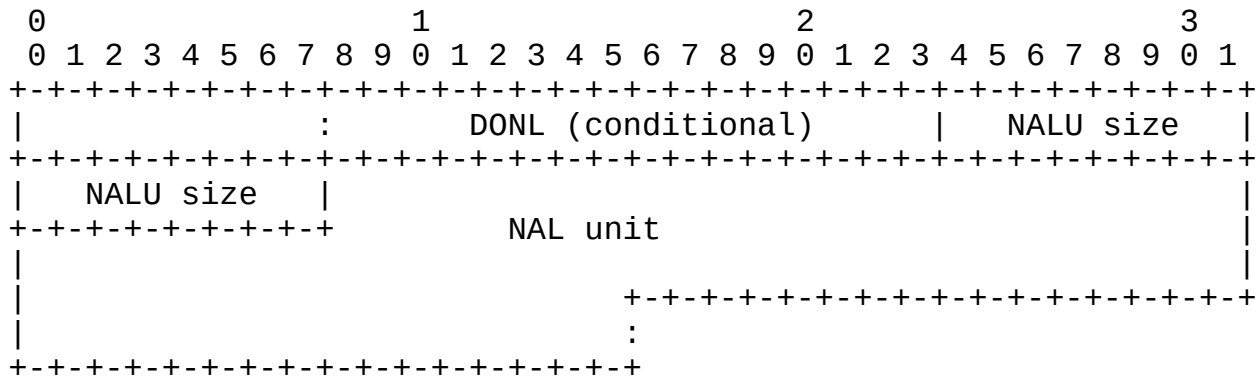
NOTE: double check #48 against post-geneva [VVC] spec

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 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.



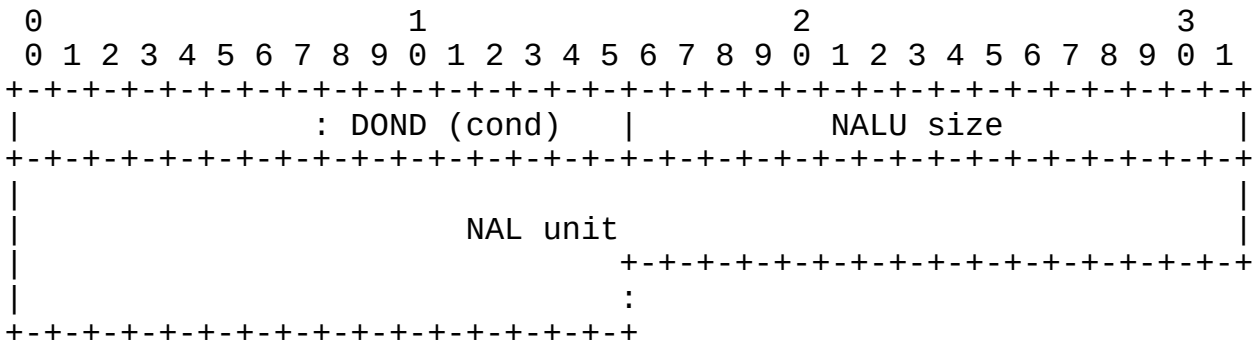
The Structure of the First Aggregation Unit in an AP

Figure 5

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.



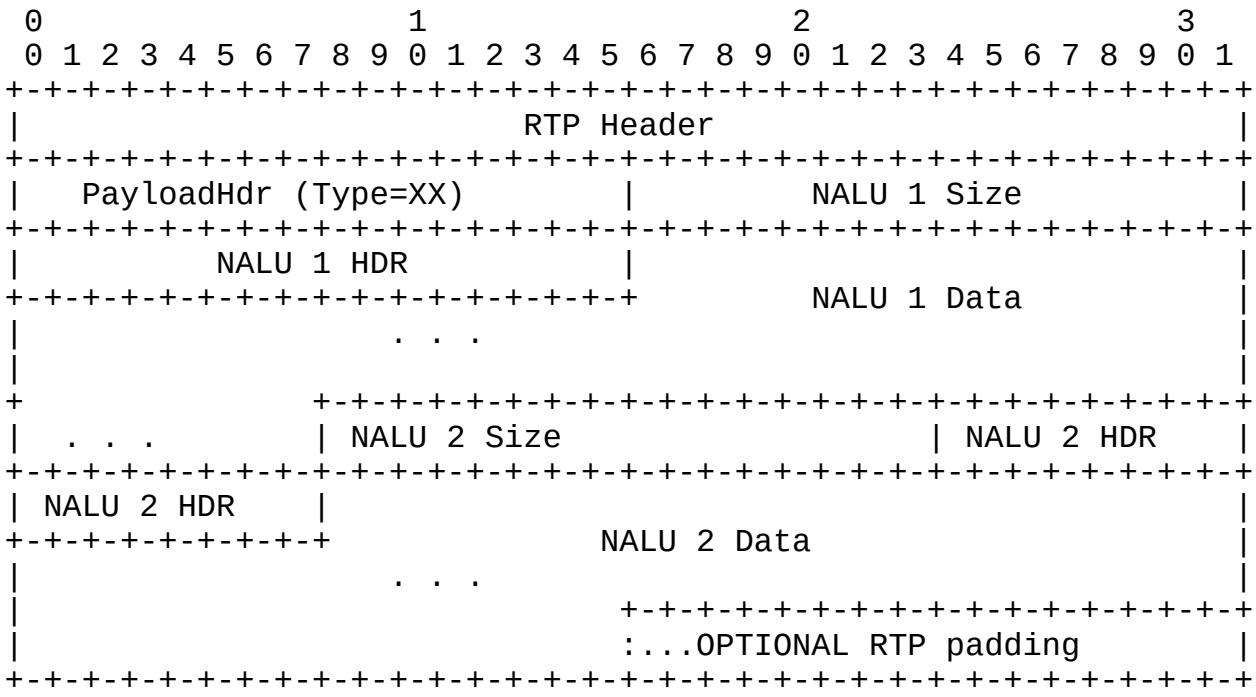
The Structure of an Aggregation Unit That Is Not the First Aggregation Unit in an AP

Figure 6

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.

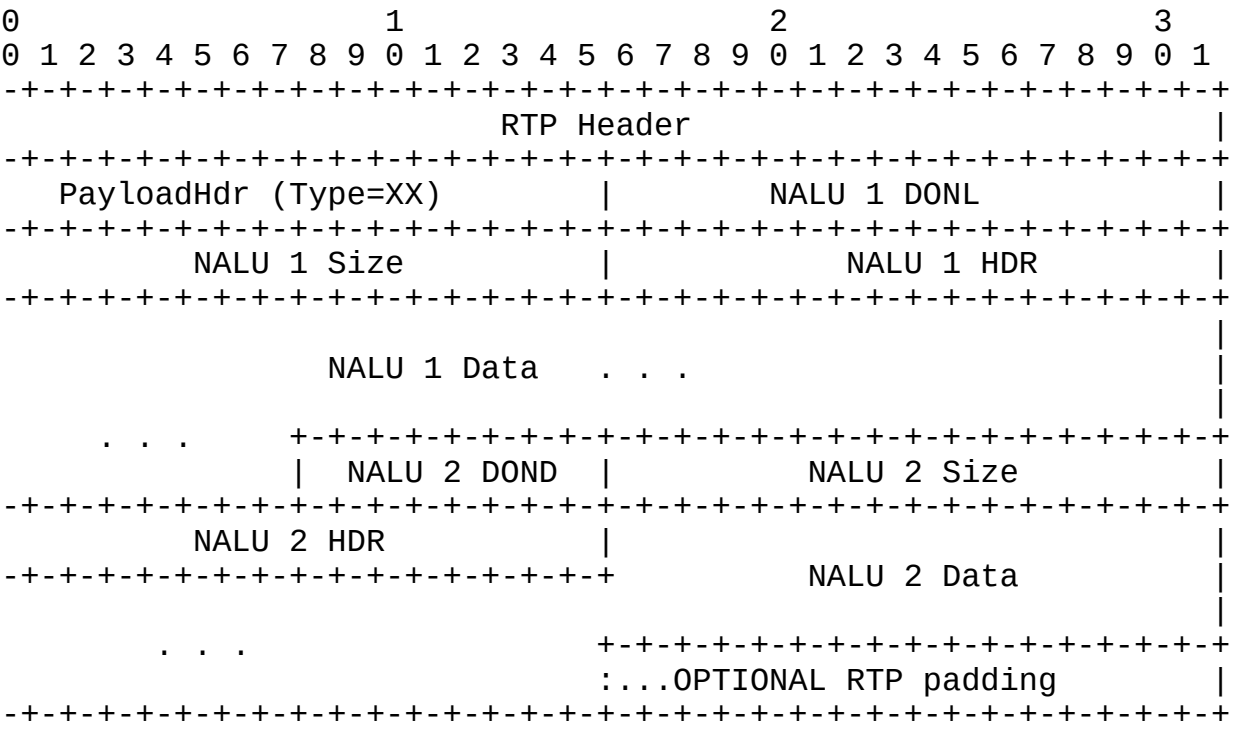
Figure 7 presents an example of an AP that contains two aggregation units, labeled as 1 and 2 in the figure, without the DONL and DOND fields being present.



An Example of an AP Packet Containing Two Aggregation Units without the DONL and DOND Fields

Figure 7

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.



An Example of an AP Containing Two Aggregation Units with the DONL and DOND Fields

Figure 8

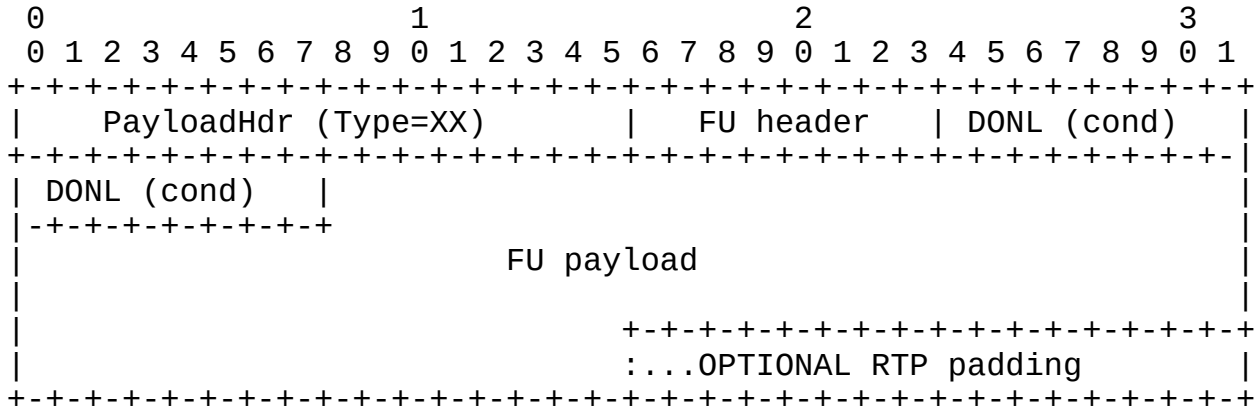
4.3.3. Fragmentation Units

Fragmentation Units (FUs) are introduced to enable fragmenting a single NAL unit into multiple RTP packets, possibly without cooperation or knowledge of the HEVC [RFC7798] 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.

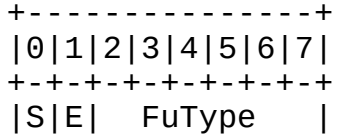


The Structure of an FU

Figure 9

The fields in the payload header are set as follows. The Type field MUST be equal to XX. 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.



The Structure of FU Header

Figure 10

The semantics of the FU header fields are as follows:

S: 1 bit

When set to 1, 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 0.

E: 1 bit

When set to 1, 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 0.

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 1 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 1 to indicate a syntax violation.

4.4. 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].
- 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:

- AbsDon[n] greater than AbsDon[m] indicates that NAL unit n follows NAL unit m in NAL unit decoding order.
- When AbsDon[n] is equal to AbsDon[m], the NAL unit decoding order of the two NAL units can be in either order.

- 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 Rulesumber

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 XX, inclusive, may be passed to the decoder. NAL-unit-like structures with NAL unit type values in the range of XX to XX, 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 hereafter called the de-packetization buffer in this section. Receivers should also prepare for transmission delay jitter; that is, 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

Placeholder

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 [RFC4585] , and the FIR message is defined in [RFC5104] .

8.1. Picture Loss Indication (PLI)

As specified in RFC 4585, Section 6.3.1, the reception of a PLI 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 a [VVC] 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)

For further study. Maybe remove as there are no known implementations of SDLI in H.265 based systems

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 the encoder side that a particular picture that has been decoded relatively earlier is correct and still present in the decoded picture buffer; it 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 [VVC] 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 [VVC], even a decoded picture buffer size of two picture storage buffers would work for the approach described in the previous paragraph.

the text below is copy-paste from RFC 7798. If we keep the RPSI message, it needs adaptation to the [VVC] syntax. Doing so shouldn’t be too hard as the [VVC] reference picture mechanism is not too different from the H.265 one.

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 [VVC] 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 [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 Framework: Why RTP Does Not Mandate a Single Media Security Solution" [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 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. [VVC] 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 HEVC [RFC7798], [VVC] includes a user data Supplemental 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. [VVC] leaves the handling of this SEI message to the receiving system. In order to avoid harmful side effects rganization the user data SEI message, decoder implementations cannot naively 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 mechanisms available in [VVC] are temporal scalability, and spatial/SNR 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) or higher spatio-SNR layers (as indicated by interpreting the VPS) until the sending bitrate drops to an acceptable range.

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 Considertaions

Placeholder

12. Acknowledgements

Large parts of this specification share text with the RTP payload format for HEVC [RFC7798], RFC 7798. We thank the authors of that specification for their excellent work. We also thank BD Choi for his contribution towards the [VVC] descriptive text.

13. References

13.1. Normative References

[ISO23090-3]
ISO and IEC, "Versatile video coding -- not yet published", August 2020.

[RFC2119] Bradner, S., "Key words for use in RFCs to Indicate Requirement Levels", BCP 14, RFC 2119, DOI 10.17487/RFC2119, March 1997, <https://www.rfc-editor.org/info/rfc2119>.

[RFC3550] Schulzrinne, H., Casner, S., Frederick, R., and V. Jacobson, "RTP: A Transport Protocol for Real-Time Applications", STD 64, RFC 3550, DOI 10.17487/RFC3550, July 2003, <https://www.rfc-editor.org/info/rfc3550>.

[RFC3551] Schulzrinne, H. and S. Casner, "RTP Profile for Audio and Video Conferences with Minimal Control", STD 65, RFC 3551, DOI 10.17487/RFC3551, July 2003, <https://www.rfc-editor.org/info/rfc3551>.

[RFC3711] Baugher, M., McGrew, D., Naslund, M., Carrara, E., and K. Norrman, "The Secure Real-time Transport Protocol (SRTP)", RFC 3711, DOI 10.17487/RFC3711, March 2004, <https://www.rfc-editor.org/info/rfc3711>.

[RFC4566] Handley, M., Jacobson, V., and C. Perkins, "SDP: Session Description Protocol", RFC 4566, DOI 10.17487/RFC4566, July 2006, <https://www.rfc-editor.org/info/rfc4566>.

[RFC4585] Ott, J., Wenger, S., Sato, N., Burmeister, C., and J. Rey, "Extended RTP Profile for Real-time Transport Control Protocol (RTCP)-Based Feedback (RTP/AVPF)", RFC 4585, DOI 10.17487/RFC4585, July 2006, <https://www.rfc-editor.org/info/rfc4585>.

[RFC5104] Wenger, S., Chandra, U., Westerlund, M., and B. Burman, "Codec Control Messages in the RTP Audio-Visual Profile with Feedback (AVPF)", RFC 5104, DOI 10.17487/RFC5104, February 2008, <https://www.rfc-editor.org/info/rfc5104>.

[RFC5124] Ott, J. and E. Carrara, "Extended Secure RTP Profile for Real-time Transport Control Protocol (RTCP)-Based Feedback (RTP/SAVPF)", RFC 5124, DOI 10.17487/RFC5124, February 2008, <https://www.rfc-editor.org/info/rfc5124>.

[VVC] ITU-T, "Versatile video coding - JVET-O2001-vE, available from http://phenix.it-sudparis.eu/jvet/doc_end_user/documents/15_Gothenburg/wg11/JVET-O2001-v14.zip", August 2019.

13.2. Informative References

[CABAC] Sole, J., Joshi, R., Nguyen, N., Ji, T., Karczewicz, M., Clare, G., Henry, F., and A. Duenas, "Transform coefficient coding in HEVC", IEEE Transactions on Circuts and Systems for Video Technology Vol. 22 No. 12 pp. 1765-1777, DOI 10.1109/TCSVT.2012.2223055, December 2012.

[Girod99] Girod, B. and F. Faerber, "Feedback-based error control for mobile video transmission", Proceedings of the IEEE Vol. 87, No. 10, pp. 1707-1723, DOI 10.1109/5.790632, October 1999.

[MPEG2S] IS0/IEC, "Information technology - Generic coding of moving pictures and associated audio information - Part 1: Systems", ISO International Standard 13818-1, 2013.

[RFC6051] Perkins, C. and T. Schierl, "Rapid Synchronisation of RTP Flows", RFC 6051, DOI 10.17487/RFC6051, November 2010, <https://www.rfc-editor.org/info/rfc6051>.

[RFC6184] Wang, Y., Even, R., Kristensen, T., and R. Jesup, "RTP Payload Format for H.264 Video", RFC 6184, DOI 10.17487/RFC6184, May 2011, <https://www.rfc-editor.org/info/rfc6184>.

[RFC6190] Wenger, S., Wang, Y., Schierl, T., and A. Eleftheriadis, "RTP Payload Format for Scalable Video Coding", RFC 6190, DOI 10.17487/RFC6190, May 2011, <https://www.rfc-editor.org/info/rfc6190>.

[RFC7201] Westerlund, M. and C. Perkins, "Options for Securing RTP Sessions", RFC 7201, DOI 10.17487/RFC7201, April 2014, <https://www.rfc-editor.org/info/rfc7201>.

[RFC7202] Perkins, C. and M. Westerlund, "Securing the RTP Framework: Why RTP Does Not Mandate a Single Media Security Solution", RFC 7202, DOI 10.17487/RFC7202, April 2014, <https://www.rfc-editor.org/info/rfc7202>.

[RFC7798] Wang, Y., Sanchez, Y., Schierl, T., Wenger, S., and M. Hannuksela, "RTP Payload Format for High Efficiency Video Coding (HEVC)", RFC 7798, DOI 10.17487/RFC7798, March 2016, <https://www.rfc-editor.org/info/rfc7798>.

Appendix A. Change History

draft-zhao-payload-rtp-vvc-00 ..... initial version

Authors' Addresses

Shuai Zhao
Tencent
2747 Park Blvd.
Palo Alto, CA 94306
US

Email: shuaiizhao@tencent.com

Stephan Wenger
Tencent
2747 Park Blvd.
Palo Alto, CA 94306
US

Email: stewe@stewe.org