3GPP TSG-SA WG4 Meeting #131-bis-eS4-250601r01

Online, 11 – 17 April 2025

**Source: Xiaomi**

**Title: Pseudo-CR on Video Decoder API and System Integration updates**

**Spec: 3GPP TS 26.265 v1.0.0**

**Agenda item: 9.5**

**Document for: Agreement**

**1. Introduction**

This contribution proposes updates to the Video Decoder API as well as the Systems functions.

The r01 versions implements the following changes:

* Reordering of changes by clause order of TR (despite lack of relationship between the changes in this order)
* Updating every changes against the draft integrated document [26265-102-rm.docx](https://www.3gpp.org/ftp/tsg_sa/WG4_CODEC/TSGS4_131-bis-e/Inbox/Drafts/Video/26265-102-rm.docx)
* Adding a brief motivation in each change in addition to the general motivation

**2. Reason for Change**

Addressing the case of multi-layer and multi-bitstream video representation, especially for stereo video.

The changes 1 to 4 and 5 to 6 can be addressed separately.

**3. Proposal**

It is proposed to agree the following changes to 3GPP TS 26.265 v1.0.0.

\* \* \* First Change \* \* \* \*

Motivations:

* Adding Access Unit definition which is used in 7.2.1.
* Creating the an intermediate concept between bitstream and CVS such this concept can accommodate layer in case of layer coding. The proposal is to reuse elementary stream as known from system integration, cf analysis below.
* Analysis of terminology:
	+ HEVC:
		- **Elementary stream** (only 3 occurrences in the HEVC spec):
			* A sequence of one or more **bitstreams:**
				+ A sequence of bits, in the form of a NAL unit stream or a byte stream, that forms the representation of coded pictures and associated data forming one or more **coded video sequences (CVSs):**

A sequence of **access units** that consists in a given order.

(general) A set of NAL units […] and contain exactly one coded picture with nuh\_layer\_id equal to 0.

(multiview) A set of NAL units [...] and contain at most one *coded picture* with any specific value of nuh\_layer\_id.

* + System (14496-1/ISOBMFF/DASH/CMAF)
		- **Elementary stream**
			* (part 1) consecutive flow of mono-media data from a single source entity to a single destination entity on the compression layer. More than one elementary stream may be connected to a single decoder (e.g., in a decoder of a scalable audio-visual object)
			* (NALU FF) sequence of one or more bitstreams of the applicable video standard
		- **access units**:
			* (part 1) smallest individually accessible portion of data within an elementary stream to which unique timing information can be attributed
			* (DASH) unit of a *media stream* (3.1.29) with an assigned Media Presentation time
			* (ISOBMFF) media data pertaining to a particular composition time in a media stream, usually carried in one sample of a media track.

# 3 Definitions of terms, symbols and abbreviations

## 3.1 Terms

For the purposes of the present document, the terms given in TR 21.905 [1] and the following apply. A term defined in the present document takes precedence over the definition of the same term, if any, in TR 21.905 [1].

**Access Unit:** Smallest individually accessible portion of data within an Elementary Stream to which unique timing information can be attributed.

**Bitstream:** One or more Elementary Streams formatted in the same sequence of bits.

**Coded Video Sequence:** A sequence of bits that consists of a series of coded frames and any associated metadata (required for decoder and rendering initialization) and conforms to a specific video encoding format and aligns with a certain Operation Point, as defined in this document. Such coded video sequence (CVS) has no decoding dependency on any other prior CVS and consists, in decoding order, of information specifying the characteristics or format of the encoded video data, a single intra random access coded frame followed by zero or more dependent, on the intra random access coded frame, coded frames, and a series of associated coded metadata.

Editor’s Note: Needs to be completed.

**Chroma:** a sample array or single sample representing one of the two colour difference signals related to the primary colours, represented by the symbols *Cb* and *Cr*.

**Elementary Stream:** One or more Coded Video Sequences that conform to a specific video encoding format and aligns with a certain Operation Point.

**Hero Eye**: The default eye in a stereo (stereoscopic) video pair, often determined by tags set by the cameras used to capture the video.

**Luma:** a sample array or single sample representing the monochrome signal related to the primary colours (denoted with the symbol *Y*),

**Operation Point:** A collection of discrete combinations of different video representation formats, including spatial and temporal resolutions, colour mapping, transfer functions, and the encoding format.

**Receiver:** A device capable of decoding and rendering any bitstream that is conforming to a certain Operation Point.

\* \* \* 2nd Change \* \* \* \*

Motivations:

* Clarifying that there can be multiple signal at the input of the encoder
* Clarifying that there can be multiple signals at the output of the decoder
* Creating a new “observation point” after the rendering before the display buffer. For this point, there would not be any conformance on the signal bits but definition of a hypothetical reference signal, e.g. in case of stereoscopic content, it would be expected to have there two signals 0left and right view) regardless of the type of signal out of the decoder, i.e. frame-packed or Multiview.

# 4 Context and Definitions

## 4.1 Motivation

Video codecs, encoders, and decoders are core components of 3GPP services. At the same time, video encoders and decoders, residing on 3GPP User Equipment (UE) and defined in 3GPP specifications, also provide interoperability points for third-party services. Video capabilities are predominantly independent of the service in use. This specification addresses the definition of video capabilities and operating points such that 3GPP service specifications as well as third-party service providers can refer to the interoperability points defined in this specification.

The present specification makes use some of the concepts recommended in TR 26.857 [2], i.e. the concept of Media Service Enablers.

## 4.2 Reference architectures and definitions

In order to define the normative aspects of this specification, reference architectures are defined. The core architecture is provided in Figure 4.2-1. The workflow addresses the generation of a *video bitstream* from one or more video signals using a *video encoder* as well as the decoding of a video bitstream by a *video decoder* and providing the resulting decoded video as well as associated metadata to a rendering and display process. The resulting video can be composed of one or more video signals. The number of video signals as input of the video encoder is the greater of equal than the number of video signals as output of the decoder. After rendering, the resulting video signals are sent to the display buffer for presentation. The video encoder as well as the video decoder may be configured to certain operations indicated by APIs in Figure 4.2-1. These APIs are not normatively specified but serve as an example reference to configure encoders and decoders as documented in Annex [A].



Figure 4.2-1 Reference architecture for video operating points and capabilities

Video encoders produce *Coded Video Sequences,* as defined in clause 3.1, referred to as *Bitstreams*.

An intra random access coded frame, together with the associated metadata, forms a Random Access Point (RAP) that permits to initialize decoding of the coded video sequence.

The decoder is provided with access units which correspond to pieces of the Bitstream that can be processed by the decoder to regenerate decoded video frames.

Figure 4.2-2 provides an overview of the data model and the definitions in this specification.



Editor’s Note: This figure is for illustrative purposes, informative and may be moved to an Annex.

Figure 4.2-2 Data model

In this case, configuration information is coded into metadata, that can be provided to the decoder in order to initialize the decoding of the CSVs included in the Bitstream.

A more system-centric architecture is provided in Figure 4.2-3. The workflow addresses the generation of a *transport stream* from a video signal using a *video encoder* and a *packager*. The package may include for example timing and metadata information. The de-packaging and decoding of the *transport stream* by a de-packager and a *video decoder*, respectively, allows for providing the resulting video signal as well as associated metadata to a rendering and display process. Again, the packager/encoder as well as the de-packager/decoder may be configured to certain operations indicated by APIs in Figure 4.2-2.



Figure 4.2-3 Reference architecture for system operating points and capabilities

Based on this introduction, the following terms are defined:

**- Operating Point:** A collection of different possible video formats including spatial and temporal resolutions, colour mapping, transfer functions, etc. and a video encoding format.

**- Bitstream**: A compressed media representation presented as a sequence of bits

- that forms the representation of any coded pictures and associated metadata data,

- this sequence of bits is formed by one or more CVSs and each CSV has identical metadata

- the sequence of bits conforms to a particular video coding specification/format and one or more Operating Points.

- comprised by access units that serve as units to be provided to decoders for regenerating frames.

**- Receiver**: A device that can ingest and decode any Bitstream that is conforming to a particular video coding specification and Operating Point, and optionally render it.

In addition, on system level the following terms are defined:

**- System Operating Point:** A collection of different possible video formats including spatial and temporal resolutions, colour mapping, transfer functions, etc., a video encoding and a packaging format.

**- Transport Stream:** A packaged media bitstream that conforms to a particular video coding and packaging specification/format and one or more Operating Points.

**- System Receiver:** A receiver that can de-package and decode any system bitstream that is conforming to a particular System Operating Point, and optionally render it.

NOTE: A reference architecture for multiple decoders is for further study.

System Operating Points are not defined in this specification but are left for mappings to specific delivery protocols such as RTP for MTSI, CMAF/DASH for 5G Media Streaming, or ISO BMFF for Messaging Services. However, this specification provides mapping principles to delivery protocols.

\* \* \* 3rd Change \* \* \* \*

Motivations:

* Creating at the concept of multi-signal representation for stereoscopic content since it is by nature two “signals” (from the offline it could also be renamed multi-component signal) and moving the “stereoscopic video” from table 4.4.2-1 to this new table 4.4.3-1/

## 4.4 Video representation formats

### 4.4.1 Overview

This clause defines video representation formats in the context of media delivery in 3GPP. For this purpose, a set of video signal parameters are defined in clause 4.4.2, with the restriction on what is defined in 3GPP media delivery. Based on the defined video signal parameters, clause 4.4.3 defines a set of video representation formats.

NOTE: These clause does not specify whether these parameters and formats are required, recommended or suggested to be supported. This aspect is left to specific service specifications or external specifications to refer to the parameters and formats defined in this clause.

### 4.4.2 Video signal parameters

Video signals considered in this specification are represented by a sequence of pictures, where a *picture* can represent either an array of *luma* samples in a monochrome format or an array of luma samples and two corresponding arrays of *chroma* samples in a 4:2:0, 4:2:2, or 4:4:4 colour format. Only *progressive* signals are considered. A component refers to an array or single sample from one of the three arrays (luma and two chroma) that compose a picture. The Luma component represents a sample array or single sample representing the monochrome signal related to the primary colours (denoted with the symbol *Y*), and a chroma component represents a sample array or single sample representing one of the two colour difference signals related to the primary colours, represented by the symbols *Cb* and *Cr*.

Video signals are typically described by a set of parameters that are required for the proper rendering of the decoded signal. Table 4.4.2-1 documents typical video signal parameters and provides a definition and/or reference.

Table 4.4.2-1 Video Signal Parameters

|  |  |  |  |
| --- | --- | --- | --- |
| Parameter | Definition | 3GPP restrictions | Service or Application restrictions |
| Spatial Resolution width | The number of active samples per line for the luma component.Example values are 1280 or 1920 for HD, and 3840 for UHD.NOTE: The width does not restrict the encoding resolution to fixed values. Cropping parameters can be indicated that prescribe decoders the need to remove spatial video samples in a partially filled coding block that are not intended for presentation. | No restrictions | Restrictions possible |
| Spatial Resolution height | The number of active lines per picture for the luma component.Example values are 720 or 1080 for HD, and 2160 for UHD.NOTE: The height does not restrict the encoding resolution to fixed values. Cropping parameters can be indicated that prescribe decoders the need to remove spatial video samples in a partially filled coding block that are not intended for presentation. | No restrictions | Restrictions possible |
| Scan Type | Indicates the source scan type of the pictures as defined in clause 7.3 of Rec. ITU-T H.273.Typical value is progressive | Progressive only |  |
| Chroma format indicator | Indicates whether the picture has only a luma component or that the picture has three colour components that consist of a luma component and two associated chroma components, such that the width and height of each chroma component are the width and height of the luma component divided by a factor defined by the chroma format as defined in Rec. ITU-T H.274, clause 7.3.  | 4:2:0 |  |
| Bit depth | Indicates the bit depth for the samples of the luma component and the samples of the two associated chroma components.Note that in general, the bit depth of the luma component and of the two associated chroma components may differ.Typical values are 8 or 10 bits. | 8 or 10 bitsLuma and chroma components shall not differ |  |
| Colour primaries  | Indicates the chromaticity coordinates of the source colour primaries as specified in clause 8.1 of Rec. ITU-T H.273.Typical values are 1 to refer to Rec. ITU-R BT.709-6 [bt709] or 9 to refer to Rec. ITU-R BT.2020-2 and Rec. ITU-R BT.2100-2.  | BT.709 or BT.2020/BT.2100 |  |
| Transfer Characteristics | Either indicates the reference opto-electronic transfer characteristic function of the source picture as a function of a source input linear optical intensity input or indicates the inverse of the reference electro-optical transfer characteristic function as a function of an output linear optical intensity as defined in clause 8.2 of Rec. ITU-T H.273.Typical values are 1 to refer to Rec. ITU-R BT.709-6, 14 to refer to Rec. ITU-R BT.2020-2 (10 bit), 16 to refer to the Rec. ITU-R BT.2100-2 perceptual quantization (PQ) system, or 18 to refer to the Rec. ITU-R BT.2100-2 hybrid log-gamma (HLG) system | BT.709, BT.2020 SDR, BT.2100 PQ, or BT.2100 HLG |  |
| Matrix Coefficients | Describes the matrix coefficients used in deriving the luma and chroma signals from the green, blue, and red primaries. A video full range flag may be supplied with this parameter specifying the scaling and offset values applied in association with the Matrix coefficients. For detailed definition refer to clause 8.2 of Rec. ITU-T H.273.Typical values are 1 to refer to the non constant luminance YCbCr representation in Rec. ITU-R BT.709-6 or 9 to refer to the non constant luminance YCbCr representations in Rec. ITU-R BT.2020-2 and Rec. ITU-R BT.2100-2. | YCbCr BT.709, YCbCr BT.2020, orYCbCr BT.2100 |  |
| Frame rate | Typical values, using frames per second, are: 120, 120/1.001, 100, 60, 60/1.001, 50, 30, 30/1.001, 25, 24, 24/1.001 | No restrictions | services may only permit a restricted subset |
| Frame packing | Indicates a frame packing arrangement, if present, as defined in clause 8.4 of Rec. ITU-T H.273. | Typically restricted to no frame packing. | Some applications may use frame packing. |
| Projection | Indicates a projection, if present, as defined in Rec. ITU-T H.274, clause 7.3, and typically refers to packing arrangements in clause 8.6 of Rec. ITU-T H.274. | Typically restricted to no projection. | Some applications may use projections. |
| Sample aspect ratio | Indicates width-to-height aspect ratio of the luma samples of the associated pictures as defined in clause 7.3 of Rec. ITU-T H.273.Typical value is 1 | No specific restrictions, but 1 is expected. |  |
| Chroma sample location type | Specifies the location of the chroma samples relative to the luma samples for frames as defined in Rec. ITU-T H.273, clause 8.7.Typical values are 0 (chroma samples are horizontally co-sited with and vertically centered between the first luma sample at the top-left corner and the first two luma samples at the top-left corner, respectively) or 2 (chroma samples are co-sited with the luma sample at the top-left corner). Note that a value of 1 is common for still images. | No specific restrictions, but 0 is expected if not present. For HDR the value is typically set to 2. |  |
| Range | Specifies how luma and chroma samples are represented in digital video as defined in Rec. ITU‑T H.273, clause 8.3 using the parameter VideoFullRangeFlag. For video applications only the value set to 0 is used, i.e. the video range or restricted range is applied where the luma values range from 16 to 235 in an 8-bit system, and chroma values range from 16 to 240. For 10-bit systems, the values are multiplied by 4. Note that for still images full range (value set to 1) is commonly used. | No specific restrictions, but 0 is expected if not present. |  |
|  |  |  |  |

### 4.4.3 Multi-signal video representation types

Some video experiences are concurrently displaying more than one video signals. In this case, the video representation format describes each video signal individually with the parameters defined in Table 4.4.2-1. Additionally, the video signals in the same video representation are typically further constrained for the proper rendering of the video representation.

The video signals to be displayed can be delivered in either of the following forms:

* As a single video using frame packing as defined in Table 4.4.2-1.
* As multiple video signals.

Table 4.4.3-1 documents multi-signal video representation types.

Table 4.4.3-1 Multi-signal Video Representation Types

|  |  |  |  |
| --- | --- | --- | --- |
| Parameter | Definition | 3GPP restrictions | Service or Application restrictions |
| Stereoscopic Video | Visual media may be stereoscopic, in which case a view is available to be presented to the left eye and another view is available to be presented simultaneously to the right eye. The presentation of both the left and right views allows for an effect known as stereopsis, which can be defined as "the perception of depth produced by the reception in the brain of visual stimuli from both eyes in combination; binocular vision." For signal representations, [3dtv] recommends that the Left and Right eyes comply to regular image formats such as Rec. ITU-R BT.709 and any necessary 3D-specific metadata is incorporated with the data. Hence, for stereoscopic video, two synchronized video signals are available, each with identical format parameters (such as the ones defined in this table). NOTE: When distributing the signal, some systems may use different resolutions for one of the views.Additional metadata that may be added with stereoscopic video:- “Hero eye” is the default eye in a stereo (stereoscopic) video pair, often determined by tags set by the cameras used to capture the video. If so signaled, this indicates that the other stereo eye view is derived from the specified stereo eye and may be useful when choosing which eye to use in a monoscopic viewing environment. There is no requirement that either of the two eyes (or views) is tagged as the hero eye, in which case no hero eye tagging may be present.  |  |  |

\* \* \* 4th Change \* \* \* \*

Motivations:

* Text improvement for codecs
* Decoupling video decoder API and player API as done in CTA-5003, even if we don’t keep this reference

## 4.6 Reference API parameters

### 4.6.1 Introduction

When media is played back, the decoder and the playback pipeline need to be initialized. For this purpose, certain parameters are required. In CTA-5003 [DPC], a media playback model is described that is aligned with HTML 5.1 and the <video> element, as well as the Media Source Extensions.

### 4.6.2 Video Decoder API Parameters

Video decoders are typically accessed by API parameters. The parameters are used for the following purposes:

- to identify the capability of the device in order to check whether the signal can be played back

- to initialize the decoding and playback platform to allocate the resources for decoding and rendering

Table 4.6.2-1 provide relevant parameters for Video Decoder APIs.

Table 4.6.2-1 Video Decoder API Parameters

|  |  |  |
| --- | --- | --- |
| Parameter | Restrictions | Status |
| media type | Specifies the media type of the component, in this case video. | required |
| codecs | Specifies through a well-defined string the codec parameters which the encoded video signal is compliant to. | required |
| video format parameters | Specifies additional video format parameters as defined in Table 4.4.2.1 to describe the signal and to initialize the encoder. | optional |

Editor’s Note: The capability of such API for decoding and playback of multilayer content, e.g. for stereoscopic content needs to be documented.

### 4.6.3 Video Encoder API Parameters

Video encoder API parameters are for further study.

### 4.6.4 Player API Parameters

Media players are typically configurable via API parameter. The main purpose of the API are:

* For video components, to create one or more display windows to display the decoded video signal
* To bind a media source, possibly remote, to the one or more created display windows.

Table 4.6.2-2 Display Window Object Parameters

|  |  |  |
| --- | --- | --- |
| Parameter | Restrictions | Status |
| width | Specifies the width of a video player window, in pixels | required |
| height | Specifies the width of a video player window, in pixels. | required |
| video format parameters | Specifies additional video format parameters as defined in Table 4.4.2-1 to describe the signal. | optional |

\* \* \* 5th Change \* \* \* \*

Motivations:

* Most of original 601 is integrated in [26265-102-rm.docx](https://www.3gpp.org/ftp/tsg_sa/WG4_CODEC/TSGS4_131-bis-e/Inbox/Drafts/Video/26265-102-rm.docx).
* Text improvement for RAP to change from bit position to byte position since RAP in bitstream are typically at byte position.

7.2 Functional Definitions

### 7.2.1 General

##### 7.2.1.1 Summary

This clause defines functional definitions for system integration in Table 7.2.1.1-1. The remainder of this

Table 7.2.1.1-1 Functional Definitions

|  |  |  |
| --- | --- | --- |
| Term | Summary | Details |
| Codec String | A single value identifying the codec indicated to render the content in the Bitstream as defined in IETF RFC 6381. | 7.2.1.2 |
| Decoder Configuration | a data structure storing essential parameters needed for decoding and rendering a video stream. | 7.2.1.3 |
| Random Access Point | A byte position in the Bitstream, for which in combination with the Decoder Configuration, the Bitstream can be randomly accessed, i.e. in decoding order the Bitstream carries sufficient information to access the media in the stream. | 7.2.1.4 |
| Access Unit (AU) | See Clause 3.1 |  |
| Coded access unit (CAU) | bits corresponding to an Access Unit | 7.2.1.5 |
| Random Access CAU | A CAU that starts with a random access point | 7.2.1.6 |

\* \* \* End of Changes \* \* \* \*