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

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.

**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 \* \* \* \*

## 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

Based on CTA-5003 [DPC], Table 4.6.2-1 provide relevant parameters that need to be attached to the content, in order to establish media playback properly, and serve as an API. 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 Video Decoder API Parameters

|  |  |  |
| --- | --- | --- |
| Parameter | Restrictions | Status |
| width | Specifies the width of the display window of the video player, in pixels. | required |
| height | Specifies the height of the display window of the video player , in pixels. | required |
| media type | Specifies the media type of the component, in this case video. | required |
| media format | Specifies the format of the media component, for example mp4. | required |
| media profiles | Specifies the profile of the media format, for example 'cmfc'. | optional |
| 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 video signal and to initialize the decoder. | optional |

NOTE: The width and height parameters either match the aspect ratio of the video or are set to a width and height values which correspond to a full-screen display window.

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

## 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

Based on CTA-5003 [DPC], Table 4.6.2-1 provide relevant parameters that need to be attached to the content, in order to establish media playback properly, and serve as an API. 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 Video Decoder API Parameters

|  |  |  |
| --- | --- | --- |
| Parameter | Restrictions | Status |
| display windows | Specifies the parameters of the display windows of the video player. | required |
| media type | Specifies the media type of the component, in this case video | required |
| format | Specifies the format of the media, for example mp4 | required |
| profiles | Specifies the profile of the format, for example 'cmfc' | optional |
| codecs | Specifies through a well-defined string the codec used for the signal | required |

Table 4.6.2-2 Display Window Object Parameters

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

NOTE: The Media API in [DPC] assumes a single display window attached to the video player.

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

## 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 bits  Luma 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, or YCbCr 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. |  |
| Frame packing | Specifies whether the video signal is a composite video signal made of sub video signal. Additional metadata describes the spatial arrangement of each sub video signal within the main video signal such that a player can extract the different sub video signal after decoding the pictures of the main video signal. Video representations using multiple video signals are defined in Clause 4.4.3 |  |  |

### 4.4.3 Multi-signal video representation types

Some video experiences are concurrently displaying more than one video signal. In this case, the video representation format describes each video signal 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 4.4.2-1 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 \* \* \* \*

## 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].

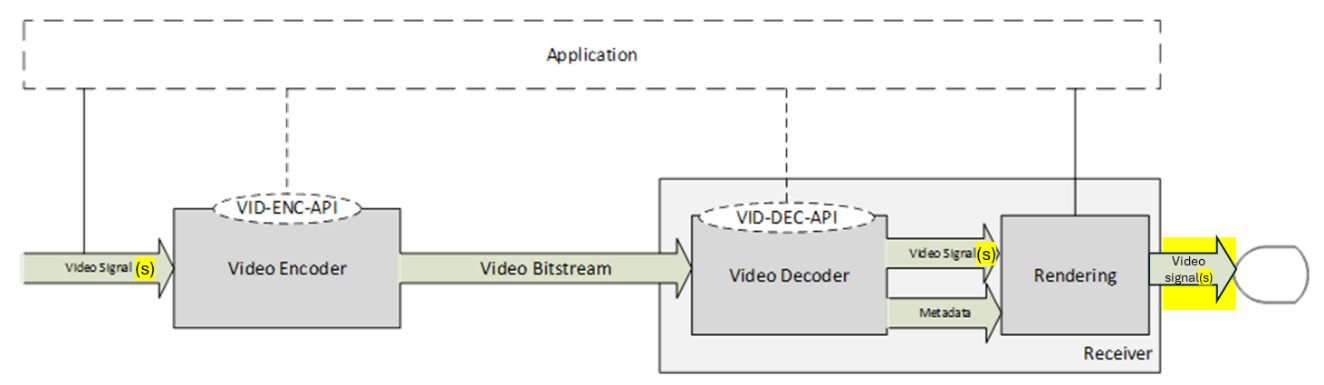


Figure 4.2-1 Reference architecture for video operating points and capabilities

A more system-centric architecture is provided in Figure 4.2-2. 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 4.2-2 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 conforms to a particular video coding specification/format and one or more Operating Points.

**- 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.

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

# 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 conforms to a specific video encoding format and a single Representation format.

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*.

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

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

**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.

\* \* \* 6th Change \* \* \* \*

### 7.2.1 General

This clause defines functional definitions for system integration in Table 7.2.1-1.

Table 7.2.1-1 Functional Definitions

|  |  |
| --- | --- |
| Term | Definition |
| **Codec String** | A single value identifying the codec parameter which the Bitstream complies to as defined in IETF RFC 6381. |
| **Decoder Configuration** | A data structure containing parameters for initialising the decoder. |
| **Random Access Point** | A byte position in the Bitstream from which, in combination with the Decoder Configuration, the Bitstream can be decoded, i.e. the Bitstream carries sufficient information from this byte position to decode the coded video data. |
| **Access Unit** | See Clause 3.1 |
| **Random Access Unit** | An access unit that starts with a random access point |

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