The high-level syntax (HLS) designs in VVC

Ye-Kui Wang
Principal Scientist, Bytedance

September 2020

Outline

- Versatile Video Coding (VVC) – the freshly new video coding standard
- System and transport interfaces of video codecs
  - *System diagram and protocol stack*
  - *What is high-level syntax (HLS)*
  - *Why HLS*
- An introduction to VVC HLS features
  - *VVC bitstream structure and NAL unit types*
  - *Random access support*
  - *Parameter sets, picture header, slice header*
  - *POC and reference picture management*
  - *Tiles, WPP, slices, subpictures*
  - *Reference picture resampling (RPR)*
  - *Scalability*
  - *Profile, tier, level (PTL)*
  - *Hypothetical reference decoder (HRD)*
  - *DCI, VUI, SEI*

Versatile Video Coding (VVC) – the freshly new video coding standard

字节跳动 ByteDance

Versatile Video Coding (VVC) – the freshly new video coding standard

- VVC is the new video coding standard finalized by the JVET of ITU-T and ISO/IEC in July 2020.
- Technically identical twin text: ITU-T H.266 | ISO/IEC 23090-3
  - *In the ITU-T publication process and the ISO/IEC approval process*
  - *Latest text in JVET-S2001 (http://phenix.int-evry.fr/jvet/doc_end_user/current_document.php?id=10399)*
- Versatile SEI messages for coded video bitstreams
  - *Twin text: ITU-T H.274 | ISO/IEC 23002-7*
  - *Independent SEI messages and VUI, specification not needed for core decoding process, could be used with VVC or other video standards*
  - *Latest text in JVET-S2007 (http://phenix.int-evry.fr/jvet/doc_end_user/current_document.php?id=10407)*

Performance of VVC (PSNR)

VTM9 compared to HEVC-HM, "common test conditions" (CTC)
Random Access is most important in storage, streaming, broadcast

	Random Access				
	Over HM-16.20				
	Y	U	V	EncT	DecT
Class A1	−38.74%	−37.19%	−44.34%	884%	186%
Class A2	−43.13%	−39.74%	−38.35%	999%	199%
Class B	−34.74%	−46.77%	−44.61%	935%	189%
Class C	−29.90%	−30.58%	−32.56%	1212%	199%
Class E					
Overall	−35.93%	−39.13%	−40.09%	1004%	193%
Class D	−27.64%	−26.48%	−26.11%	1326%	194%
Class F	−41.55%	−44.78%	−46.09%	689%	163%

- UHD average >40% (PSNR) – both luma and chroma
- Reasonable complexity tradeoff

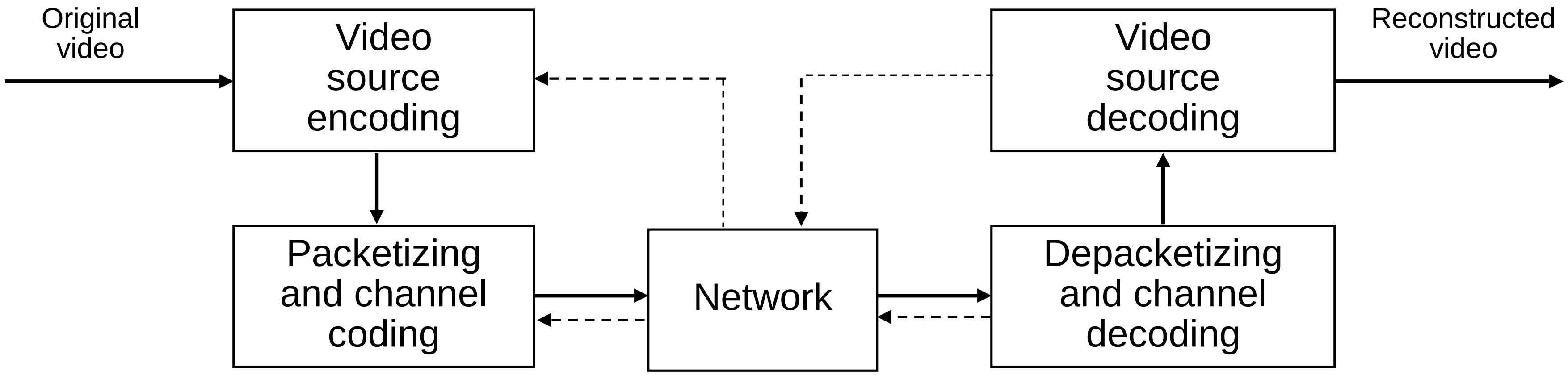
Versatility of VVC

- Set of core coding tools provides excellent performance for various types of video:
  - *Camera captured and computer generated content*
  - *Standard and high dynamic range*
  - *Various colour formats, including 4:4:4 and wide gamut*
  - *360° video, multiview video (including depth maps), MPEG's point cloud compression*
  - *Lossless coding support*
- Flexible high-level syntax designs for additional functionalities:
  - *Flexible access mechanisms, including localized access per subpictures*
  - *Extraction and merging at bitstream level*
  - *Layers and sublayers, including low-complex scalability modes (spatial scalability per reference picture resampling)*

Slide is courtesy of Jens—Rainer Ohm of RWTH and Gary Sullivan of Microsoft

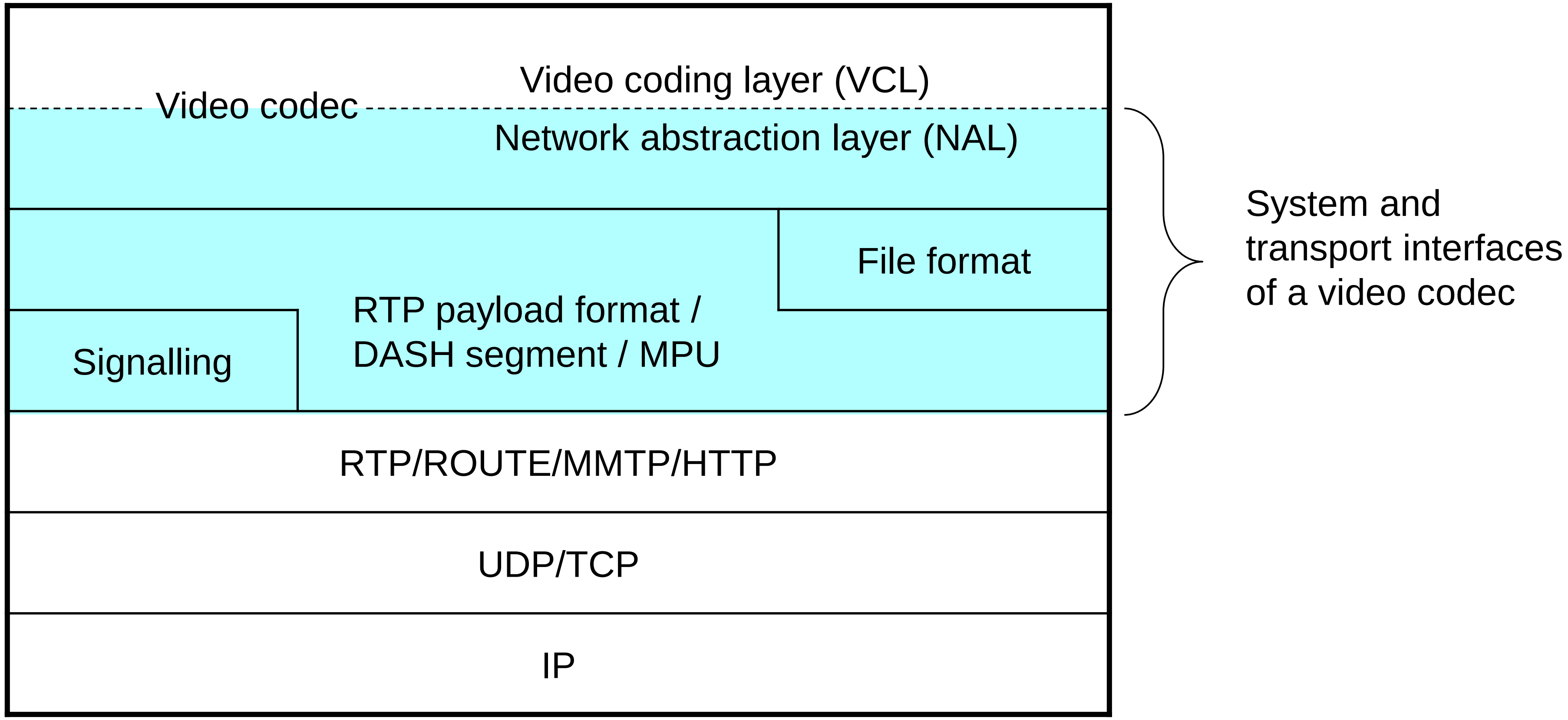
System and transport interfaces of video codecs

Diagram of typical video communication systems



- The video bitstream may also come from a file stored according to a file format standard.
- Encoding, packetization and channel coding should be optimized according to the network conditions and receiver preferences and capabilities.

Protocol stack of typical video communication and application systems



What is high-level syntax (HLS)?

- The high-level syntax (HLS) is an integral part of a video codec.
  - *E.g., roughly half of all the 300 pages of the HEVCv1 spec are on HLS topics*
- HLS is also referred to as the Network Abstraction Layer (NAL).
- As the name of NAL implies, the key purpose of the HLS design is
  - *to provide a (generic) interface of a video codec to (various) networks/systems (e.g. DASH, video conferencing, TV broadcast) where the video codec is used.*

- HLS topics include
  - *Bitstream structure and coded data units structures*
  - *Sequence and picture level parameters signalling*
  - *Random access and stream adaptation*
  - *Decoded picture management (incl. reference picture management)*
  - *Profile and level specification and signalling*
  - *Buffering model*
  - *Parallel processing*
  - *Temporal scalability*
  - *Byte stream format*
  - *Extensibility and backward compatibility*
  - *Error resilience*
  - *Signalling of supplemental information*

From the spec or the JVET points of view

- HLS consists of all syntaxes down to the slice header, inclusive
  - *NAL unit header (NUH)*
  - *Parameter sets (VPS, SPS, PPS, APS)*
    - Video usability information (VUI)
  - *Picture header (PH)*
  - *Slice header (SH)*
  - *Decoding capability information (DCI, used to be DPS)*
  - *Supplemental enhancement information (SEI)*
- CTU-level and lower-level syntaxes are not HLS

- Each HLS functionality involves syntaxes of different levels, and different HLS functionalities often have overlaps, thus forming a multi-dimension matrix, e.g.,
  - *Decoded picture management related syntaxes are present in NUH, VPS, SPS, PPS, PH, and SH*
  - *There are syntaxes in almost all levels of HLS working together for providing random access functionalities*
  - *There are also syntaxes in almost all levels of HLS for HRD (the buffering model)*

Why HLS?

- HLS provides friendliness of a video codec to video application systems
  - *The NAL unit header design and the byte stream format design enable simple and efficient use of VVC in both RTP based transmissions (e.g. in video conferencing) and MPEG-2 TS based transmission (e.g. in TV broadcasting).*
  - *The video parameter set and sequence parameter set provide a "big picture" of what the bitstream contains and how it can be used, thus application systems can use the information for session negotiation etc.*

- HLS provides flexible random accessing and stream adaptation capabilities while keeping high coding efficiency
  - *The design of HEVC/VVC IRAP picture types allows efficient coding and signalling of random access points, such that application systems using HEVC/VVC can easily perform random access (e.g. seeking in streaming, joining a TV program) and stream adaptation (e.g. in DASH).*
    - E.g., using CRA pictures for random access would provide about 6% coding gain compared to IDR pictures.
    - Although the same coding structure as CRA pictures is possible in AVC, the AVC signalling of CRA is more difficult to access by the systems layer.

Why HLS?

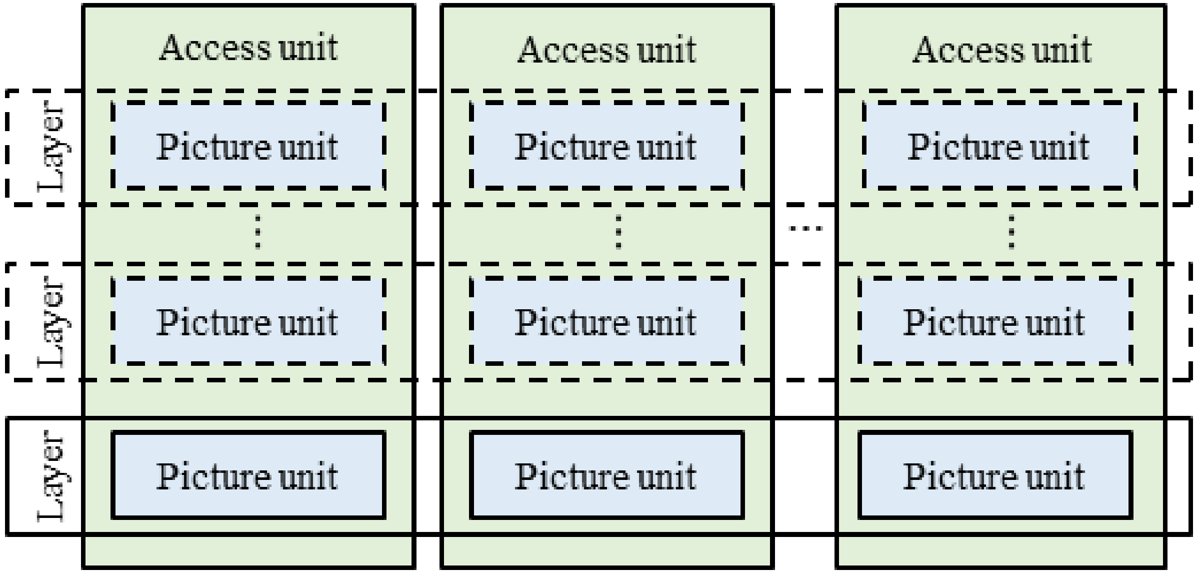
- HLS provides error resilience while keeping high coding efficiency, e.g.:
  - *The design of VVC RPL provides efficient yet error-robust management of reference pictures, at the same time enabling minimizing the required decoded picture buffer.*
  - *The design of VVC parameter sets allows an efficient yet error-robust way of transmitting video header information in an out-of-band manner.*
  - *Scalability and temporal scalability allows for UEP such that the base sub-layer can be better protected, which is widely used in video conferencing.*

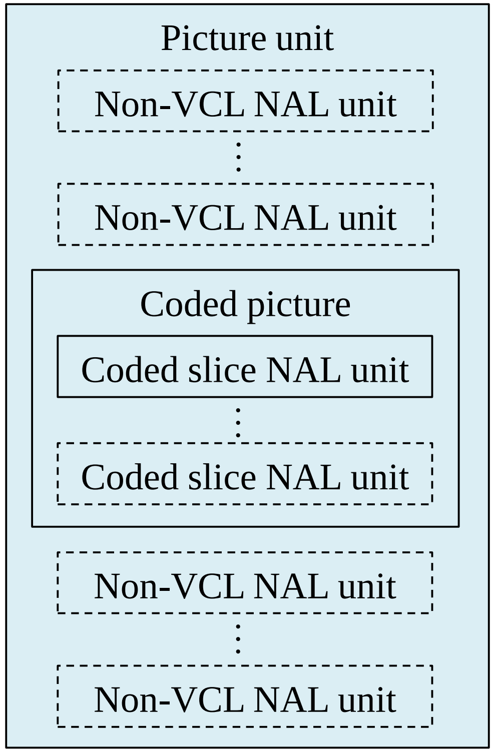
- HLS ensures interoperability
  - *Through defining and signalling of profile, tier, level, and buffering parameters*
  - *A decoder can have serious problems, incl. crashing, if the bitstream violates some of the conformance requirements*

- HLS provides extensibility and backward compatibility
  - *Extensibility designs in the first version of a spec enable extending the codec in a backward compatible manner*
  - *Backward compatibility in a new version of a spec enables legacy decoders to appropriately handle bitstreams generated by encoders implemented per the new version*

An introduction to VVC HLS features

VVC bitstream structure





- Dashed boxes indicate optionally present structures.
- Note that this is just an example of a typical structure. There are many flexibilities not shown.
- VCL NAL units are slices. Non-VCL NAL units contain other data than slices.

Figures are courtesy of Miska M. Hannuksela of Nokia

VVC NAL unit structure

nal_unit(NumBytesInNalUnit) {	Descriptor
nal_unit_header( )	
NumBytesInRbsp = 0	
for(i = 2; i < NumBytesInNalUnit; i++)	
if(i + 2 < NumBytesInNalUnit && next_bits(24) = = 0x000003) {	
rbsp_byte[NumBytesInRbsp++]	b(8)
rbsp_byte[NumBytesInRbsp++]	b(8)
i += 2	
emulation_prevention_three_byte /* equal to 0x03 */	f(8)
} else	
rbsp_byte[NumBytesInRbsp++]	b(8)
}	

nal_unit_header( ) {	Descriptor
forbidden_zero_bit	f(1)
nuh_reserved_zero_bit	u(1)
nuh_layer_id	u(6)
nal_unit_type	u(5)
nuh_temporal_id_plus1	u(3)
}	

- The start code emulation prevention mechanism is only NEEDED for the byte stream format (Annex B).
- The byte stream format is only used for the MPEG-2 Transport Stream (MPEG-2 TS) environment, which is used in conventional TV broadcast systems (still widely used today), not for other application systems like Internet video streaming (Netflix, Xigua Video, etc.) or conservation applications (Zoom, FaceTime, etc.)
- However, to avoid different video bitstream formats, the start code emulation presentation mechanism is always used. This has been the case for H.264/AVC, H.265/HEVC, and still for H.266/VVC. Maybe since the next generation video coding standard, we can get rid of burden, at least for most applications like Internet video streaming.

nal_unit_type	Name of nal_unit_type	Content of NAL unit and RBSP syntax structure	NAL unit type class
0	TRAIL_NUT	Coded slice of a trailing picture or subpicture* slice_layer_rbsp( )	VCL
1	STSA_NUT	Coded slice of an STSA picture or subpicture* slice_layer_rbsp( )	VCL
2	RADL_NUT	Coded slice of a RADL picture or subpicture* slice_layer_rbsp( )	VCL
3	RASL_NUT	Coded slice of a RASL picture or subpicture* slice_layer_rbsp( )	VCL
4..6	RSV_VCL_4.. RSV_VCL_6	Reserved non-IRAP VCL NAL unit types	VCL
7 8	IDR_W_RADL IDR_N_LP	Coded slice of an IDR picture or subpicture* slice_layer_rbsp( )	VCL
9	CRA_NUT	Coded slice of a CRA picture or subpicture* slice_layer_rbsp( )	VCL
10	GDR_NUT	Coded slice of a GDR picture or subpicture* slice_layer_rbsp( )	VCL
11	RSV_IRAP_11	Reserved IRAP VCL NAL unit type	VCL
12	OPI_NUT	Operating point information operating_point_information_rbsp( )	non-VCL
13	DCI_NUT	Decoding capability information decoding_capability_information_rbsp( )	non-VCL
14	VPS_NUT	Video parameter set video_parameter_set_rbsp( )	non-VCL
15	SPS_NUT	Sequence parameter set seq_parameter_set_rbsp( )	non-VCL
16	PPS_NUT	Picture parameter set pic_parameter_set_rbsp( )	non-VCL
17 18	PREFIX_APS_NUT SUFFIX_APS_NUT	Adaptation parameter set adaptation_parameter_set_rbsp( )	non-VCL
19	PH_NUT	Picture header picture_header_rbsp( )	non-VCL
20	AUD_NUT	AU delimiter access_unit_delimiter_rbsp( )	non-VCL
21	EOS_NUT	End of sequence end_of_seq_rbsp( )	non-VCL
22	EOB_NUT	End of bitstream end_of_bitstream_rbsp( )	non-VCL
23 24	PREFIX_SEI_NUT SUFFIX_SEI_NUT	Supplemental enhancement information sei_rbsp( )	non-VCL
25	FD_NUT	Filler data filler_data_rbsp( )	non-VCL
26 27	RSV_NVCL_26 RSV_NVCL_27	Reserved non-VCL NAL unit types	non-VCL
28..31	UNSPEC_28.. UNSPEC_31	Unspecified non-VCL NAL unit types	non-VCL

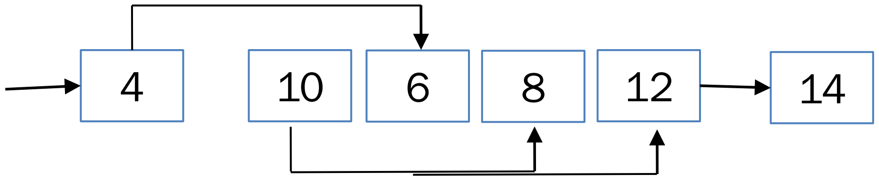
VVC NAL unit types

- 12 VCL NAL unit types
    - 8 specified
    - 4 reserved
- 20 non-VCL NAL unit types
    - 14 specified
    - 2 reserved
    - 4 unspecified

- The unspecified NAL unit types are not to be specified the video coding specification, even in future versions. These are intended for use by systems and application specifications, such as the VVC file format and the VVC RTP payload format.

Random access support

- Random access
  - *Needed for seeking, stream adaptation, tuning-in etc.; these are fundamental must-have functionalities for most video applications*
  - *Also useful for error resilience*
- VVC specifies two types of IRAP pictures, IDR and CRA, using 3 NAL unit types, and one type of GDR.
- An example CRA picture with associated RASL and RADL pictures



Picture 10:	*CRA*
Picture 6:	*RASL*
Picture 8:	*RADL*
Pictures 12, 14:	*Trailing*

- As in HEVC, VVC provides mechanisms to enable the specification of conformance of bitstreams with RASL pictures being discarded, thus to provide a standard-complaint way to enable systems components to discard RASL pictures when needed.
- GDR signalling in AVC and HEVC uses the recovery point SEI message, and in VVC now uses a NAL unit type, and the recovery point count is signalled in the PH. This allows specifying full conformance of GDR operations, e.g., a bitstream can contain only GDR pictures as the random access points, without a single IRAP picture.
- GDR can be used for improved error resilience as well as end-to-end delay optimization. Nowadays the latter is much more important, hence that was the main reason for making GDR a much more normative feature in VVC than in HEVC and AVC.

VPS, SPS, APS, PH, and SH

Video parameter set (VPS)

- VPS provides a "big picture" of a bitstream, including
  - *Number of layers, number of sublayers*
  - *Layer dependency info*
  - *Info of output layer sets (OLSs)*
  - *Profile, tier, and level (PTL) of the operation points (OPs)*
    - An OP is a temporal subset of an output layer set (OLS).
  - *DPB parameters*
  - *HRD parameters*
- Many of the VPS information can be used as the basis for systems usages like session negotiation and content selection.
- VPS was introduced for multi-layer bitstreams.
- Single-layer bitstreams do not have to have VPS in the bitstream (this is different from HEVC), and single-layer decoders may ignore VPSs when present in the bitstream (same as in HEVC).
- Updating a VPS (changing the content while reusing the VPS ID) in a bitstream is not disallowed. However, typically VPSs won’t be updated, and can be transmitted out-of-band.
  - *Out-of-band transmission means not transmitted together with the coded slices.*
  - *One example is transported as Session Description Protocol (SDP) parameters in applications using the Real-time Transport Protocol (RTP) for media transport, e.g., video telephony and video conferencing type of applications, including Zoom, FaceTime, etc.*
  - *Another example is signalling of the parameters in the sample descriptions (instead of as part of media samples) in ISO base media file forma files (a.k.a. mp4 files), or in the Media Presentation Description (MPD) of DASH media presentations (instead of as part of the media segments).*

Sequence parameter set (SPS)

- SPS – conveys sequence-level information shared by all pictures in an entire coded layer video sequence (CLVS)
  - *PTL (for single-layer OLSs)*
  - *Video format (max width, max height, colour format, bit depth), …*
  - *Subpicture info*
  - *Tool/feature on/off flags*
    - Including WPP on/off (in HEVC this is in the PPS)
  - *Coding/prediction/transform block structures and hierarchies*
  - *RPL candidates*
  - *DPB parameters (for single-layer OLSs)*
  - *HRD parameters (for single-layer OLSs)*
  - *VUI*
  - …
- Many of the SPS information can also be used as the basis for systems usages like session negotiation and content selection, for single-layer video content.
- Updating an SPS (changing the content while reusing the SPS ID) in a bitstream is not disallowed. However, typically SPSs won't be updated, and can be transmitted out-of-band.

Picture parameter set (PPS)

- PPS – conveys picture-level information shared by all slices of a picture, and that are not changing frequently across pictures thus typically shared by many pictures
  - *Feature on/off flags*
  - *Picture width and height (in SPS for HEVC, in PPS for VVC due to ARC/RPR)*
  - *RPR scaling window*
  - *Layout of tiles*
  - *Layout of rectangular slices*
  - *Default numbers of active RPL entries, initial QP, default DBF parameters, etc.*
  - *WP flags*
  - *RPL/DBF/SAO/ALF/QP delta/WP info in PH or SH flags*
  - *…*
- Updating an PPS (changing the content while reusing the PPS ID) in a bitstream is not disallowed. However, typically PPSs won’t be updated, and can be transmitted out-of-band.

Adaptation parameter set (APS)

- APS – conveys slice-level information that may be shared by multiple slices of a picture, and/or by slices of different pictures, but can change frequently across pictures and the total number of variants can be high thus not suitable for inclusion into the PPS
  - *ALF parameters*
  - *LMCS parameters*
  - *Scaling list parameters*
- Updating an APS (changing the content while reusing the APS ID) in a bitstream is not disallowed, and typically APSs would be updated, and hence APS are typically not transmitted out-of-band, although they may be transmitted in a separate, time-synchronized file format track / DASH representation / RTP stream etc.

Picture header (PH)

- Picture header – conveys information for a particular picture
  - *IRAP/GDR picture indications*
  - *Inter/intra slices allowed flags*
  - *POC LSB and (optionally) MSB*
  - *Picture-level RPL/DBF/SAO/ALF/QP delta/WP info*
  - *Coding partitioning info*
    - Max MTT depth, diff max BT min QT, …
  - *Virtual boundaries*
  - *Picture-level collocated picture info*
  - *Picture-level tool on/off flags*
  - …
- There is always one and only one PH structure for each picture.
- The PH structure is either in its own NAL unit or directly included in the SH.
  - It can be in the SH only if in the entire CLVS each picture has only one slice and all PHs are in SHs.
- It was mainly designed for saving of the signalling overhead for cases of multiple slices per picture, which would be almost always the case in practical 360º video applications and often in many other applications such as ultralow delay applications.
- To enable lightweight Bitstream Extraction And Merging (BEAM) without rewriting the PHs, highly coordinated encoding is needed such that the same PH parameters would be used for time-synchronized pictures that would be merged into one picture.

Slice header (SH)

- Slice header – conveys information for a particular slice
  - *The entire PH slices of CLVSs containing only single-slice pictures*
  - *Subpicture ID, slice address*
  - *Number of tiles in slice (for raster-scan slices)*
  - *Slice type*
  - *Output of prior pictures flag*
  - *Slice-level RPL/DBF/SAO/ALF/QP delta/WP info*
  - *Slice-level collocated picture info*
  - *Tile/WPP entry offsets*
    - Optional for tile entry offsets, and optional for WPP entry offsets
    - Both are mandatory in HEVC
  - …

POC and reference picture management

POC

- In VVC (and HEVC), POC is basically used to as a picture ID, for identification of pictures in many parts of the decoding process, including DPB management, part of which is reference picture management.
- SPS syntax

sps_log2_max_pic_order_cnt_lsb_minus4	ue(v)

- Picture header syntax

ph_pic_order_cnt_lsb	u(v)

- POC calculation

```
if( ( ph_pic_order_cnt_lsb <  prevPicOrderCntLsb )  &&
                ( ( prevPicOrderCntLsb − ph_pic_order_cnt_lsb )  >=  ( MaxPicOrderCntLsb / 2 ) ) )
        PicOrderCntMsb = prevPicOrderCntMsb + MaxPicOrderCntLsb
else if( ( ph_pic_order_cnt_lsb  >  prevPicOrderCntLsb )  &&
                ( ( ph_pic_order_cnt_lsb − prevPicOrderCntLsb )  >  ( MaxPicOrderCntLsb / 2 ) ) )
        PicOrderCntMsb = prevPicOrderCntMsb − MaxPicOrderCntLsb
else
        PicOrderCntMsb = prevPicOrderCntMsb

PicOrderCntVal = PicOrderCntMsb + ph_pic_order_cnt_lsb
```

Reference picture management (RPM)

- RPM is a core HLS functionality that is necessary for each and every video codec that uses inter prediction.
- It manages storage and removal of reference pictures into and from the DPB, and puts reference pictures in the optimal order in the reference picture lists (RPLs).
- To be able know which pictures are to be removed from the DPB, the refence picture marking process (making a reference picture as “used for short-term reference”, “used for long-term reference”, or “unused for reference”) is needed.
- RPM in AVC has two modes, the implicit sliding window process and the explicit memory management control operation (MMCO) process.
- RPM in HEVC is based on a mechanism named reference picture set (RPS).
  - *The most fundamental difference with the RPS concept compared to the sliding window plus MMCO process of AVC is that, for each particular slice a complete set of the reference pictures that are used by the current picture or any subsequent picture is provided.*
  - *Thus, a complete set of all pictures that must be kept in the DPB for use by the current or future picture is signalled.*
  - *This is different from the AVC scheme where only relative changes to the DPB are signalled.*
  - *With the RPS concept, basically no information from earlier pictures in decoding order is needed to maintain the correct status of reference pictures in the DPB.*

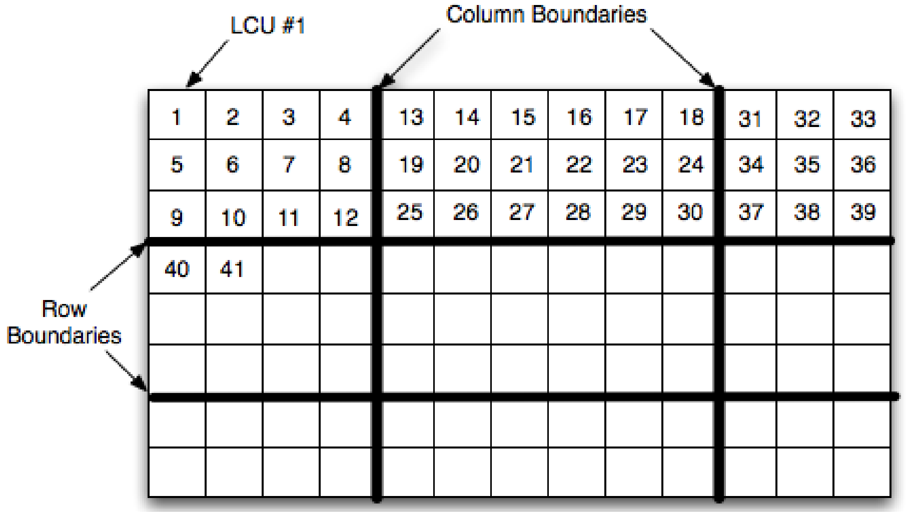
Reference picture management (RPM) in VVC

- RPM in VVC is close to that in HEVC than that in AVC, and is based on direct signalling of reference picture lists (RPLs).
- Two RPLs, list 0 and list 1, are directly signalled and derived. They are not based on RPS as in HEVC or the sliding window plus MMCO process as in AVC.
- Reference picture marking is directly based on RPLs 0 and 1, utilizing both active and inactive entries in the RPLs, while only active entries may be used as reference indices in inter prediction of CTUs.
- Information for derivation of the two RPLs is signalled by syntax elements and syntax structures in the SPS, the PPS, the PH, and the SH. Predefined RPL structures are signalled in the SPS, for use by referencing in the PH/SH.
- The two RPLs are generated for all types of slices, i.e., B, P, and I slices.
- The two RPLs are constructed without using an RPL initialization process or a RPL modification process.

Tiles, WPP, slices, subpictures

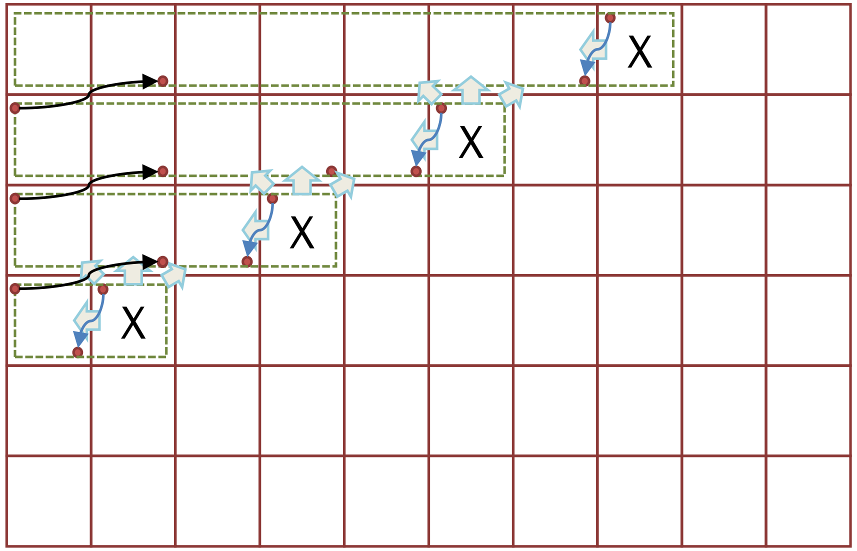
Tiles

- The feature of tiles was introduced in HEVC, mainly for parallel processing purposes. It later also gets extensively used in 360^{0} video applications, in the form of Motion-Constrained Tile Sets (MCTSs), for viewport-dependent 360^{0} video delivery optimization.
- The final design of tiles (as of now) in VVC is basically the same as in HEVC, but the signalling is changed (to be more efficient).
- Tiles define horizontal and vertical boundaries that partition a picture into tile columns and rows.
- Decoding order: tile raster scan within the picture, CTU raster scan within a tile
- Similar to slices, tiles break in-picture prediction dependencies as well as entropy decoding dependencies. However, they do not need to be included into individual NAL units (same as WPP in this regard).
- Each tile can be processed by one processor/ core, and the inter-processor and inter-core communication required for in-picture prediction between processing units for decoding neighboring tiles is limited to slice header and loop filtering info.



Wavefront parallel processing (WPP)

- WPP in VVC is similar as WPP in HEVC, except that the CTU row delay is reduced from two CTUs to one CTU.
  - *When WPP is on, each tile is partitioned into single rows of CTUs.*
  - *Entropy decoding and prediction are allowed to use data from CTUs in other partitions within a tile.*
  - *Parallel processing is possible through parallel decoding of CTU rows, where the start of the decoding of a CTU row is delayed by one CTU in VVC (and two CTUs in HEVC; **the figure below is WPP in HEVC**), so to ensure that data related to a CTU above the subject CTU is available before the subject CTU is being decoded.*
  - *Using this staggered start (like a wavefront), parallelization is possible with up to as many processors/cores as the picture contains CTU rows.*
  - *Because in-picture prediction between neighboring CTU rows within a picture is permitted, the required inter-processor and inter-core communication to enable in-picture prediction can be substantial.*

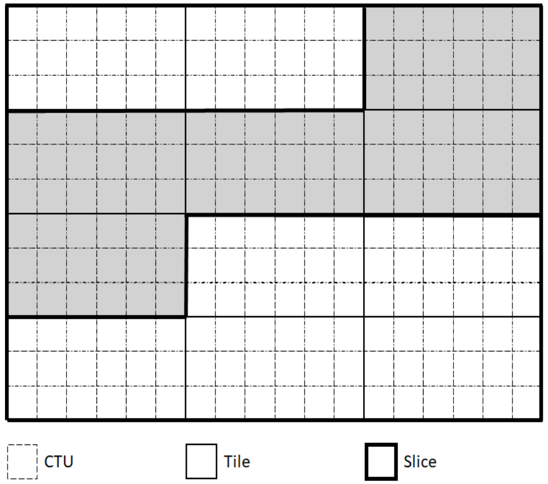


Slices

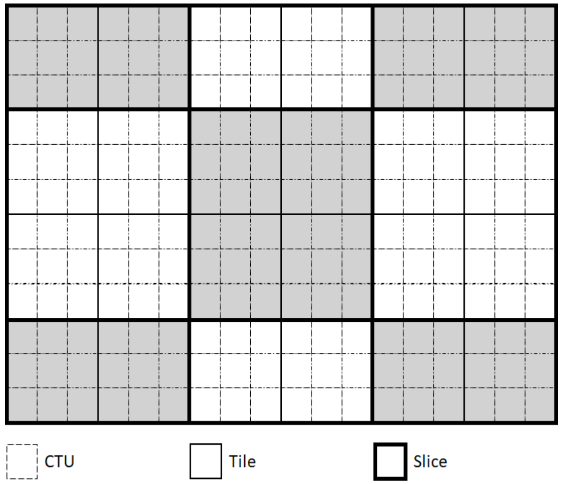
- The feature of slices was introduced mainly for Maximum Transfer Unit (MTU) size matching, an error resilience purpose. Thus slices in AVC and HEVC were based on MBs and CTUs, respectively.
- In VVC, the conventional slices based on CTUs or MBs have been removed, and its main use is no longer for MTU size matching, but rather for subpicture level access and ultralow delay.
  - *Slice based error concealment has become practically impossible, due to the ever-increasing number and efficiency of in-picture and inter-picture prediction mechanisms that make it harder for machines to estimate the quality of an error-concealed picture, and cause the coding efficiency loss due to splitting a picture into multiple slices more significant.*
  - *Network conditions become significantly better while at the same time techniques for dealing with packet losses have become significantly improved, such that conversational applications rely on system/transport-level error resilience (e.g., retransmission, FEC) and/or picture-based error resilience tools (feedback based error resilience, insertion of IRAPs, scalability with higher protection level of the base layer, and so on).*
  - *Consequently, it is very rare that a picture that cannot be correctly decoded is passed to the decoder, and when such a rare case occurs, the system can afford to wait for an error-free picture to be decoded and available for display without frequent and long period of picture freezing.*

Slices in VVC

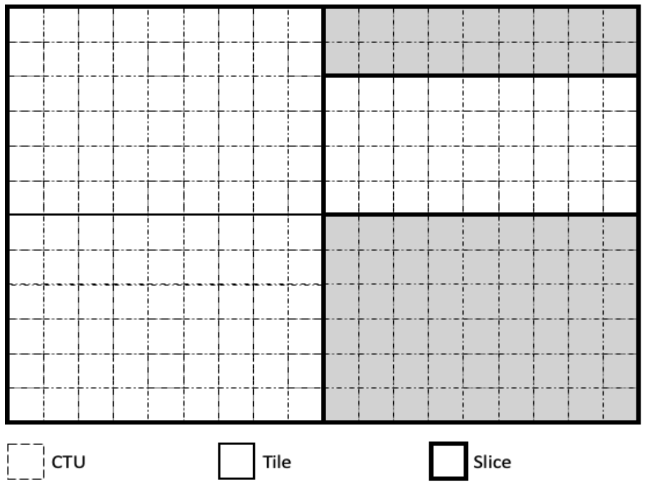
- Slices in VVC have two modes: rectangular slices (when rect_slice_flag is equal to 1) and raster-scan slices (when rect_slice_flag is equal to 0).
- Rectangular slices are always in a rectangular shape. A raster-scan slice may or may not be in a rectangular shape.
- A raster-scan slice consists of one or more complete tiles in tile raster scan order
- A rectangular slice consists of either one or more complete tiles or one or more complete CTU rows within a tile.
- The layout of rectangular slices is signalled in the PPS, based on the layout of tiles.
- Information of the tiles included in a raster-scan slice is signalled in the slice header.



A picture partitioned into 12 tiles and 3 raster-scan slices



A picture partitioned into 24 tiles and 9 rectangular slices

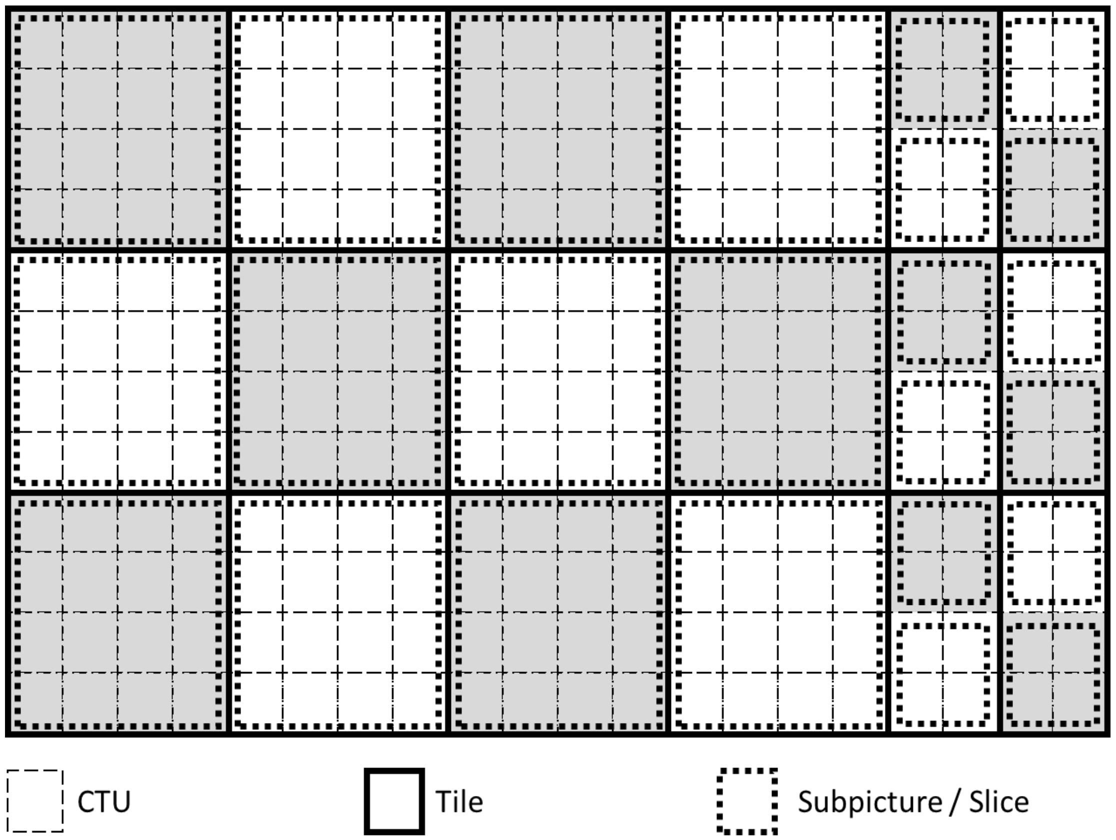


A picture partitioned into 4 tiles and 4 rectangular slices (note that the top-right tile is split into two rectangular slices)

Subpictures

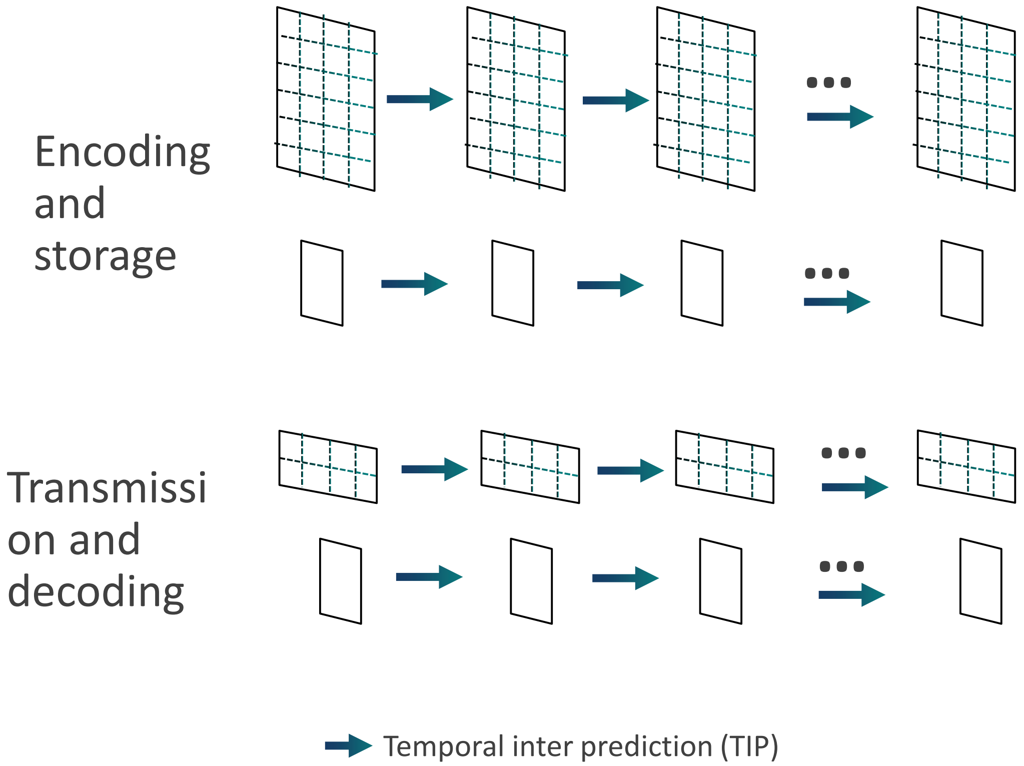
- The subpictures feature was newly introduced in the development of VVC.
- Functionally, subpictures are the same as the motion-constrained tile sets (MCTSs) in HEVC.
  - *They both allow independent coding and extraction of a rectangular subset of a sequence of coded pictures, for use cases like viewport-dependent 360º video streaming optimization and region of interest (ROI) applications.*
  - *One key difference is that the subpictures feature in VVC allows motion vectors of a coding block pointing outside of the subpicture even when the subpicture is extractable (i.e., the sps_subpic_treated_as_pic_flag[i] is equal to 1), thus allowing padding at subpicture boundaries in this case, similarly as at picture boundaries.*
    - This allows higher coding efficiency compared to the non-normative motion constraints applied for MCTSs.
  - *Another difference is the extraction of one or more subpictures from a CLVS does not need to change any bits in the VCL NAL units or PHs.*
    - In MCTS extraction, the slice headers, particularly the slice address values, need to be changed.
    - Note that in both extraction cases, parameter sets need to be rewritten.
- The layout of subpictures in VVC is signalled in the SPS, thus constant within a CLVS.
- The subpicture ID mapping may be either constant within a CLVS (in that case signalled in the SPS) or allowed to change across pictures within a CLVS (in the case signalled in the PPS).
- Each subpicture consists of one or more complete (rectangular) slices.

Example: tiles, rectangular slices, and subpictures



A picture that is partitioned into 18 tiles, 24 slices and 24 subpictures

Typical 360^{o} video delivery with subpictures



Encoding and storage

Transmission and decoding

Temporal inter prediction (TIP)

- The video sequence is coded as multiple independent/simulcast bitstreams of different resolutions.
- The higher resolution is coded with extractable subpictures
- The lower resolution bitstream can have less RAPs
- Only part of the higher-resolution bitstream needs to be transmitted/decoded/rendered

Decoding order of data units within a picture

- Decoding order of subpictures and slices
  - *The order of the VCL NAL units within a coded picture is constrained as follows:*
    - For any two coded slice NAL units A and B of a coded picture, let subpicIdxA and subpicIdxB be their subpicture level index values, and sliceAddrA and sliceddrB be their slice_address values.
    - When either of the following conditions is true, coded slice NAL unit A shall precede coded slice NAL unit B:
      - *subpicIdxA is less than subpicIdxB.*
      - *subpicIdxA is equal to subpicIdxB and sliceAddrA is less than sliceAddrB.*
  - *Note that in addition to the above, the availability rule applies for subpictures and slices:*
    - The shapes of the subpictures shall be such that each subpicture, when decoded, shall have its entire left boundary and entire top boundary consisting of picture boundaries or consisting of boundaries of previously decoded subpictures.
    - The shapes of the slices of a picture shall be such that each CTU, when decoded, shall have its entire left boundary and entire top boundary consisting of a picture boundary or consisting of boundaries of previously decoded CTU(s).
- Decoding order of tiles within a slice
  - *In the tile raster scan order of the tiles in the slice*
- Decoding order of CTUs within a tile
  - *In CTU raster scan order of the CTUs in the tile*
- The enabling of WPP does not have an impact on the decoding order of CTUs.

Reference picture resampling (RPR)

Reference picture resampling (RPR) in VVC

- PRR, also known as adaptive resolution change (ARC), allows changing of the picture spatial resolution within a CLVS.

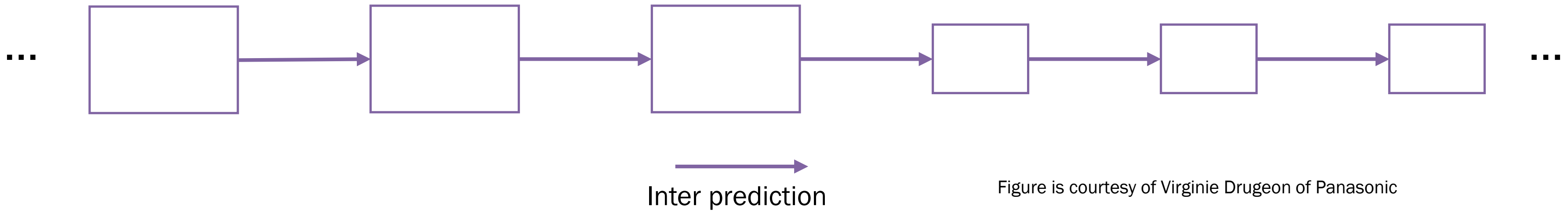


Figure is courtesy of Virginie Drugeon of Panasonic

- RPR provides higher coding efficiency for adaptation of the spatial resolution and bitrate, in both conversational (Zoom, FaceTimeetc.) and streaming applications (Internet video streaming like Netflix, Xigua Video, etc.).
- RPR also can be used in application scenarios wherein zooming of the entire video region or some region of interest is needed.
- The support of RRP mainly involves the following design changes:
  - *The picture resolution and the corresponding conformance window are moved from the SPS to the PPS, with the max picture resolution signalled in the SPS.*
  - *Resampling (both up-sampling and down-sampling) filters are specified.*
    - The scaling window is signalled in the PPS for calculation of the resampling ratio.
  - *Each picture store in the DPB (for single-layer OLSs) has the size same as the max picture resolution.*

Scalability

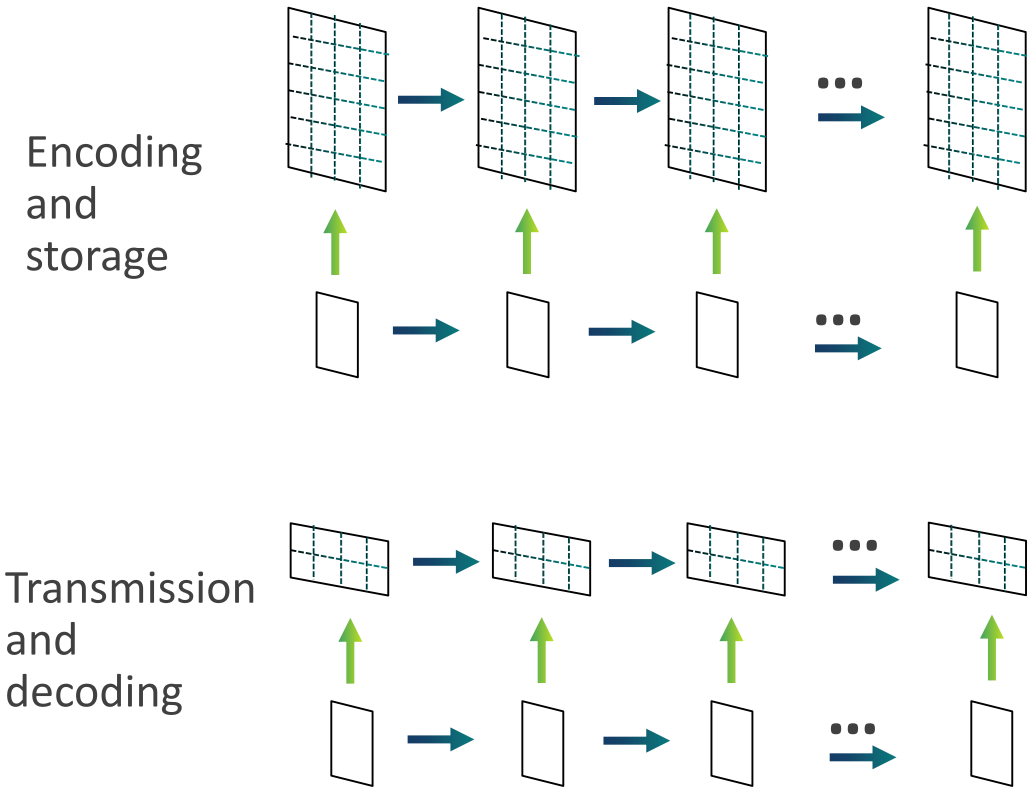
Temporal scalability support in VVC

- The ability to provide different independently decodable temporal subsets (e.g., corresponding to different frame rates) of a video bitstream
- Similar as in HEVC, including
  - *Signalling of temporal ID in the NAL unit header*
    - Media-aware network elements (MANEs) can utilize the temporal ID in the NAL unit header for stream adaptation purposes based on temporal scalability
    - Such adaptation is reported widely used in conversational applications to cope with network condition changes
  - *Restriction that pictures of a particular temporal sublayer cannot be used for inter prediction reference by pictures of a lower temporal sublayer*
  - *Sub-bitstream extraction process*
  - *Requirement that each sub-bitstream extraction output of an appropriate input must be a conforming bitstream*

Multi-layer scalability support in VVC

- Scalability enables the use of one multi-layer bitstream to serve multiple classes of devices (screen size, bandwidth, etc.), which is more efficient than simulcast (to use multiple independently-coded single-layer bitstreams).
- Thanks to the support of PRR, scalability support in VVC comes for free from signal processing (i.e., low-level) coding tool point of view, as upsampling needed for spatial scalability support uses the RPR upsampling filter.
- Of course, high-level syntax changes are needed for scalability support.
- Different from the scalability supports in any earlier video coding standards, the design of VVC scalability has been made to be friendly to single-layer decoder designs.
  - *The decoding capability for multi-layer bitstreams are specified in a manner as if there were only a single layer in the bitstream.*
    - E.g., the decoding capability, such as DPB size, is specified in a manner that is independent of the number of layers.
  - *Basically, a single-layer decoder design does not need much change to be able to decode multi-layer bitstreams.*
  - *Because of this, hopefully scalability would be supported in the Main 10 profile in VVC version 1. Let's see what would actually happen at the April and July JVET meetings.*
- When there are multiple layers, the picture in each layer and the associated non-VCL NAL units are referred to a picture unit (PU). The decoding order of PUs in an AU are in increasing order of their layer ID value.

Improved 360º video delivery with subpictures and scalability



Encoding and storage

Transmission and decoding

➡ Temporal inter prediction (TIP)

⬆ Inter-layer prediction (ILP)

- The video sequence is coded as a multi-layer **scalable** bitstream, with both TIP and ILP used
- The Enhancement Layer (EL) is split into **subpictures**
- The Base Layer (BL) does not need to be split into subpictures
- The BL can have less frequent random access points (RAPs) than the EL
- Only part of an EL needs to be transmitted and decoded

字节跳动 ByteDance

PTL, HRD

Profile, tier, and level (PTL)

- Specifies conformance, for interoperability (IOP)
- Profile: A set of selected tools with certain restrictions
- Tier: Mainly for the addition of the video contribution applications that have higher bitrate values than video distribution applications.
- Level: Restrictions on the bitstream (values of syntax elements and their arithmetic combinations, e.g., spatial resolution, and bitstream statistics, e.g., bitrate values and variations)
- A decoder implementation can be claimed to conform to a particular combination of PTL
  - *E.g., a decoder conforming to Main 10 profile, Main tier, and Level 4.1 shall be able to decode any bitstream conforming to the Main profile, Main tier, and Level 4.1 or lower.*
- A bitstreams can be claimed to conform to a particular combination of PTL
  - *E.g., a bitstream conforming to Main 10 profile, Main tier, and Level 4.1 shall be decodable by any decoder conforming to Main profile, Main or High tier, and Level 4.1 or above*
- The definitions of decoder conformance, bitstream conformance, and tier and level restrictions are heavily dependent on the HRD specification
- The PTL syntax structure also includes sub-profile signalling, as well as a long list of general constraint information (GCI) fields.
  - *The general constraint flags are new in VVC, driven by enabling to turn off certain tool in the future in case that tool becomes licensing-unfriendly.*
  - *The introduction of the new sub-profile signalling in VVC was driven by the same purpose.*

Six profiles in VVC v1

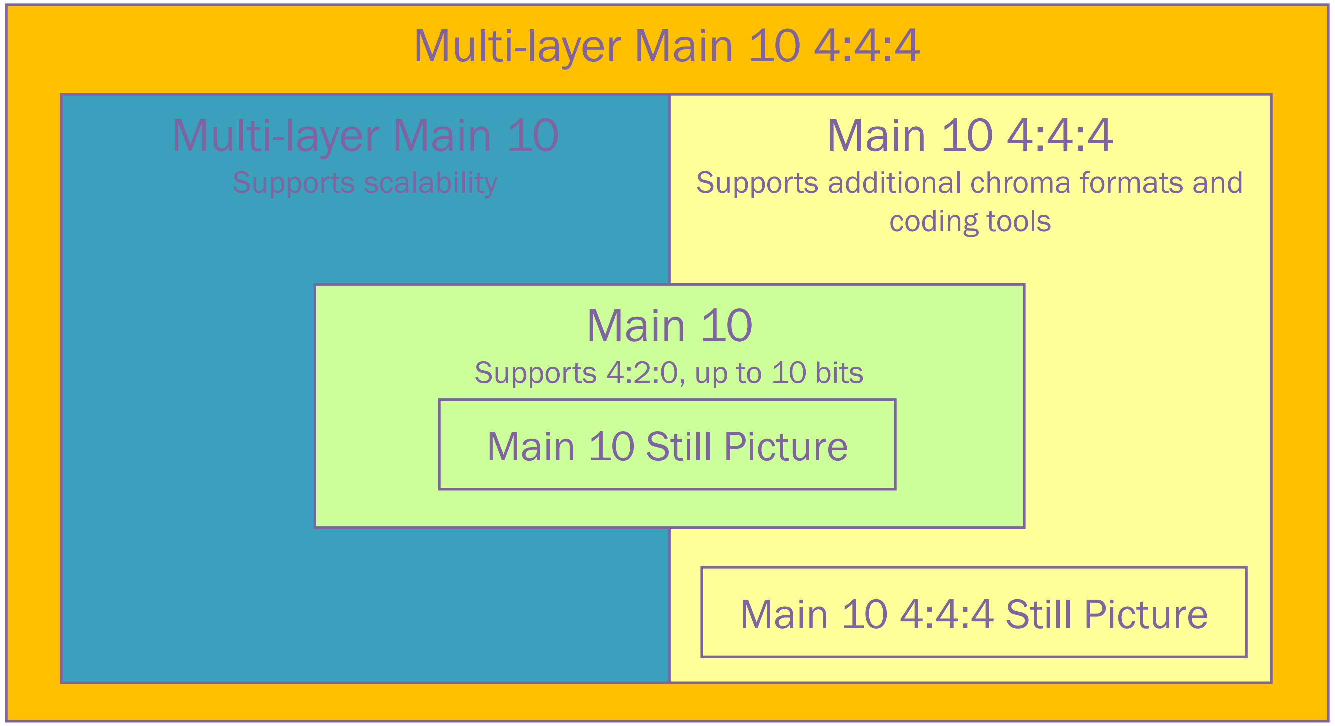

Multi-layer Main 10 4:4:4
Multi-layer Main 10
Supports scalability
Main 10 4:4:4
Supports additional chroma formats and coding tools
Main 10
Supports 4:2:0, up to 10 bits
Main 10 Still Picture
Main 10 4:4:4 Still Picture

Figure is courtesy of Virginie Drugeon of Panasonic

1. Main 10 profile: monochrome and 4:2:0, bit depth of 8 to 10 bits, 1 layer only
2. Main 10 Still Picture profile: based on the Main 10 profile, but 1 picture only
3. Main 10 4:4:4 profile: based on the Main 10 profile, but further supports 4:2:2 and 4:4:4
4. Main 10 4:4:4 Still Picture profile: based on the Main 10 4:4:4 profile, but 1 picture only
5. Multilayer Main 10 profile, based on the Main 10 profile, but >= 1 layer
6. Multilayer Main 10 4:4:4 profile, based on the Main 10 4:4:4 profile, >= 1 layer

Tiers and levels in VVC v1

- Similar as in HEVC
  - *Excluding the so-called unlimited level, 13 levels (1.0, 2.0, 2.1, 3.0, 3.1, 4.0, 4.1, 5.0, 5.1, 5.2, 6.0, 6.1 and 6.2) are specified.*
  - *Typical (Main tier) levels and interoperability points:*
    - Level 3.1: 1280×720@30fps
    - Level 4.0/4.1: 2048×1080@30/60fps
    - Level 5.0/5.1/5.2: 4096×2160@30/60/120fps
    - Level 6.0/6.1/6.2: 8192×4320@30/60/120fps
- But with a few differences, including:
  - *The maximum number of tiles rows MaxTilesRows was replaced with the maximum number of tiles per AU MaxTilesPerAu.*
  - *The maximum number of slices per picture MaxSlicesPerPicture was replaced by the maximum number of slices per AU MaxSlicesPerAu.*
  - *The highest picture rate for the High tier has been changed from 300 picture per second to 960 pictures per second.*
    - The highest picture rate for the Main tier remains to be 300 picture per second.

HRD

- Hypothetical reference decoder
- Buffering model, both coded picture buffer (CPB) and decoded picture buffer (DPB)
- Directly imposing constraints on different timing, buffer sizes, and bit rates
- Indirectly imposing constraints on bitstream characteristics/statistics
- Both CPB and DPB behaviors (mathematically) specified
- Five basic parameters
  - *Initial CPB removal delay, CPB size, bit rate, initial DPB output delay, DPB size*
- Specifying bitstream conformance and decoder conformance
- Typically required at **the encoder side** to guarantee bitstream conformance
  - *Two types of bitstream/HRD conformance points (Type I and Type II)*
  - *Two types of decoder conformance ("timing" decoder and output order decoder)*

Ultralow delay support through DU based HRD

- HEVC/VVC specify a decoding-unit-level HRD operation, for support of the so-called ultralow delay.
- The mechanism specifies a standard-complaint way to enable delay reduction below one picture interval.
- Decoding-unit-level CPB and DPB parameters may be signalled, and utilization of these information for the derivation of CPB timing (wherein the CPB removal time corresponds to decoding time) and DPB output timing (display time) is specified.
- Decoders are allowed to operate the HRD at the conventional AU-level, even when the DU-level HRD parameters are present.

DCI, VUI, SEI

Decoding capability information (DCI) NAL unit

- This is a new feature, not present in earlier standards like HEVC and AVC.
- This was initially introduced as decoding parameter set (DPS).
- Later changed from a parameter set to be a standalone NAL unit, and the name was also changed.
- The purpose of DCI/DPS is to indicate the highest required decoding capability for the entire bitstream (instead of for a CVS or CLVS like VPS or SPS).
  - *It's persistency scope is therefore the entire bitstream.*
    - Consequently, all DCI NAL units in a bitstream shall have the same content.
    - If present in the bitstream, a DCI NAL unit shall be present at least in the first AU.
  - *It can include more than one PTL syntax structures.*
    - A PTL syntax structure in a DCI NAL unit does not include sublayer level info.
- Information in the DCI NAL unit is just metadata, not needed for the decoding process.
  - *Therefore, it'd also be OK if the information is signalled in an SEI message instead.*

Video usability information (VUI)

- Similar as in HEVC and AVC, VUI carries information on
  - *Sample aspect ratio (SAR)*
  - *Overscan*
  - *Colour primaries*
  - *Transfer characteristics*
  - *Matrix coefficients*
  - *Full range or not*
  - *Sample location types*
- And also similar as in HEVC and AVC, VUI is carried in the SPS.
- There is a big editorial difference of VUI in VVC compared to HEVC and AVC: it's included in a separate specification than the main VVC spec.
  - *Currently that spec includes only VUI and those SEI messages that do not affect the HRD specification*

SEI messages

- Normative but optional
- No effect to decoding process, but affect conformance, and some affect the HRD specification
- SEI messages affecting the HRD specification are specified in Annex D of the main VVC spec (e.g., JVET-S2001)
  - Buffering period
  - Picture timing
  - Decoding unit information
  - Scalable nesting
  - Subpicture level information
- Other SEI messages are specified in the VSEI spec (e.g., JVET-S2007)
  - Filler payload
  - User data registered by Recommendation ITU-T T.35
  - User data unregistered
  - Film grain characteristics
  - Frame packing arrangement
  - Parameter sets inclusion indication
  - Decoded picture hash
  - Mastering display colour volume
  - Content light level information
  - Dependent random access point indication
  - Alternative transfer characteristics information
  - Ambient viewing environment
  - Content colour volume
  - Equirectangular projection
  - Generalized cubemap projection
  - Sphere rotation
  - Region-wise packing
  - Omnidirectional viewport
  - Frame-field information
  - Sample aspect ratio information

Thanks!
Questions, comments, and suggestions?