

Title: Modifications to High-Level Syntax and Semantics

Status: Input Document to JVT

Purpose: Proposal

Author(s) or Contact(s): Miska M. Hannuksela
Nokia Mobile Software
P.O. Box 88
33721 Tampere
Finland

Tel: +358 40 521 2845
Email: miska.hannuksela@nokia.com

Ye-Kui Wang
TICSP
Tampere Univ. of Technology
P.O. Box 553
FIN-33101 Tampere
Finland

+358 3 3115 3972
wyk@ieee.org

Young-Kwon Lim
net&tv Inc.
5th Floor Himart Building
1007-46 Sadang-Dong Dongjak-Gu
Seoul, 156-090,
Korea

+82 2 581 2305
young@netntv.co.kr

Source: Nokia Corporation, Tampere University of Technology, net&tv Inc.

1. SUMMARY

Section 2 of the contribution proposes replacement of the picture header with the picture parameter set. We believe that this improves error resiliency and makes the operation of gateways easier.

Section 3 proposes signaling of motion vectors over picture boundaries in the parameter set. The signaling can be used to save memory in decoder implementations.

Section 4 proposes a simplification to the definition of the first byte of the NAL unit syntax.

Section 5 proposes straightforward changes to the slice header syntax.

Section 6 presents a high-level architecture description of the JVT codec.

Proposed agenda category: 1b) Bug fix & clean-up, not completely uncontroversial

Keywords for agenda allocation: high-level syntax and semantics

2. REPLACEMENT OF PICTURE HEADER

One of the key goals in the JVT high-level syntax design was to facilitate efficient interworking between different types of network. Another key goal was to have an error-robust syntax for error-prone networks. An embodiment of the latter key idea is the parameter set concept, which ensures independent decodability of slices. In other words, a slice depends only on a parameter set that is preferably conveyed reliably.

During the Fairfax meeting, picture headers were introduced into the JVT syntax. This has at least two negative consequences related to the goals presented in the previous paragraph:

1. Picture headers cannot be conveyed in a reliable channel, but rather they are always conveyed in the same data channel as coded video data. Consequently, the only way to protect picture headers better than coded video data is to repeat them. We feel that the earlier parameter set based design had more flexibility and gave more possibilities in terms of error resiliency.
2. Introducing picture headers enables coding of slices that are not decodable without the presence of the corresponding picture header. The picture header concept tempts encoders connected to reliable networks to code slices without a picture header copy at the beginning. This makes the operation of gateways converting coded JVT data from a reliable network to error-prone network harder, because they should modify the packet stream so that each packet contains a copy of the picture header.

Picture header is a good location for frequently changing information remaining unchanged within one picture. In some coding and transmission environments, sending picture parameters similarly to picture header is desirable. However, there are video communication systems where very few sets of picture parameters are used during the whole session. Allowing these sets to be transmitted reliably out of band would be desirable instead of continuous redundant transmission with each coded picture. We think that it is possible to meet both goals with the same concept and syntax described below.

We propose that infrequently changing parameters are collected into the parameter set similarly to the JVT CD. The referred parameter set and its values must remain the same within one independent GOP.

Parameters that are allowed to change on picture basis but which remain the same throughout all slices of a picture are generally collected into the picture parameter set structure. Picture parameter sets are not necessarily associated to one picture only but the same picture parameter set may be referred from multiple pictures. This is one major difference from the picture header concept where the picture header was coded separately for each picture. A picture parameter set contains an identifier, and the identifier value is included in the slice header to indicate the picture parameters used in the slice. Picture parameter sets may be conveyed out of band, which is another major difference compared to picture headers.

The proposed contents of the picture parameter set are as follows:

- Picture parameter set identifier.
- Flexible macroblock ordering information, which previously occurred in the parameter set. It may be desirable to use different macroblock allocation maps for different types of pictures, for example. A checkerboard macroblock allocation map may be good for I pictures, whereas P pictures are typically conveyed in one RTP packet in (very) low bit-rates, and therefore no FMO is necessary in them.
- Picture structure.

- Reference picture selection layer. Defines multi-frame buffer operations for the entire picture similarly to the picture header of the JVT CD.
- Adaptive bi-prediction coefficients.
- The parameters related to the direct mode of B pictures/slices are expected to change during the Klagenfurt meeting. It remains to be seen in which syntax unit the upcoming direct mode parameters suit best.

The following parameter occurring in the picture header of the JVT CD is left out from the picture parameter set and included in the slice header:

- `frame_num` changes after each stored frame. It identifies the frame to which the slice belongs. If `frame_num` were included in the picture parameter set, each stored picture would have its own picture parameter set, which would essentially cause the picture parameter set being a conventional picture header.

The syntax of the picture parameter set is presented below (modifications to data types are highlighted with yellow background). The semantics of the fields remain the same as in the JVT CD.

<code>picture_parameter_set_rbsp() {</code>	Category	Descriptor
<code> picture_parameter_set_id</code>	3	<code>e(v)</code>
<code> picture_structure</code>	3	<code>e(v)</code>
<code> rps_layer()</code>		
<code> num_slice_groups_minus1</code>	3	<code>e(v)</code>
<code> if(num_slice_groups_minus1 > 0) { /* use of Flexible MB Order */</code>		
<code> mb_allocation_map_type</code>	3	<code>e(v)</code>
<code> if(mb_allocation_map_type == 0)</code>		
<code> for(i=0; i<=max_slice_group_id; i++)</code>		
<code> run_length</code>	3	<code>e(v)</code>
<code> else if(mb_allocation_map_type == 2)</code>		
<code> for(i=0; i<num_mbs_in_picture; i++)</code>		
<code> slice_group_id</code>	3	<code>u(v)</code>
<code> }</code>		
<code> direct_mv_scale_fwd</code>	3	<code>e(v)</code>
<code> direct_mv_scale_bwd</code>	3	<code>e(v)</code>
<code> direct_mv_scale_divisor</code>	3	<code>e(v)</code>
<code> explicit_bipred_weight_indicator</code>	3	<code>e(v)</code>
<code> if(explicit_bipred_weight_indicator > 1)</code>		
<code> adaptive_B_prediction_coeff_table()</code>	3	
<code> rbsp_trailing_bits()</code>	3	
<code>}</code>		

(The type of `num_slice_groups_minus1` is changes from `u( 3 )` to `e( v )` to allow future extensions in the number of slice groups. Correspondingly, the type of `slice_group_id` is changed from `u( 3 )` to `u( v )`, where the length is the minimum number of bits that can be used to represent `num_slice_groups_minus1` with an unsigned integer.)

We would also like to remark that

- Pre-defined parameter sets as proposed in JVT-C078 may help to reduce the bit-rate needed to convey parameter sets.
- The slice header as proposed in this document contains an identifier of both the parameter set and the picture parameter set. It is possible to get rid of the other identifier by nesting the identifiers as proposed in JVT-C078. However, the bit-rate saving was considered as negligible.
- The parameter set identifiers are coded with the normal entropy coding method. However, the probability distribution of the identifiers is probably far from the optimal one for the entropy coder. Thus, in order to save bits, we recommend considering the entropy coding method proposed in JVT-C078 for parameter set identifiers.
- If the JVT foresees that a situation where only few parameter change frequently compared to an earlier parameter set, it should be verified if a considerable amount of bits could be saved by designing a "tagged" parameter set NAL unit. In other words, the NAL unit would contain a pointer to a reference parameter set that is used as a basis for the new one to be defined. The reference parameter set could be a default parameter set defined in the standard or any earlier parameter set. The NAL unit would also contains indications which parameters are changed relative to the reference parameter set and, of course, the values of the parameters.

3. ADDITIONS TO PARAMETER SET

As motion vectors over picture boundaries provide only a moderate compression gain in small picture sizes (contribution VCEG-M34 reports about 1 % coding gain in QCIF and about 6 % coding gain in CIF), we propose that the possibility of having motion vectors over pictures boundaries is signaled in the parameter set. If motion vectors over picture boundaries are not used, decoder may allocate remarkably smaller amount of memory for an uncompressed picture compared to a picture where boundary pixels are extrapolated.

motion_vectors_over_picture_boundaries	0	u(1)
---	---	--------

4. THE FIRST BYTE OF THE NAL UNIT

Before the Fairfax meeting, the definition of the first byte of the NAL unit syntax was simple:

NAL unit type	Error flag	Code
SSP	0	0x10
SSP	1	0x11
DPA	0	0x20
DPA	1	0x21
DPB	0	0x30
DPB	1	0x31
DPC	0	0x40
DEC	1	0x41
SEIP	0	0x50
SEIP	1	0x51
PUP	0	0x60
PUP	1	0x61
CP	0	0x70
CP	1	0x71

The primary purpose of the first byte was to be a kind of an interface between the video coding level and the transport coding level. The first byte contained that was useful for both the JVT

codec and for the transport system. Fixed-length coding was applied in the first byte in order to enable parsing easily in any network element. Entities in the transport system were able to utilize the information in the first byte or even manipulate it. For example,

- Data partition A packets could be prioritized over data partition B and C packets.
- Gateways could compose or split compound packets (CPs).
- Any network element could set the error flag.

During the Fairfax meeting, the design of the first byte was changed much. We think that the original purpose of the first byte, i.e., to convey information important to the network elements, was forgotten in some cases. In the following, we summarize the Fairfax changes in the first byte and comment on their relevancy for network elements:

- `picture_header_flag`. Picture headers or similar units should be prioritized in transport compared to normal slice data.
- `non_stored_content_flag` signals whether the content of the EBSP belongs to a picture that is not stored in the multi-frame buffer. Network elements may use this flag for traffic shaping and prioritization.
- Picture header NAL units for each picture type. Picture headers should be prioritized in transport compared to normal slice data. However, we think that the concept of picture headers can be replaced with picture parameter sets as explained in section 2.
- Indication of picture type for each slice type (e.g., intra slice of intra picture, intra slice of IDR picture, and intra slice of mixed picture). We do not believe that the picture type information can be used in a meaningful way in network elements. All stored pictures are approximately equally important subjectively, and any picture type can either be stored or non-stored.

If picture parameter sets are conveyed in-band and used similarly to picture headers, there is a need to associate them tightly with one or more slices of the corresponding pictures. Otherwise, if a picture header were conveyed in a separate NAL unit from slice data, they are likely to be encapsulated into different transport packets of a packet-oriented network, and the picture header may arrive later than the slice data packet due to transport delay jitter. Thus, we want to maintain the possibility to multiplex a picture parameter set with a slice or data partition A into the same NAL unit.

Supplemental Enhancement Information (SEI) is sometimes tightly coupled with a particular picture or slice. An example of such an SEI message is the clock timestamp. Thus, we propose that an SEI RBSP can be encapsulated into the same NAL unit as a slice or a data partition A.

We propose the following NAL unit syntax (modifications are highlighted with yellow background):

<code>nal_unit(NumBytesInEBSP) {</code>	Category	Descriptor
<code> nal_unit_type</code>		<code>u(4)</code>
<code> picture_parameter_set_flag</code>		<code>u(1)</code>
<code> sei_flag</code>		<code>u(1)</code>
<code> error_flag</code>		<code>u(1)</code>
<code> non_stored_content_flag</code>		<code>u(1)</code>
<code> for(I=0; i<NumBytesInEBSP; i++)</code>		
<code> ebsp[i]</code>		<code>b(8)</code>
<code>}</code>		

The semantics of `nal_unit_type` are defined below:

Code	NAL Unit Type (<code>nal_unit_type</code>)	Value of picture_paramete r_set_flag and sei_flag
0x0	Reserved	0
0x1	Slice	Any
0x2	DPA	Any
0x3	DPB	0
0x4	DPC	0
0x5	IDR	Any
0x6	Supplemental Enhancement Information	0
0x7	Parameter Set Information	0
0x8	Picture Parameter Set	0
0x9 – 0xF	Reserved	0

Note that only slice, data partition A, and IDR NAL units can have '1' bits in `picture_parameter_set_flag` and/or `sei_flag`. If `picture_parameter_set_flag` is '1', the NAL unit starts with a picture parameter set EBP, and the EBP of the indicated type follows. If both `picture_parameter_set_flag` and `sei_flag` are '1', an SEI EBP follows the picture parameter set EBP and precedes the EBP of the indicated type. If `picture_parameter_set_flag` is '0' and `sei_flag` is '1', the NAL unit starts with an SEI EBP followed by the EBP of the indicated type. `picture_parameter_set_flag` and `sei_flag` for other NAL unit types than slice, DPA, and IDR are '0', and non-zero values are reserved for future use.

5. SLICE HEADER

Due to the proposed changes in the parameter set syntax and in picture-level parameters, a couple of changes in the slice header are necessary:

- `picture_parameter_set_id` points to the picture parameter set that shall be used when decoding the slice.
- `frame_num` is defined the same way as in the JVT CD.
- `coding_type` indicates the coding type (I, P, B, SI, SP) of the slice.

In addition, fixed-length coding of `first_mb_in_slice` is proposed to save bits. See JVT-C079 for an analysis of bit-rate savings. The length of the `first_mb_in_slice` field is the minimum length of

an unsigned integer that can represent $(\text{frame\_width\_in\_MBs\_minus1} + 1) * (\text{frame\_height\_in\_MBs\_minus1} + 1) - 1$.

Furthermore, `num_mbs_in_slice` is changed to `num_mbs_in_slice_minus1`, because slices of zero macroblocks are not allowed.

Altogether, the slice header syntax is as follows (modifications are highlighted with yellow background):

slice_header() {	Category	Descriptor
parameter_set_id	4	e(v)
picture_parameter_set_id	4	e(v)
frame_num	4	u(v)
coding_type	4	e(v)
first_mb_in_slice	4	u(v)
if(coding_type() == Inter coding_type() == Bipred) {		
num_ref_pic_active_fwd_minus1	4	e(v)
if(coding_type() == Bipred)		
num_ref_pic_active_bwd_minus1	4	e(v)
}		
rps_layer()		
slice_qp_minus26 /* relative to 26 */	4	e(v)
if(coding_type() == SP coding_type() == SI) {		
if(coding_type() == SP)		
sp_for_switch_flag	4	u(1)
slice_qp_s_minus26 /* relative to 26 */	4	e(v)
}		
if(entropy_coding_mode == 1)		
num_mbs_in_slice_minus1	4	e(v)
}		

6. ARCHITECTURE

Figure 1. illustrates the high level architecture of AVC|H.264 decoder. To fulfill two major requirements, independently decodable slices and easy interchange of contents over the network gateways, two types of data structure, bitstream and packet, are defined.

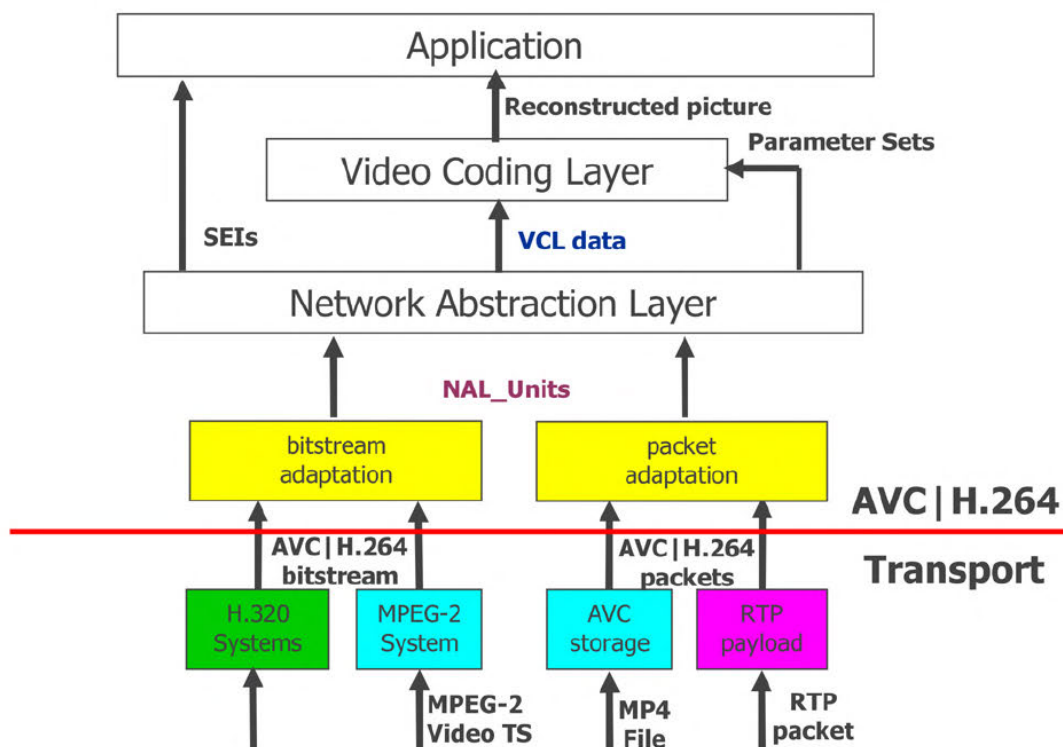


Figure 1. High level architecture

AVC|H.264 bitstream format is defined for the transport layers based on bitstreams such as ITU-T H.320 or ISO/IEC 13818-1. It is defined as a data structure very similar to ISO/IEC 13818-2|ITU-T H.262 elementary stream including start codes to identity boundaries and types of concatenated data elements. For the packet-oriented transports such as RTP, AVC|H.264 packet structure is also defined.

Bitstreams or packets form the transport layers are processed appropriately at each transport specific adaptation layers to produce a common data structure, NAL_Unit. It is defined as a common container having short header for various information needed for decoding process and accurate presentation of reconstructed pictures. It is a data entity commonly existing in the bitstream and the packets.

NAL is a layer consuming the NAL_Units to produce separate data entities such as VCL data, parameter sets and SEI messages. VCL produces reconstructed pictures by consuming VCL data containing encoded information about the slices. Parameter sets are used to configure VCL by supplying high level coding conditions above slice layer. SEI messages are consumed by VCL or by the applications. SEI messages contain information that is not necessary to decode VCL data correctly but is helpful for decoding, presentation, or other purposes.

(Append for Proposal Documents)

JVT Patent Disclosure Form

International Telecommunication Union
Telecommunication Standardization Sector



International Organization for Standardization



International Electrotechnical Commission



Joint Video Coding Experts Group - Patent Disclosure Form

(Typically one per contribution and one per Standard | Recommendation)

Please send to:

JVT Rapporteur Gary Sullivan, Microsoft Corp., One Microsoft Way, Bldg. 9, Redmond WA 98052-6399, USA
Email (preferred): Gary.Sullivan@itu.int Fax: +1 425 706 7329 (+1 425 70MSFAX)

This form provides the ITU-T | ISO/IEC Joint Video Coding Experts Group (JVT) with information about the patent status of techniques used in or proposed for incorporation in a Recommendation | Standard. JVT requires that all technical contributions be accompanied with this form. *Anyone* with knowledge of any patent affecting the use of JVT work, of their own or of any other entity ("third parties"), is strongly encouraged to submit this form as well.

This information will be maintained in a "living list" by JVT during the progress of their work, on a best effort basis. If a given technical proposal is not incorporated in a Recommendation | Standard, the relevant patent information will be removed from the "living list". The intent is that the JVT experts should know in advance of any patent issues with particular proposals or techniques, so that these may be addressed well before final approval.

This is not a binding legal document; it is provided to JVT for information only, on a best effort, good faith basis. Please submit corrected or updated forms if your knowledge or situation changes.

This form is *not* a substitute for the ITU ISO IEC Patent Statement and Licensing Declaration, which should be submitted by Patent Holders to the ITU TSB Director and ISO Secretary General before final approval.

Submitting Organization or Person:

Organization name Nokia Corporation, net&tv Inc.

Mailing address _____

Country _____

Contact person _____

Telephone _____

Fax _____

Email _____

Place and date of submission _____

Relevant Recommendation | Standard and, if applicable, Contribution:

Name (ex: "JVT") JVT

Title Modifications to High-Level Syntax and Semantics

Contribution number JVT-D094

(Form continues on next page)

Disclosure information – Submitting Organization/Person (choose one box)



2.0 The submitter is not aware of having any granted, pending, or planned patents associated with the technical content of the Recommendation | Standard or Contribution.

or,

The submitter (Patent Holder) has granted, pending, or planned patents associated with the technical content of the Recommendation | Standard or Contribution. In which case,



2.1 The Patent Holder is prepared to grant – on the basis of reciprocity for the above Recommendation | Standard – a free license to an unrestricted number of applicants on a worldwide, non-discriminatory basis to manufacture, use and/or sell implementations of the above Recommendation | Standard.



2.2 The Patent Holder is prepared to grant – on the basis of reciprocity for the above Recommendation | Standard – a license to an unrestricted number of applicants on a worldwide, non-discriminatory basis and on reasonable terms and conditions to manufacture, use and/ or sell implementations of the above Recommendation | Standard.

Such negotiations are left to the parties concerned and are performed outside the ITU | ISO/IEC.



2.2.1 The same as box 2.2 above, but in addition the Patent Holder is prepared to grant a “royalty-free” license to anyone on condition that all other patent holders do the same.



2.3 The Patent Holder is unwilling to grant licenses according to the provisions of either 2.1, 2.2, or 2.2.1 above. In this case, the following information must be provided as part of this declaration:

- patent registration/application number;
- an indication of which portions of the Recommendation | Standard are affected.
- a description of the patent claims covering the Recommendation | Standard;

*In the case of any box **other than 2.0** above, please provide the following:*

Patent number(s)/status _____

Inventor(s)/Assignee(s) _____

Relevance to JVT _____

Any other remarks: _____

(please provide attachments if more space is needed)

(form continues on next page)

Third party patent information – fill in based on your best knowledge of relevant patents granted, pending, or planned by other people or by organizations other than your own.

Disclosure information – Third Party Patents (choose one box)

☒

3.1 The submitter is not aware of any granted, pending, or planned patents *held by third parties* associated with the technical content of the Recommendation | Standard or Contribution.

☐

3.2 The submitter believes third parties may have granted, pending, or planned patents associated with the technical content of the Recommendation | Standard or Contribution.

For box 3.2, please provide as much information as is known (provide attachments if more space needed) - JVT will attempt to contact third parties to obtain more information:

3rd party name(s)

Mailing address

Country

Contact person

Telephone

Fax

Email

Patent number/status

Inventor/Assignee

Relevance to JVT

Any other comments or remarks: