|  |  |
| --- | --- |
| **Joint Collaborative Team on Video Coding (JCT-VC)**  **of ITU-T SG16 WP3 and ISO/IEC JTC1/SC29/WG11**  12th Meeting: Geneva, CH, 14–23 Jan. 2013 | Document: JCTVC-L01802 |

|  |  |  |  |
| --- | --- | --- | --- |
| *Title:* | **Profile, tier, level and operation points signaling in the VPS extension** | | |
| *Status:* | Input Document to JCT-VC | | |
| *Purpose:* | Proposal | | |
| *Author(s) or Contact(s):* | Jill Boyce 433 Hackensack Ave  Hackensack, NJ 07601 | Tel: Email: | +1 201 478 6145 jill@vidyo.com |
| *Source:* | Vidyo | | |

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

# Abstract

A method for profile\_tier\_level and operation point signaling in the VPS extension is proposed, similar to the previous proposal in JCTVC-K0204. The operation points signaled in the base VPS for HRD parameters are utilized, and additional operation points may be signaled in the VPS extension. For each operation point, level information is sent, and profile and tier information may either be explicitly sent or may be inferred by referencing the previously sent profile and tier for another operation point.

In the r1 version, the additional operation points signaling is removed, and changes were made to the syntax table for profile\_tier\_level signaling.

In the r2 version, a change was made to the syntax table to match the variant recommended for adoption by the BoG.

# Background

## Base specification

The base specification contains signaling of operation points for the HRD.

The relevant portion of the VPS is copied below.

|  |  |
| --- | --- |
| **vps\_max\_nuh\_reserved\_zero\_layer\_id** | u(6) |
| **vps\_num\_op\_sets\_minus1** | ue(v) |
| for( i = 1; i <= vps\_num\_op\_sets\_minus1; i++ ) |  |
| operation\_point\_set( i ) |  |
| **vps\_num\_hrd\_parameters** | ue(v) |
| for( i = 0; i < vps\_num\_hrd\_parameters; i++ ) { |  |
| **hrd\_op\_set\_idx**[ i ] | ue(v) |
| if( i > 0 ) |  |
| **cprms\_present\_flag**[ i ] | u(1) |
| hrd\_parameters( cprms\_present\_flag[ i ], vps\_max\_sub\_layers\_minus1 ) |  |
| } |  |

The operation point set syntax function is also copied below.

|  |  |
| --- | --- |
| operation\_point\_set( opIdx ) { | **Descriptor** |
| for( i = 0; i <= vps\_max\_nuh\_reserved\_zero\_layer\_id; i++ ) |  |
| **op\_layer\_id\_included\_flag**[ opIdx ][ i ] | u(1) |
| } |  |

## Draft extension design in JCTVC-K1007

JCTVC-K1007 contains a draft design of the VPS extension, and includes in its list of topics for future study efficient signaling of profile\_tier\_level in the vps\_extension( ). Because of time constraints at the Shanghai meeting, several contributions at that meeting related to profile signaling were not fully evaluated, so a very simple brute force profile\_tier\_level signaling was included in the JCTVC-K1007 draft design’s VPS extension, which is copied below.

|  |  |
| --- | --- |
| for( i = 1; i <= vps\_max\_layers\_minus1 ; i++ ) |  |
| profile\_tier\_level( 1, vps\_max\_sub\_layers\_minus1 ) |  |

# Proposal

A profile\_tier\_level and operation point signaling solution is proposed, which is similar to that originally proposed as one of several aspects of JCTVC-K0204. It is assumed that it is desirable to have a profile/tier/level indication for each operation point, rather than each layer.

Profile/tier/level for each of the signaled operation points is either explicitly signaled or inferred by referencing another operation point.



## Profile\_tier\_level signaling

A coding efficiency savings method is proposed for profile\_tier\_level signaling, based upon the assumption that several operation points may use the same profile and tier but different levels. For example, consider a spatial scalable bitstream with 3 spatial layers, with layer 2 referencing layer 1, and layer 1 reference the layer 0 base layer. Both layer 1 and layer 2 are spatial scalable enhancement layers and will likely use the same scalable profile indication, but will have different levels.

A loop over all totalOpSets of the signaled operation points is proposed, in which a profile presence flag is coded. If the profile presence flag is set, all of profile, tier, and level are explicitly coded for the operation point. If the profile present flag is not set for a particular operation point, an index to a reference operation point’s profile and tier are coded rather than explicitly coding the profile and tier, and only the level is explicitly coded.

When the referenced op is the 0-th operation point, the base layer’s profile and tier are inferred.

The syntax below is proposed to be added in the VPS extension, replacing the existing profile\_tier\_level signaling in the JCTVC-K1007 draft design.

|  |  |
| --- | --- |
| for( i = 1; i <= vps\_num\_op\_sets\_minus1; i ++ ) { |  |
| **vps\_profile\_present\_flag[** i **]** | u(1) |
| if (!vps\_profile\_present\_flag[ i ]) |  |
| **profile\_op\_ref\_minus1[** i **]** | ue(v) |
| profile\_tier\_level( vps\_profile\_present\_flag[ i ],  vps\_max\_sub\_layers\_minus1) |  |
| **}** |  |

**vps\_profile\_present\_flag**[i **]** equal to 1 specifies the profile and tier information for operation point i is present in the profile\_tier\_level( ) syntax structure.vps\_profile\_present\_flag[ i ] equal to 0 specifies that profile and tier information for operation point i is not present in the profile\_tier\_level( ) syntax structure and is inferred.

## **profile\_op\_ref\_minus1**[ i ] indicates that the profile and tier information for the i-th operation point is inferred to be equal to the profile and tier information from the (profile\_op\_ref\_minus1[ i ] + 1)-th operation point. Complete proposed VPS extension syntax

|  |  |
| --- | --- |
| vps\_extension( ) { | Descriptor |
| while( !byte\_aligned( ) ) |  |
| **vps\_extension\_byte\_alignment\_reserved\_one\_bit** | u(1) |
| **avc\_base\_codec\_flag** | u(1) |
| **scalability\_mask** | u(16) |
| for( i = 0; i <NumScalabilityTypes; i++ ) { |  |
| **dimension\_id\_len\_minus1**[ i ] | u(3) |
| } |  |
| **vps\_nuh\_layer\_id\_present\_flag** | u(1) |
| // layer specific information |  |
| for( i = 1; i <= vps\_max\_layers\_minus1; i++ ) { |  |
| // mapping of layer ID to scalability dimension IDs |  |
| if( vps\_nuh\_layer\_id\_present\_flag ) |  |
| **layer\_id\_in\_nuh**[ i ] | u(6) |
| for( j = 0; j <= num\_dimensions\_minus1; j++ ) ~~{~~ |  |
| **dimension\_id**[ i ][ j ] | u(v) |
| } |  |
|  |  |
|  |  |
| for( i = 0; i <= vps\_num\_op\_sets\_minus1; i ++ ) { |  |
| **vps\_profile\_present\_flag[** i **]** | u(1) |
| if (!vps\_profile\_present\_flag[ i ]) |  |
| **profile\_op\_ref\_minus1[** i **]** | ue(v) |
| profile\_tier\_level( vps\_profile\_present\_flag[ i ],  vps\_max\_sub\_layers\_minus1) |  |
| **}** |  |
| for( i = 1; i <= vps\_max\_layers\_minus1; i++ ) { |  |
| // layer dependency |  |
| **num\_direct\_ref\_layers**[ i ] | u(6) |
| for( j = 0; j < num\_direct\_ref\_layers[ i ]; j++ ) |  |
| **ref\_layer\_id**[ i ][ j ] | u(6) |
| } |  |
| } |  |

# Patent rights declaration(s)

**Vidyo may have IPR relating to the technology described in this contribution and, conditioned on reciprocity, is prepared to grant licenses under reasonable and non-discriminatory terms as necessary for implementation of the resulting ITU-T Recommendation | ISO/IEC International Standard (per box 2 of the ITU-T/ITU-R/ISO/IEC patent statement and licensing declaration form).**