



JCTVC-L0180: Profile, tier, level and operation points signaling in the VPS extension

Jill Boyce

Vidyo



Background

- HEVC specification contains operation point signaling in the VPS

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)	
}	

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)
}	

Background

- **JCTVC-K1007 contains draft design of the VPS extension**
 - Efficient signaling of profile_tier_level in the vps_extension() included in list of topics for further study encouraged
 - Because of time constraints at the Shanghai meeting, several contributions at that meeting related to profile signaling were not fully evaluated
 - Very simple brute force profile_tier_level signaling was included in the JCTVC-K1007 draft design's VPS extension

for(i = 1; i <= vps_max_layers_minus1 ; i++)	
profile_tier_level(1, vps_max_sub_layers_minus1)	

Proposal

- **Similar to previous proposal in JCTVC-K0204, which was not fully reviewed because of time constraints in Shanghai**
 1. Signaling of operation points
 2. Signaling of profile_tier_level for each operation point

Signaling of operation points

- **Make use of operation points signaled in base VPS**
- **Optionally send additional operation points in VPS extension**

vps_num_additional_op_sets	ue(v)
totalOpSets = vps_num_op_sets_minus1 + 1 + vps_num_additional_op_sets	
for (i = vps_num_op_sets_minus1 + 1; i < totalOpSets; i++)	
operation_point_set (i)	

num_additional_operation_points specifies the number of additional operation point sets for which **operation_point_set()** syntax structures are present in the video parameter set extension.

profile_tier_level signaling

- In VPS, replace per level profile_tier_level() signaling loop with per operation point signaling loop
- Optionally send profile/tier explicitly for each operation point, or refer to a previously sent profile/tier

for(i = 1; i <= totalOpSets; i ++) {	
vps_profile_present_flag[i]	u(1)
if (!vps_profile_present_flag[i])	
profile_op_ref[i]	u(8)
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[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[i]-th operation point. When profile_op_ref[i] equals 0, the profile and tier information for the i-th operation point is inferred to be equal to the profile and tier information of the base layer coded in the video parameter set RBSP.

Discussion and Conclusion

- **Preferable to have profile_tier_level signaling per operation point, rather than per layer**
- **Proposed method provides some coding efficiency advantages, and is simple**
- **Assumes that level will be different for all layers, but profile/tier may be repeated across layers**
 - For example, spatial scalable bitstream with 3 spatial layers
 - Layer 2 references layer 1, and layer 1 references the layer 0 base layer
 - Layer 1 and layer 2 are both spatial scalable enhancement layers, will likely use the same scalable profile (and tier) indication, but have different levels