



G-PCC: Editors' FDIS questions (v2)

G-PCC Editors

Resolved questions

1 & 2. Frame Index Attribute

Question: are frame_idx values relative or absolute?

Question: Is a monotonic constraint on frame_idx values?

Response: Both interpretations are valid:

- frame_idx may be absolute or relative
- frame_idx is not related to FrameCtr
- frame_idx is not required to be monotonic across the sequence
- Use two attribute types to distinguish the absolute and relative cases

4. LoD attribute coding

Intra prediction

Intra prediction uses either average or direct prediction

- Direct prediction signals the index in a list of neighbours
- If the signalled index is out of bounds zero prediction is used

Question: Should conformance prohibit out of bounds indices?

Response: Define decoder behaviour, Prohibit out of bound index value

5. Scalable attribute coding

Division in quantization weight calculation

The weight derivation is:

```
for (lod = 0; lod < NumLods - 1; lod++) {  
    weight = !lod ? 1 : (geom_num_points_minus1 + 1) / LodSize[lod]  
    for (coeffIdx = 0; coeffIdx < LodCoeffCnt[lvl]; coeffIdx++)  
        CoeffWeight[LodCoeffIdx[lvl][coeffIdx]] = weight * 256  
}
```

Question: This is the only integer division operation, should it use the Div() approximation?

Question: Should the fractional bits be discarded?

- NB: the SW is: `double weight = numPoints / LodSize[lod]`

Response: Use integer division, discard fractional bits

6. Scalable attribute coding

Extended inter search range equal to 0

In current SW:

- **Non-scalable case:** inter_lod_search_range=0 is a **full range** extended search
- **Scalable case:** inter_lod_search_range=0 **disables** the extended search

Question: Is this intended?

- NB: full range search may be specified with by a large range value.

Response: Align definitions, inter_lod_search_range=0 disables extended search

7. Scalable attribute coding

Extended inter search centre

Search centre is index of:

- first found neighbour (unless scalable enabled), or
- nearest co-located point if no neighbours found

Question: Is it intended for the behaviour to change when scalable enabled?

Response: Align behaviour: remove condition on scalable

- The two methods are intended to be the same apart from scalable specifics

8. Centroid based LoD subsampler for decimation

SW Question

The centroid based decimator picks one point from a group of $2 \times 2 \times 2$ blocks. Blocks are added to the group until the group contains at least N points.

The tmc13 encoder enforces the configuration " $N \leq 8$ "

Question: Is this check correct? Should it be a spec limit?

- NB: A group may contain up to $N + 7$ points
- If the group contains 8 or more points, the last point is selected (w/o centroid)

Response: Always calculate centroid (spec), fix SW optimisation to match

- Remove SW cfg limit on lodSamplingPeriod

9. Centroid based LoD subsampler for decimation

Consistency

Subsampling rate (N) is configured In current SW:

- **Centroid:** $\text{lodIndex} + \text{lod_dist2}$
- **Distance:** $\text{lodIndex} + \text{lod_dist2} + \text{lod_dist2_offset}$

Question: Why is `lod_dist2_offset` omitted for centroid case?

- NB: in centroid case, `lod_dist2_offset` may be signalled in slice header

Response: Align definitions: add `lod_dist2_offset`

Unresolved questions

3. Attribute coding

Maximum number of components

Question: What is the maximum value of attribute_dimension_minus1?

- RAHT and LoD attribute coders support only $0 < \text{AttrDim} \leq 3$
- Raw attribute data unit supports $\text{AttrDim} > 0$

Response: Leave undefined, permit $\text{AttrDim} > 3$ for raw case

10. HLS: clarifying unique_points_flag

Confusing HLS

The unique_points_flag is similarly named to the unique_points_constraint_flag.

- unique_points_constraint_flag indicates that there are no duplicate points
- unique_points_flag indicates that the duplicate point count is **not** signalled

Normally we assert when something is present, rather than assert that it isn't.

- Suggestion: Rename **unique_points_flag** to **duplicate_points_enabled_flag**
- NB: this **inverts** the sense of the flag and avoids "if(**!**unique_points_flag)"

Response: Apply suggestion

11. Octree stream offsets

Issue: Adoption requires fix

Adoption of octree sub-stream offset removal (m55576) results in signalling issue with chunked entropy streams

- NB: this wasn't an issue prior to the reversal of the bypass sub-stream order

Possible fixes (from most to least efficient):

A. Fix concatenation (by adjusting a chunk)

B. Restoring the offsets

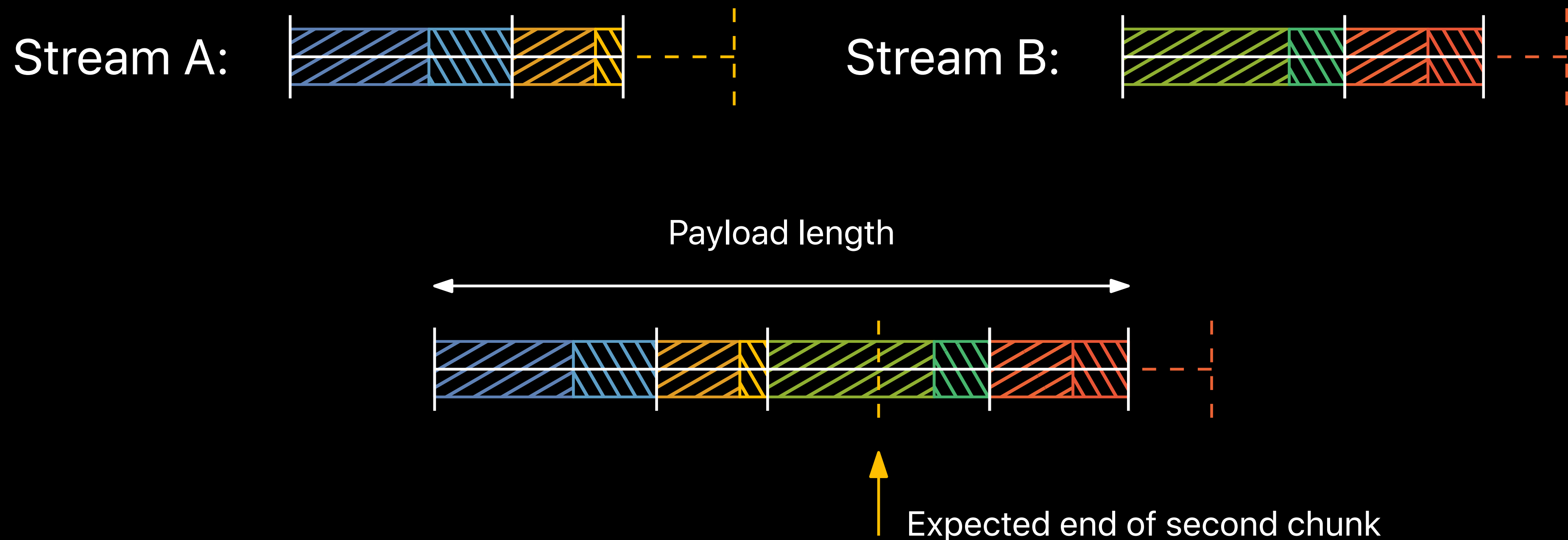
C. Fix concatenation (by adding padding)

Response: Use (A) — To be checked by AHG; AHG to make alternative recommendation if issues remain

11. Octree stream offsets

Issue: Adoption requires fix

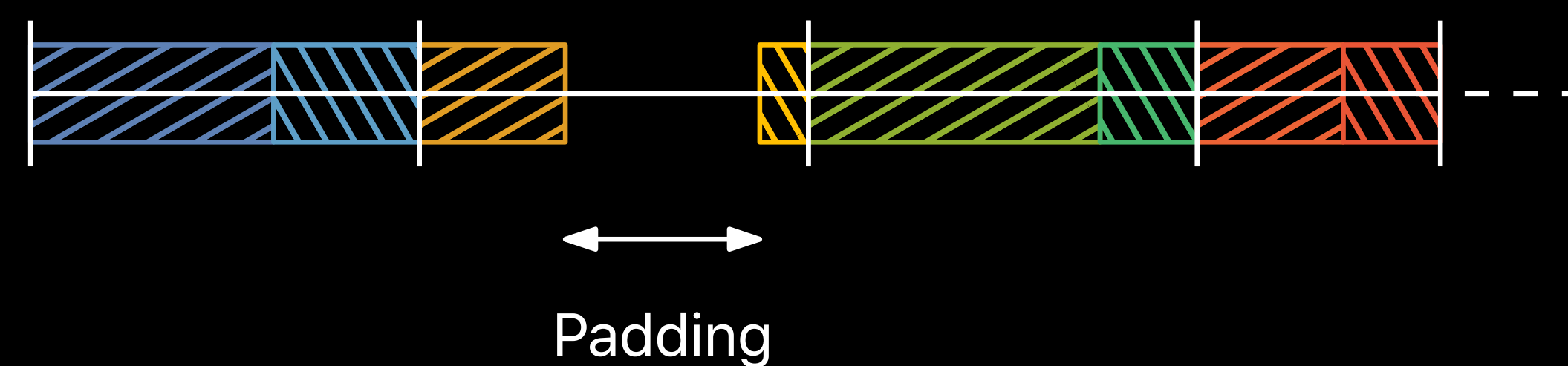
Adoption of octree sub-stream offset removal (m55576) results in signalling issue with chunked entropy streams



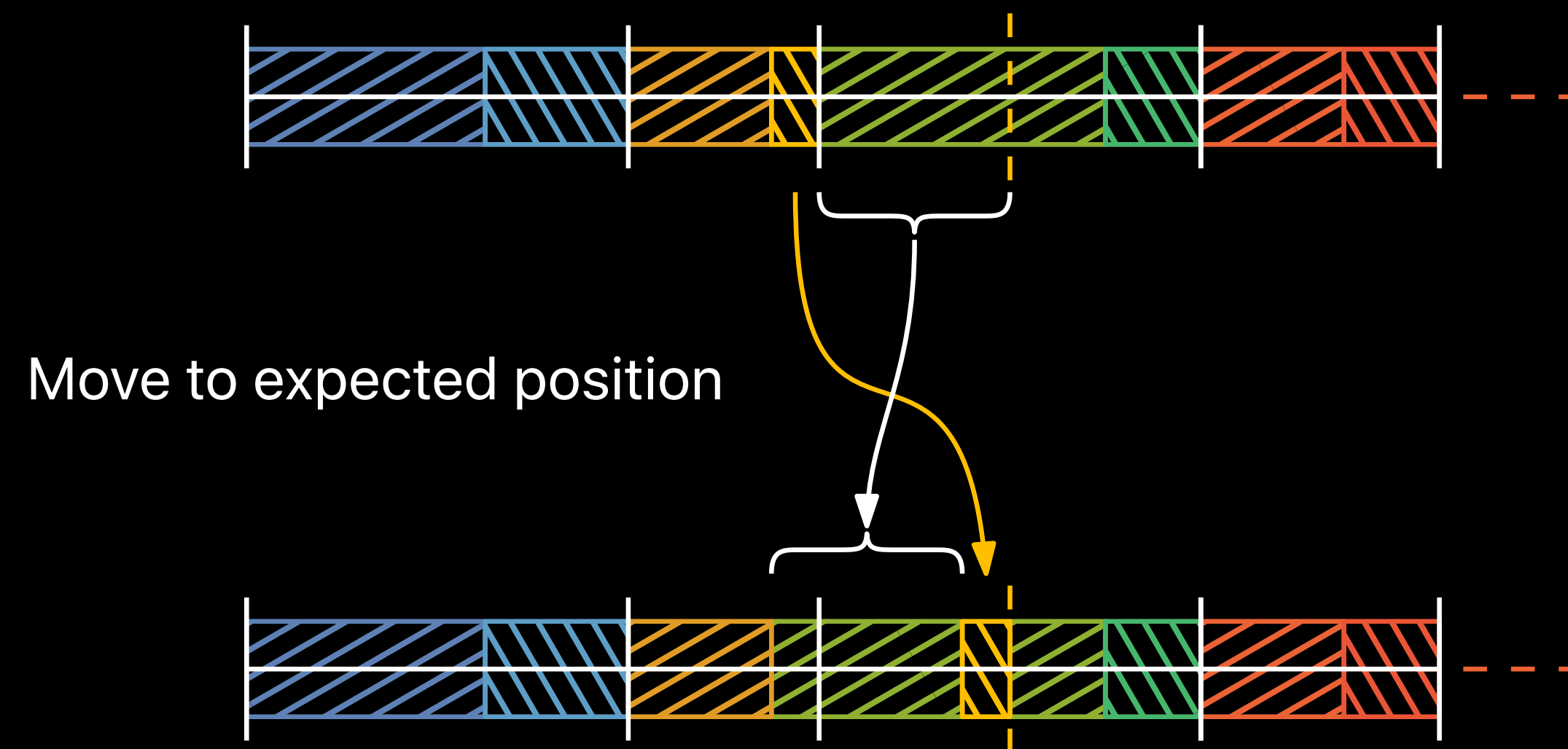
11. Octree stream offsets

Issue: Adoption requires fix

Fix concatenation (by adjusting a chunk)



Fix concatenation (by adding padding)



12. In-tree geometry quantisation depth signalling

Forgotten adoptions

Software release notes (v9.0-rc1)

m52521: Signalling quantisation depth on a per-level basis.

This adoption fundamentally conflicts with another higher-priority adoption (m52400 planar mode interaction fix) made at the same time.

This topic should be re-reviewed at the next meeting.

Was not re-reviewed. However, the adoption of m53677 removed the conflict.

Question: Should we do anything?

Response: Maintain adoption, if AHG finds any issue, delete.

13. Sign bit values

Consistency

- Sign flag indicates if a signed value is positive or negative
- Some uses are inconsistent (se(v) vs. flags)
 - Original intention was $\text{sign} = 1$ to indicate negative value: $(1 - 2 \times \text{sign}) \times \text{value}$
- Sign bit should occur at end of value, not immediately after foo_abs_gt0_flag
 - Simplifies description of syntax element binarisation

Response: a) use $\text{sign}=1$ for negative values, b) put sign bit at end of syntax element

