

The background is a solid red color. It is decorated with various digital and network-related motifs. On the left side, there are faint, semi-transparent icons of a code editor with a '</>' symbol, a waveform graph, and a circular target-like graphic. In the center, there are faint binary digits (0s and 1s) and some illegible text fragments. On the right side, there is a prominent network diagram consisting of numerous white dots (nodes) connected by thin white lines, with a bright white starburst effect at one of the nodes.

AHG17: Removal of Bricks

JVET-P1004

**T. Hellman, W. Wan, M. Zhou, B. Heng,
P. Chen**

Geneva, October 2019 meeting

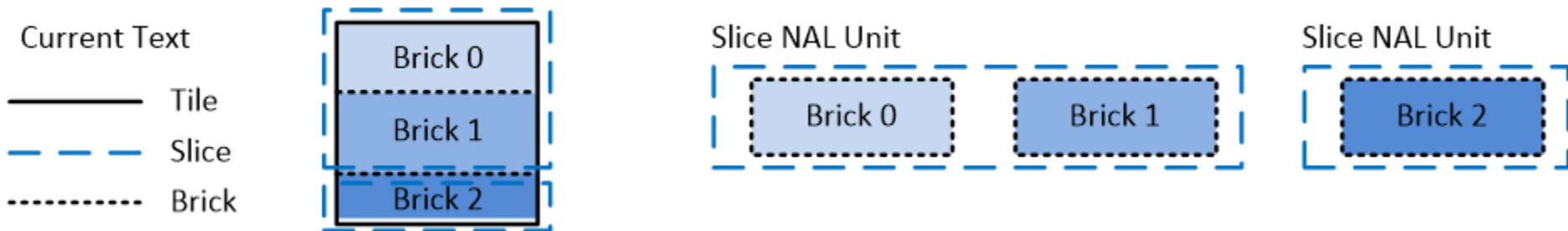


Introduction

- What is the motivation behind multi-brick slices within tiles?
- Contribution claims the primary benefit of multi-brick slices is partitioning
- Equivalent partitioning functionality is already provided by rectangular slices
- Propose to remove brick definitions because:
 - Additional layer of brick partitioning appears to be unnecessary
 - Specification easier to understand

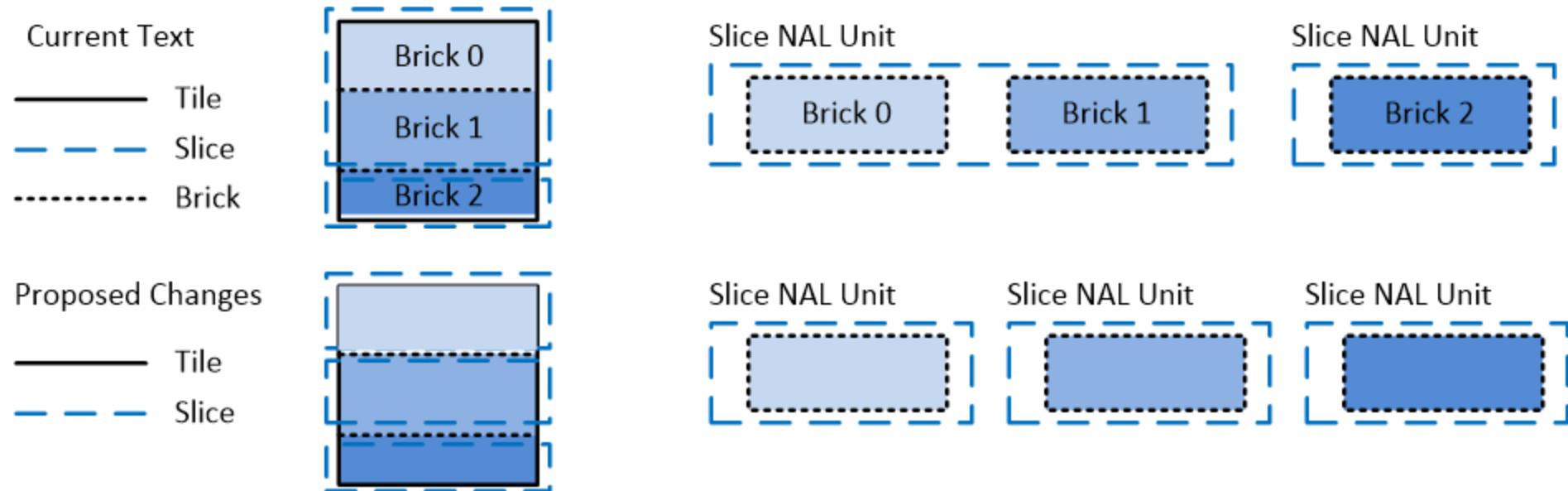
Multi-brick slices

- A slice consists of either a number of complete tiles or only a consecutive sequence of complete bricks of one tile.
- Example of slices composed of bricks in one tile:



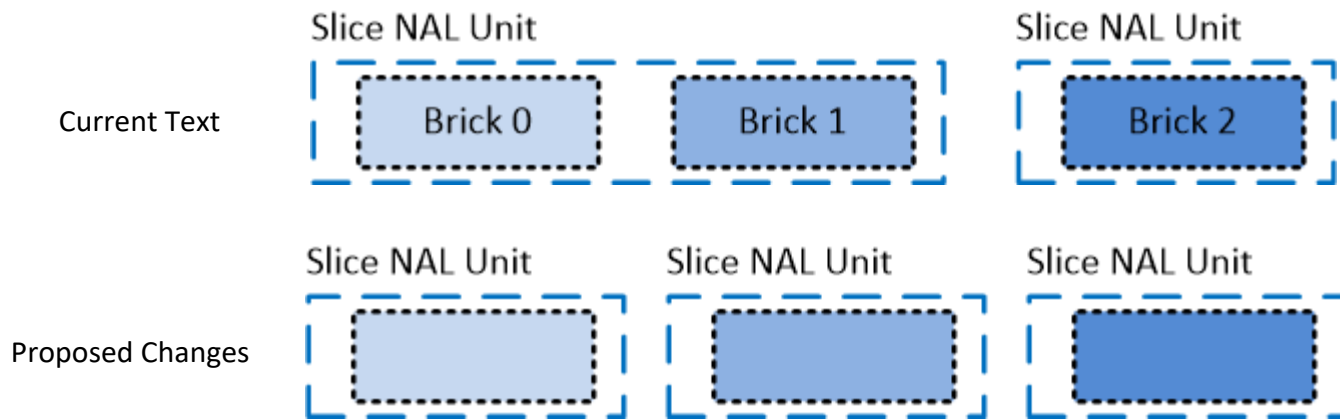
Rectangular slices

- Rectangular slices represent a superset of brick partitioning
- No loss in partitioning options if each brick is its own slice



Brick benefits besides partitioning?

- Note bricks inside a multi-brick slice are constrained to be within the same (slice) NAL unit
- Unclear what is the benefit of defining a brick partitioning layer



Proposal

- Modify the concepts in the text defined using bricks to use slices instead
- Requested by group to provide specification text changes incorporating elements in JVET-P0096 as well as concepts in JVET-P0240
 - JVET-P0096: Single signalling mode instead of separate uniform and explicit tile/brick spacing modes
 - JVET-P0240: Dimensions and locations of rectangular slices be signalled directly by using tile information rather than bricks.

Changes beyond JVET-P0096 & JVET-P0240-v2

- Minimal syntax changes
 - PPS:
 - loop_filter_across_bricks → loop_filter_across_tiles
 - Slice header:
 - num_bricks_in_slice_minus1 → num_tiles_in_slice_minus1
 - Slice data:
 - end_of_brick_one_flag → end_of_slice_one_bit and end_of_tile_one_bit



BROADCOM[®]

connecting everything[®]