

COMMENTS ON TEST MODEL DEVELOPMENT (JVET-J0081)

M. Zhou, W. Wan, T. Hellman, P. Chen (Broadcom)
O. Hugosson, D. Symes, A. Duenas (ARM)
E. Chai (RealTek)

Recommendations for tool selection for an initial test model

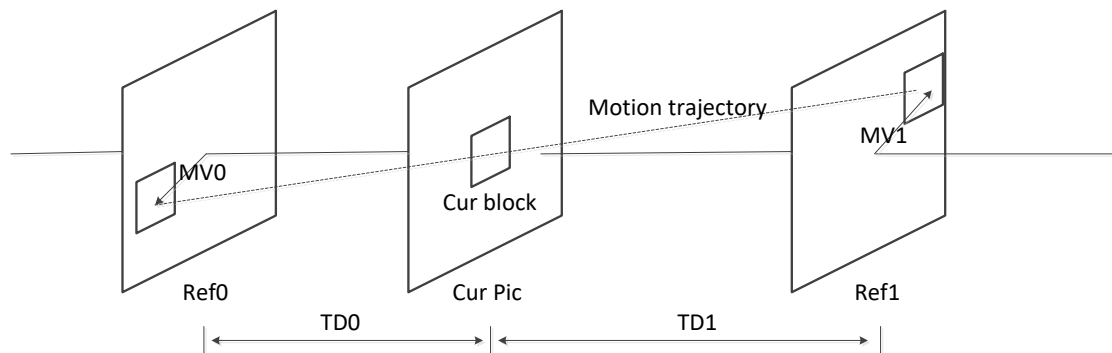
- 1) Quadtree plus binary tree and triple tree (QTBTT) appears to be a good starting point for block partitioning
- 2) Suggest setting the maximum TU size to 64x64 (luma)/32x32 (chroma) as the initial starting point with further study on the impact of larger TU sizes of 128x128 (luma)/64x64 (chroma) in CEs
- 3) Inter coding tools (e.g. template matching in FRUC, uni-directional LIC, diffusion filter for inter prediction) which require access to the reconstructed samples from neighboring blocks have a substantial implementation impact so it is recommended not to include them in the test model
- 4) Neural Network (e.g. CNN) based tools have a substantial implementation impact for a decoder so it is recommended to further study the tradeoff between coding efficiency and complexity before deciding whether to include them in the test model

Inter Coding Tools using Neighboring Reconstructed Samples

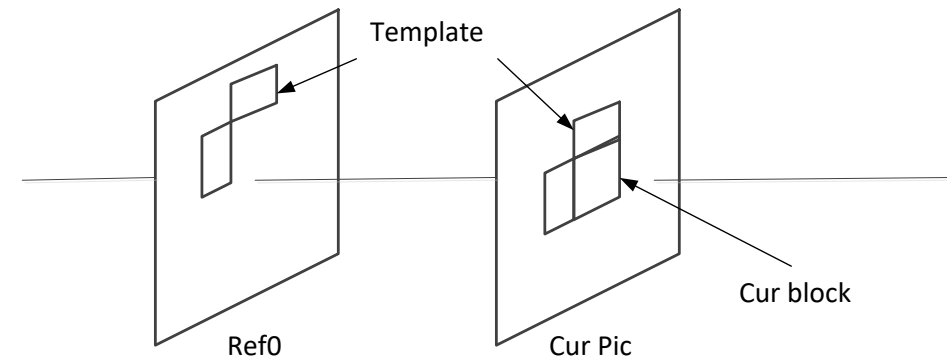
1. There several coding tools in the CfP responses which require access to the reconstructed samples from left and top neighboring blocks
2. Some of such examples are
 - 1) Template matching in FRUC for uni-prediction
 - 2) Template matching based MV candidate reordering
3. We see those tools as no-goers, since they break the decoder pipeline architecture
 - 1) The high latency dependency loop prevents an efficient implementation
 - 2) No high-throughput decoder (e.g. 4K@60) possible w/ those tools

Example: Pattern Matched Motion Vector Derivation (FRUC)

- Bilateral matching uses only list0/list1 reference blocks
- Uni-directional template matching accesses reconstructed samples from neighboring blocks
 - Reconstructed samples of top/left neighboring blocks may not be available due to pipeline delay



Bilateral matching

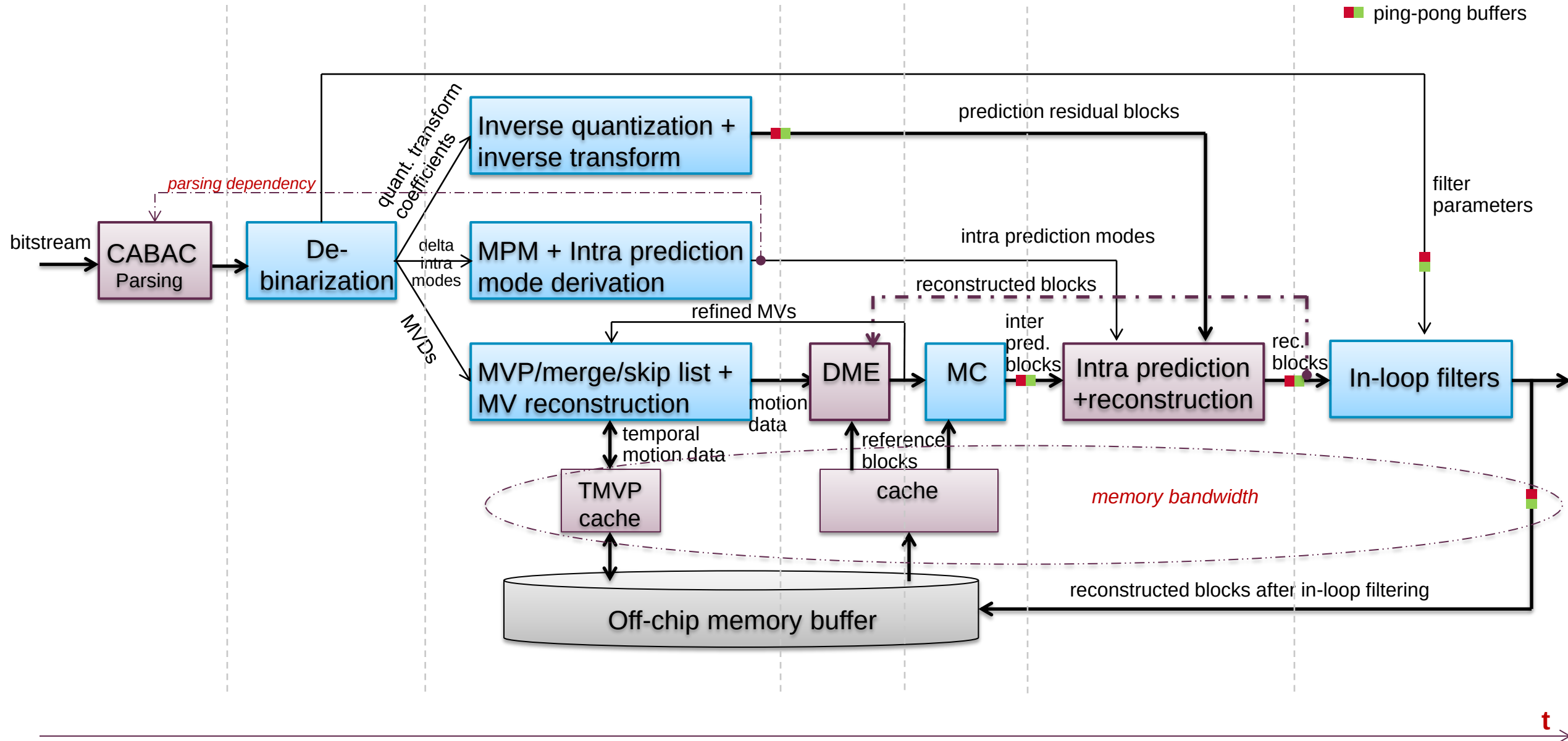


Template matching

Maximum and Minimum TU size

1. TU size up to 256x256 and down to 4x1/1x4 has been seen in CfP responses
2. The maximum TU size dictates the local buffer size of decoder HWAs (Hardware Accelerators)
3. HEVC has 32x32 (luma)/16x16 (chroma) maximum TU size, which enables 32x32 based decoder pipeline
4. 256x256 TU will increase decoder local buffer size by 64x
5. 256x256 transform itself is very costly
6. Small TU and intra blocks are likely to cause throughput issue

Decoder Partitioning w/o Template Matching



Decoder Partitioning w/ template matching

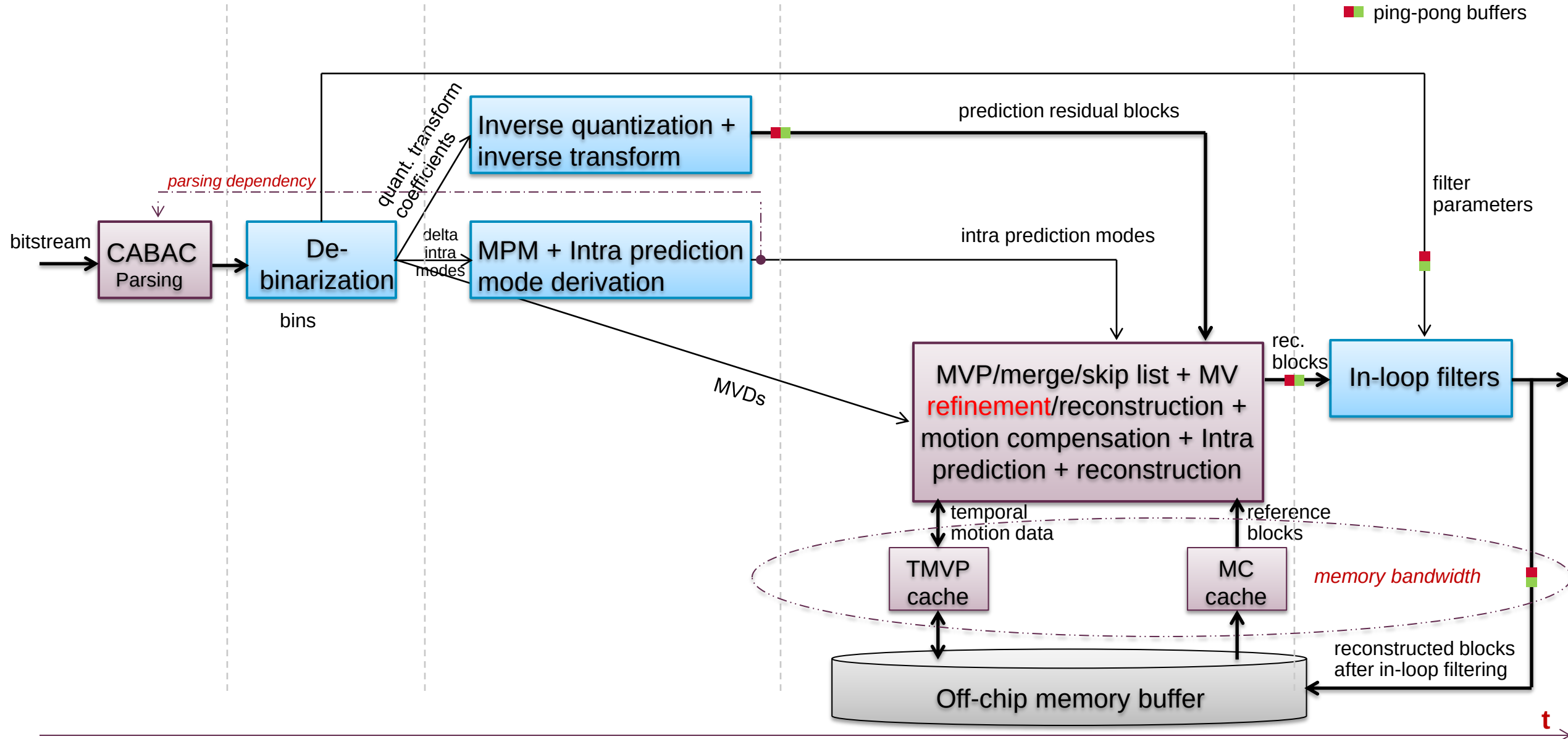


Diagram of HEVC Decoder

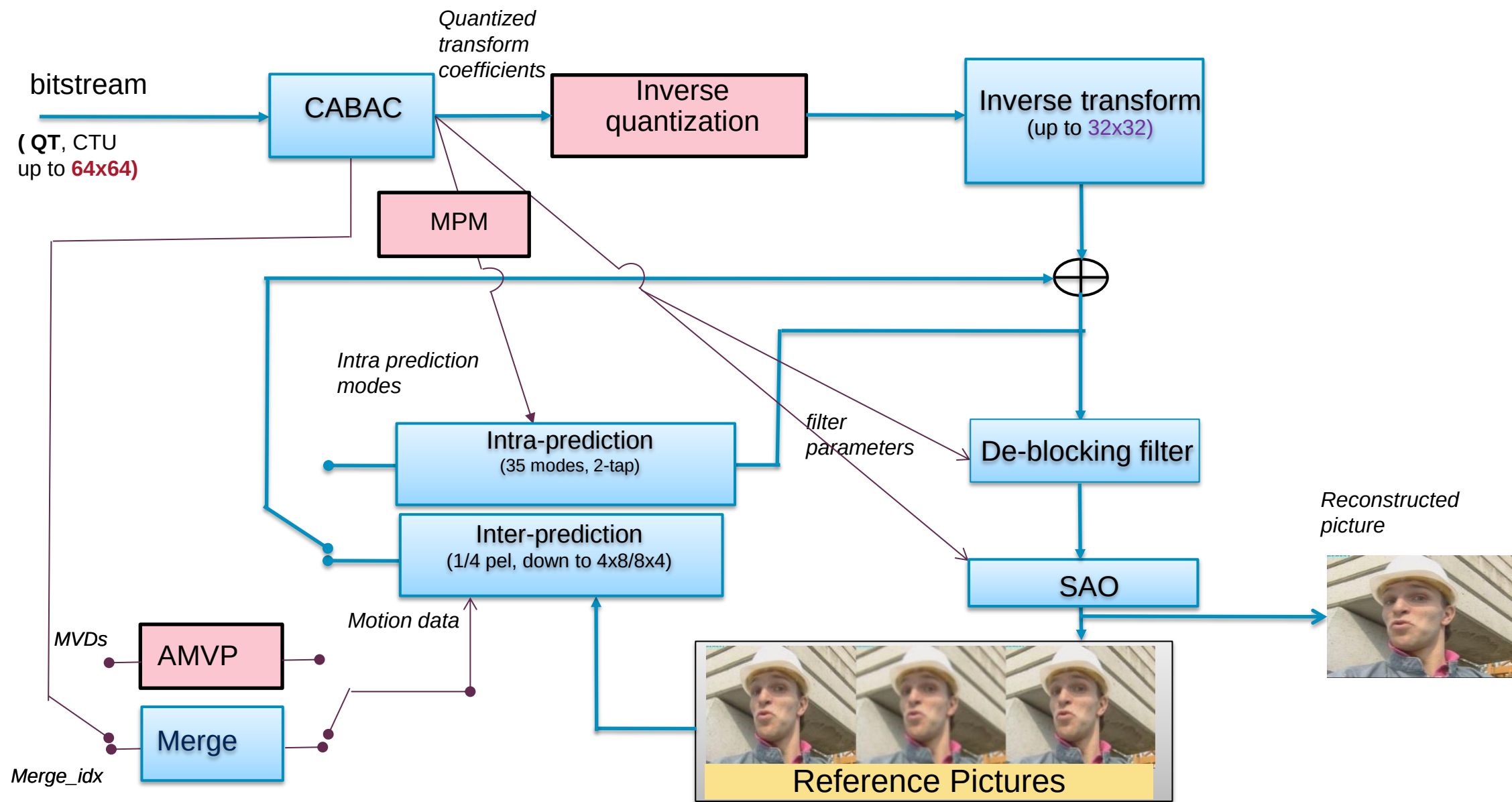
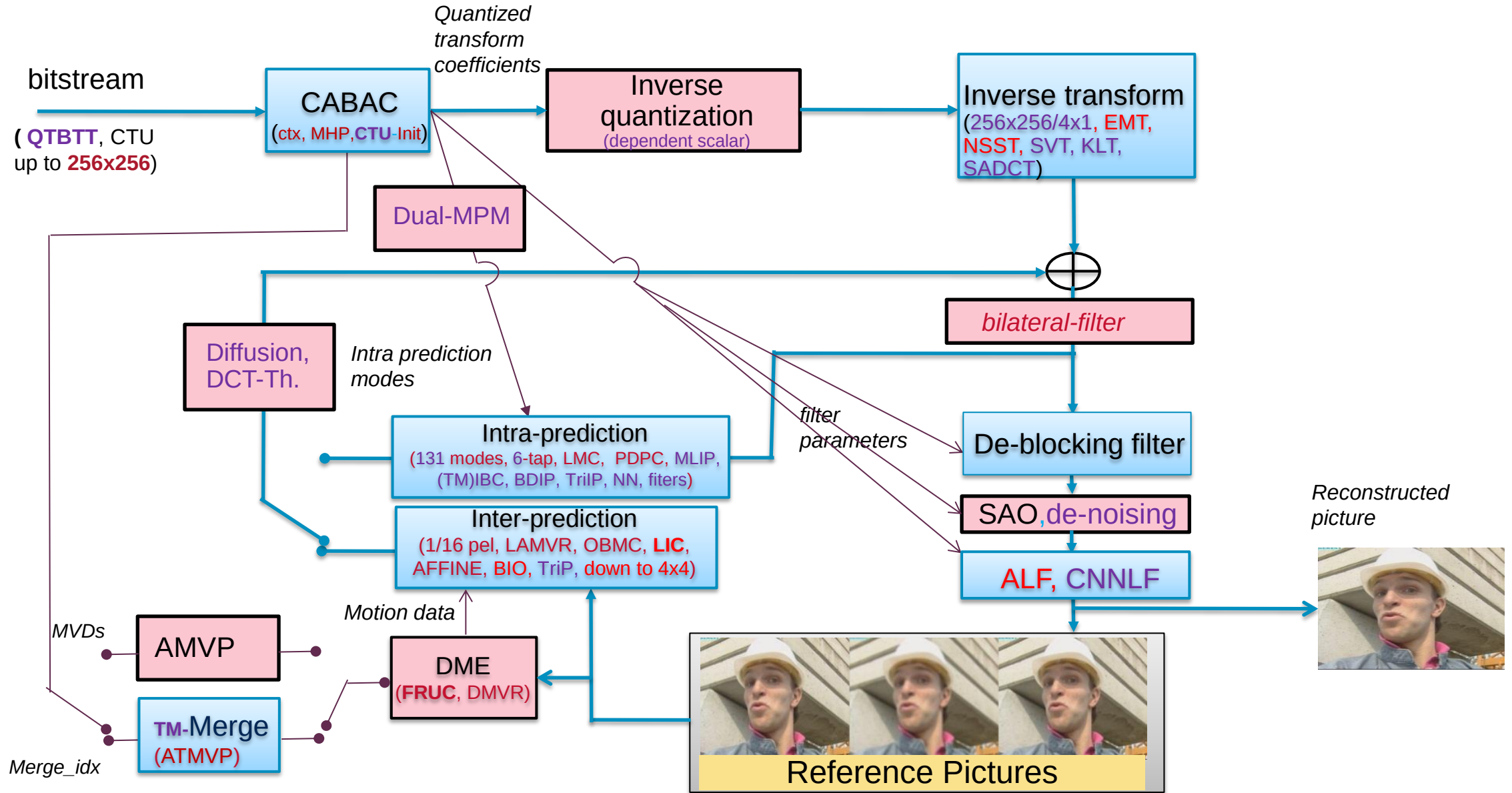


Diagram of Video Decoder w/ CfP Tools



Comments on the long term standardization development (1/2)

1. Small **transform and intra prediction block** sizes less than 4x4 should be avoided for both luma and chroma
2. The MPM list derivation may need to be decoupled from **CABAC** parsing to avoid parsing dependency issues limiting throughput.
3. CABAC context initialization from previous frames should be carefully studied as it makes it harder to parallelize across multiple cores.
4. New coding tools inside the **intra prediction critical loop**, such as bilateral filter, multiple line intra prediction, PDPC + planar, LMC, boundary filters smoothing/sharpening filters, and etc., should be closely studied as there is concern over throughput and/or implementation cost issues with these tools.

Comments on the long term standardization development (2/2)

5. **Decoder side motion refinement and inter prediction tools**, such as bilateral matching in FRUC, DMVR, BIO and bi-prediction LIC, should be closely studied as there is concern over memory bandwidth consumption, throughput and/or implementation cost with these tools. For example,
 - Block to block serial dependency and memory latency issues for bilateral FRUC needs to be seriously investigated
 - Memory latency caused by the feedback of refined motion vectors to the MV reconstruction process also needs investigation
6. 4x4 based motion compensation such as currently being done in JEM7.0 is a memory bandwidth concern.
7. Non-separable 8x8 secondary transforms is another significant cost issue that should be addressed.