

ABSTRACT**EXPLOITING FRAME TO FRAME COHERENCY IN A SORT-MIDDLE ARCHITECTURE**

Pixel values that were computed in a previous frame may be reused for the current frame, operating in a sort-middle architecture. A hash or some other compact representation of all the data used in a tile, including all triangles, uniforms, textures, shaders, etc. is computed and stored for each tile. When rendering the next frame, that compact representation is once again computed for each tile. In a sort-middle architecture, there is a natural break point just before rasterization. At this break point, the compact representation may be compared to the compact representation computed in the previous frame for the same tile. If those compact representations are the same, then there is no need to render anything for this tile. Instead, the contents of the color buffer or other buffers of the previous frame of the tile may be moved to the same buffer of the tile for the current frame.

Fig. 2