Work Graphs based Denoising for Real-Time Ray Tracing

Felix Kawala, Michael Wittmann, Fabian Wildgrube, Paul Trojahn, Dominik Baumeister, Matthäus Chajdas, Richard Membarth
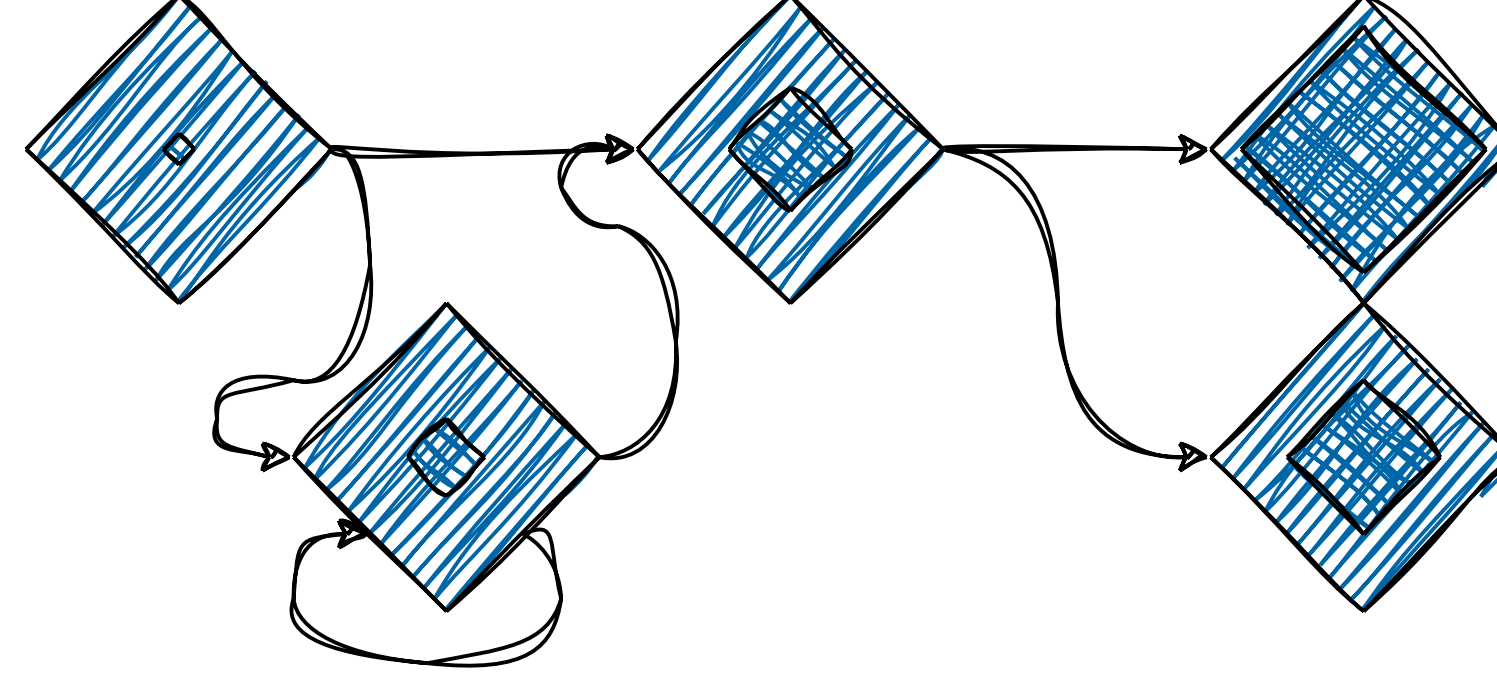
Work Graphs

D3D12 Work Graphs is a new API for GPU autonomy.

D3D12 Work Graphs was recently introduced to DirectX to simplify the use of highly variable complex pipelines.

A graph containing a program is constructed on the CPU and dispatched to the GPU. From that point, the GPU can independently and dynamically schedule all work for the program.

Benefits:

- Enables dynamic scheduling entirely on the GPU
- Introduces new options for cache reuse
- Reduces launch overhead
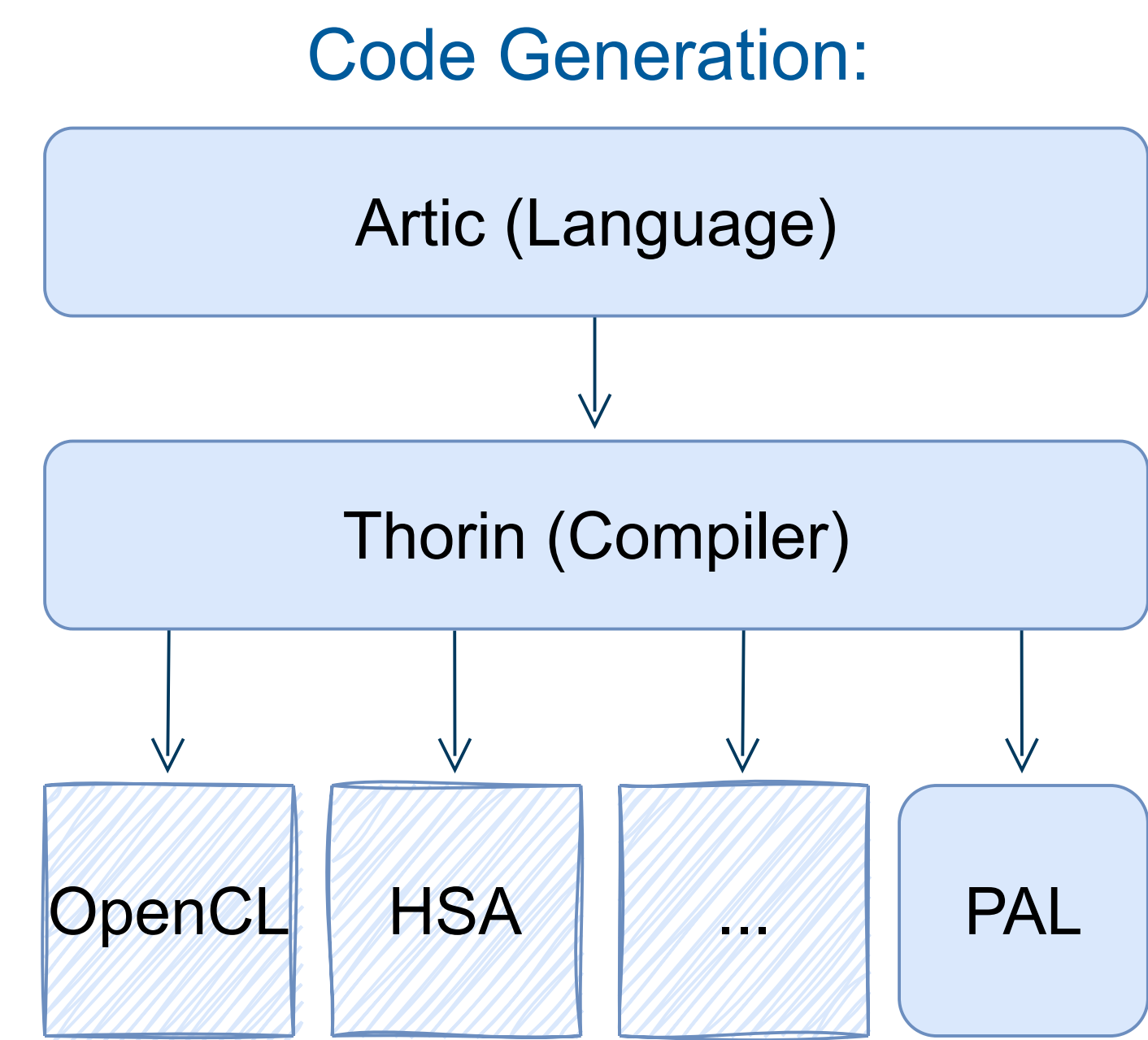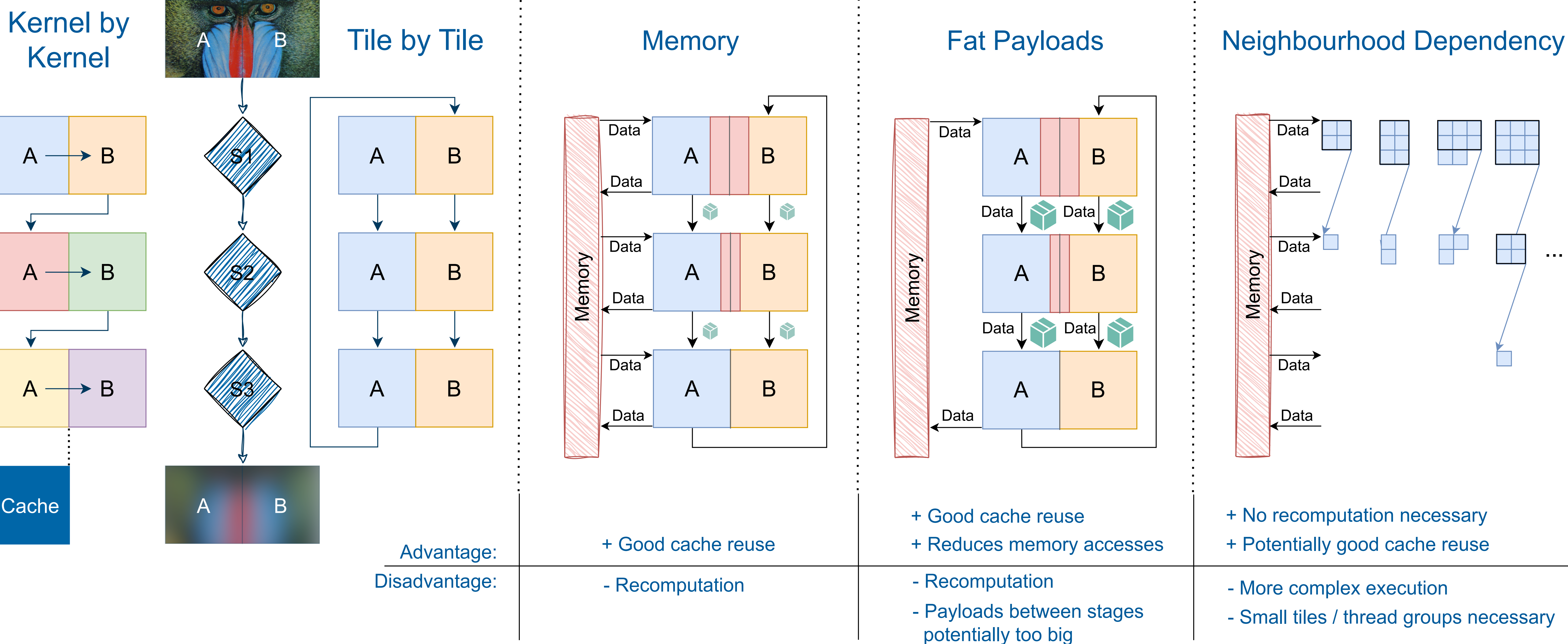- Simplifies development

AnyDSL

We leverage the research compiler/language AnyDSL with Work Graphs support.

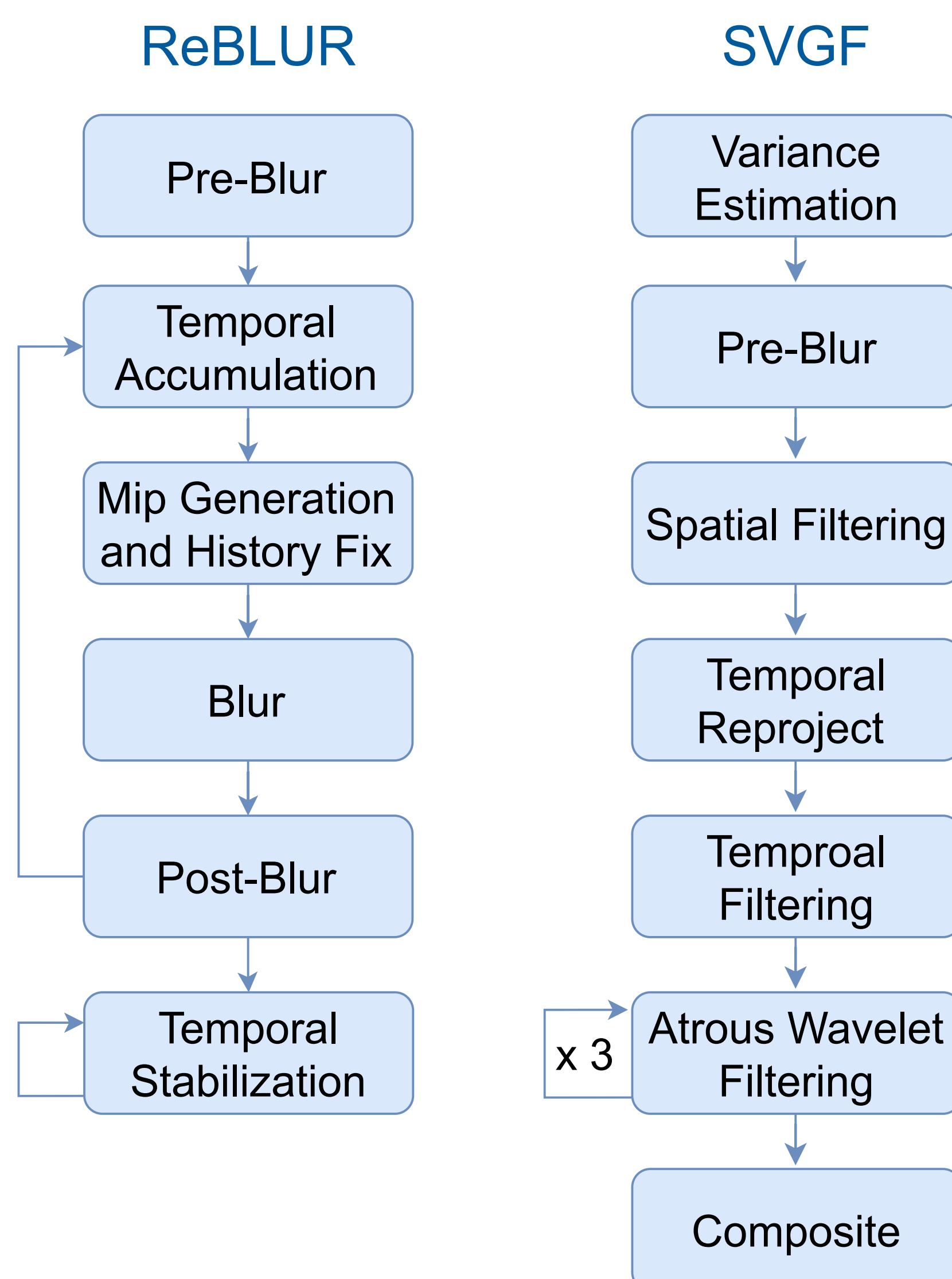This allows us to use a single language to write the CPU and GPU code.

AnyDSL provides Work Graphs support on AMD GPUs via PAL.

Code Generation:

Artic (Language) → Thorin (Compiler) → OpenCL | HSA | ... | PAL

Cache Reuse utilizing a Tile by Tile Schedule

Kernel by Kernel

Cache

Tile by Tile

S1 → S2 → S3

Memory

Advantage: + Good cache reuse

Disadvantage: - Recomputation

Fat Payloads

+ Good cache reuse
+ Reduces memory accesses

- Recomputation
- Payloads between stages potentially too big

Neighbourhood Dependency

+ No recomputation necessary
+ Potentially good cache reuse

- More complex execution
- Small tiles / thread groups necessary

Work Loads

ReBLUR

Pre-Blur → Temporal Accumulation → Mip Generation and History Fix → Blur → Post-Blur → Temporal Stabilization

SVGF

Variance Estimation → Pre-Blur → Spatial Filtering → Temporal Reproject → Temproal Filtering → (x 3) Atrous Wavelet Filtering → Composite

Idea

Implement real-time denoising for ray tracing as a workload.

Use tile-based scheduling approaches.

Improve performance through better cache reuse.

To illustrate:

src: "ReBLUR: A Hierarchical Recurrent Denoiser" Figure 49-1

Preliminary Results

SVGF: Kernel by Kernel
Total Execution Time [µs]

Windows Reference	AnyDSL Compute	AnyDSL Work Graphs
4065	3569	4421