Profiling GPU Shaders for Profile-Guided Optimizations

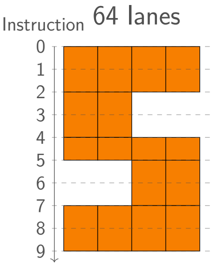
Sebastian Neubauer
Technische Universität München

July 25, 2019

GPUs

Hardware

- SIMD-units with 64 lanes
- Diverging control flow by masking lanes (SIMT)
- AMD Radeon VII has 240 SIMD units



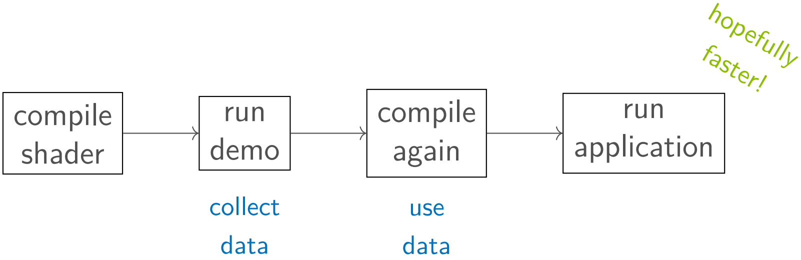
Vulkan

Software

- Graphics and compute standard for GPUs
- Shaders are loaded in SPIR-V
- Compilation to ISA happens in driver

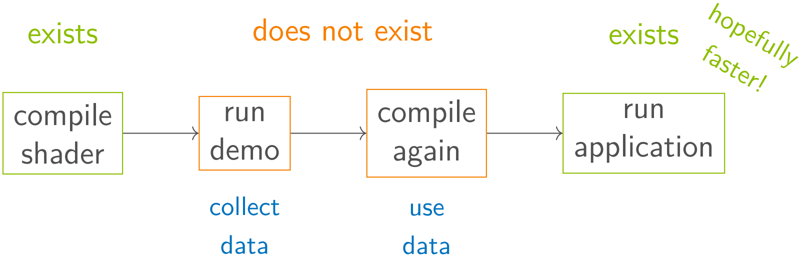
Workflow

Profile-Guided Optimization



Current State

Profile-Guided Optimization



exists

does not exist

exists

hopefully faster!

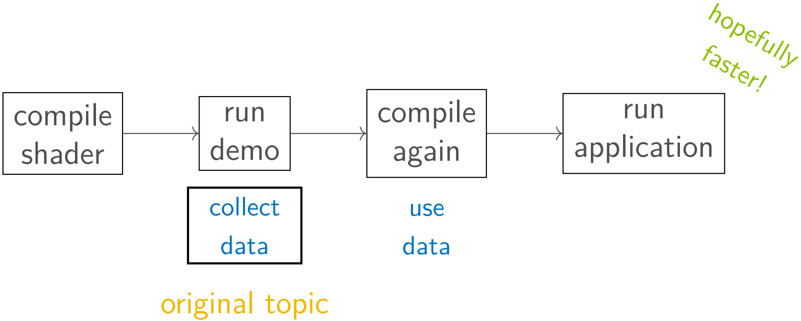
compile shader → run demo → compile again → run application

collect data

use data

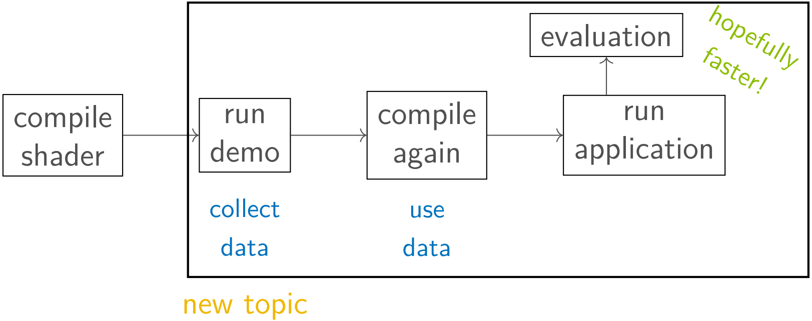
This Thesis

Profile-Guided Optimization



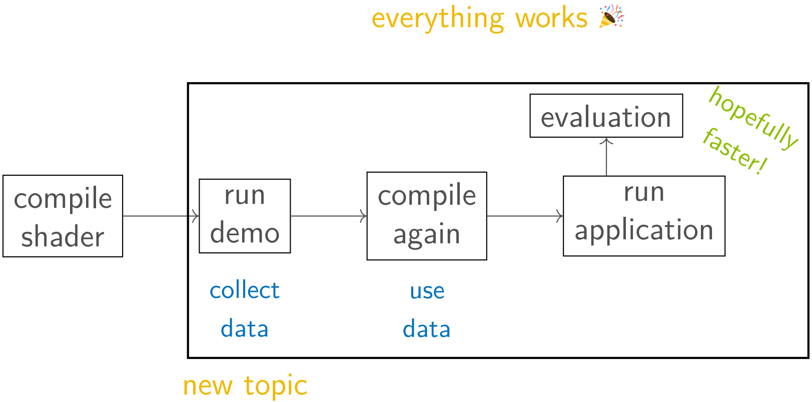
This Thesis

Profile-Guided Optimization

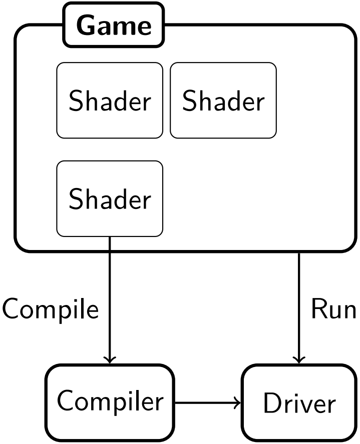


This Thesis

Profile-Guided Optimization

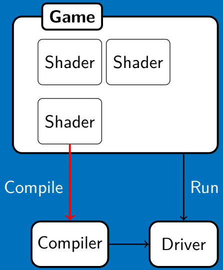


Basic Block Counting



Basic Block Counting

GLSL/SPIR-V

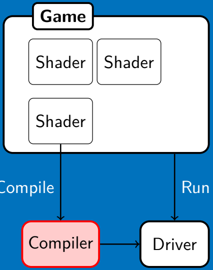


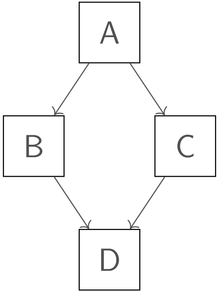
- GLSL gets precompiled to SPIR-V
- SPIR-V is passed to driver

```
if (inputPos.x < 0.5) {
    outColor = vec4(1.0, 0.0, 0.0, 1.0);
} else {
    outColor = vec4(0.0, 0.0, 1.0, 1.0);
}
```

Basic Block Counting

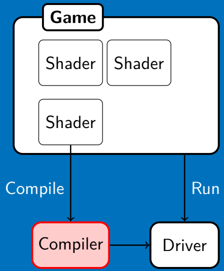
CFG





Basic Block Counting

CFG



Game

Shader

Shader

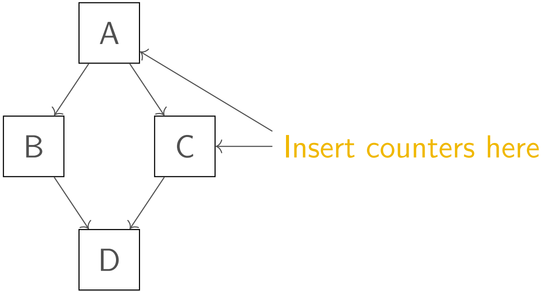
Shader

Compile

Run

Compiler

Driver



A

B

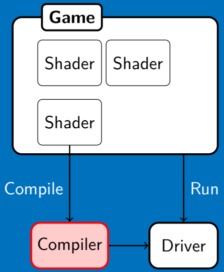
C

D

Insert counters here

Basic Block Counting

CFG


Game

Shader

Shader

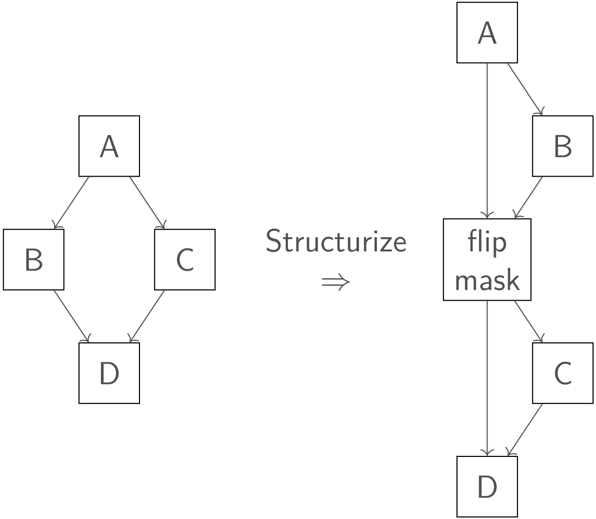
Shader

Compile

Run

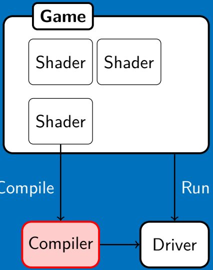
Compiler

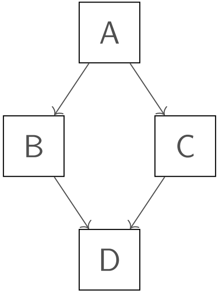
Driver



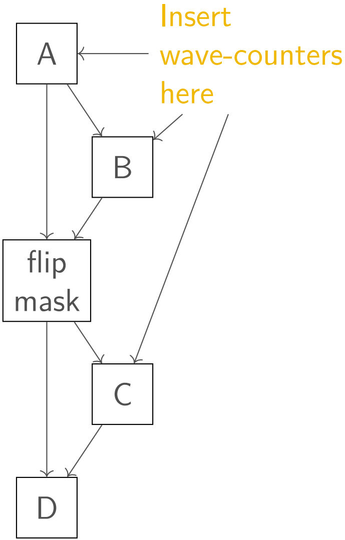
Basic Block Counting

CFG



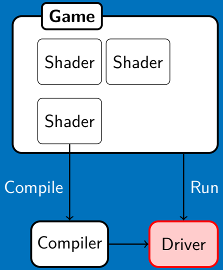


Structurize ⇒



Basic Block Counting

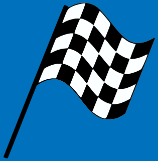
Save Counter



- Counters are saved when the pipeline is destroyed and every 10 s
- Fetch counters from GPU memory
- Write counters and metadata to file

Basic Block Counting

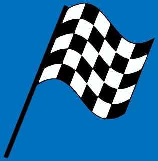
Result



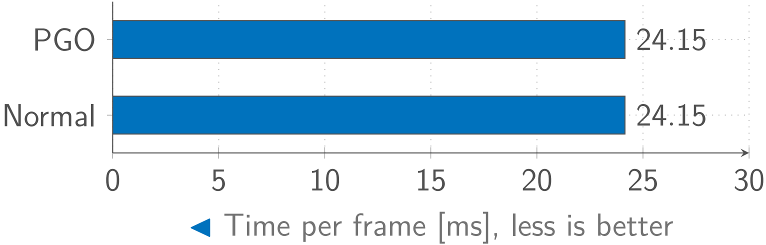
- Declares pixel shader as *hot* and vertex shader as *unlikely*
- Changes basic block ordering

Basic Block Counting

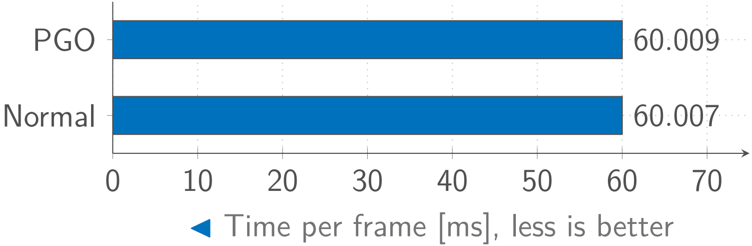
Result



Dota 2: No change apart from compilation speed on first run



Ashes of the Singularity: (60.007 ± 0.004) ms vs (60.009 ± 0.003) ms



My Work

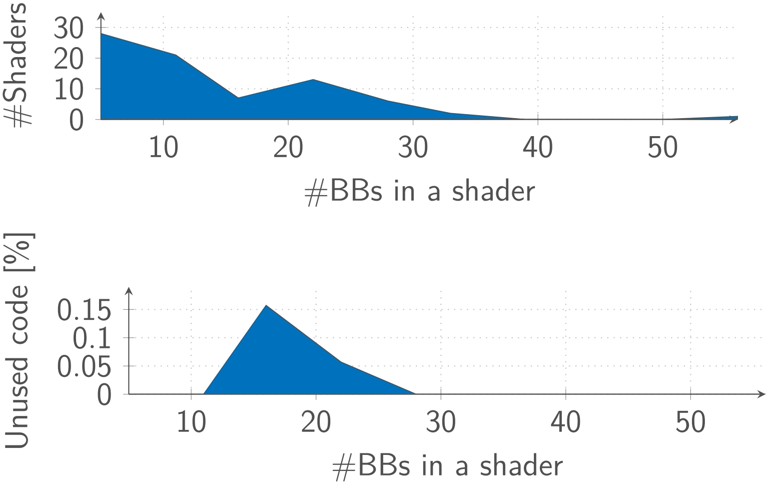
- Enable atomic basic block counters in LLVM
- Implement ELF loading and relocations in AMDVLK
- Write result files from driver
- Apply PGO per wave instead of per thread
- Fix bugs in LLVM (with PGO on GPUs)

Future Work

- Find dynamically uniform variables
- Create some interesting statistics, e.g. unused basic blocks, uniform branches
- More benchmarks
- (More optimizations)

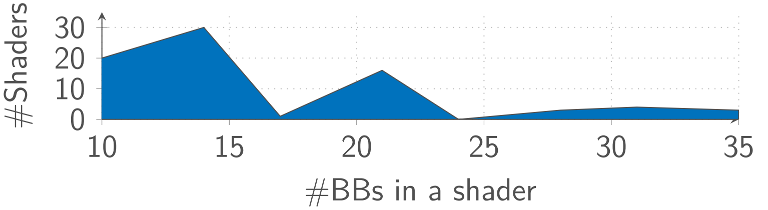
Dead Code

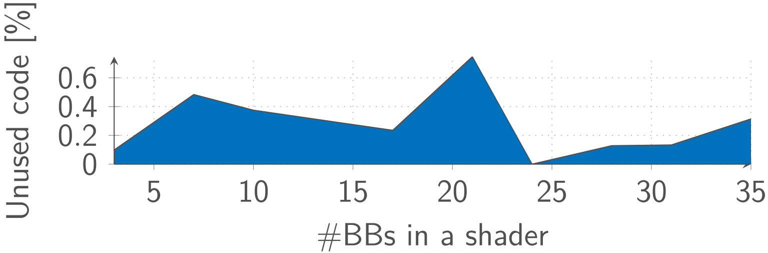
Ashes of the Singularity



Dead Code

Dota





Questions?