

Vulkan: Graphics and Compute Compilation on GPU

Time: 8:00am-12:00pm 2/16/2019

Vulkan is a new industry standard for graphics and compute API that aims to support high efficiency, cross-platform access to modern GPUs ranging from PCs and consoles to mobile phones and embedded platforms. Vulkan has been presented in various computer graphics and game conferences, but has limited exposure in systems and compiler conferences.

This tutorial will first introduce the Vulkan API and SPIR-V shader language. It will next discuss GPU compiler performance techniques for both graphics and compute, especially for mobile GPUs. We will attempt to make comparisons and contrasts of the concepts in Vulkan and GPU compiler with existing work in various domains, and share our experience in an industry implementation of mobile graphics compiler.

Most coverage of GPU in systems or compiler conferences are about *General-Purpose GPU* (GPGPU). This tutorial wants to introduce more attention to general graphics on GPU. A major goal (and our earnest hope) of this tutorial is to inspire new inter-disciplinary systems and compiler research beyond GPGPU by introducing general graphics and Vulkan's development to a broader community outside the computer graphics crowd.

The tutorial will not cover game development nor computer graphics techniques.

Vulkan API and SPIR-V shader language – General introduction	Alexander Bakst
Compiling for graphics – Performance techniques and issues	Chu-Cheow Lim
Compiling for compute – Performance techniques and issues	Chunling Hu

Prerequisite knowledge:

- Basic understanding of computer graphics pipeline
- General knowledge of CPU compiler

Special requirement: NA

Biography:

Chu-Cheow Lim is a Director of Engineering at Qualcomm and leads a distributed performance / verification team in the GPU compiler team. His prior work includes CPU compiler optimizations and backend, research on speculative multithreading and development of a GPGPU C-for-media compiler. His research interests are in high performance computing, compilers and performance topics in various application domains. He received his PhD from University of California at Berkeley and MSc and BSc from Stanford University.

Ruihao Zhang is a Director of Engineering at Qualcomm and leads compiler development to support different graphics and compute APIs, such as Vulkan, OpenGL ES, D3D and OpenCL, in the GPU compiler

team. His prior work includes code generation, compiler optimizations, logical verification and debugger support for embedded systems. He received his MS from Purdue University and BS from Tsinghua University.

Chunling Hu is a software engineer of the GPU compiler team at Qualcomm. Her work focuses on graphics/compute performance on mobile GPUs. Her previous work includes research on runtime techniques (task scheduling, power saving, etc.) for high performance computing on heterogeneous systems and development of C-for-media compiler for GPU. She received her PhD in Computer Science from Rutgers University and BS from Beijing University of Posts&Telecommunications.

Alexander Bakst is a software engineer in the GPU compiler team at Qualcomm. His work focuses on support for the Vulkan API. He received his PhD in Computer Science from the University of California, San Diego, where he studied algorithmic software verification. He received his BS and MEng from the Massachusetts Institute of Technology.