A3VK: Vulkan in Animal3D

Ben Hubner
Cameron Schneider

Advanced Rendering
SP2021

Project Goals

Rendering

- Recreate existing OpenGL functionality
- Create easy-to-use Vulkan wrapper for rendering functionality
- **Optimize passes**

Data Abstraction & Interfacing

Abstraction

- Abstract VK data types & functions
- **Wrap VK informational structs**

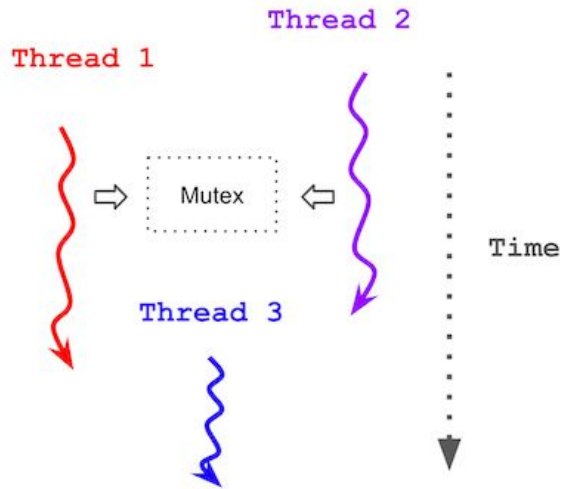
Interfaces

- A3D Types to VK types
- Resource loading and setup
- **Per-Plugin API selection**

Advanced Features

GPU Multithreading

- VK offers Multithreading
  - Very Difficult
- Only makes an impact in certain situations
- Could be used better for computation tasks



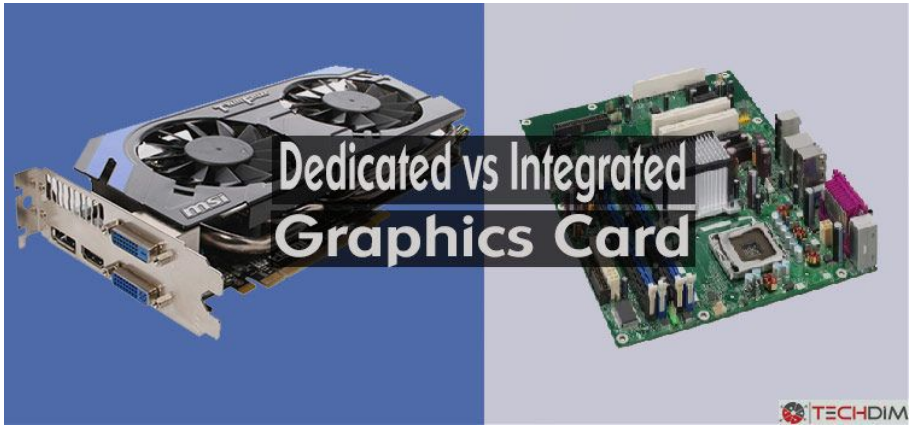
Compute Tasks

- Exposes GPU to general task running
- Lots of control over hardware
- Lots of parallelization opportunities

Multi-GPU Computing

- Vulkan setup allows access to all graphics devices

- Individual devices running tasks
  - Hardware restrictions



Applications & P.O.I

Physics & Simulation

- Compute shaders & matrix buffers
- A parallelized, GPU accelerated physics solver interface
- Optimized GPU data uploading via octree matrices & parallelized upload thread

Compute Tasks & Management (non-rendering)

Tasks

- Post-processing passes via compute
- Pre-vertex shader matrix setup in batches
- Physics computations?

Job System

- Manage execution of tasks
- Organize task execution into priority queue
  - Need to manage resources based on device
- Can user determine execution priority?

Questions