



1


Transition: CUDA ↔ OpenCL



Oregon State University
Mike Bailey
mjb@cs.oregonstate.edu



This work is licensed under a [Creative Commons Attribution-NonCommercial-NoDerivatives 4.0 International License](https://creativecommons.org/licenses/by-nc-nd/4.0/)



Oregon State University
Computer Graphics

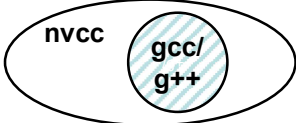
transition_cuda_opengl.pptx mjb - May 8, 2022

1

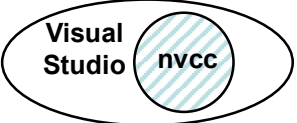
2

CUDA Summary


- CPU and GPU programs exist in the same file
 - Can share #defines
 - Can share information on the GPU function calling sequence
- Nvidia-only
- Much utility code provided (linear algebra, machine learning, etc.)
- Well-respected in the research community
- Need special compiler options




Linux



Windows



Oregon State University
Computer Graphics



Oregon State University
Computer Graphics

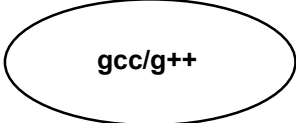
mjb - May 8, 2022

2


3

OpenCL Summary


- CPU and GPU programs exist in separate files
 - Must be sure to set #defines the same
 - You must provide information on the GPU function calling sequence
- Runs on Nvidia GPUs, AMD CPUs/GPUs, Intel CPUs/GPUs, FPGAs, ...
- Little utility code provided
- Code looks a lot like GLSL compute shader code
- Well-respected in the production community
- Need no special compiler options (GPU code compiled in the driver)




Linux



Windows



Oregon State University
Computer Graphics



Oregon State University
Computer Graphics

mjb - May 8, 2022

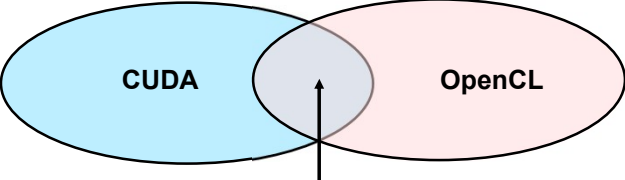
3

4


What's Unique and What's Common?

- CPU and GPU programs exist in the same file
 - Can share #defines
 - Can share information on the GPU function calling sequence
- Nvidia-only
- Much utility code provided (linear algebra, machine learning, etc.)
- Well-respected in the research community
- Need special compiler options


- CPU and GPU programs exist in separate files
 - Must be sure to set #defines the same
 - You must provide information on the GPU function calling sequence
- Runs on Nvidia GPUs, AMD CPUs/GPUs, Intel CPUs/GPUs, FPGAs, ...
- Little utility code provided
- Code looks a lot like GLSL compute shader code
- Well-respected in the production community
- Need no special compiler options (GPU code compiled in the driver)



- Allocate data space in GPU memory
- Transfer data from CPU to GPU
- Execute a kernel to compute on that data
- Transfer data back from the GPU to the CPU



Oregon State University
Computer Graphics



Oregon State University
Computer Graphics

mjb - May 8, 2022

4