

# Uni.lu HPC School 2020

## PS7: Introduction to GPU programming with CUDA (Part I)

High Performance  
Computing &  
Big Data Services



Uni.lu High Performance Computing (HPC) Team

F. Pinel

University of Luxembourg (UL), Luxembourg

<http://hpc.uni.lu>



## Latest versions available on Github:



UL HPC tutorials:

<https://github.com/ULHPC/tutorials>

UL HPC School:

<http://hpc.uni.lu/hpc-school/>

PS7 tutorial sources:

[ulhpc-tutorials.rtf.d.io/en/latest/cuda/](http://ulhpc-tutorials.rtf.d.io/en/latest/cuda/)





# Objectives

- Minimum CUDA knowledge to start development
  - ↳ Single GPU, Unified Memory
- On the **iris** cluster
  - ↳ interactive and passive jobs
- Error handling
- Profiling

## Out of scope (but useful)

- Streams
- Multi-GPU
- Profiling for performance improvement (information available)
- Fine-tuning memory access

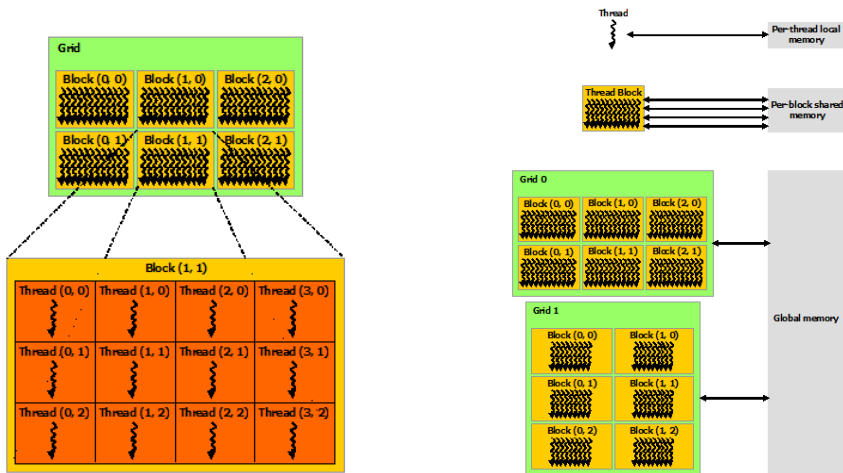
# Software overview

GPU Computing Applications					
Libraries and Middleware					
cuDNN TensorRT	cuFFT, cuBLAS, cuRAND, cuSPARSE	CULA MAGMA	Thrust NPP	VSIPL, SVM, OpenCurrent	PhysX, OptiX, iRay  MATLAB Mathematica
Programming Languages					
C	C++	Fortran	Java, Python, Wrappers	DirectCompute	Directives (e.g., OpenACC)
CUDA-enabled NVIDIA GPUs					
Turing Architecture (Compute capabilities 7.x)	DRIVE/JETSON AGX Xavier	GeForce 2000 Series	Quadro RTX Series	Tesla T Series	
Volta Architecture (Compute capabilities 7.x)	DRIVE/JETSON AGX Xavier			Tesla V Series	
Pascal Architecture (Compute capabilities 6.x)	Tegra X2	GeForce 1000 Series	Quadro P Series	Tesla P Series	
Maxwell Architecture (Compute capabilities 5.x)	Tegra X1	GeForce 900 Series	Quadro M Series	Tesla M Series	
Kepler Architecture (Compute capabilities 3.x)	Tegra K1	GeForce 700 Series GeForce 600 Series	Quadro K Series	Tesla K Series	
	EMBEDDED	CONSUMER DESKTOP, LAPTOP	PROFESSIONAL WORKSTATION	DATA CENTER	

# Hardware/Software overview



# Thread Hierarchy and Memory overview



# Accelerating Applications with CUDA



NVIDIA

CUDA C++ Programming Guide

Your Turn!

## Hands-on GPU programming with CUDA

► [url](#) ◀ | [github](#) | [src](#)

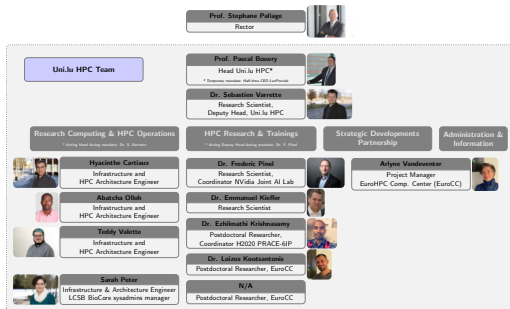
- **Access to a GPU-equipped node** of the Iris cluster si-gpu
  - ↪ Load the CUDA development kit
- **Writing and compile** GPU application nvcc
  - ↪ Launch parallel kernels
- Understanding **CUDA thread hierarchy**
  - ↪ Accelerating for loops
- Understanding **Memory allocation**
  - ↪ application to array manipulation on both the host and device
  - ↪ management of data sets larger than the grid
- **Performance evaluation** and **Profiling** of your GPU application.



Thank you for your attention...

# Questions?

High Performance Computing @ Uni.lu



University of Luxembourg, Belval Campus  
Maison du Nombre, 4th floor  
2, avenue de l'Université  
L-4365 Esch-sur-Alzette  
mail: [hpc@uni.lu](mailto:hpc@uni.lu)

<https://hpc.uni.lu>