



5-day Course on GPU Computing at IT-Designers GmbH

(17–21 October 2016)

Applied Parallel Computing LLC is delivering GPU training courses since 2009. Several dozens of courses have been organized all over Europe, both for commercial and academic customers. We work in close partnership with NVIDIA, CUDA Centers of Excellence and Tesla Preferred Partners. In addition to trainings, our company provides GPU porting/optimization services and [CUDA certification](#).

All corresponding presentations and code samples will be available to attendees from the beginning of each training day.

Hands-ons: All discussed topics will be accompanied with practical sessions, using NVIDIA CUDA 7.5 toolkit. Exercises will be conducted on the remote GPU server accessible from attendees' personal laptops.

Day 1: Introduction to GPU Computing and CUDA

Morning (09:00-12:30)

09:00-10:30: lecture

- An overview of GPU performance in various applications
- Brief intercomparison of different types of accelerators
- Key programming principles to achieve high GPU performance

10:30-12:30: [hands-on session](#)

- Login into laptops
- Understanding GPU compute capabilities, *deviceQuery*
- Overview of CUDA applications development using Eclipse IDE
- Exercise 1: Compile & deploy poissonGL CUDA demo in Eclipse IDE

Afternoon (13:30-17:00)

13:30-15:00: lecture

- CUDA principles and CUDA implementation for C++
- Analogies between MPI+OpenMP and CUDA programming models
- The first CUDA program explained
- CUDA compute grid, examples
- Realistic CUDA application example (wave propagation code)
- Basic optimization techniques

15:00-17:00: [hands-on session](#)

- Exercise 2: Write & deploy a simple CUDA program
- Exercise 3: More control on CUDA compute grid
- Exercise 4: Write & deploy a meaningful image processing tool in CUDA

Day 2: Introduction to OpenACC

Morning (09:00-12:30)

09:00-10:30: lecture

- Advantages of OpenACC.
- Execution model: gangs, workers and vectors – three levels of coarse-grain and fine-grain parallelism. SIMD instructions.
- OpenACC memory model: host and accelerator address spaces.
- OpenACC directive syntax in C and Fortran. Main directives: *parallel* and *kernels* – offloading code regions to accelerator, *loop* – detailed parallelization parameters for each loop.

10:30-12:30: [hands-on session](#)

- PGI compiler for NVIDIA and Radeon GPUs. GCC open-source OpenACC compiler.
- Examples of *vector addition* and *reduction* explained. Comparison to CUDA and OpenMP4.

Afternoon (13:30-17:00)

13:30-15:00: lecture

- Understanding PGI OpenACC compiler output. Compiler flags and environment variables for detailed analysis and performance reports. Compiler limitations in dependencies tracking, enforcing data independence.
- Organizing data persistence regions using *data* directives.
- Data clauses: *deviceptr*, *copy*, *copyin*, *copyout*, *create*, *delete*, *present*, *present_or_(-copy, -copyin, -copyout, -create)*.

15:00-17:00: [hands-on session](#)

- Profiling GPU kernels in OpenACC application. *Time* option, *PGI_ACC_TIME* environment variable. Profiling with *pgprof*.
- **Hands-on:** “fill-in” exercise on implementing wave propagation stencil in OpenACC (*wave13pt*). Adding OpenACC directives step by step.

Day 3: Thrust (STL for GPU) and GPU-enabled libraries

Morning (09:00-12:30)

09:00-10:30: lecture

- Rapid GPU development with Thrust
- Transforms and functors
- Placeholders and tuples
- Performance considerations, optimization
- Interop between Thrust and CUDA C

10:30-12:30: [hands-on session](#)

- Case study: porting interpolation routine to GPUs with Thrust in financial modelling application
- **Hands-on:** “fill-in” exercise on key-value sorting with Thrust (*sort_by_key*)
- **Hands-on:** “fill-in” exercise on implementing custom Thrust transform (*saxpy*)
- **Hands-on:** solving linear system with CUSP library (*cusp_cg*)

Afternoon (13:30-17:00)

13:30-15:30: lecture

- CUBLAS, MAGMA, CUBLAS-XT and CUSPARSE libraries
- AmgX – Introduction into Krylov and multigrid solvers library
- CUFFT and CURAND libraries
- CUDNN – Introduction into Deep Neural Network library

15:30-17:00: [hands-on session](#)

- **Hands-on:** solving Poisson equation with CUFFT
- **Hands-on:** solving linear system with AmgX library (*amgx_precond*)

Day 4: GPU memory hierarchy and advanced CUDA programming

Morning (09:00-12:30)

09:00-10:30: lecture

- GPU memory types
- Shared memory
- GPU caches hierarchy and mode switches
- Automatic texture cache (Kepler GK110)
- Unified virtual address space (UVA) in CUDA 6.0
- Streams and asynchronous data transfers

10:30-12:30: [hands-on session](#)

- **Hands-on:** “fill-in” exercise on reduction with and without shared memory
- **Hands-on:** getting additional performance using automatic texture cache

Afternoon (13:30-17:00)

13:30-15:00: lecture

- Dynamic parallelism
- Dynamic memory allocation in CUDA threads
- Compiling & linking relocatable device code
- CUDA C++ compiler pipeline, PTX assembler, SASS
- Understanding “-Xptxas -v” reports

15:00-17:00: [hands-on session](#)

- **Hands-on:** efficient GPU implementation of CFD problem solver with adaptive grid refinement (case study of dynamic parallelism)

Day 5: GPU code optimization, profiling and debugging

Morning (09:00-12:30)

09:00-10:30: lecture

- PCI-E optimizations: streams, asynchronous data transfers
- An overview of Fermi, Kepler and Maxwell GPU architectures
- GPU optimizations: compute grid, coalescing, divergence, unrolling, vectorization, maxrregcount, aligning, floating-point constants
- Overview of *NVIDIA Visual Profiler*
- Overview of *nvprof* (command line profiler)
- Common practices of identifying performance hazards in GPU application using NVIDIA Visual Profiler

10:30-13:30: [hands-on session](#)

- **Hands-on:** step by step profile and optimize the bilinear interpolation kernel

Afternoon (13:30-17:00)

13:30-15:00: lecture

- Principles and terminology
- GNU Debugger (*gdb*)
- CUDA-enabled GNU Debugger (*cuda-gdb*)
- GPU memory checker (*cuda-memcheck*)
- Debugging SASS without the source code

15:00-17:00: [hands-on session](#)

- **Hands-on:** live demonstration of *cuda-gdb* debugger on a sample application
- **Hands-on:** live demonstration of Eclipse IDE Nsight Edition built-in debugger on a sample application

Location

Entennest 2, 73730 Esslingen am Neckar, Germany

