

## GPU Technology Conference 2010 Sessions on Programming Languages & Techniques (subject to change)

*IMPORTANT: Visit [www.nvidia.com/qtc](http://www.nvidia.com/qtc) for the most up-to-date schedule and to enroll into sessions to ensure your spot in the most popular courses.*

---

### **2143 - CUDA Fortran Programming for NVIDIA GPUs**

An introduction to programming NVIDIA GPUs using CUDA Fortran. Suitable for expert Fortran or CUDA C programmers who need to extract maximum performance from GPUs using an explicit GPU Fortran programming model. Introduces the CUDA Fortran language, and through examples, illustrates how to explicitly program GPUs in native Fortran 95/03 through creation of GPU kernel subroutines, management of host and device memory, definition of CUDA grids and thread blocks, launching kernels, and use of the CUDA Fortran runtime API. This talk includes a live component with a Windows laptop containing an NVIDIA GPU and the PGI CUDA Fortran compiler.

Speaker: Brent Leback, The Portland Group

Topics: Tools & Libraries, High Performance Computing, Programming Languages & Techniques

Time: Wednesday, September, 22nd, 15:30 - 15:50

### **2296 - CUDA Optimization for Ninjas: A Case Study of High-Performance Sorting**

In this presentation, we use our implementation for high performance radix sorting as a case study for illustrating advanced design patterns and idioms. These techniques have allowed us to demonstrate Fermi sorting rates that exceed 1.0 billion 32-bit keys per second (and over 770 million key-value pairs per second), making it the fastest fully-programmable micro-architecture for this genre of sorting problems.

Although the CUDA programming model is elegantly decoupled from any particular hardware configuration, we present techniques for exploiting knowledge of the NVIDIA GPU machine model in order to produce more efficient implementations. Our design patterns enable the compiler to specialize a single program text for a variety of architectures, resulting in target code that “fits” the underlying hardware significantly better than more general approaches. In particular, we discuss strategies for kernel fusion, warp-synchronous programming, flexible granularity via meta-programming, algorithm serialization, and data-movement tuning.

Speaker: Duane Merrill, NVIDIA

Topic: Programming Languages & Techniques

Time: To Be Determined

## **2167 - Designing a Geoscience Accelerator Library Accessible from High Level Languages**

Explore a library for geoscience applications on CUDA and OpenCL platforms. Target applications span atmosphere, ocean, geomorphology and porous media flows. These areas are linked by common numerical techniques encapsulated in our library. We will review the scope of the library, its meta-programming approaches, and its key design attributes. We will also demonstrate its support for multi-GPU parallelism within and across address spaces and provide examples of its use from high level languages including C, Fortran, and Python.

Speakers: Chris Hill, M.I.T, Alan Richardson, M.I.T

Topics: Programming Languages & Techniques, Algorithms & Numerical Techniques, Computational Fluid Dynamics, Tools & Libraries

Time: Wednesday, September, 22nd, 17:00 - 17:50

## **2260 - DirectCompute (Pre-Conference Tutorial)**

Learn how to use the DirectCompute API to solve GPU computing problems. This tutorial will introduce the DirectCompute API, cover the recommended best practices for GPU programming, and go over examples of how to use this API efficiently and effectively to solve compute-intensive problems.

Speaker: Eric Young, NVIDIA

Topic: Programming Languages & Techniques

Time: Monday, September, 20th, 14:30 - 15:50

## **2281 - Domain-Specific Languages**

Computer graphics has introduced several domain-specific languages (DSLs) that enable high performance and parallelism for narrow problem domains - RenderMan, Cg, GLSL, and recently OpenRL and OptiX. We think that similar approaches can benefit other areas of GPU computing - visualization, animation, physics simulation, or scientific data analysis. In this talk, we present Shadie, a domain-specific shading language for rapid development of complex custom volume

visualizations in radiation oncology. The shaders are written in a high-level Python-like language and translated to CUDA for efficiency. We will explain how you can develop your own DSLs using source-to-source translation and a suitable backend library.

Speaker: Hanspeter Pfister

Topics: Programming Languages & Techniques, General Interest

Time: Wednesday, September, 22nd, 15:00 - 15:50

## **2067 - Experiences with Code Optimizations for High Performance GPGPU Programs**

Attend this session to learn and share code optimizations to achieve high performance GPU computing. We will cover code transformations for memory coalescing, workload management at both thread and thread-block levels, and different ways to handle memory partition conflicts. We will also discuss Integration of code optimizations into a compiler.

Speakers: Huiyang Zhou, North Carolina State University, Yi Yang, North Carolina State University

Topic: Programming Languages & Techniques

Time: Tuesday, September, 21st, 16:00 - 16:50

## **2011 - Fundamental Performance Optimizations for GPUs**

This presentation covers the major CUDA optimizations. Topics will include: maximizing memory throughput, kernel launch configuration, using shared memory, and improving GPU/CPU interaction. While C for CUDA is used for illustration, the concepts covered will apply equally to programs written with OpenCL and DirectCompute APIs.

Speaker: Paulius Micikevicius, NVIDIA

Topics: Programming Languages & Techniques, Tools & Libraries

Time: Wednesday, September, 22nd, 17:00 - 18:00

## **2294 - GPU.NET with TidePowerd**

Join TidePowerd for a demonstration of GPU.NET, our innovative new product which dramatically cuts the time needed to develop and maintain a GPU-based application by extending Microsoft's .NET Framework onto GPUs. With GPU.NET, your device-accelerated code can be written in any .NET-supported language (e.g., C#, F#, IronPython) and called like

any other method - so it's easy to create new GPU-based applications without having to retrain your developers. You'll learn how to use GPU.NET to quickly develop a financial calculator in C#, use the built-in Visual Studio unit-testing tools to ensure the correctness of the code, and seamlessly deploy the application into a mixed Windows / Linux environment. We'll also discuss how GPU.NET expands the frontiers of GPU computing into lucrative new markets such as business intelligence, database processing, and data visualization.

Speaker: Jack Pappas, TidePowerd

Topic: Programming Languages & Techniques

Time: Wednesday, September, 22nd, 17:00 - 17:50

## **2010 - Implementing Stereoscopic 3D in Your Applications (Pre-Conference Tutorial)**

Let's dive into the 3rd dimension. This talk presents a comprehensive technical overview of NVIDIA's stereo technology and tools. After a complete introduction to NVIDIA's stereo technology, we will then explore in more detail production techniques for the new artistic space of effects and creativity offered by 3D stereo. The take away of this session will be a solid understanding of NVIDIA's stereo technology and how to take best advantage of it.

Speakers: Samuel Gateau, NVIDIA, Steve Nash, NVIDIA

Topics: Programming Languages & Techniques, Stereoscopic 3D

Time: Monday, September, 20th, 16:00 - 17:20

## **2131 - Introduction to CUDA C (Pre-Conference Tutorial)**

Starting with a background in C or C++, learn everything you need to know in order to start programming in CUDA C. Beginning with a "Hello, World" CUDA C program, explore parallel programming with CUDA through a number of hands-on code examples. Examine more deeply the various APIs available to CUDA applications and learn the best (and worst) ways in which to employ them in applications. Master the first half of the book "CUDA by Example" as taught by the author, pointing you on a trajectory to complete the second half on your own after course completion.

Speaker: Jason Sanders, NVIDIA

Topic: Programming Languages & Techniques

Time: Monday, September, 20th, 14:30 - 15:50

## **2004 - Languages, APIs and Development Tools for GPU Computing (Pre-Conference Tutorial)**

Get a head start on the conference with this first-day introduction to key technologies for GPU Computing. This 90-minute tutorial session will cover the key features and differences between the major programming languages, APIs and development tools available today. Attendees will also learn several high level design patterns for consumer, professional and HPC applications, with practical programming considerations for each.

Speaker: Will Ramey, NVIDIA

Topics: Programming Languages & Techniques, Tools & Libraries

Time: Monday, September, 20th, 13:00 - 14:20

## **2028 - Mathematica for GPU Programming**

Mathematica is widely used in scientific, engineering, mathematical fields and education. In this session, new tools for general GPU programming in the next release of Mathematica are presented. These tools build on top of Mathematica's technology which provides a simple, yet powerful, interface to the large base of compiling tools. Applications of CUDA and OpenCL from within Mathematica will be presented. These examples will provide a general overview of the powerful development environment for GPU programming that Mathematica can offer not just for researchers but for anybody with basic knowledge of Mathematica and GPU programming.

Speaker: Ulises Cervantes-Pimentel, Wolfram Research

Topics: Programming Languages & Techniques, Algorithms & Numerical Techniques, Imaging, Tools & Libraries

Time: Tuesday, September, 21st, 14:00 - 14:50

## **2024 - NVIDIA Acceleration Engines with NVIDIA SceniX (Pre-Conference Tutorial)**

Learn how NVIDIA SceniX is a rapid start to building state of the art, realtime 3D applications, and how raytracing can be combined with raster graphics for new levels of interactive realism.

Speakers: Phillip Miller, NVIDIA, Holger Kunz, NVIDIA

Topics: Programming Languages & Techniques, Computer Vision, Ray Tracing

Time: Monday, September, 20th, 13:00 - 14:20

## **2161 - NVIDIA Quadro Digital Video Pipeline Overview**

This session will provide an overview of the Quadro Digital Video Pipeline. It will cover a description of the DVP components, application architectures software architectures, and programming resources available.

Speaker: Thomas True, NVIDIA

Topics: Computer Graphics, Video Processing, Programming Languages & Techniques

Time: Tuesday, September, 21st, 16:00 - 16:50

## **2127 - OpenGL (Pre-Conference Tutorial)**

This session will discuss the latest OpenGL features offered by NVIDIA for both Quadro and Geforce line of products. Learn more about OpenGL 4 as well as NVIDIA specific OpenGL extensions.

Speakers: Barthold Lichtenbelt, NVIDIA, Mark Kilgard, NVIDIA

Topic: Programming Languages & Techniques

Time: Monday, September, 20th, 16:00 - 17:20

## **2124 - Operating System Abstractions for GPU Programming**

GPGPU frameworks such as CUDA improve programmability, but GPU parallelism remains inaccessible in many application domains. This session argues that poor OS support causes this problem. OSES do not provide the kind of high-level abstractions for GPUs that applications expect for other resources like CPUs and file systems. We advocate reorganizing kernel abstractions to support GPUs as first-class computing resources, with traditional guarantees such as fairness and isolation. We demonstrate shortcomings in Windows 7 GPU support, and show that better OS abstractions can accelerate interactive workloads like gesture recognition by a factor of 10X over a CUDA implementation.

Speakers: Christopher Rossbach, Microsoft Research, Emmett Witchel, University of Texas at Austin

Topics: Programming Languages & Techniques, Tools & Libraries

Time: Thursday, September, 23rd, 10:00 - 10:50

## **2212 - Parallel Nsight for Accelerated DirectX 11 Development** **[Advanced]**

Parallel Nsight is NVIDIA's new development environment for graphics and GPU computing. In this advanced session, you will learn how Parallel Nsight can accelerate debugging and profiling of Direct3D 11 applications.

Attendees will learn how to debug Direct3D frames and HLSL shaders using Parallel Nsight's powerful Graphics Inspector and Debugger which allows developers to inspect Direct3D resources and state, set breakpoints in HLSL shaders, examine shader variables, and see which graphics primitives are live on the GPU.

Attendees will also learn how to use the Frame Profiler to capture and mine performance information, and easily pinpoint bottlenecked GPU units.

Speaker: Simon Barrett, NVIDIA

Topics: Programming Languages & Techniques, Computer Graphics

Time: Tuesday, September, 21st, 17:00 - 17:50

## **2151 - Parallel Nsight: Analyzing and Optimizing Massively Parallel Applications [Advanced]**

Life altering products that provide early detection of breast cancer or simulate molecular behavior, accelerating drug discovery, are becoming reality thanks to the power of the GPU. As these technologies become mainstream, mainstream tools are required to support these development efforts. NVIDIA Parallel Nsight delivers the power of the GPU within the familiar environment of Microsoft Visual Studio. In this session, you will learn advanced techniques for visualizing your application's workloads and performance characteristics across the CPU, GPU, and operating system, and explore the depths of Parallel Nsight profilers, including GPU performance counters and how to use them.

Speaker: Sebastien Domine, NVIDIA

Topics: Tools & Libraries, Programming Languages & Techniques

Time: Tuesday, September, 21st, 16:00 - 16:50

## **2150 - Parallel Nsight: Debugging Massively Parallel Applications [Advanced]**

Data parallel algorithms that provide real-time financial options pricing or identification of hidden oil reserves are utilizing the massively parallel nature of the GPU for industry changing performance gains. Developers require industry standard development tools to create the software that accomplishes these parallel tasks. NVIDIA Parallel Nsight delivers the power of

the GPU within the familiar environment of Microsoft Visual Studio. In this session, you will learn advanced techniques for debugging CUDA C/C++ and DirectCompute code using Parallel Nsight, including conditional and data breakpoints as well as out of bound GPU memory access detection.

Speaker: Sebastien Domine, NVIDIA

Topics: Tools & Libraries, Programming Languages & Techniques

Time: Tuesday, September, 21st, 14:00 - 14:50

## **2023 - Processing Device Arrays with C++ Metaprogramming**

I will describe tricks for building APIs using C++ metaprogramming that generate custom kernels for complex manipulation of device-side arrays in CUDA. Using a variation of Expression Templates, multiple operations can be fused into a single kernel that executes with reasonable efficiency.

Speaker: Jonathan Cohen, NVIDIA Research

Topics: Programming Languages & Techniques, Tools & Libraries

Time: Thursday, September, 23rd, 11:00 - 11:50

## **2159 - Programming the NVIDIA Digital Video Pipeline with Direct3D (Pre-Conference Tutorial)**

Learn how to program the NVIDIA Quadro Digital Video pipeline using Direct3D. This session will provide an overview of the SDK, discuss device control, data transfers, performance measuring and tuning, ancillary data and application design considerations.

Speaker: Thomas True, NVIDIA

Topics: Programming Languages & Techniques, Video Processing, Computer Graphics

Time: Monday, September, 20th, 14:30 - 15:50

## **2158 - Programming the NVIDIA Digital Video Pipeline with OpenGL (Pre-Conference Tutorial)**

This tutorial session teaches attendees how to program the NVIDIA Quadro Digital Video Pipeline with OpenGL. It will go in-depth into the techniques and recommended practices.

Speaker: Thomas True, NVIDIA

Topics: Programming Languages & Techniques, Video Processing, Computer Graphics

Time: Monday, September, 20th, 13:00 - 14:20

## **2278 - Strategies for Code Encapsulation in GPU Implementations**

Code encapsulation is a common technique used to reduce code complexity that a given programmer has to understand. It allows the use of increasingly complex systems of hardware, software, and algorithms to tackle increasingly difficult scientific problems. Unfortunately, code encapsulation is not easily attainable in current GPU environments. We will share our OpenCL development experiences for achieving partial encapsulation in GPU implementations, and discuss best practices in this area.

Speaker: Brian Cole, OpenEye Scientific Software

Topics: Programming Languages & Techniques, High Performance Computing, Life Sciences

Time: Thursday, September, 23rd, 09:00 - 9:20