SCALING CUDA C++ APPLICATIONS TO MULTIPLE NODES

DEEP LEARNING INSTITUTE

Present-day high-performance computing (HPC) and deep learning applications benefit from, and even require, cluster-scale GPU compute power. Writing CUDA® applications that can correctly and efficiently utilize GPUs across a cluster requires a distinct set of skills. In this workshop, you'll learn the tools and techniques needed to write CUDA C++ applications that can scale efficiently to clusters of NVIDIA GPUs.

You'll do this by working on code from several CUDA C++ applications in an interactive cloud environment backed by several NVIDIA GPUs. You'll gain exposure to a handful of multi-GPU programming methods, including CUDA-aware Message Passing Interface (MPI), before proceeding to the main focus of this course, **NVSHMEM™**.

NVSHMEM is a parallel programming interface based on OpenSHMEM that provides efficient and scalable communication for NVIDIA GPU clusters. NVSHMEM creates a global address space for data that spans the memory of multiple GPUs and can be accessed with fine-grained GPU-initiated operations, CPU-initiated operations, and operations on CUDA streams. NVSHMEM's asynchronous, GPU-initiated data transfers eliminate synchronization overheads between the CPU and the GPU. They also enable long-running kernels that include both communication and computation, reducing overheads that can limit an application's performance when strong scaling. NVSHMEM has been used to great effect on systems such as the Summit supercomputer located at the Oak Ridge Leadership Computing Facility (OLCF), the Lawrence Livermore National Laboratory's Sierra supercomputer, and the NVIDIA DGX™ A100.

Learning Objectives

By participating in the workshop, you'll:

- Learn several methods for writing multi-GPU CUDA C++ applications
- Use a variety of multi-GPU communication patterns and understand their tradeoffs
- Write portable, scalable CUDA code with the single-program multiple-data (SPMD) paradigm using CUDA-aware MPI and NVSHMEM
- Improve multi-GPU SPMD code with NVSHMEM's symmetric memory model and its ability to perform GPU-initiated data transfers
- Get practice with common multi-GPU coding paradigms like domain decomposition and halo exchanges
- Explore scaling considerations for a variety of GPU-cluster configurations

Overview

Duration	8 hours
Price	**Contact us for pricing**
Prerequisites	- Intermediate experience writing CUDA C/C++ applications Suggested materials to satisfy the prerequisites: - **Fundamentals of Accelerated Computing with CUDA C/C++** - **Accelerating CUDA C++ Applications with Multiple GPUs** - **Accelerating CUDA C++ Applications with Concurrent Streams** - **Scaling Workloads Across Multiple GPUs with CUDA C++**
Tools, libraries, and frameworks	CUDA, MPI, NVSHMEM

Assessment type	Skills-based coding assessment Students must refactor a single-GPU 1D wave function solver to be GPU-cluster-ready with NVSHMEM.
Certificate	Upon successful completion of the assessment, participants will receive an NVIDIA DLI certificate to recognize their subject matter competency and support professional career growth.
Hardware requirements	Desktop or laptop computer capable of running the latest version of Chrome or Firefox. Each participant will be provided with dedicated access to a fully configured, GPU-accelerated workstation in the cloud.
Language	English

Workshop Outline

Introduction (15 mins)	> Meet the instructor. > Create an account at **courses.nvidia.com/join**
Multi-GPU Programming Paradigms (120 minutes)	**Survey multiple techniques for programming CUDA C++ applications for multiple GPUs using a Monte Carlo approximation of a CUDA C++ program.** > Use CUDA to utilize multiple GPUs. > Learn how to enable and use direct peer-to-peer memory communication. > Write an SPMD version with CUDA-aware MPI.
Break (60 minutes)	
Introduction to NVSHMEM (120 minutes)	**Learn how to write code with NVSHMEM and understand its symmetric memory model.** > Use NVSHMEM to write SPMD code for multiple GPUs. > Utilize symmetric memory to let all GPUs access data on other GPUs. > Make GPU-initiated memory transfers.
Break (15 minutes)	
Halo Exchanges with NVSHMEM (90 minutes)	**Practice common coding motifs like halo exchanges and domain decomposition using NVSHMEM, and work on the assessment.** > Write an NVSHMEM implementation of a Laplace equation Jacobi solver. > Refactor a single GPU 1D wave equation solver with NVSHMEM. > Complete the assessment and earn a certificate.
Final Review (45 minutes)	> Learn about application tradeoffs on GPU clusters. > Review key learnings and answer questions. > Complete the workshop survey.
Next Steps	**Continue learning with these DLI trainings:** > **Fundamentals of Deep Learning for Multi-GPUs** for those interested in cluster-scale deep learning. > **High Performance Computing with Containers** for HPC programmers looking to improve the portability of their cluster-scale applications. > **Fundamentals of Accelerated Computing with CUDA Python** for CUDA C++ programmers who would like to extend their CUDA knowledge to Python.

Why Choose NVIDIA Deep Learning Institute for Hands-On Training?

- Access workshops from anywhere with just your desktop/laptop and an internet connection. Each participant will have access to a fully configured, GPU-accelerated server in the cloud.
- Obtain hands-on experience with the most widely used, industry-standard software, tools, and frameworks.
- Learn how to build deep learning and accelerated computing applications for industries, such as healthcare, robotics, manufacturing, accelerated computing, and more.
- Gain real-world experience through content designed in collaboration with industry leaders, such as the Children's Hospital of Los Angeles, Mayo Clinic, and PwC.
- Earn an NVIDIA DLI certificate to demonstrate your subject matter competency and support your career growth.

For the latest DLI workshops and trainings, visit **www.nvidia.com/dli**

For questions, contact us at **nvdli@nvidia.com**

NVIDIA | DEEP LEARNING INSTITUTE