White Paper

NVIDIA DGX Station

The First Personal AI Supercomputer

Abstract

NVIDIA® DGX Station™ (Figure 1) is the world's first personal supercomputer for leading-edge AI development. DGX Station features four NVIDIA® Tesla® V100 GPU accelerators fully connected through NVIDIA® NVLink™, the NVIDIA high-performance GPU interconnect, and is powered by DGX software. The Tesla V100 GPU accelerator features the new Tensor Core architecture to help deliver unprecedented levels of performance not seen in any AI workstation prior. Offering whisper-quiet, breakthrough performance, DGX Station gives computational scientists, data scientists, and AI researchers the fastest start in deep learning, HPC, and data science from the convenience of their desks. Unlike other platforms, DGX Station software provides highly optimized libraries and containers designed for maximized deep learning performance leveraging its four Tesla V100 GPUs, providing a flexible, versatile and scalable platform for running deep learning workloads for research and in production.

Figure 1 NVIDIA DGX Station

1 INTRODUCTION

Deep learning is quickly changing virtually every industry and is having a large impact on the economics of businesses:

- The number of GPU deep learning developers has leapt 25 times in just two years [Huang, 2016]. In 2017 alone, NVIDIA's CUDA has been downloaded 2.4 million times.
- The number of active US startups developing AI systems has increased 14x since 2000 [AI Index, Page 16].
- Facebook CTO Mike Schroepfer noted that they have deployed more than 40 PFLOPs of GPU capability in house to support deep learning across their organization [Schroepfer 2016:6:54].
- Organizations are looking to the new-generation of HPC systems to provide the performance, reliability, and flexibility that they need. IDC estimates that for every $43 spent on HPC, users can generate $515 in revenue, making HPC adoption a highly lucrative investment [Wheat, 2017].

The ever-increasing computational power required for running deep learning workloads is driving the need for advanced workstations that make use of hardware accelerators to turbocharge the parallelized algorithms. According to IDC [NVIDIA Corporation, 2017a], worldwide accelerated computing infrastructure revenue will grow from $2.5 billion in 2016 to $6.8 billion in 2021, a CAGR of 21.9%. The on-premises portion of this market is forecast to grow from $1.6 billion in 2016 to $3.4 billion in 2021, at a CAGR of 16.3%. IDC expects an increasing portion of this market to be served by new form factors, including desktops and workstations. Workstations were designed for those professionals who don't want to wait in a queue for IT to approve the compute cycles in a server. They are meant to be placed in a professional's office and used as a deskside-based supercomputer at their fingertips.

To satisfy this insatiable need for high performance, GPU accelerated computing, NVIDIA designed DGX Station™ (shown in Figure 1), which is the fastest personal AI supercomputer for deep learning training. NVIDIA's goal with DGX Station™ was to create the world's fastest platform for training deep neural networks that can be deployed quickly and run quietly by deep learning researchers and data scientists under their desks. The architecture of DGX Station™ draws on NVIDIA's experience in the field of high-performance computing and knowledge gained from optimizing deep learning frameworks on NVIDIA GPUs.

2 NVIDIA DGX STATION ARCHITECTURE

DGX Station is a deep learning workstation architected for high performance, multi-GPU neural network training equivalent or better than what's traditionally found in a data center, now placed at the developer's fingertips. The core of the system is a complex of four NVIDIA® Tesla® V100 GPUs in a fully connected NVLink™ topology, described in Section 2.2. Using mixed-precision multiply and accumulate operations, the new Tensor Core architecture enables Volta V100 to deliver the performance required to train large neural networks. In addition to the four GPUs, DGX Station™ includes one 20-core CPU, fast local storage (3 SSDs configured in RAID 0), a water-cooling system for the GPUs, dual 10 GbE networking, all balanced to optimize throughput and deep learning training time.

Figure 2 shows the DGX Station™ system components.

1. GPUs
4X NVIDIA Tesla® V100 16 GB/GPU
500 TFLOPS (Mixed Precision)
20,480 Total NVIDIA CUDA® Cores
2,560 Tensor Cores

2. SYSTEM MEMORY
256 GB RDIMM DDR4

3. GPU INTERCONNECT
NVIDIA NVLink™,
Fully Connected 4-Way

4. STORAGE
Data: 3 x 1.92 TB SSD RAID 0
OS: 1 x 1.92 TB SSD

5. CPU
Intel Xeon E5-2698 v4
2.2 GHz 20-Core

6. NETWORKING
2X 10 GbE

7. DISPLAYS
3X DisplayPort,
4K Resolution

8. COOLING
Water-Cooled

9. POWER
1500 W

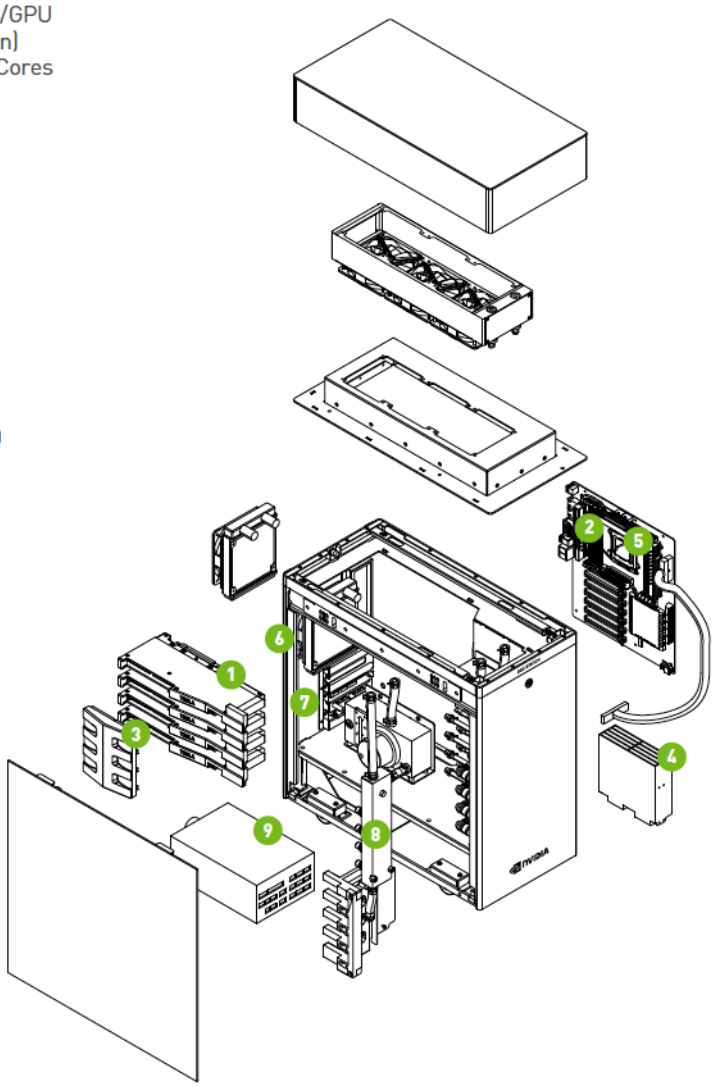

Figure 2 DGX Station™ components

2.1 NVIDIA Tesla V100

Tesla V100 (Figure 3) is the latest NVIDIA accelerator, designed for high performance computing and deep learning applications [NVIDIA Corporation 2017d]. The Tesla V100 accelerator features the GV100 GPU, which incorporates 80 streaming multiprocessors (SMs), each with:

- 8 Tensor Cores;
- 64 single-precision (FP32) cores;
- 64 integer (INT32) cores;
- 32 double-precision (FP64) cores;
- 256KB of register file (RF);
- Up to 96KB of shared memory (configurable).

Figure 3 The Tesla V100 Accelerator

Tesla V100 peak[1] computational throughput is:

- 125 Tensor TFLOP/s [NVIDIA Corporation 2017d];
- 15.7 TFLOP/s for FP32;
- 7.8 TFLOP/s for FP64.

To support this high computational throughput, Tesla V100 incorporates HBM2 (High Bandwidth Memory version 2). V100 includes 16GB of HBM2 stacked memory with 900 GB/s of bandwidth; significantly higher than the bandwidth of GDDR5 RAM. Because HBM2 memory is stacked memory located on the same physical package as the GPU, it provides considerable space savings compared to traditional GDDR5, which enables high-density GPU workstations like DGX Station, including its water-cooling blocks.

Each Tesla V100 in DGX Station has 4 NVLink connections each capable of 50 GB/s of bidirectional bandwidth, for an aggregate of up to 200 GB/s bidirectional bandwidth. NVLink and the DGX Station interconnect topology and its implications are discussed in detail in Section 3.

The PCIe links between the GPUs and CPUs provide access to system memory to enable working set and dataset streaming to and from the GPUs. The system memory capacity is four times the GPU memory capacity to enable simplified buffer management and balance for deep learning workloads. While twice the GPU memory footprint would normally be sufficient to manage background data moves and double buffering, four times gives greater flexibility for managing in-memory working sets and streaming data movement. In addition to the 256 GB of system memory, the four Tesla V100 GPUs have a total of 64 GB HBM2 memory with net GPU memory bandwidth of 4 × 900 GB/s = 3.6 TB/s.

1. Based on GPU Boost Clock.

Some key compute features of Tesla V100:

New Streaming Multiprocessor (SM) Architecture Optimized for Deep Learning

Volta features a major new redesign of the SM processor architecture that is at the center of the GPU. The new Volta SM is 50% more energy efficient than the previous generation Pascal design, enabling major boosts in FP32 and FP64 performance in the same power envelope. New Tensor Cores designed specifically for deep learning deliver up to 12x higher peak TFLOPS for training and 6x higher peak TFLOPS for inference. With independent parallel integer and floating-point data paths, the Volta SM is also much more efficient on workloads with a mix of computation and addressing calculations. Volta's new independent thread scheduling capability enables finer-grain synchronization and cooperation between parallel threads. Finally, a new combined L1 data cache and shared memory unit significantly improves performance while also simplifying programming.

Second-Generation NVIDIA NVLink™

The second generation of NVIDIA's NVLink high-speed interconnect delivers higher bandwidth, more links, and improved scalability for multi-GPU and multi-GPU/CPU system configurations. DGX Station uses four NVLink links with a total bandwidth of 200 GB/sec. to deliver greater scalability for ultra-fast deep learning training.

HBM2 Memory: Faster, Higher Efficiency

Volta's highly tuned 16 GB HBM2 memory subsystem delivers 900 GB/sec peak memory bandwidth. The combination of both a new generation HBM2 memory from Samsung, and a new generation memory controller in Volta, provides 1.5x delivered memory bandwidth versus Pascal GP100, with up to 95% memory bandwidth utilization running many workloads.

Volta Multi-Process Service

Volta Multi-Process Service (MPS) is a new feature of the Volta GV100 architecture providing hardware acceleration of critical components of the CUDA MPS server, enabling improved performance, isolation, and better quality of service (QoS) for multiple compute applications sharing the GPU. Volta MPS also triples the maximum number of MPS clients from 16 on Pascal to 48 on Volta.

Enhanced Unified Memory and Address Translation Services

GV100 Unified Memory technology includes new access counters to allow more accurate migration of memory pages to the processor that accesses them most frequently, improving efficiency for memory ranges shared between processors.

Cooperative Groups and New Cooperative Launch APIs

Cooperative Groups is a new programming model introduced in CUDA 9 for organizing groups of communicating threads. Cooperative Groups allows developers to express the granularity at which threads are communicating, helping them to express richer, more efficient parallel decompositions. Basic Cooperative Groups functionality is supported on all NVIDIA GPUs since Kepler. Pascal and Volta include support for new cooperative launch APIs that support synchronization amongst CUDA thread blocks. Volta adds support for new synchronization patterns.

Volta Optimized Software

New versions of deep learning frameworks such as Caffe2, MXNet, CNTK, TensorFlow, and others harness the performance of Volta to deliver dramatically faster training times and higher multi-node training performance. Volta-optimized versions of GPU accelerated libraries such as cuDNN, cuBLAS, and TensorRT leverage the new features of the Volta GV100 architecture to deliver higher performance for both deep learning inference and High Performance Computing (HPC) applications. NVIDIA CUDA Toolkit version 9.0 includes new APIs and support for Volta features to provide even easier programmability.

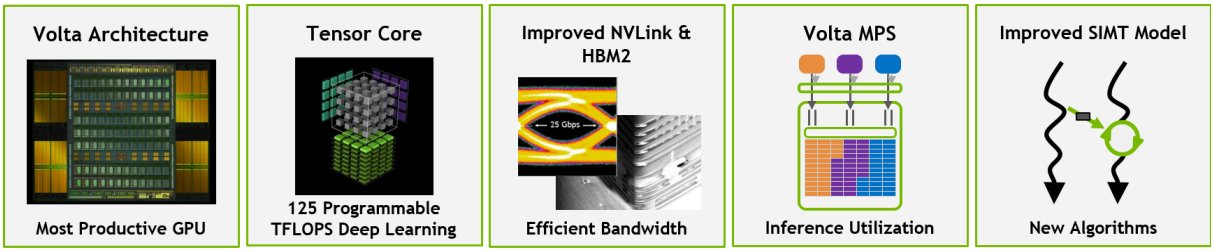

Figure 4 New Technologies in Tesla V100

2.2 Second-Generation NVIDIA NVLink™

NVLink is NVIDIA's high-speed interconnect technology first introduced in 2016 with the Tesla P100 accelerator. NVLink provides significantly more performance for both GPU-to-GPU system configurations compared to using PCIe interconnects. Tesla V100 introduces the second generation of NVLink, which provides higher link speeds, more links per GPU, CPU mastering, cache coherence, and scalability improvements.

2.3 Water-Cooling System for the GPUs

The water-cooling system for the GPUs in the DGX Station captures 90% of the GPUs' Thermal Design Power (TDP). This level of efficiency allows for whisper-quiet operation with better performance at higher TDP and more than twice as much noise abatement compared to air cooling. Its design enables both the radiator and other components to be cooled by a single set of fans.

The water-cooled design of DGX Station allows each Tesla V100 GPU board to match the performance of a V100 SXM2 module in a NVIDIA DGX-1 rack-mountable server. This means NVIDIA DGX systems provide higher performance out-of-the-box than air-cooled solutions.

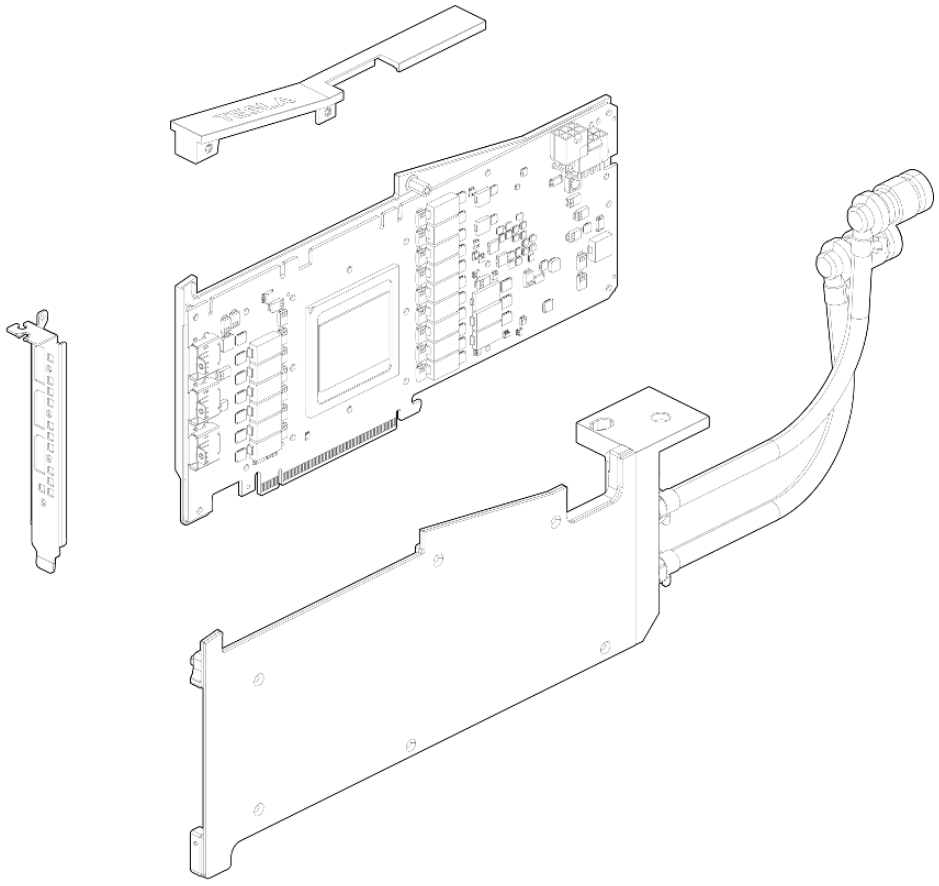

Figure 5 NVIDIA Tesla V100 for DGX Station, with the water-block assembly, exploded view

2.4 GPU and System Memory

The PCIe links between the GPUs and the CPU enable access to the CPU's bulk DRAM to enable working set and dataset streaming to and from the GPUs. The CPU memory capacity is configured with four times the GPU memory capacity, to enable simplified buffer management and balance for deep learning workloads. While twice the GPU memory footprint would normally be sufficient to manage background data moves and back buffering, four times gives greater flexibility for managing in-memory working sets and streaming data movement. In addition to the 256 GB of system RDIMM DDR4, each Tesla V100 GPU includes 16 GB of HBM2 co-packaged memory with memory bandwidth of 900 GB/s, yielding a total of 64 GB HBM2 memory with net GPU memory bandwidth of 3600 GB/s.

NVIDIA GPU copy engines transfer data between multiple GPUs or between GPUs and CPUs. In previous-generation GPUs, performing copy engine transfers (which are like DMA transfers) could cause fatal faults if either the source or destination memory addresses were not mapped in the GPU page tables. The prior copy engines required both source and destination memory regions to be pinned (non-pageable).

The new Volta GV100 GPU copy engines can generate page faults for addresses that are not mapped into the page tables. The memory subsystem can then service the page faults, mapping the addresses into the page table, after which the copy engine can perform the transfer. This is an important enhancement, because pinning memory for multiple copy engine operations between multiple processors can substantially reduce available memory. With hardware page faulting, addresses can be passed to the copy engines without worrying if they are resident, and the copy process just works.

2.5 Other Workstation Components

Efficient, high-bandwidth streaming of training data is critical to the performance of DGX Station™ as a deep learning system, as is reliable, low-failure-rate storage. Each system comes configured with a single 1.92 TB boot OS SSD, and three 1.92 TB SSDs (5.76 TB total) configured as a RAID 0 striped volume for performance. External storage can be added to DGX Station through two eSATA ports as well as two USB 3.1 ports. Alternatively, a Network Attached Storage (NAS) device can be connected to a 10 GbE LAN port.

In addition to the four GPUs, DGX Station includes:

- A server-class Intel Xeon CPU (Intel Xeon E5-2698 v4 2.2 GHz) for overall system throughput and performance
- Three DisplayPort™ connectors on one of the NVIDIA Tesla V100 GPU cards in the DGX Station, enabling up to three displays at 4K resolution to be connected to the DGX Station
- Two 10GBASE-T (RJ45) Ethernet ports for fast networking access to and from DGX Station

DGX Station's power consumption is dynamic based on workload. The TDP of DGX Station is 1,500 W, but is equipped with a power supply rated at 1,600 W. For ease of deployment, DGX Station is designed to be operated at temperatures between 10°C and 30°C (50°F and 86°F).

3 MULTI-GPU WITH NVLINK

Workstations with two or more GPUs per CPU are becoming common as developers increasingly expose and leverage the available parallelism in their applications. While dense GPU systems provide a great vehicle for scaling single-node performance, multi-GPU application efficiency can be constrained by the performance of the PCIe (Peripheral Component Interconnect Express) bus connections between GPUs. Similarly, data center applications are growing outside the box, requiring efficient scaling across multiple interconnected systems. To address both of these needs, DGX Station incorporates the new NVLink high-speed GPU interconnect for multi-GPU scalability within a system.

Given that communication is an expensive operation, developers must overlap data transfers with computation or carefully orchestrate GPU accesses over PCIe interconnect to maximize performance. As GPUs get faster and GPU-to-CPU ratios climb, a higher-performance GPU interconnect is warranted.

This challenge motivated the creation of the NVLink high-speed interconnect, which enables NVIDIA Pascal GPUs to connect to peer GPUs and/or to NVLink-enabled CPUs or other devices within a node. NVLink supports the GPU ISA, which means that programs running on NVLink-connected GPUs can execute directly on data in the memory of another GPU as well as on local memory. GPUs can also perform atomic memory operations on remote GPU memory addresses, enabling much tighter data sharing and improved application scaling.

The second generation of NVLink allows direct load/store/atomic access from the CPU to each GPU's HBM2 memory. Coupled with a new CPU mastering capability, NVLink supports coherency operations allowing data reads from graphics memory to be stored in the CPU's cache hierarchy. The lower latency of access from the CPU's cache is key for CPU performance. While P100 supported peer GPU atomics, sending GPU atomics across NVLink and completed at the target CPU was not supported. NVLink adds support for atomics initiated by either the GPU or the CPU. Support for Address Translation Services (ATS) has been added allowing the GPU to access the CPU's page tables directly. A low-power mode of operation for the link has been added allowing for significant power savings when the link is not being heavily used.

The increased number of links, faster link speed, and enhanced functionality of second- generation NVLink, combined with Volta's new Tensor Cores, results in significant increases in deep learning performance in multi-GPU Tesla V100 systems over systems with Tesla P100 GPUs.

3.1 DGX NVLink Network Topology for Efficient Application Scaling

High-performance applications typically scale their computations in one of two ways, known as strong scaling and weak scaling[2]. Strong scaling measures the improvement in time to solution when

2. Note that "weak" is not an inferior form of scaling to "strong" scaling. Both are important metrics in practice.

increasing the number of parallel processors applied to a fixed total problem size. With perfect strong scaling, the speedup achieved would be equal to the number of processors used.

Weak scaling, on the other hand, measures the improvement in time to solution when increasing the number of parallel processors applied to a problem of fixed size per processor. In other words, the problem size is increased along with the number of processors. Here the execution time tends to remain fairly constant as the problem size (and the number of processors) increases. Perfect weak scaling, then, implies that the time to solution did not increase by scaling up the problem linearly with the number of processors.

As individual processors and clusters of processors get ever wider (having ever more parallel processing elements), the benefit of weak scaling for some problems diminishes—eventually these problems may run out of parallelism. It is at this point that these problems are forced into the realm of strong scaling. But in reality, while most applications do exhibit some degree of strong scaling, it is usually not perfectly linear.

A key reason for this is the cost of communication. Strong-scaling a problem onto an increasing number of processors gives each processor progressively less work to do, and increases the relative cost of communicating among those processors. In the strong-scaling regime, fast interconnects and communication primitives tuned for those interconnects are essential.

To provide the highest possible computational density, DGX Station™ includes four NVIDIA Tesla V100 accelerators. Application scaling on this many highly parallel GPUs is hampered by today's PCIe interconnect. NVLink provides the communications performance needed to achieve good (weak and strong) scaling on deep learning and other applications. Each Tesla V100 GPU in DGX Station has four NVLink connection points, each providing a point-to-point connection to another GPU at a peak bandwidth of 25GB/s. Multiple NVLink connections can be bonded together, multiplying the available interconnection bandwidth between a given pair of GPUs. The result is that NVLink provides a flexible interconnect that can be used to build a variety of network topologies among multiple GPUs. V100 also supports 16 lanes of PCIe 3.0. In DGX Station™, these are used for connecting between the CPUs and GPUs. PCIe is also used for accessing the high-speed networking interface cards.

The design of the NVLink network topology for DGX Station™ aims to optimize a number of factors, including the bandwidth achievable for a variety of point-to-point and collective communications primitives, the flexibility of the topology, and its performance with a subset of the GPUs. During the design, NVIDIA engineers modeled projected strong and weak scaling of a variety of applications, such as deep learning, sorting, Fast Fourier Transforms (FFT), molecular dynamics, graph analytics, computational fluid dynamics, seismic imaging, ray tracing, and others. This paper focuses on the analysis of scaling of deep learning training.

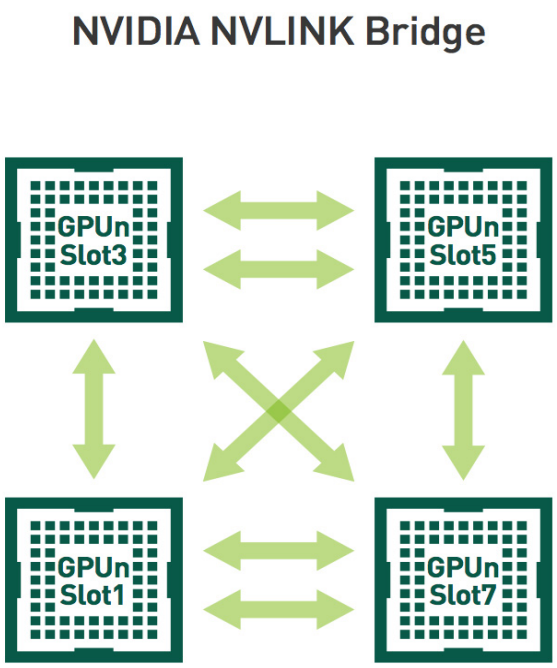

Figure 6 NVIDIA NVLink Bridge

3.2 Scaling Deep Learning Training on NVLink

Deep neural networks learn many layers of abstraction, in a hierarchy of simple to complex concepts. The strength of deep models is the ability to learn complex distributed representations from massive amounts of training data. A deep neural network is trained by feeding it input and letting it compute layer-by-layer to generate output for comparison with a known correct answer. After computing the error at the output, this error flows backward through the net by back-propagation. At each step backward the model parameters are tuned in a direction that tries to reduce the error using one of many numerical optimization methods, such as stochastic gradient descent (SGD). This process sweeps over the data, improving the model as it goes.

Training deep neural networks in parallel across multiple GPUs and/or multiple nodes requires distributing one of the following:

- The input data (“data parallel”): In data-parallel approaches, separate workers must periodically resynchronize the gradients with respect to the model that are calculated during back-propagation such that the model parameters are kept in sync across workers. This amounts to an all-reduce operation.
- The model being trained (“model parallel”): Model-parallel approaches may either elect one worker at a time to broadcast its gradients with respect to the input data, or they may use an all-gather of (successive subsets of) the data gradients so that all workers’ outbound bandwidths are utilized concurrently.
- A hybrid of the two [Wu et al. 2015][Krizhevsky 2014].

In weak-scaling the training of a deep neural network, the global effective SGD minibatch size increases as the number of GPUs increases. Perhaps unsurprisingly, weak-scaled approaches have high parallel efficiency, even with relatively slow interconnections among GPUs. However, the minibatch size can only be scaled to a certain point before the convergence of the SGD optimization is negatively impacted. The relative tolerance of various networks to increased amounts of weak scaling varies with the network.

To demonstrate scaling of training performance on DGX Station from 1, to 2, to 4 Tesla V100 GPUs, the bars in Figure 7 represent training performance in images per second for the ResNet-50 deep neural network architecture using the 17.09 DGX optimized containers for each of the three framework displayed (NVCaffe, Caffe2, and MXNet). These benchmark numbers were achieved using mixed precision with the new Tensor Cores available in V100, and show the linear scalability of V100s connected via NVLink.

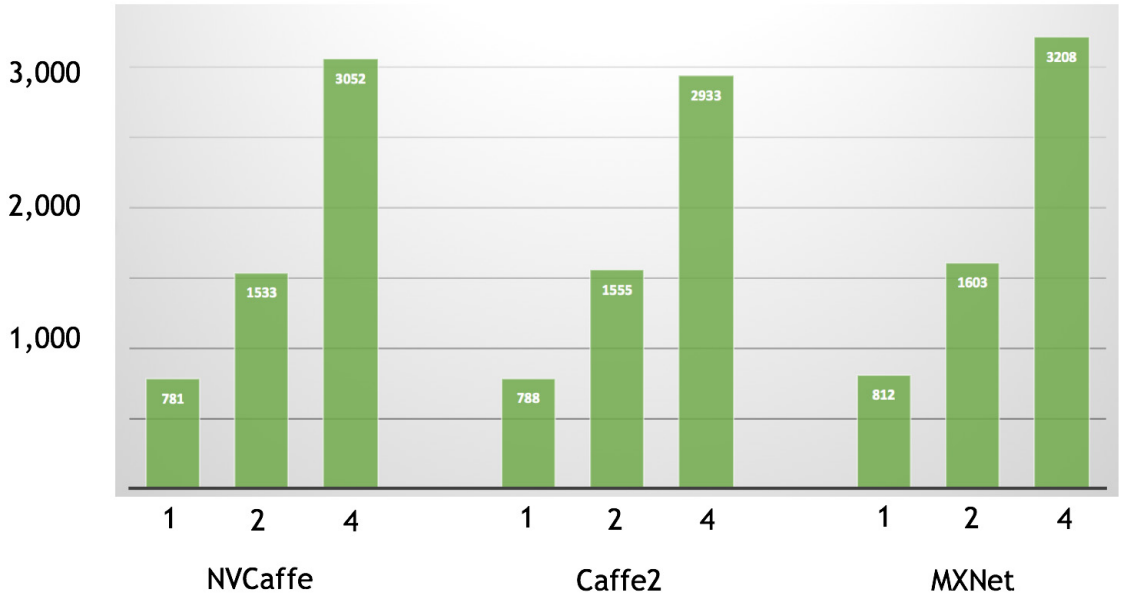

Figure 7 *DGX Station scalability with 4x Tesla V100. ResNet-50 Training, Volta with Tensor Core (mixed precision), 17.09 DGX optimized container.*
Score: Images per Second

In addition, the similar scores across the three different deep learning frameworks highlight the work the NVIDIA engineering team is doing for all major frameworks.

4 DGX STATION SOFTWARE STACK FOR DEEP LEARNING

The DGX Station™ software stack for deep learning has been built to run deep learning workloads at scale.

Note: DGX Station can also be used for High Performance Computing (HPC) and accelerated analytics workloads. See Section 6 for details.

A key goal is to enable practitioners to deploy deep learning frameworks and other applications on DGX Station™ with minimal setup effort. The design of the platform software is centered around an operating system based on Ubuntu Desktop with appropriate developer software and drivers installed on the workstation, and provisioning of all application software and additional SDK software with the Docker Engine Utility for NVIDIA GPUs.

Optimized deep learning Docker containers can be obtained through NVIDIA GPU Cloud (NGC). Containers available for DGX Station™ include multiple optimized deep learning frameworks, the NVIDIA DIGITS deep learning training application, third party accelerated analytics solutions, HPC containers, and the NVIDIA CUDA Toolkit. Figure 8 shows the DGX Station™ Deep Learning Software Stack.

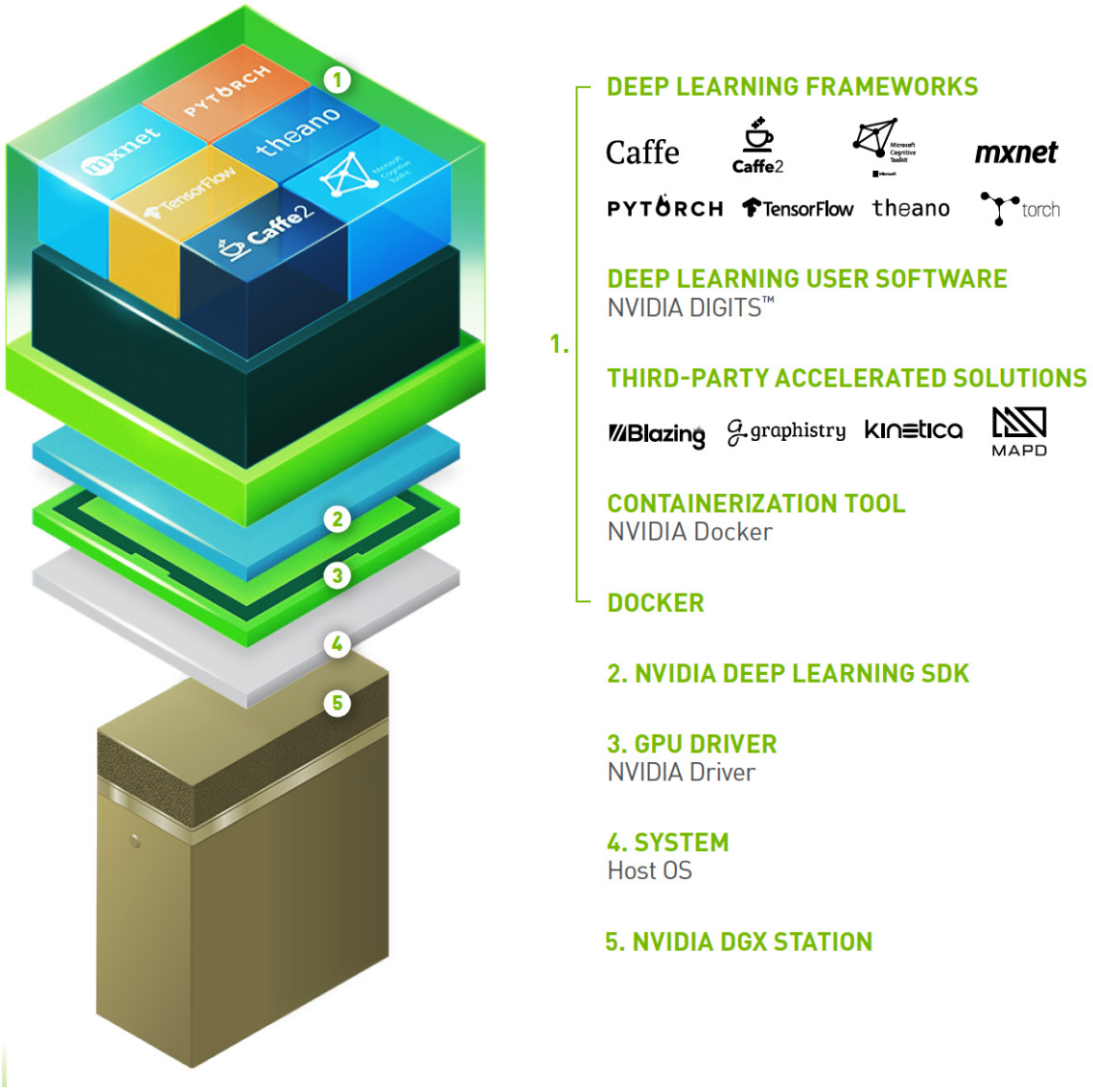

Figure 8 The DGX Station™ Deep Learning Stack

This software architecture has many advantages:

- Because each deep learning framework is in a separate container, each framework can use different versions of libraries such as libc, cuDNN, and others, and not interfere with each other.
- This also means that different frameworks, with different workloads and test procedures, can be run simultaneously on DGX Station by assigning different GPUs to each container.
- As deep learning frameworks are improved for performance or bug fixes, new versions of the containers are made available in the DGX Container Registry.
- The system is easy to maintain, and the OS image stays clean, because applications are not installed directly on the OS.

- Security updates, driver updates and OS patches can be delivered seamlessly.

The NGC container registry features containers tuned, tested, certified, and maintained by NVIDIA for the top deep learning frameworks to provide high multi-GPU performance on DGX Station.

The remainder of this section covers the key components of the DGX Station software stack (above the GPU Compute Software Driver) in detail. Section 5 provides details of optimizations to deep learning frameworks for DGX Station, and Section 6 describes details on HPC and accelerated analytics containers.

4.1 NVIDIA CUDA Toolkit

CUDA is a parallel computing platform and programming model created by NVIDIA to give application developers access to the massive parallel processing capability of GPUs. CUDA is the foundation for GPU acceleration of deep learning as well as a wide range of other computation- and memory-intensive applications ranging from astronomy, to molecular dynamics simulation, to computational finance. Today, there are over 500 GPU-accelerated applications that leverage the CUDA parallel computing platform [NVIDIA 2017b]. DGX Station is not only the fastest platform for deep learning in a workstation form factor, but the most advanced CUDA platform for a wide variety of GPU-accelerated applications that you can run under your desk.

The NVIDIA CUDA Toolkit provides a comprehensive environment for C and C++ developers building GPU-accelerated applications. The CUDA Toolkit includes NVCC, the CUDA C++ compiler for NVIDIA GPUs, a suite of libraries of GPU-accelerated algorithms, debugging and profiling tools, examples, and comprehensive programming guides and documentation. While the CUDA Toolkit comes directly installed on DGX Station as part of the Ubuntu-based operating system, it is also provided as an NVIDIA Docker container image which can be used as the base layer for any containerized CUDA application (as Figure 9 shows). In addition, the full CUDA Toolkit is embedded in every Deep Learning Framework container image.

4.2 NVIDIA Deep Learning SDK

NVIDIA provides a complete suite of GPU-accelerated libraries built on top of the CUDA parallel computing platform. The following libraries provide GPU-accelerated primitives for deep neural networks:

- **CUDA Basic Linear Algebra Subroutines library (cuBLAS):** cuBLAS is a GPU-accelerated version of the complete standard BLAS library that delivers significant speedup running on GPUs. The cuBLAS generalized matrix-matrix multiplication (GEMM) routine is a key computation used in deep neural networks, for example in computing fully connected layers.
- **CUDA Deep Neural Network library (cuDNN):** cuDNN is a GPU-accelerated library of primitives for deep neural networks. cuDNN provides highly tuned implementations of standard routines such as forward and backward convolution, pooling, normalization, and activation layers.

When deployed using the NVIDIA Docker containers for DGX Station, deep learning frameworks are automatically configured to use parallel routines optimized for the Tesla V100 architecture in DGX Station.

The NVIDIA Collective Communication Library (NCCL, pronounced “Nickel”) is a library of multi-GPU MPI-compatible collective communication primitives that are topology-aware and can be easily integrated into applications. NCCL is designed to be light-weight, depending only on common C++ and CUDA libraries. NCCL can be deployed in single-process or multi-process applications, handling required inter-process communication transparently. The NCCL API is designed to be familiar to anyone with experience using MPI collectives such as broadcast, reduce, gather, scatter, all-gather, all-reduce, or all-to-all that are optimized to achieve high bandwidth over PCIe and NVLink high-speed interconnect.

NVIDIA Docker containers for DGX Station include a version of NCCL that optimizes these collectives for the DGX Station architecture’s four-GPU second generation NVLink. When deployed using these containers, deep learning frameworks such as Caffe2, PyTorch, CNTK (Microsoft Cognitive Toolkit), and TensorFlow automatically use this version of NCCL when run on multiple GPUs.

4.3 Docker Engine Utility for NVIDIA GPUs

Over the last few years, there has been a dramatic rise in the use of software containers for simplifying deployment of data center applications at scale. Containers encapsulate an application’s dependencies to provide reproducible and reliable execution of applications and services without the overhead of a full virtual machine.

A Docker container is a mechanism for bundling a Linux application with all of its libraries, configuration files, and environment variables so that the execution environment is always the same, on whatever Linux system it runs and between instances on the same host (see Figure 9). Docker containers are user-mode only, so all kernel calls from the container are handled by the host system kernel. DGX Station uses Docker containers as the mechanism for deploying deep learning frameworks and other application software.

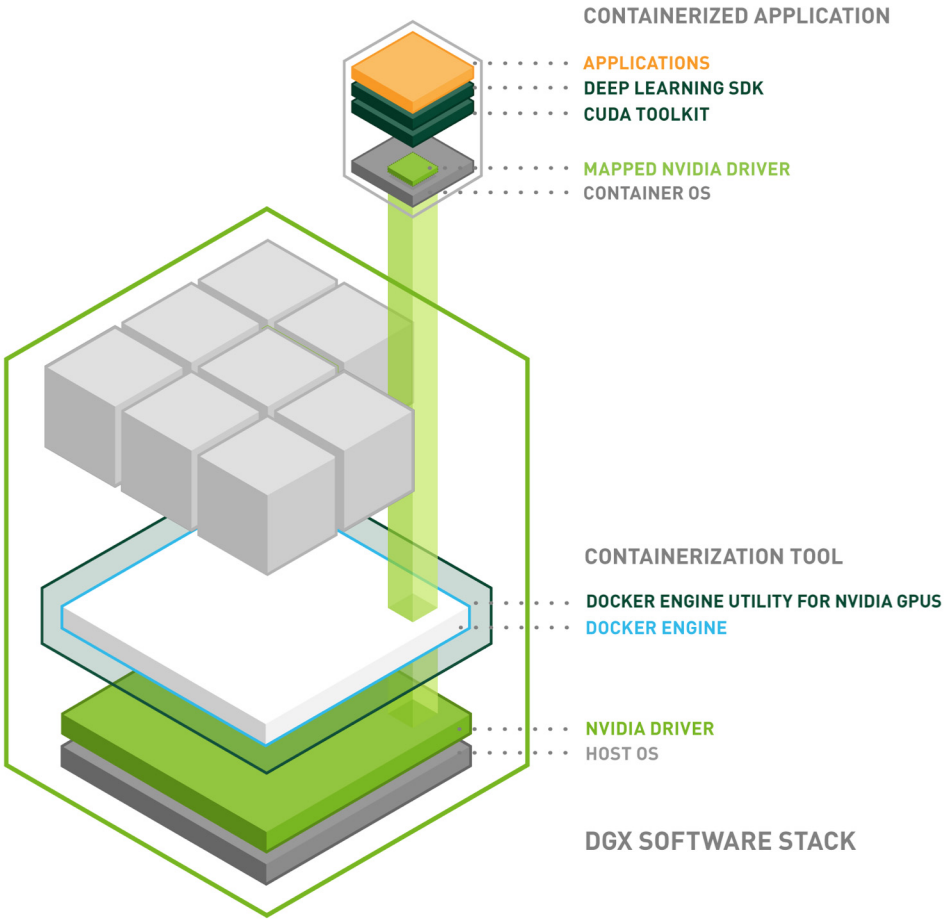

Figure 9 *Docker containers encapsulate application dependencies to provide reproducible and reliable execution. NVIDIA Docker mounts the NVIDIA driver into the containerized application image, enabling deployment of GPU-accelerated applications across any Linux GPU server with NVIDIA Docker support.*

Docker containers are platform- and hardware-agnostic, and achieve this with separation of user mode code (in the container) from kernel mode code (accessed on the host). This separation presents a problem when using specialized hardware such as NVIDIA GPUs, because GPU drivers consist of a matched set of user-mode and kernel-mode modules. An early work-around to this problem was to fully install the NVIDIA drivers inside the container and map in the character devices corresponding to the NVIDIA GPUs (for example, **`/dev/nvidia0`**) on launch. This solution is brittle because the version of the host driver must exactly match the version of the driver installed in the container. This requirement drastically reduced the portability of these early containers, undermining one of Docker's more important features.

To enable portability in Docker images that leverage GPUs, NVIDIA developed Docker Engine Utility for NVIDIA GPUs [NVIDIA Corporation 2015], an open-source project that provides the `nvidia-docker` command-line tool to mount the user-mode components of the NVIDIA driver and the GPUs into the Docker container at launch, as Figure 9 shows. For this to work, it is essential that the developer does not install an NVIDIA driver into the Docker image at docker build time.

The `nvidia-docker` tool is essentially a wrapper around Docker that transparently provisions a container with the necessary components to execute code on the GPU.

4.4 NVIDIA Container Access

NVIDIA provides a library of pre-integrated docker containers for use with DGX Station and other compatible systems. This library of NVIDIA GPU-optimized software eliminates the complexity typically associated with deploying deep learning software and is updated monthly, making it simple to get started and to stay up to date with the most popular deep learning frameworks.

At a high level, there are two steps to use these containers on DGX Station.

1 First, you will use a web browser and register with NVIDIA GPU Cloud (NGC).

   There is no charge to access and use the NVIDIA containers on NGC.

   After you are registered, you will have access to the complete library of GPU-optimized containers, including NVIDIA-optimized deep learning frameworks such as TensorFlow and PyTorch, third-party managed HPC applications, NVIDIA HPC visualization tools, third party accelerated analytics software, and NVIDIA's programmable inference accelerator, NVIDIA TensorRT.

   The NGC user interface provides detailed instructions for logging in to the container registry from your system, and the exact command needed to pull each container.

   All these containers are described in more detail in Section 5.

2 After you have set up your NGC account and chosen which container to pull, simply open a shell on your DGX Station, log in to the NGC container registry with the provided credentials, and enter the appropriate `docker pull` command.

   This command is provided on the container's page in the NVIDIA GPU Cloud user interface, so you can easily copy and paste it.

   The container will download to your DGX Station. After the download is complete, the container is ready to run.

For more details, see the NGC Container User Guide at:

http://docs.nvidia.com/ngc/ngc-user-guide/

5 DEEP LEARNING FRAMEWORKS AND TOOLS FOR DGX STATION

The NVIDIA Deep Learning SDK accelerates widely used deep learning frameworks such as NVCaffe, Caffe2, Microsoft Cognitive Toolkit, MXNet, TensorFlow, Theano, PyTorch, Torch, and TensorRT by optimizing them for DGX products.

The DGX Station software stack provides containerized versions of these frameworks optimized for the system. These frameworks, including all necessary dependencies, are pre-built, tested, and ready to run. For users who need more flexibility to build custom deep learning solutions, each framework container image also includes the framework source code to enable custom modifications and enhancements, along with the complete software development stack described in Section 4.

Most deep learning frameworks have begun to merge support for half-precision training techniques that exploit Tensor Core calculations in Volta. Some frameworks include support for FP16 storage and Tensor Core math. To achieve optimum performance, you can train a model using Tensor Core math and FP16 mode on some frameworks. For information about which frameworks are optimized for Volta, see the Training with Mixed Precision User Guide at:

http://docs.nvidia.com/deeplearning/sdk/mixed-precision-training/

Also note that because NVIDIA constantly improves these frameworks, it is good practice to check the release notes for each of these containers for latest information:

http://docs.nvidia.com/deeplearning/dgx/

5.1 NVCaffe

Caffe[3] is a deep learning framework made with flexibility, speed, and modularity in mind. It was originally developed by the Berkeley Vision and Learning Center (BVLC) and by community contributors.

NVIDIA Caffe [NVIDIA Corporation, 2017c], also known as NVCaffe, is an NVIDIA-maintained fork of BVLC Caffe tuned for NVIDIA GPUs, particularly in multi-GPU configurations. It includes multi-precision support as well as other NVIDIA-enhanced features and offers performance specially tuned for the NVIDIA DGX Station.

NVCaffe supports single and multi-GPU execution.

3. http://caffe.berkeleyvision.org/

The following list summarizes NVIDIA's NVCaffe optimizations and changes for DGX products.

- 16-bit (half) floating point train and inference support.
- Mixed-precision support. It allows storing and/or computing data in either 64-, 32- or 16-bit formats. Precision can be defined for every layer (forward and backward passes maybe different, too), or it can be set for the whole Net.
- A parallelized parser and image transformer for improved I/O performance.
- Full utilization of Volta Tensor Cores for faster 16-bit training.
- Integration with cuDNN v7.
- Integration with CUDA 9.
- Automatic selection of the best cuDNN convolution algorithm.
- Integration with latest version of NCCL for improved multi-GPU scaling.
- Optimized GPU memory management for data and parameters storage, I/O buffers and workspace for convolutional layers.
- Parallel data parser and transformer for improved I/O performance.
- Parallel back-propagation and gradient reduction on multi-GPU systems.
- Fast solvers implementation with fused CUDA kernels for weights and history update.
- Multi-GPU test phase for even memory load across multiple GPUs.
- Backward compatibility with BVLC Caffe and NVCaffe 0.15.
- Extended set of optimized models (including 16-bit floating point examples).

5.2 Caffe2

Caffe2[4] is a deep-learning framework designed to easily express all model types, for example, convolutional neural networks (CNNs), recurrent neural networks (RNNs), and more, in a friendly Python-based API, and execute them using a highly efficient C++ and CUDA backend.

Caffe2's flexible API lets users define models for inference or training using expressive, high-level operations. The Python interface allows easy control and visualization of the inference or training process.

Caffe2 supports single- and multi-GPU execution, along with multi-node execution.

The following list summarizes NVIDIA's Caffe2 optimizations and changes for DGX products.

- Integration with cuDNN v7.
- Integration with CUDA 9.
- Integration with latest version of NCCL.

4. https://research.fb.com/downloads/caffe2/

5.3 Microsoft Cognitive Toolkit

The Microsoft Cognitive Toolkit[5] (also known as CNTK) is a unified deep-learning toolkit that allows users to easily realize and combine popular model types such as feed-forward deep neural networks (DNNs), convolutional neural networks (CNNs), and recurrent neural networks (RNNs). Microsoft Cognitive Toolkit implements stochastic gradient descent (SGD) learning with automatic differentiation and parallelization across multiple GPUs and servers[6]. Microsoft Cognitive Toolkit can be called as a library from Python or C++ applications, or executed as a standalone tool by using the BrainScript model description language.

NVIDIA and Microsoft worked closely to accelerate the Microsoft Cognitive Toolkit on GPU-based systems such as DGX Station and Azure N-Series virtual machines. This combination offers startups and major enterprises alike tremendous ease of use and scalability because a single framework can be used first to train models on premises with the DGX Station and later to deploy those models at scale on a cluster of DGX-1 systems or in the Microsoft Azure cloud[7].

Microsoft Cognitive Toolkit supports single-GPU and multi-GPU execution.

The following list summarizes NVIDIA's Microsoft Cognitive Toolkit optimizations and changes for DGX systems.

- Integration with cuDNN v7.
- Integration with CUDA 9.
- Integration with latest version of NCCL with NVLink support for improved multi-GPU scaling.
- Integration of Volta hardware support.
- Image reader pipeline improvements allow AlexNet [Krizhevsky et al. 2012] to train at over 12,000 images/second.
- Reduced GPU memory overhead for multi-GPU training by up to 2 GB per GPU.
- Dilated convolution support.

5.4 MXNet

MXNet[8] is a deep learning framework designed for both efficiency and flexibility, which allows you to mix symbolic and imperative programming to maximize efficiency and productivity. At the core of MXNet is a dynamic dependency scheduler that automatically parallelizes both symbolic and imperative operations on the fly. A graph optimization layer on top of the scheduler makes symbolic

5. https://www.microsoft.com/en-us/research/product/cognitive-toolkit/
6. Source: https://github.com/Microsoft/CNTK#what-is-the-microsoft-cognitive-toolkit
7. For information on using Microsoft Cognitive Toolkit in Azure see https://docs.microsoft.com/en-us/cognitive-toolkit/CNTK-on-Azure
8. http://mxnet.io

execution fast and memory efficient. MXNet is portable and lightweight, and scales to multiple GPUs and multiple machines.

MXNet supports single and multi-GPU execution.

The following list summarizes NVIDIA's MXNet optimizations and changes for DGX products.

- Integration with cuDNN v7.
- Integration with CUDA 9.
- Integration with latest version of NCCL.
- Improved input pipeline for image processing.
- Optimized embedding layer CUDA kernels.
- Optimized tensor broadcast and reduction CUDA kernels.
- Support for Tensor Core in convolutions, deconvolutions and fully connected layers on Volta.
- Support for mixed precision training with SGD optimizer.
- Streamlined FP16 examples for image classification.
- Optimized input pipeline for image processing.

5.5 TensorFlow

TensorFlow[9] is an open-source software library for numerical computation using data flow graphs. Nodes in the graph represent mathematical operations, while the graph edges represent the multidimensional data arrays (tensors) that flow between them. This flexible architecture lets you deploy computation to one or more CPUs or GPUs in a desktop, server, or mobile device without rewriting code.

TensorFlow was originally developed by researchers and engineers working on the Google Brain team within Google's Machine Intelligence research organization for the purposes of conducting machine learning and deep neural networks research. The system is general enough to be applicable in a wide variety of other domains, as well.

For visualizing TensorFlow results, the TensorFlow Docker image also contains TensorBoard (https://www.tensorflow.org/get_started/summaries_and_tensorboard). TensorBoard is a suite of visualization tools. For example, you can view the training histories as well as an image of the network model.

TensorFlow supports single- and multi-GPU execution.

9. https://www.tensorflow.org

The following list summarizes NVIDIA's TensorFlow optimizations and changes for DGX products.

- Integration with cuDNN v7.
- Integration with CUDA 9.
- Replacement of `libjpeg` with `libjpeg-turbo`.
- Integration with latest version of NCCL with NVLink support for improved multi-GPU scaling.
- Support for the ImageNet preprocessing script.

5.6 Theano

Theano[10] is a Python library that allows you to efficiently define, optimize, and evaluate mathematical expressions involving multi-dimensional arrays. Theano has been powering large-scale computationally intensive scientific investigations since 2007.

Theano supports single and limited multi-GPU execution.

The following list summarizes NVIDIA's Theano optimizations and changes for DGX products.

- Integration with cuDNN v7.
- Integration with CUDA 9.
- Integration with latest version of NCCL.
- Runtime code generation: evaluate expressions faster.
- Extensive unit-testing and self-verification: detect and diagnose many types of errors.

5.7 PyTorch

PyTorch[11] is a Python package that provides two high-level features:

- Tensor computation (like NumPy) with strong GPU acceleration
- Deep neural networks (DNNs) built on a tape-based autograd system

You can reuse your favorite Python packages such as NumPy, Scipy and Cython to extend PyTorch when needed.

PyTorch supports single and multi-GPU execution.

10. http://deeplearning.net/software/theano/
11. http://pytorch.org/

The following list summarizes NVIDIA's PyTorch optimizations and changes for DGX products.

- Supports Tensor Core operations for convolutions and GEMMs on Volta hardware.
- The examples directory contains examples of ImageNet and LSTM training scripts that uses FP16 data, as well as how to do training with FP16.
- Matrix multiplication on fp16 inputs uses Tensor Core math when available.
- A custom batch normalization layer is implemented to use cuDNN for batch normalization with FP16 inputs.
- Integration with cuDNN v7 with support for Tensor Core math when available.
- Integration with CUDA 9.
- Integration with latest version of NCCL.

5.8 Torch

Torch[12] is a scientific computing framework with wide support for deep learning algorithms. Torch is easy to use and efficient, thanks to an easy and fast scripting language, Lua, and an underlying C/CUDA implementation. Torch offers popular neural network and optimization libraries that are easy to use yet provide maximum flexibility to build complex neural network topologies.

Torch supports single and multi-GPU execution.

The following list summarizes NVIDIA's Torch optimizations and changes for DGX products.

- Integration with cuDNN v7.
- Integration with CUDA 9.
- Integration with latest version of NCCL with NVLink support for improved multi-GPU scaling.
- Buffering of parameters to be communicated by NCCL to reduce latency overhead.
- cuDNN bindings for recurrent networks (RNN, GRU, LSTM), including persistent versions which greatly improve the performance of small batch training.
- Dilated convolution support.
- Support for 16- and 32-bit floating point (FP16 and FP32) data input to cuDNN routines.
- Support for operations on FP16 tensors (using FP32 arithmetic).

12. http://torch.ch

5.9 DIGITS

The NVIDIA Deep Learning GPU Training System (DIGITS)[13] puts the power of deep learning into the hands of engineers and data scientists.

DIGITS can be used to rapidly train highly accurate deep neural network (DNNs) for image classification, segmentation, and object detection tasks. DIGITS simplifies common deep learning tasks such as managing data, designing and training neural networks on multi-GPU systems, monitoring performance in real time with advanced visualizations, and selecting the best performing model from the results browser for deployment. DIGITS is completely interactive so that data scientists can focus on designing and training networks rather than programming and debugging.

The following list summarizes NVIDIA's DIGITS optimizations and changes for DGX products.

- DIGITS runs on top of NVCaffe, Torch, and TensorFlow frameworks which are optimized for DGX.
- Integration with cuDNN v7.
- Integration with CUDA 9.
- Integration with latest version of NCCL.

5.10 TensorRT

NVIDIA TensorRT™ is a high-performance deep learning inference optimizer and runtime that delivers low latency, high-throughput inference for deep learning applications on NVIDIA GPUs. TensorRT takes a network definition and optimizes it by merging tensors and layers, transforming weights, choosing efficient intermediate data formats, and selecting from a large kernel catalog based on layer parameters and measured performance.

After you have trained a neural network, you can optimize and deploy the model for GPU inferencing with TensorRT. The TensorRT container provides an easy to use container for TensorRT development. The container allows for the TensorRT samples to be built, modified and executed. For more information about optimizing and deploying using TensorRT, see the Deep Learning SDK Documentation at:

http://docs.nvidia.com/deeplearning/sdk/

The TensorRT API includes importers for trained deep learning models in a variety of standard formats. Once imported, TensorRT can optimize a network and generate a run-time engine for deployment. TensorRT also includes an infrastructure that allows you to perform inference using reduced-precision arithmetic to take advantage of the high performance and efficiency reduced-precision capabilities of Pascal and Volta GPUs.

13. https://developer.nvidia.com/digits

TensorRT has both C++ and Python APIs. The C++ API allows developers to import, calibrate, generate and deploy networks using C++. Networks can be imported directly from NVCaffe or from other frameworks via the UFF format. They may also be created programmatically by instantiating individual layers and setting parameters and weights directly.

TensorRT 3.0 introduces a Python API to allow developers to easily parse models (for example from NVCaffe, TensorFlow, NumPy compatible frameworks) and generate and run PLAN files within Python-based development environments. Currently, all TensorRT functionality except for INT8 calibrators and RNNs is exposed via the Python API. The Python API also introduces compatibility with NumPy arrays for layer weights and GPU-resident input and output data (via PyCUDA). The Python API provides a set of utility functions to address common tasks including parsing NVCaffe and UFF models and writing PLAN files.

6 HPC AND ACCELERATED ANALYTICS APPLICATIONS ON DGX STATION

6.1 HPC Applications

Docker containers for some of the most popular HPC applications are available for easy application deployment.

6.1.1 NAMD

NAMD (NAnoscale Molecular Dynamics) is a production-quality molecular dynamics application designed for high-performance simulation of large biomolecular systems. One of the most popular NAMD use case is gain insight on the interaction between the dynamic HIV structure and the human protein to help develop new therapeutic drugs.

6.1.2 LAMMPS

Large-scale Atomic/Molecular Massively Parallel Simulator (LAMMPS) is a classical molecular dynamics code to model atoms or, more generically, as a parallel particle simulator at the atomic, meso, or continuum scale. It has potentials for solid-state materials (for example, metals and semiconductors), soft matter (such as, biomolecules and polymers) and coarse-grained or mesoscopic systems. LAMMPS is used to simulate polymer based nanocarriers for targeted drug discoveries.

6.1.3 GROMACS

GROMACS is a molecular dynamics application designed to simulate Newtonian equations of motion for systems with hundreds to millions of particles. GROMACS simulates biochemical molecules like proteins, lipids, and nucleic acids that have a lot of complicated bonded interactions.

6.1.4 RELION

RELION (REgularized LIkelihood OptimizatioN) implements an empirical Bayesian approach for analysis of electron cryo-microscopy (Cryo-EM). Specifically it provides methods of refinement of singular or multiple 3D reconstructions as well as 2D class averages. RELION is an important tool in the study of living cells.

RELION is comprised of multiple steps to cover the entire single-particle analysis workflow. Steps include beam-induced motion-correction, CTF estimation, automated particle picking, particle extraction, 2D class averaging, 3D classification, high-resolution refinement in 3D. RELION can process movies generated from direct-electron detectors, apply final map sharpening and perform local-resolution estimation.

6.1.5 GAMESS

The General Atomic and Molecular Electronic Structure Systems (GAMESS) program simulates molecular quantum chemistry, allowing users to calculate various molecular properties and dynamics. It is used to computational chemistry calculations including DFT which is used to understand the potential of a drug molecule binding with proteins for drug discovery.

6.2 Analytics Applications

The first three available applications are H2O.ai's Driverless AI, Kinetica, and MapD

6.2.1 H2O's Driverless AI

Driverless AI by H2O.ai is a machine learning platform that automates many of the most difficult data science and machine learning workflows, such as feature engineering, model validation, model tuning, model selection and model deployment. The community version of Driverless AI is a fully functional version with a 30-day trial on DGX systems and supported NGC platforms. Customers can achieve up to 40x speedups on GPU-accelerated algorithms vs. on CPUs.

6.2.2 Kinetica

Kinetica is a GPU-accelerated, in-memory analytics database that delivers truly real-time deep insights with unprecedented performance for ingesting, processing and visualizing data in motion and at rest. Up to 100x faster than traditional databases, Kinetica also converges Artificial Intelligence and Business Intelligence, allowing businesses to directly apply machine learning in-database to effectively engage customers and find new business opportunities. In addition, Kinetica comes with a native geospatial and visualization pipeline for interactive location-based analytics. Customers interested in exploring the power and potential benefits of Kinetica's next-gen data platform can access the software using a free 90-day trial through Kinetica's website or as part of the NVIDIA registry on DGX and NGC.

6.2.3 MapD

MapD is a GPU-accelerated platform with an open-source SQL engine called MapD Core and an integrated visualization system called MapD Immerse. Open source MapD Core is now containerized on DGX systems and NGC. Customers considering purchasing a license can access the software for a free trial for 30 days on Amazon Web Services or as part of the NVIDIA registry on DGX and NGC. Data scientists using MapD experience unparalleled analytic speed, constant innovation from the open source community, and interactive visual exploration of the data used to build machine learning models.

7 RESULTS: DGX STATION FOR HIGHEST DEEP LEARNING PERFORMANCE AT YOUR DESK

7.1 Volta Architecture Performance

As general computing performance increases are flattening ("The End of Moore's Law", [Huang, 2017]), NVIDIA is able to dramatically increase performance of its GPUs with each generation. The delivery of the Volta architecture, only within a year of the launch of the previous Pascal architecture, was especially impressive. Although impressive performance gains could be achieved for different kind of computations, such as single or double precision calculations often needed for HPC workloads, the biggest performance gains can be seen in mixed precision (half and single precision) operations thanks to the new Tensor Core architecture.

The bars in Figure 10 represent training performance in images per second for the ResNet-50 deep neural network architecture using the 17.09 DGX optimized containers for each of the three framework displayed (NVCaffe, Caffe2, and MXNet). It compares the performance of using 4x P100 PCIe GPUs with 4x V100 GPUs available in DGX Station.

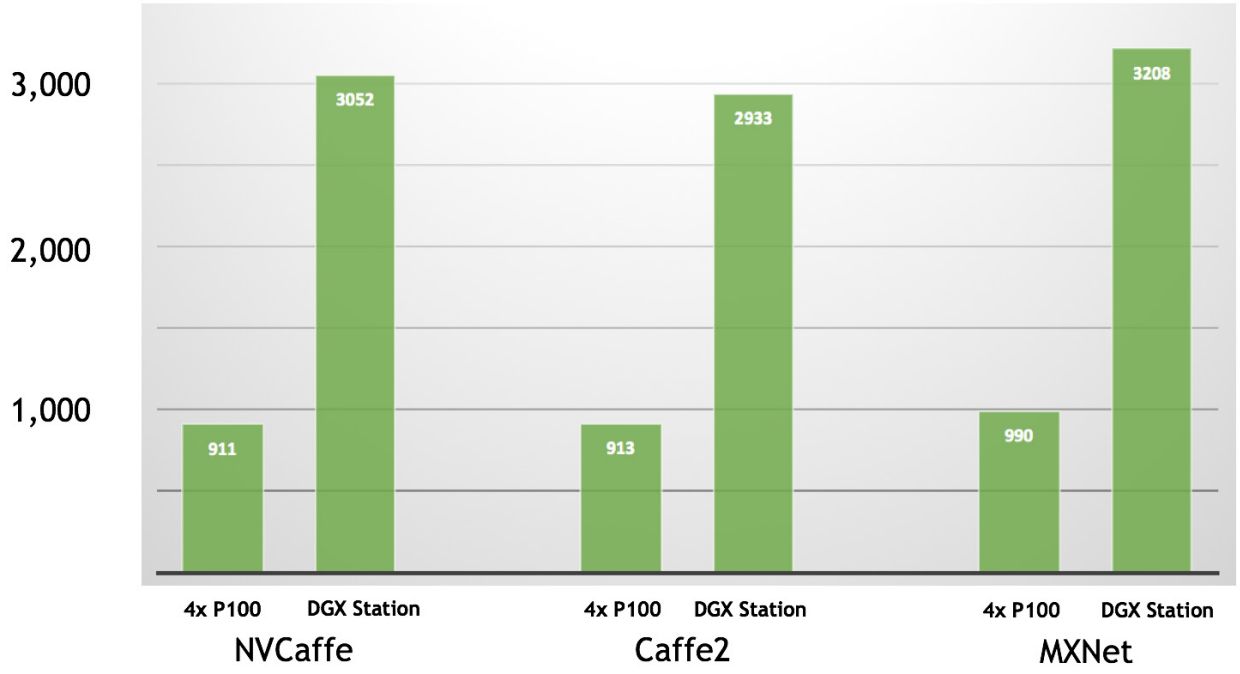

Figure 10 *DGX Station with 4x Tesla V100. ResNet-50 Training, Volta with Tensor Core (mixed precision FP16 & FP32), 17.09 DGX optimized container.*
4x Tesla P100 PCIe. ResNet-50 Training, FP32, 17.09 DGX Optimized container.
Score: Images per Second

7.2 Scalability

As shown in Section 3.2, it can be demonstrated nicely that scaling of training performance on DGX Station from 1, to 2, to 4 Tesla V100 GPUs is close to linear.

The bars in Figure 11 represent training performance in images per second for the ResNet-50 deep neural network architecture using the 17.09 DGX optimized containers for each of the three framework displayed (NVCaffe, Caffe2, and MXNet). These benchmark numbers were achieved using mixed precision with the new Tensor Cores available in V100, and show the linear scalability of V100s connected via NVLink.

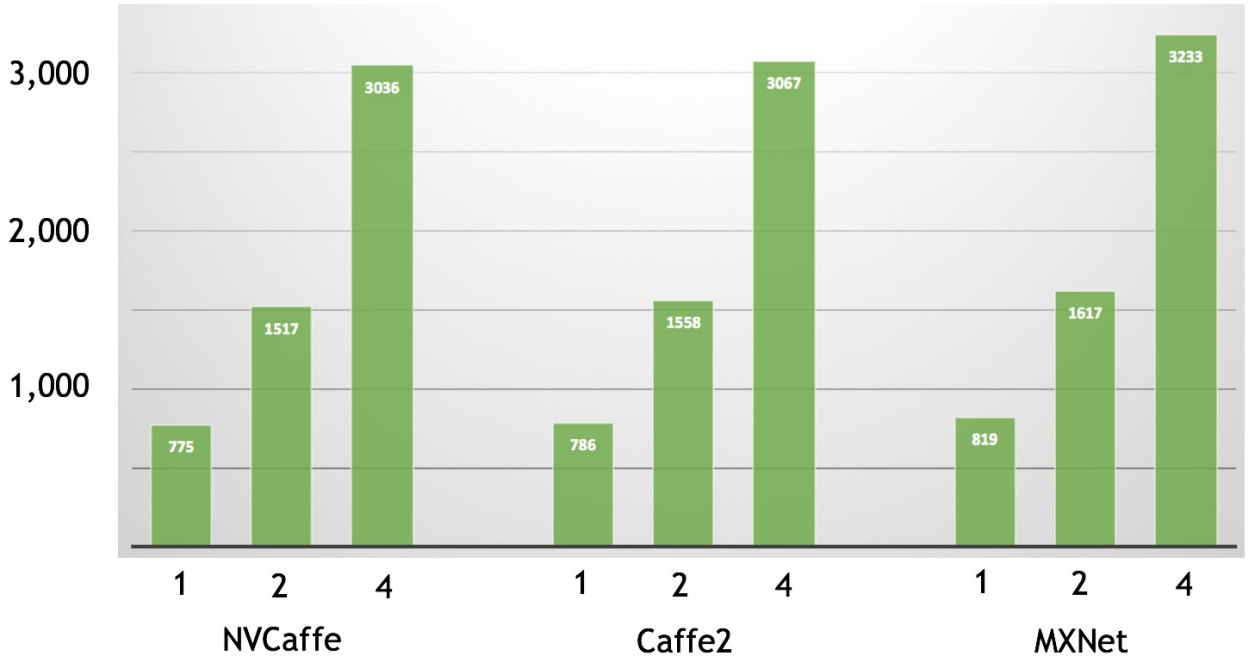

Figure 11 *DGX Station with 4x Tesla V100. ResNet-50 Training, Volta with Tensor Core (mixed precision FP16 & FP32), 17.11 DGX optimized container. DGX OS Desktop 3.1.4 with driver 384.111 Score: Images per Second*

7.3 Continuous Optimizations

Years of technology interlock between NVIDIA Engineering and the developers of the leading deep learning frameworks have resulted in steady, incremental progress, advancing the art and science of deep learning performance.

Figure 12 highlights the type of improvements that were achieved in the course of only one year.

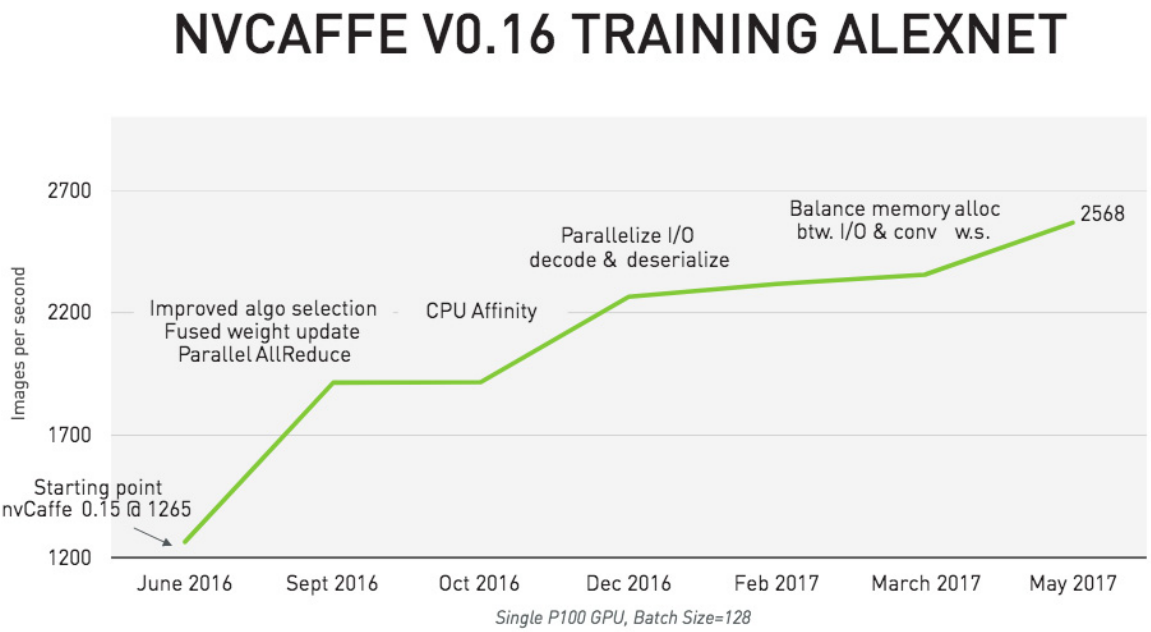

Figure 12 Performance improvements of NVCaffe, AlexNet, from 2016 to 2017

With each new framework release comes enhancements to DGX software, that have been planned early on in the development cycle, ahead of the actual framework release to the general public. These improvements typically fall into several pillars of targeted development that NVIDIA engineers have perfected over time, performance gains being one of the most important goals.

The bars in Figure 13 represent training performance in images per second for the ResNet-50 deep neural network architecture comparing the 17.09 DGX optimized containers for each of the three framework displayed (NVCaffe, Caffe2, and MXNet) with the 17.12 versions. As you can see, in only three months, NVIDIA engineers managed to achieve performance gains of about 4% to 7%.

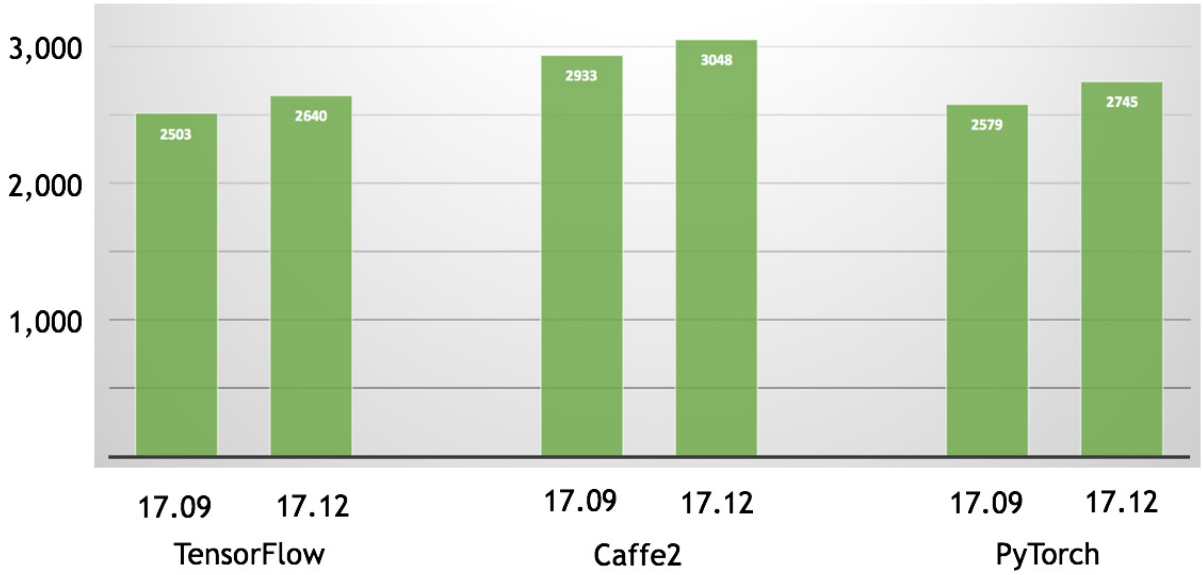

Figure 13 *DGX Station with 4x Tesla V100. ResNet-50 Training, Volta with Tensor Core (mixed precision FP16 & FP32), 17.09 & 17.12 DGX optimized containers.*
Score: Images per Second

References

Huang, J. 2016. The Intelligent Industrial Revolution.
https://blogs.nvidia.com/blog/2016/10/24/intelligent-industrial-revolution/

Huang, J. 2017. AI Is Eating Software.
https://blogs.nvidia.com/blog/2017/05/24/ai-revolution-eating-software/

Krizhevsky, A. 2014. One weird trick for parallelizing convolutional neural networks. *arXiv [cs.NE]*.
http://arxiv.org/abs/1404.5997

Krizhevsky, A., Sutskever, I., and Hinton, G.E. 2012. ImageNet Classification with Deep Convolutional Neural Networks. In: F. Pereira, C.J.C. Burges, L. Bottou and K.Q. Weinberger, eds., Advances in Neural Information Processing Systems 25. Curran Associates, Inc., 1097–1105.
http://papers.nips.cc/paper/4824-imagenet-classification-with-deep-convolutional-neural-networks.pdf

NVIDIA Corporation. 2015. Docker Engine Utility for NVIDIA GPUs. Github.
https://github.com/NVIDIA/nvidia-docker

NVIDIA Corporation. 2017a. Accelerated Workstation: Run Deep Learning Workloads at Your Desk.
http://www.nvidia.com/object/idc-spotlight.html

NVIDIA Corporation. 2017b. GPU-Accelerated Applications.
http://www.nvidia.com/content/gpu-applications/PDF/gpu-applications-catalog.pdf

NVIDIA Corporation. 2017c. *NVCaffe branch*.
https://github.com/NVIDIA/caffe

NVIDIA Corporation. 2017d. NVIDIA® Tesla® V100 GPU Architecture — The World's Most Advanced Data Center Accelerator.
http://www.nvidia.com/object/volta-architecture-whitepaper.html

Patel, C. 2017. Artificial Intelligence Index Annual Report 2017 - Takeaways for a Millennial.
https://medium.com/@chaityapatel/artificial-intelligence-index-annual-report-2017-takeaways-for-a-millennial-9fc4627b4bd2

Schroepfer, M. 2016. F8 2016 Day 1 Keynote.
https://developers.facebook.com/videos/f8-2016/keynote/

Wheat, S. 2017. Empowering Business Growth with HPC Innovation, HPC Wire.
https://www.hpcwire.com/solution_content/hpe/government-academia/empowering-business-growth-hpc-innovation/

Wu, R., Yan, S., Shan, Y., Dang, Q., and Sun, G. 2015. Deep Image: Scaling up Image Recognition. *arXiv [cs.CV]*.
http://arxiv.org/abs/1501.02876

Notice

Trademarks

Copyright