NetApp Verified Architecture

ONTAP AI – NVIDIA DGX-2 POD with NetApp AFF A800

NVA Design

David Arnette and Sung-Han Lin, NetApp
May 2019 | NVA-1135-DESIGN | Version 1.0

In partnership with

NVIDIA

NetApp

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

1 Executive Summary

NVIDIA® DGX-2™ GPU servers are enabling the next generation of machine learning (ML) and artificial intelligence (AI) applications. NetApp® AFF storage systems deliver extreme performance and industry-leading hybrid cloud data management capabilities. NetApp and NVIDIA have partnered to create the NetApp ONTAP® AI infrastructure to offer customers a turnkey solution for supporting AI and ML workloads with enterprise-class performance, reliability, and support.

This document describes a reference architecture for using NetApp AFF storage systems with NVIDIA DGX-2 servers. This design and validation of the architecture are consistent with the NetApp Verified Architecture (NVA) for NVIDIA DGX-1™ servers found in NVA-1121. This document contains the details used for this validation as well as benchmark testing results that demonstrate the performance capabilities of the solution.

The configuration used in this validation is one of many possible ONTAP AI variations, and it delivers the highest levels of performance for enterprise-scale AI and ML operations. ONTAP AI enables customers to start small and seamlessly scale from half-rack to rack to data center scale.

2 Program Summary

2.1 NetApp Verified Architecture Program

The NVA program solutions are:

- Thoroughly tested
- Prescriptive in nature
- Designed to minimize deployment risks
- Designed to accelerate time to market

This document is for NetApp and partner solutions engineers and customer strategic decision makers. It describes the architecture design considerations that were used to determine the specific equipment, cabling, and configurations required for a particular environment.

2.2 NetApp ONTAP AI Solution

The NetApp ONTAP AI converged infrastructure architecture, powered by NVIDIA DGX servers and NetApp cloud-connected storage systems, was developed and verified by NetApp and NVIDIA. It gives IT organizations a prescriptive architecture that:

- Eliminates design complexities
- Allows independent scaling of compute and storage
- Enables customers to start small and scale seamlessly
- Offers a range of storage options for various performance and cost points

NetApp ONTAP AI integrates DGX servers, NVIDIA V100 Tensor Core GPUs, and a NetApp AFF A800 storage system with state-of-the-art networking. NetApp ONTAP AI simplifies artificial intelligence deployments by eliminating design complexity and guesswork. Customers can start small and grow nondisruptively while intelligently managing data from the edge to the core to the cloud and back.

Figure 1 shows several variations in the ONTAP AI family of solutions. The AFF A800 system has been verified with three DGX-2 servers and has demonstrated sufficient performance headroom to support more DGX-2 servers without impacting storage throughput or latency. Furthermore, by adding more network switches and storage controller pairs to the ONTAP cluster, the solution can scale to multiple racks to deliver extremely high throughput, accelerating training and inferencing. This approach offers the

flexibility to alter the ratios of compute to storage independently based on the size of the data lake, the deep learning (DL) models that are used, and the required performance metrics.

Figure 1) NetApp ONTAP AI solution rack-scale architecture.

- 1x 5624Q,1x A220, 1x DGX-1 server
- 2x 5624Q, 1x A300, 2x DGX-1 servers
- 2x 5624Q,1x A700, 4x DGX-1 servers
- 2x 3232C,1x A800, 9x DGX-1 servers
- 2x 3232C,1x A800, 3x DGX-2 servers

The number of DGX servers and AFF systems per rack depends on the power and cooling specifications of the rack in use. Final placement of the systems is subject to computational fluid dynamics analysis, airflow management, and data center design.

3 Deep Learning Data Pipeline

Deep learning is the engine that enables businesses to detect fraud, improve customer relationships, optimize supply chains, and deliver innovative products and services in an increasingly competitive marketplace. The performance and accuracy of DL models are significantly improved by increasing the size and complexity of the neural network as well as the amount and quality of data that is used to train the models.

Given the massive datasets required, it is crucial to architect an infrastructure that offers the flexibility to deploy across environments. At a high level, an end-to-end DL deployment consists of three phases through which the data travels: the edge (data ingest), the core (training clusters and a data lake), and the cloud (archive, tiering, and dev/test). This is typical of applications such as the Internet of Things (IoT) for which data spans all three realms of the data pipeline.

Figure 2) Components of the edge-core-cloud data pipeline.

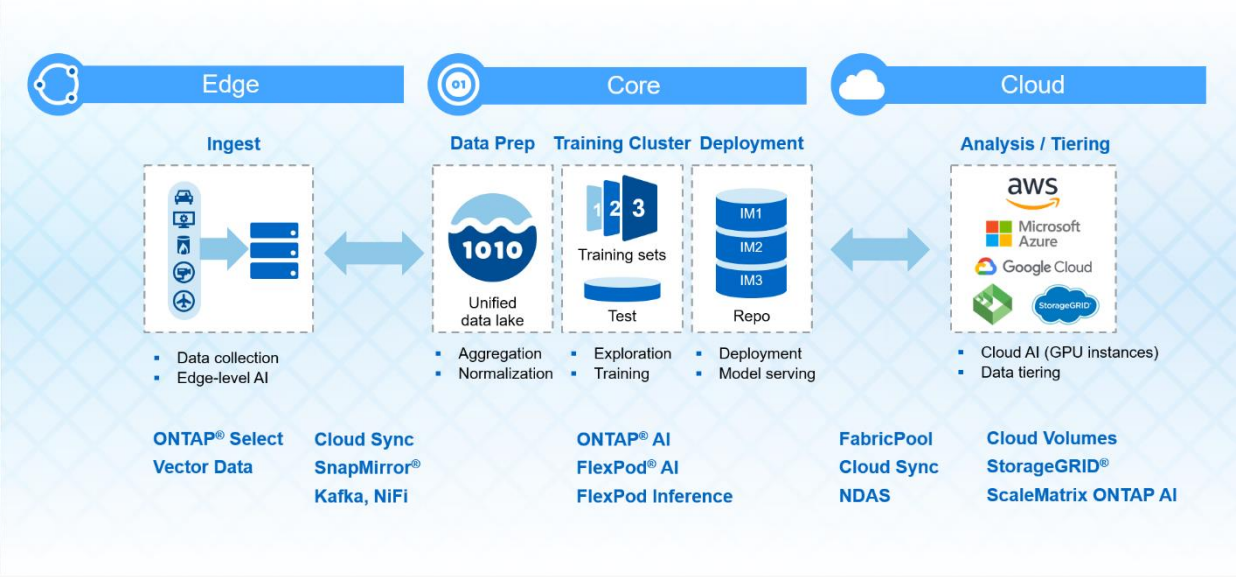


Figure 2 presents an overview of the components in each of the three realms.

- **Ingest.** Data ingestion usually occurs at the edge; for example, by capturing data streaming from autonomous cars or point-of-sale devices. Depending on the use case, an IT infrastructure might be needed at or near the ingestion point. For instance, a retailer might need a small footprint in each store that consolidates data from multiple devices.
- **Data prep.** Preprocessing is necessary to normalize and cleanse the data before training. Preprocessing takes place in a data lake, possibly in the cloud, in the form of an Amazon S3 tier or in on-premises storage systems such as a file store or an object store.
- **Training.** For the critical training phase of DL, data is typically copied from the data lake into the training cluster at regular intervals. The servers that are used in this phase use GPUs to parallelize computations, creating a tremendous appetite for data. Meeting the raw I/O bandwidth needs is crucial for maintaining high GPU utilization.
- **Deployment.** The trained models are tested and deployed into production. Alternatively, they could be fed back to the data lake for further adjustments of input weights; or in IoT applications the models could be deployed to the smart edge devices.
- **Analysis and tiering.** New cloud-based tools become available at a rapid pace, so additional analysis or development work might be conducted in the cloud. Cold data from past iterations might be saved indefinitely. Many AI teams prefer to archive cold data to object storage in either a private or a public cloud. Based on compute requirements, some applications work well with object storage as the primary data tier.

Depending on the application, DL models work with large amounts of structured and unstructured data. This difference imposes a varied set of requirements on the underlying storage system, both in terms of size of the data that is being stored and the number of files in the dataset.

Some of the high-level storage requirements include:

- The ability to store and retrieve millions of files concurrently
- Storage and retrieval of diverse data objects such as images, audio, video, and time-series data
- Delivery of high parallel performance at low latencies to meet the GPU processing speeds
- Seamless data management and data services that span the edge, the core, and the cloud

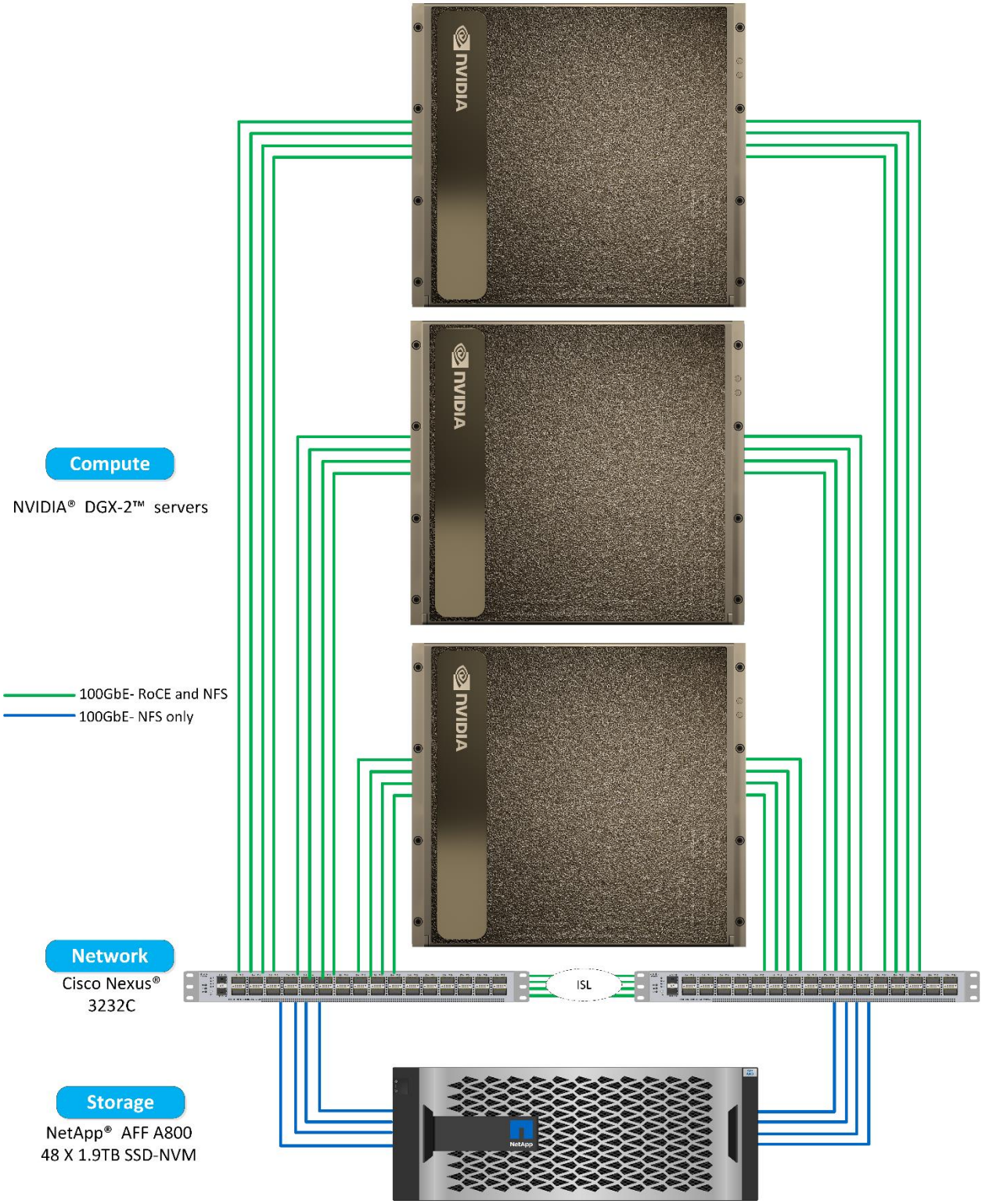
Combined with superior cloud integration and the software-defined capabilities of NetApp ONTAP, AFF systems support a full range of data pipelines that spans the edge, the core, and the cloud for DL. This document focuses on solutions for the training and inference components of the data pipeline.

4 Solution Overview

DL systems leverage algorithms that are computationally intensive and that are uniquely suited to the architecture of GPUs. Computations that are performed in DL algorithms involve an immense volume of matrix multiplications running in parallel. Advances in individual and clustered GPU computing architectures that leverage the DGX servers have made them the preferred platform for workloads such as high-performance computing (HPC), DL, and analytics. Maximizing performance in these environments requires a supporting infrastructure that can keep GPUs fed with data. Dataset access must therefore be provided at ultra-low latencies with high bandwidth.

This solution was implemented with one NetApp AFF A800 system, three DGX-2 servers, and two Cisco Nexus 3232C 100GbE switches. Each DGX-2 server is connected to the Nexus switches with eight 100GbE connections that are used for inter-GPU communications by using remote direct memory access (RDMA) over Converged Ethernet (RoCE). Traditional IP communications for NFS storage access also occur on these links. Each storage controller is connected to the network switches by using four 100GbE links. Figure 3 shows the basic solution architecture.

Figure 3) NetApp ONTAP AI verified architecture with DGX-2 servers.



4.1 NVIDIA DGX-2 Servers

The DGX-2 server is a fully integrated, turnkey hardware and software system that is purpose-built for DL workflows. Each DGX-2 server is powered by 16 V100 Tensor Core GPUs that are interconnected using NVIDIA NVLink™ and NVSwitch™ technology. NVSwitch technology provides an ultra-high-bandwidth, low-latency AI network fabric for inter-GPU communication with 2.4TB/s of bisection bandwidth. The DGX topology enables all 16 GPUs to operate as one, in a unified memory space, creating the world's first 2PFLOPS GPU accelerator. This architecture is essential for maximizing the performance needed for the most demanding data and model parallel training, eliminating the bottlenecks associated with non-NVSwitch-based topologies that cannot deliver linearity of performance as required when scaling from 1 to 16 GPUs. The DGX-2 server is also equipped with high-bandwidth, low-latency network interconnects for multinode clustering over RDMA-capable fabrics.

The DGX-2 is powered by NVIDIA GPU Cloud (NGC), a cloud-based container registry for GPU-accelerated software. NGC provides containers for today's most popular DL frameworks such as Caffe2, TensorFlow, PyTorch, MXNet, and NVIDIA TensorRT™, which are optimized for NVIDIA GPUs. The containers integrate the framework or application, necessary drivers, libraries, and communications primitives, and they are optimized across the stack by NVIDIA for maximum GPU-accelerated performance. NGC containers incorporate the CUDA™ Toolkit, which provides the CUDA Basic Linear Algebra Subroutines Library (cuBLAS), the CUDA Deep Neural Network Library (cuDNN), and much more. The NGC containers also include the NVIDIA Collective Communications Library (NCCL) for multi-GPU and multinode collective communication primitives, enabling topology awareness for DL training. NCCL enables communication between GPUs inside a single DGX-2 server and across multiple DGX-2 servers.

4.2 NetApp AFF Systems

NetApp AFF state-of-the-art storage systems enable IT departments to meet enterprise storage requirements with industry-leading performance, superior flexibility, cloud integration, and best-in-class data management. Designed specifically for flash, AFF systems help accelerate, manage, and protect business-critical data.

The NetApp AFF A800 system is the industry's first end-to-end NVMe solution. For NAS workloads, a single AFF A800 system supports throughput of 25GB/s for sequential reads and 1 million IOPS for small random reads at sub-500µs latencies. AFF A800 systems support the following features:

- Massive throughput of up to 300GB/s and 11.4 million IOPS in a 24-node cluster
- 100GbE together with 32Gb FC connectivity
- 30TB solid-state drives (SSDs) with multistream write
- High density with 2PB in a 2U drive shelf
- Scaling from 364TB (2 nodes) to 74PB (24 nodes)
- NetApp ONTAP 9.4, with a complete suite of data protection and replication features for industry-leading data management

The next best storage system in terms of performance is the AFF A700 system, supporting a throughput of 18GB/s for NAS workloads and 40GbE transport. AFF A300 and AFF A220 systems offer sufficient performance for smaller deployments at lower cost points.

4.3 NetApp ONTAP 9

ONTAP 9 is the latest generation of storage management software from NetApp that enables businesses to modernize infrastructure and transition to a cloud-ready data center. Leveraging industry-leading data management capabilities, ONTAP enables the management and protection of data with a single set of tools regardless of where that data resides. Data can also be moved freely to wherever it's needed—the

edge, the core, or the cloud. ONTAP 9 includes numerous features that simplify data management, accelerate and protect critical data, and future-proof infrastructure across hybrid cloud architectures.

Simplify Data Management

Data management is crucial to enterprise IT operations so that appropriate resources are used for applications and for datasets. ONTAP includes the following features to streamline and simplify operations and reduce the total cost of operation:

- **Inline data compaction and expanded deduplication.** Data compaction reduces wasted space inside storage blocks, and deduplication significantly increases effective capacity.
- **Minimum, maximum, and adaptive quality of service (QoS).** Granular QoS controls help maintain performance levels for critical applications in highly shared environments.
- **ONTAP FabricPool.** This feature provides automatic tiering of cold data to public and private cloud storage options including Amazon Web Services (AWS), Azure, and the NetApp StorageGRID® solution.

Accelerate and Protect Data

ONTAP delivers superior levels of performance and data protection and extends these capabilities with:

- **Performance and lower latency.** ONTAP offers the highest possible throughput at the lowest possible latency.
- **NetApp ONTAP FlexGroup.** A FlexGroup volume is a high-performance data container that can scale linearly up to 20PB/400 billion files, providing a single namespace that simplifies data management.
- **Data protection.** ONTAP provides built-in data protection capabilities with common management across all platforms.
- **NetApp Volume Encryption.** ONTAP offers native volume-level encryption with both onboard and external key management support.

Future-Proof Infrastructure

ONTAP 9 helps meet demanding and constantly changing business needs:

- **Seamless scaling and nondisruptive operations.** ONTAP supports the nondisruptive addition of capacity to existing controllers as well as to scale-out clusters. Customers can upgrade to the latest technologies such as NVMe and 32Gb FC without costly data migrations or outages.
- **Cloud connection.** ONTAP is the most cloud-connected storage management software, with options for software-defined storage (ONTAP Select) and cloud-native instances (NetApp Cloud Volumes Service) in all public clouds.
- **Integration with emerging applications.** ONTAP offers enterprise-grade data services for next-generation platforms and applications such as OpenStack, Hadoop, and MongoDB by using the same infrastructure that supports existing enterprise apps.

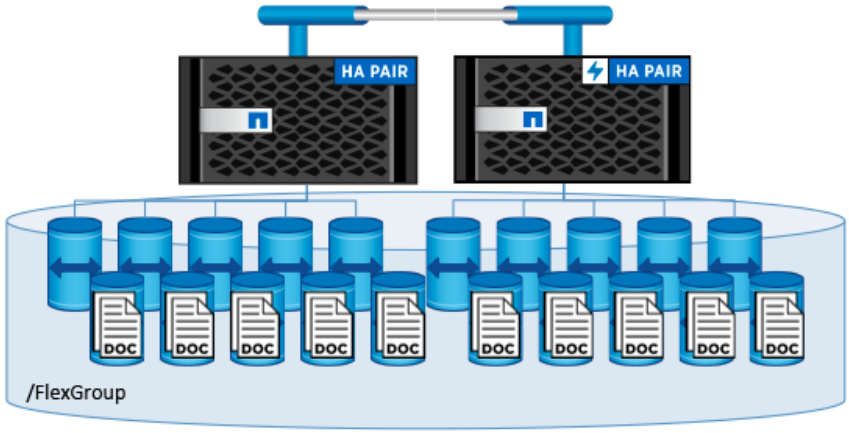
4.4 NetApp FlexGroup Volumes

The training dataset is usually a collection of potentially billions of files. Files can include text, audio, video, and other forms of unstructured data that must be stored and processed to be read in parallel. The storage system must store a large number of small files and must read those files in parallel for sequential and random I/O.

A FlexGroup volume (Figure 4) is a single namespace that is made up of multiple constituent member volumes and that is managed and acts like a NetApp FlexVol® volume to storage administrators. Files in a FlexGroup volume are allocated to individual member volumes and are not striped across volumes or nodes. They enable the following capabilities:

- FlexGroup volumes enable multiple petabytes of capacity and predictable low latency for high-metadata workloads.
- They support up to 400 billion files in the same namespace.
- They support parallelized operations in NAS workloads across CPUs, nodes, aggregates, and constituent FlexVol volumes.

Figure 4) NetApp FlexGroup volumes.



4.5 NVIDIA GPU Cloud and Trident

NVIDIA GPU Cloud (NGC) offers a catalog of fully integrated and performance engineered Docker images for deep learning that take full advantage of NVIDIA GPUs. These images include all necessary dependencies such as the CUDA Toolkit and DL libraries. These images are tested, tuned, and certified by NVIDIA for use on DGX servers. Further, to enable portability of images that leverage GPUs, NVIDIA developed NVIDIA Container Runtime for Docker, which allows the user mode components of NVIDIA drivers and GPUs to be mounted into the Docker container at launch.

Trident, from NetApp, is an open-source dynamic storage provisioner for Docker and Kubernetes. Combined with NGC and popular orchestrators such as Kubernetes and Docker Swarm, Trident enables customers to seamlessly deploy DL NGC container images onto NetApp storage, which provides an enterprise-grade experience for AI container deployments. These deployments include automated orchestration, cloning for testing and development, upgraded testing that uses cloning, protection and compliance copies, and many more data management use cases for the NGC AI and DL container images.

4.6 Cisco Nexus 3232C Network Switches

The Cisco Nexus 3232C switch (Figure 5) is a low-latency, dense, high-performance, power-efficient 100Gb/s switch that is designed for the data center. This compact, 1 rack unit (1RU) model offers wire-rate layer 2 and layer 3 switching on all ports with a latency of 450ns. This switch is a member of the Cisco Nexus 3200 platform and runs the industry-leading Cisco NX-OS software operating system, providing comprehensive features and functions that are widely deployed. The Cisco Nexus 3232C is a Quad Small Form-Factor Pluggable (QSFP) switch with 32 QSFP28 ports. Each QSFP28 port can operate at 10, 25, 40, 50, and 100Gb/s, up to a maximum of 128 ports at 25Gb/s.

Figure 5) Cisco Nexus 3232C switch.

This solution as tested consumes slightly more than half of the available ports on each network switch. Each switch could support additional DGX servers and additional storage access ports to provide more GPU power. For larger implementations, the Cisco Nexus 9500 series supports up to 576 ports of 100GbE per switch. Alternatively, a leaf-spine topology could be implemented with multiple pairs of Nexus 3000 switches that are connected into a central spine switch.

4.7 RDMA over Converged Ethernet

Direct memory access (DMA) enables hardware subsystems such as disk drive controllers, sound cards, graphics cards, and network cards to access system memory to perform data read/write without using CPU processing cycles. RDMA extends that capability by allowing network adapters to do a server-to-server data transfer between application memory by using zero-copy functionality without any OS or device driver involvement. This approach dramatically reduces CPU overhead and latency by bypassing the kernel for read/write and send/receive operations. Traditional HPC infrastructures use RDMA over InfiniBand (IB) for internode connectivity because of its high-bandwidth and low-latency features.

As Ethernet technology reaches performance levels that were previously possible only with IB, RDMA over Converged Ethernet (RoCE) enables easier adoption of these capabilities because Ethernet technologies are well understood and are widely deployed in every enterprise data center. RoCE is now available as a standard feature in many high-end network adapters, converged network adapters, and network switches. Traditional Ethernet uses a best-effort delivery mechanism for network traffic and is not suitable for the low latency and high bandwidth that are required for communications between GPU nodes. CEE enables a lossless physical-layer networking medium and the ability to optionally allocate bandwidth to any specific traffic flow on the network.

For lossless, in-order delivery of Ethernet packets, CEE networks use Priority Flow Control (PFC) and Enhanced Transmission Selection (ETS). PFC enables the sending of pause frames for each specific Class of Service (CoS) and allows limiting specific network traffic while allowing other traffic to flow freely. ETS allows specific bandwidth allocation for each CoS to provide even more granular control over network utilization.

The ability to prioritize RoCE over all other traffic allows the 100GbE links to be used for both RoCE and traditional IP traffic, such as the NFS storage access traffic that is demonstrated in this solution.

As Ethernet technology reaches performance levels that were previously possible only with IB, RoCE enables easier adoption of these capabilities because Ethernet technologies are well understood and are widely deployed in every enterprise data center. Figure 3 shows the basic solution architecture.

4.8 Automation with Ansible

Ansible is a configuration management tool from Red Hat that is quickly becoming the standard for DevOps-style system administration. Ansible accelerates time to value during deployment, and improves stability and reduces administrative overhead during daily operations. Ansible provides a declarative methodology for management of hardware and software where the administrator specifies the intended state of the configuration in a set of easy-to-read YAML files. This allows the administrator to manage the state of the infrastructure with version controls and change validation processes, just like any other software code.

Ansible was originally designed for Linux administration, but it also includes an extensible framework for management of almost any device. NetApp and Cisco offer extensive module support, enabling deployment and management of the entire ONTAP AI infrastructure by using Ansible. The infrastructure used in this validation was configured in less than 25 minutes by using Ansible modules that are publicly available in the official distribution of Ansible. For more information, see the blog post and demonstration video at https://blog.netapp.com/how-to-configure-ontap-ai-in-20-minutes-with-ansible-automation/.

5 Technology Requirements

This section covers the hardware and software used in the validation of this solution. All the testing that is documented in section 7, Solution Verification, was performed with the hardware and software described here.

5.1 Hardware Requirements

Table 1 lists the hardware components that were used to validate this solution. The components used in any particular implementation of this solution can vary based on specific workload requirements.

Table 1) Hardware requirements.

Hardware	Quantity
NVIDIA DGX-2 GPU servers	3
NetApp AFF A800 system	1 High-availability (HA) pair; includes 48x 1.92TB NVMe SSDs
Cisco Nexus 3232C network switches	2

5.2 Software Requirements

Table 2 lists the software components that were used in this validation. The software components used in any particular implementation of the solution can vary based on specific workload requirements.

Table 2) Software requirements.

Software	Version
NetApp ONTAP	9.5
Cisco NX-OS switch firmware	7.0(3)I6(1)
NVIDIA DGX OS	4.0.4 - Ubuntu 18.04 LTS
Docker container platform	18.06.1-ce [e68fc7a]
Container version	netapp_tf_19.02 based on nvcr.io/nvidia/tensorflow:19.02-py2; netapp_tf_19.03 based on nvcr.io/nvidia/tensorflow:19.03-py2 (for DALI)
Machine learning framework	TensorFlow 1.12.2 and 1.13.3
Horovod	0.15.1 and 0.16
OpenMPI	3.1.3
Benchmark software	TensorFlow benchmarks [7b9e1b4]

6 Solution Architecture

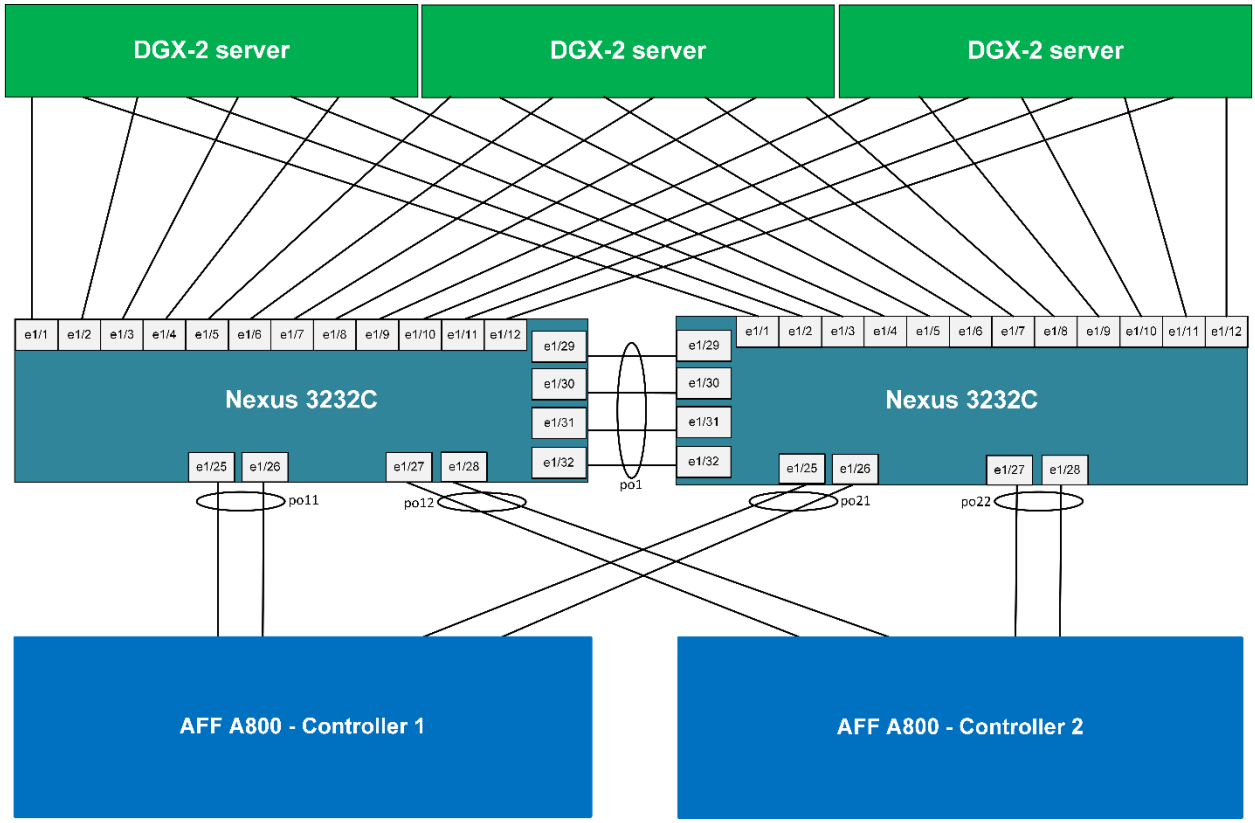
This architecture has been verified to meet the requirements for running deep learning workloads. This verification enables data scientists to deploy DL frameworks and applications on a prevalidated infrastructure, helping to eliminate risks and allowing businesses to focus on gaining valuable insights from their data. This architecture can also deliver exceptional storage performance for other HPC workloads without any modification or tuning of the infrastructure.

6.1 Network Topology and Switch Configuration

For this solution, RoCE is used in place of InfiniBand to provide the high-bandwidth, low-latency connectivity that is required for communication between DGX-2 servers. Cisco Nexus switches support RoCE by implementing Priority Flow Control (PFC), which allows users to prioritize RoCE traffic over traditional IP traffic on a shared link and allows the 100GbE links to be used for both RoCE and IP at the same time.

This architecture uses a pair of Cisco Nexus 3232C 100GbE switches for the primary intercluster and storage access network. These switches are connected to each other with four 100GbE network ports that are configured as a standard port channel. This Inter-Switch Link port channel allows traffic to flow between the switches during host or storage system link failures. Each host is connected to the Nexus switches with a pair of active-passive bonds. Also, to provide link-layer redundancy, each storage controller is connected to each Nexus switch with a two-port LACP port channel. Figure 6 shows the network switch-port configuration.

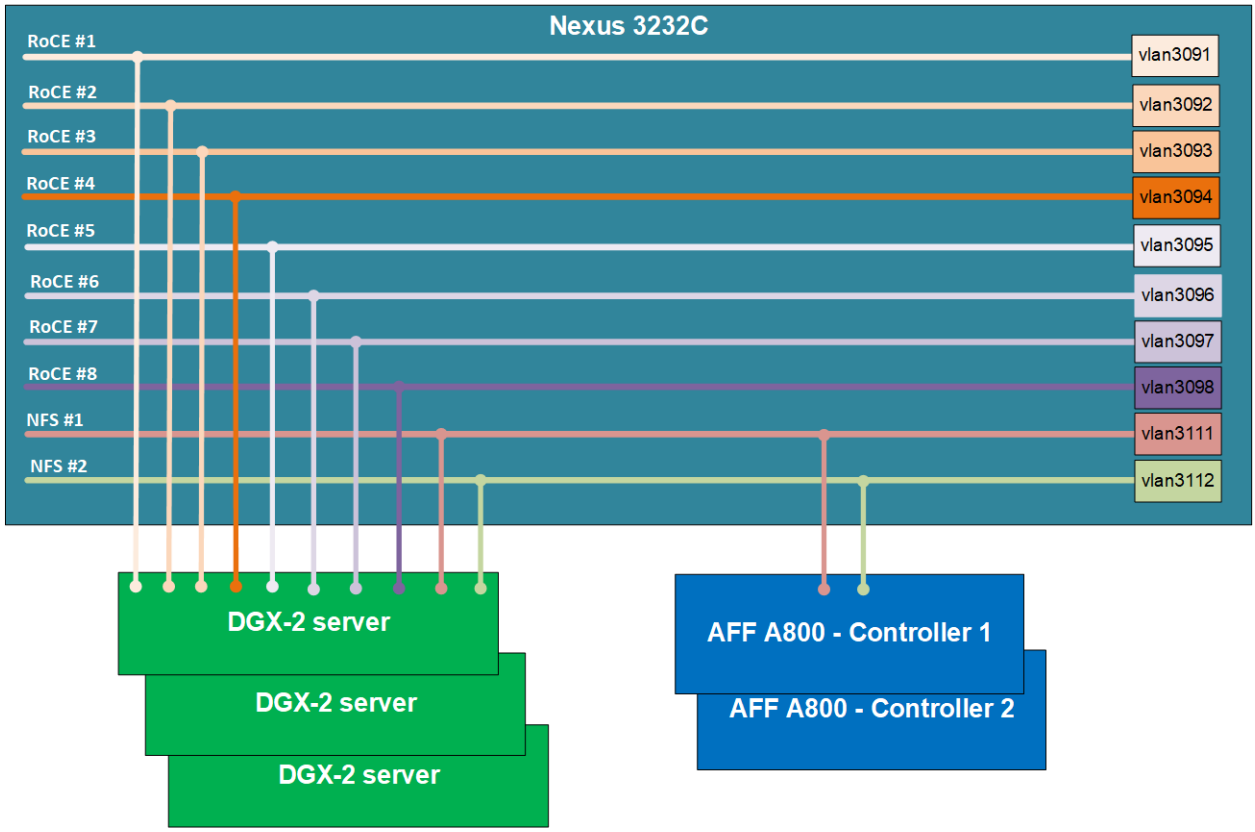
Figure 6) Network switch port configuration.



Multiple virtual LANs (VLANs) were provisioned to support both RoCE and NFS storage traffic. Eight VLANs are dedicated to RoCE traffic, and two VLANs are dedicated to NFS storage traffic. Eight discrete IP ranges are used to provide symmetrical routing for each RoCE connection, and the software stack manages these connections for bandwidth aggregation and fault tolerance. For storage access, this solution uses NFSv3, and two VLANs are used to enable multiple dedicated NFS mounts. This approach does not provide any additional fault tolerance, but it does allow multiple links to be used to increase available bandwidth. PFC is configured on each switch to assign all eight RoCE VLANs to the priority class, and the NFS VLANs are assigned to the default best-effort class. All VLANs are configured for jumbo frames with a maximum transmission unit (MTU) size of 9000.

The switch ports for DGX-2 servers are configured as trunk ports, and all RoCE and NFS VLANs are permitted. The port channels that are configured for the storage system controllers are also trunk ports, but only the NFS VLANs are permitted. Figure 7 shows the VLAN connectivity for the DGX-2 server and storage system ports.

Figure 7) VLAN connectivity for DGX-2 and storage system ports.



To provide priority service for RoCE traffic, the host network adapter assigns a Class of Service (CoS) value of 4 to traffic on each RoCE VLAN. The switch is configured with a QoS policy that provides no-drop service to traffic with this CoS value. NFS traffic is assigned the default CoS value of 0, which falls into the default QoS policy on the switch and provides best-effort service.

PFC is then enabled on each DGX-2 server port, which enables the switch port to send pause frames for specific classes of service to eliminate congestion at the switch. By using ETS to allocate 95% of the bandwidth to RoCE traffic in the event of congestion, this configuration allows dynamic resource allocation between RoCE and NFS traffic while providing priority to node-to-node communication. Bandwidth allocation can also be modified dynamically to optimize for workloads that require higher storage performance and less internode communication.

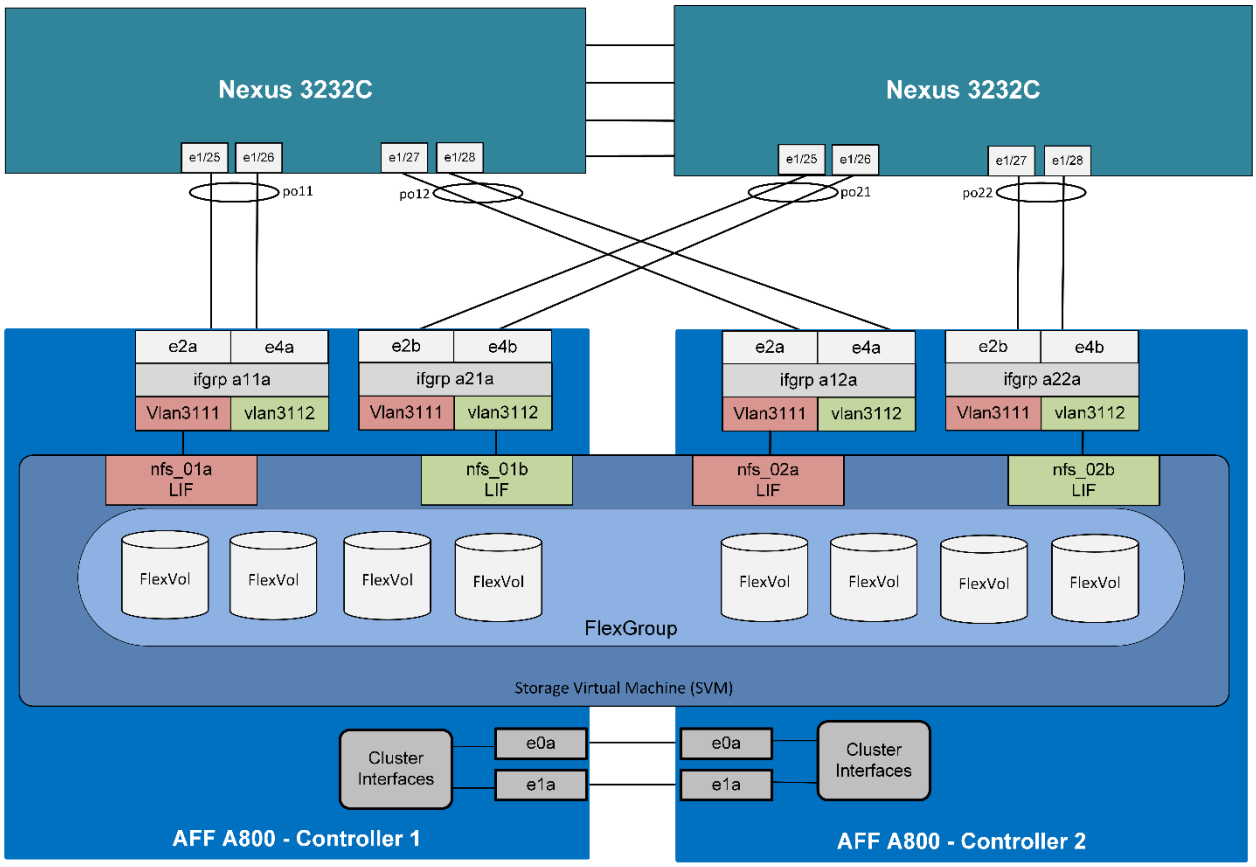
6.2 Storage System Configuration

To support the storage network requirements of any potential workload on this architecture, each storage controller is provisioned with four 100GbE ports in addition to the onboard ports that are required for storage cluster interconnection. Figure 8 shows the storage system configuration. Each controller is configured with a two-port LACP interface group (ifgrp in Figure 8) to each switch. These interface groups provide up to 200Gb/s of resilient connectivity to each switch for data access. Two VLANs are provisioned for NFS storage access, and both storage VLANs are trunked from the switches to each of

these interface groups. This configuration allows concurrent access from each host to the data through multiple interfaces, which improves the potential bandwidth that is available to each host.

All data access from the storage system is provided through NFS access from a storage virtual machine (SVM) that is dedicated to this workload. The SVM is configured with a total of four logical interfaces (LIFs), with two LIFs on each storage VLAN. Each interface group hosts a single LIF, resulting in one LIF per VLAN on each controller with a dedicated interface group for each VLAN. However, both VLANs are trunked to both interface groups on each controller. This configuration provides the means for each LIF to fail over to another interface group on the same controller so that both controllers stay active in the event of a network failure.

Figure 8) Storage system configuration.



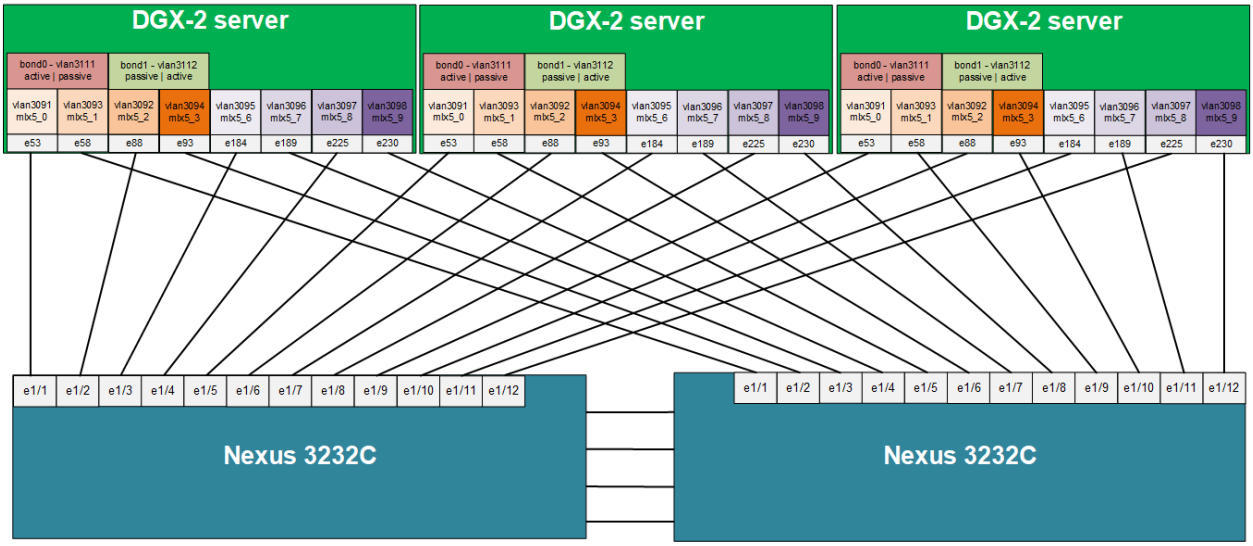
For logical storage provisioning, the solution uses a FlexGroup volume to provide a single pool of storage that is distributed across the nodes in the storage cluster. Each controller hosts an aggregate of 46 disk partitions, with both controllers sharing every disk. When the FlexGroup volume is deployed on the data SVM, a number of FlexVol volumes are provisioned on each aggregate and are then combined into the FlexGroup. This approach allows the storage system to provide a single pool of storage that can scale up to the maximum capacity of the array and provide exceptional performance by leveraging all the SSDs in the array concurrently. NFS clients can access the FlexGroup as a single mount point through any of the LIFs that are provisioned for the SVM. Capacity and client access bandwidth can be increased by simply adding more nodes to the storage cluster.

6.3 Host Configuration

For network connectivity, each DGX-2 is provisioned with eight Mellanox ConnectX5 single-port network interface cards. These cards operate at speeds of up to 100GbE and support RoCE, providing a lower-

cost alternative to IB for cluster interconnect applications. Each 100Gb port is configured as a trunk port on the appropriate switch, with eight RoCE and two NFS VLANs allowed on each. Figure 9 shows the network port and VLAN configuration of the DGX-2 hosts.

Figure 9) Network port and VLAN configuration of the DGX-2 hosts.



For RoCE connectivity, each physical port hosts a VLAN interface and IP address on one of the eight RoCE VLANs. The Mellanox drivers are configured to apply a network CoS value of 4 to each of the RoCE VLANs, and PFC is configured on the switches to guarantee priority lossless service to the RoCE class. RoCE does not support aggregating multiple links into a single logical connection, but the NCCL communication software can use multiple links for bandwidth aggregation and fault tolerance.

For NFS storage access, two active-passive bonds are created by using a link to each switch. Each bond hosts a VLAN interface and IP address on one of the two NFS VLANs, and the active port of each bond is connected to a different switch. This configuration provides up to 100Gb of bandwidth in each NFS VLAN and also provides redundancy in case of any host link or switch failure. To ensure optimal performance for the RoCE connections, all NFS traffic is assigned to the default best-effort QoS class. All physical interfaces and the bond interfaces are configured with an MTU of 9000.

To increase data access performance, multiple NFSv3 mounts are made from the DGX-2 server to the storage system. Each DGX-2 server is configured with two NFS VLANs, with an IP interface on each VLAN. The FlexGroup volume on the AFF A800 system is mounted on each of these VLANs on each DGX-2, providing completely independent connections from the server to the storage system. Although a single NFS mount is capable of delivering the performance that is required for this workload, multiple mount points are defined to enable the use of additional storage access bandwidth for other workloads that are more storage intensive.

7 Solution Verification

This section describes the testing that was done to validate this solution. Several benchmark tests were performed with various configurations to evaluate overall performance and scalability. Other tests were performed to demonstrate operations and performance with real-world software and data. This section contains the configuration details and results for each of the tests that were executed. Additional test results can be found in the appendix.

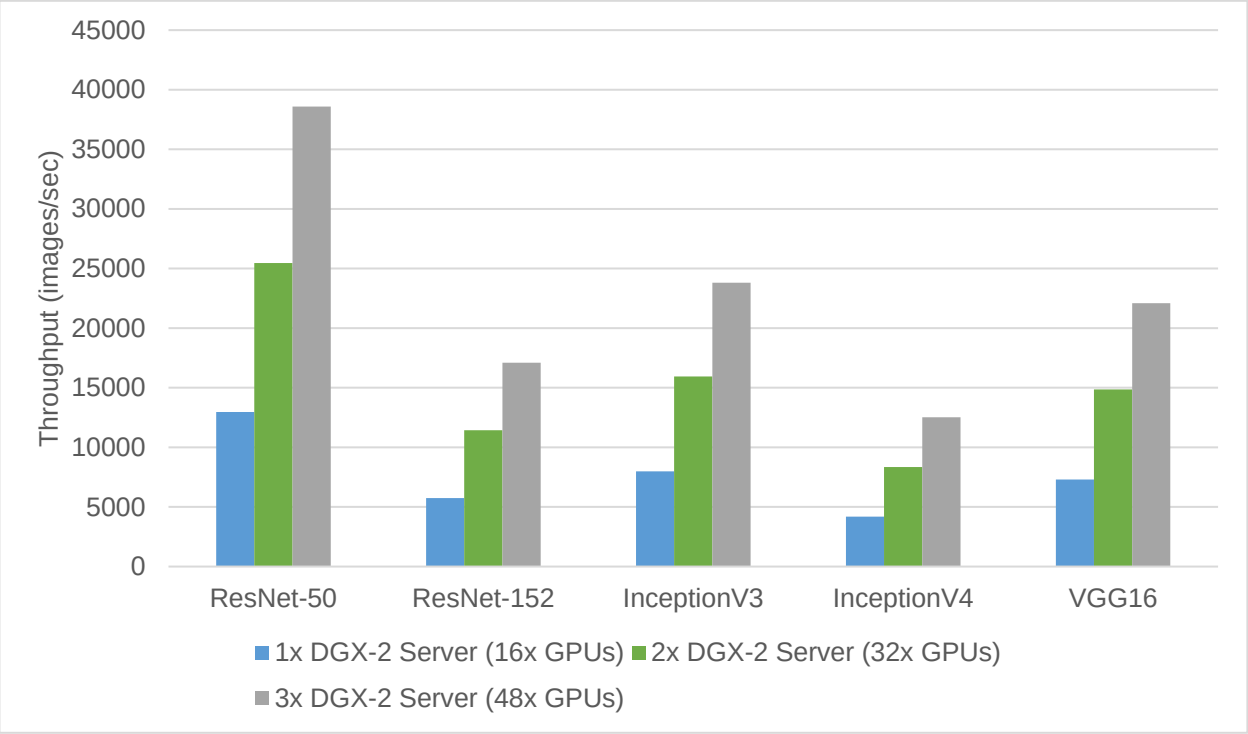
7.1 ImageNet Benchmark Testing

ImageNet is an industry-standard dataset that is often used for benchmark testing of AI and ML infrastructures. For this testing, the ImageNet dataset was duplicated 20 times in order to guarantee that the dataset was larger than DGX-2 server physical memory. The TensorFlow:19.02 container from NGC was used to do training and inferencing, and the following results are the average of three test runs.

Training Results

Figure 10 shows the results of training operations with various models. These results were generated using a batch size of 256 for ResNet-50, Resnet-152, InceptionV3, and InceptionV4, and a batch size of 192 for VGG16. For VGG16, the smaller batch size produced slightly better results.

Figure 10) Training throughput with standard ImageNet dataset.



As shown in Figure 10, the throughput of training on all models scaled linearly until the GPUs were almost completely saturated. For example, in the ResNet-50 testing, GPU utilization of a single DGX-2 server reached over 95%, as shown in Figure 11. In this case the CPU is not a bottleneck, because CPU utilization did not exceed 70%. Note that this graph shows the total utilization of all GPUs in the DGX-2 server, resulting in a scale-up to %1600 on the y axis.

Figure 11) DGX-2 server CPU and GPU utilization for ResNet-50 training.

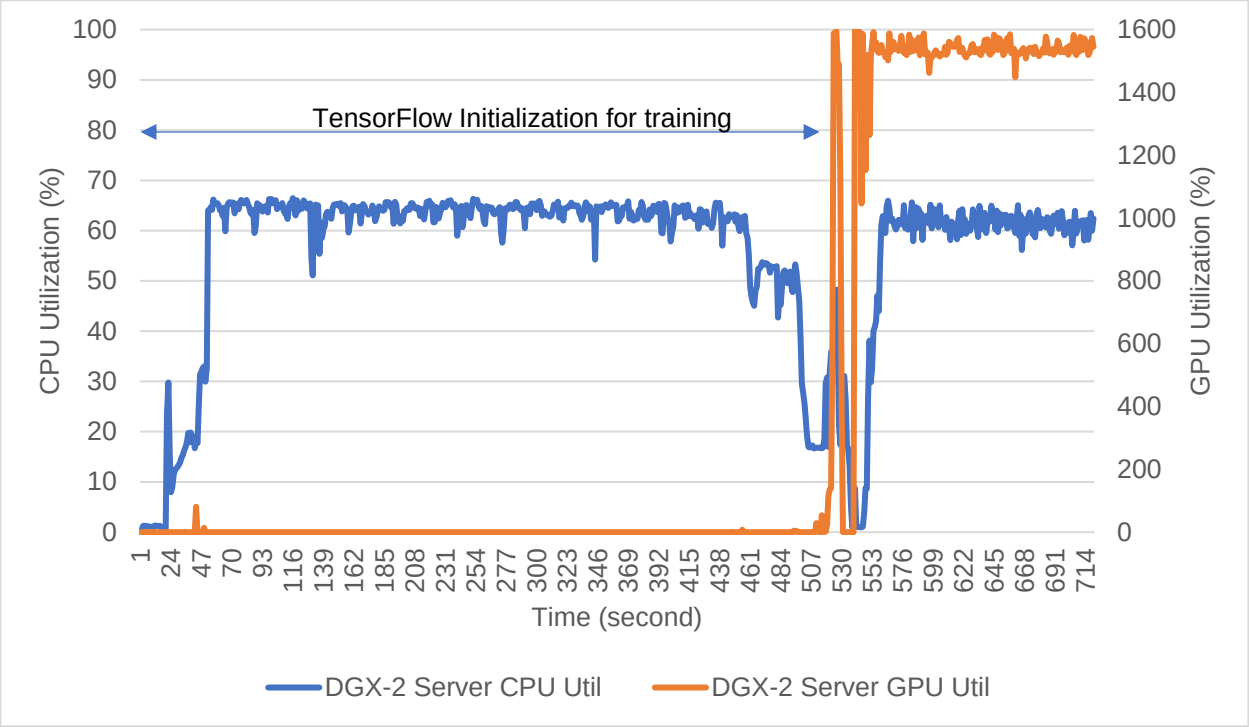
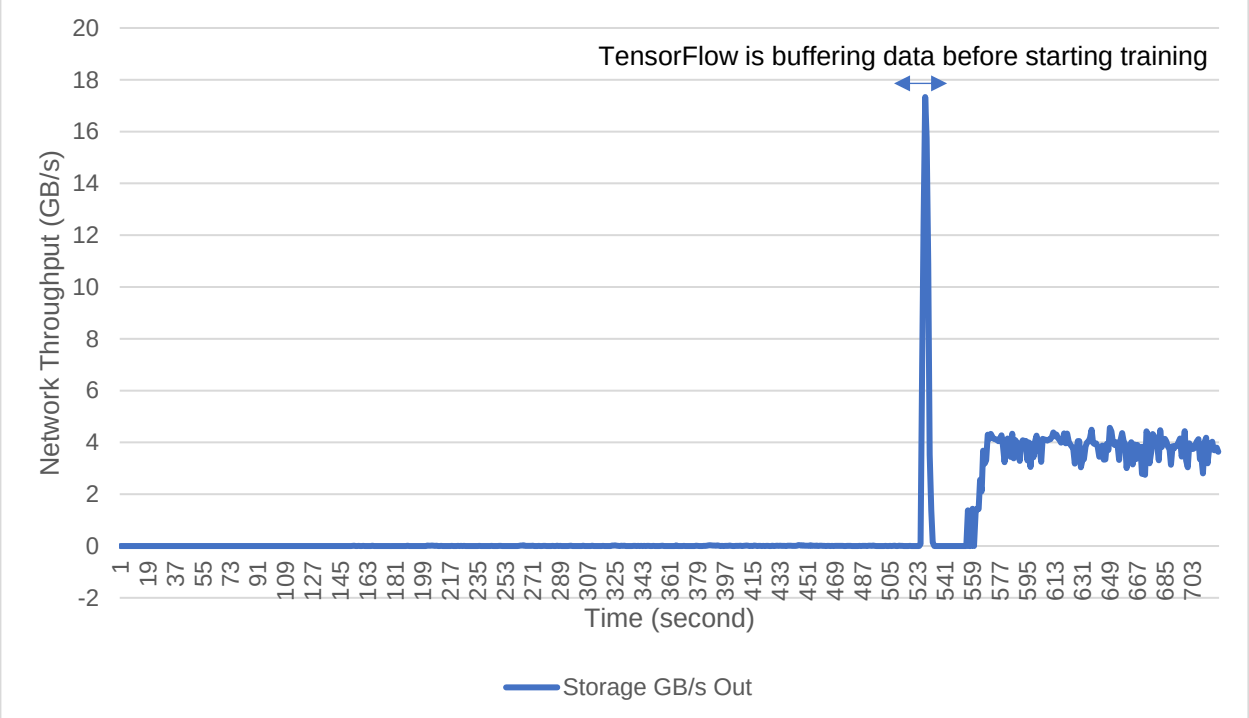


Figure 12 shows the storage system CPU utilization during ResNet-50 training. Data for this test was read completely from the storage system, and sustained throughput was less than 5GB/s, which is well within the performance capabilities of the AFF A800. Even during the period of peak bandwidth, the storage system was less than 50% utilized.

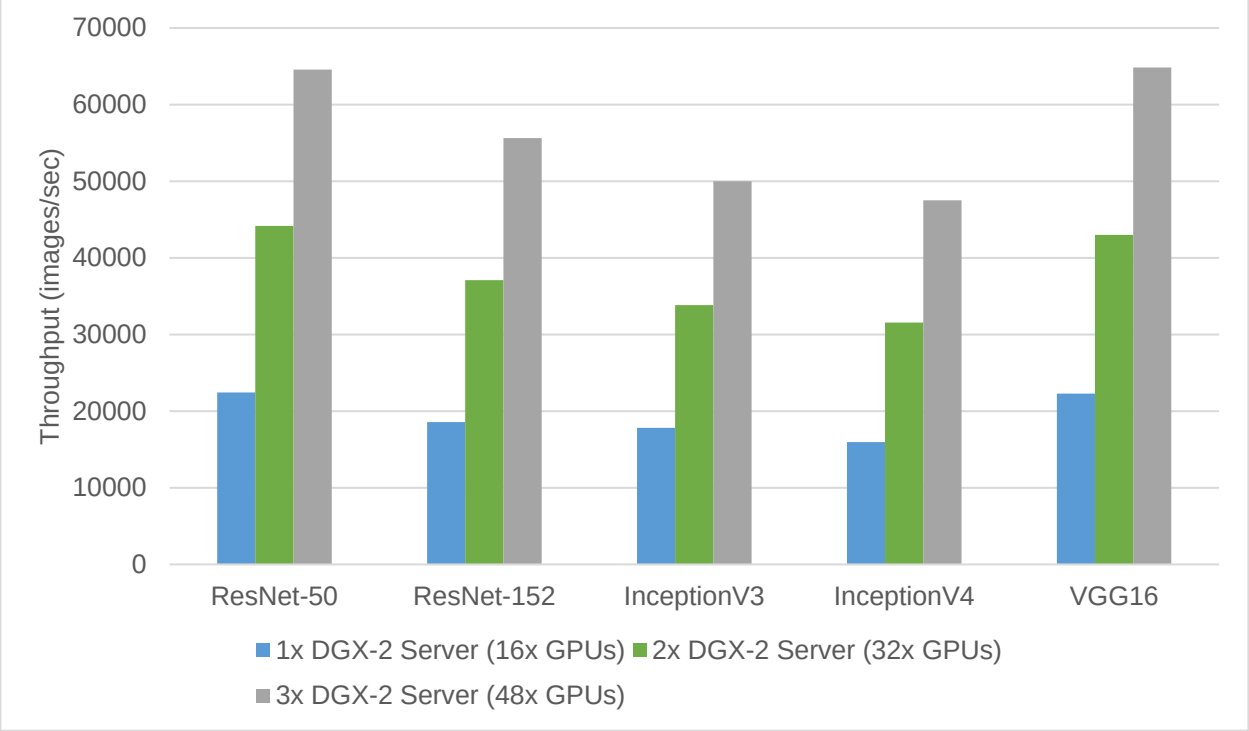
Figure 12) AFF A800 CPU utilization during ResNet-50 training.



Inferencing Results

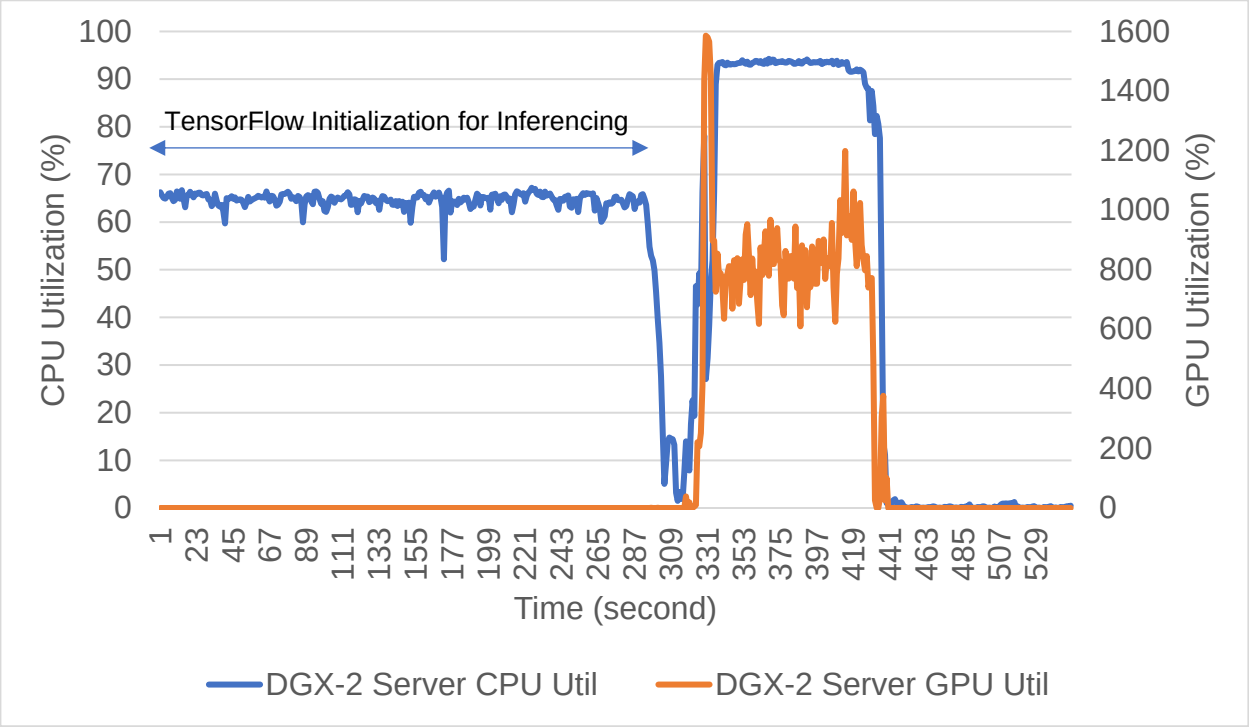
Figure 13 shows the throughput of inferencing using different models.

Figure 13) Inferencing throughput with standard ImageNet dataset.



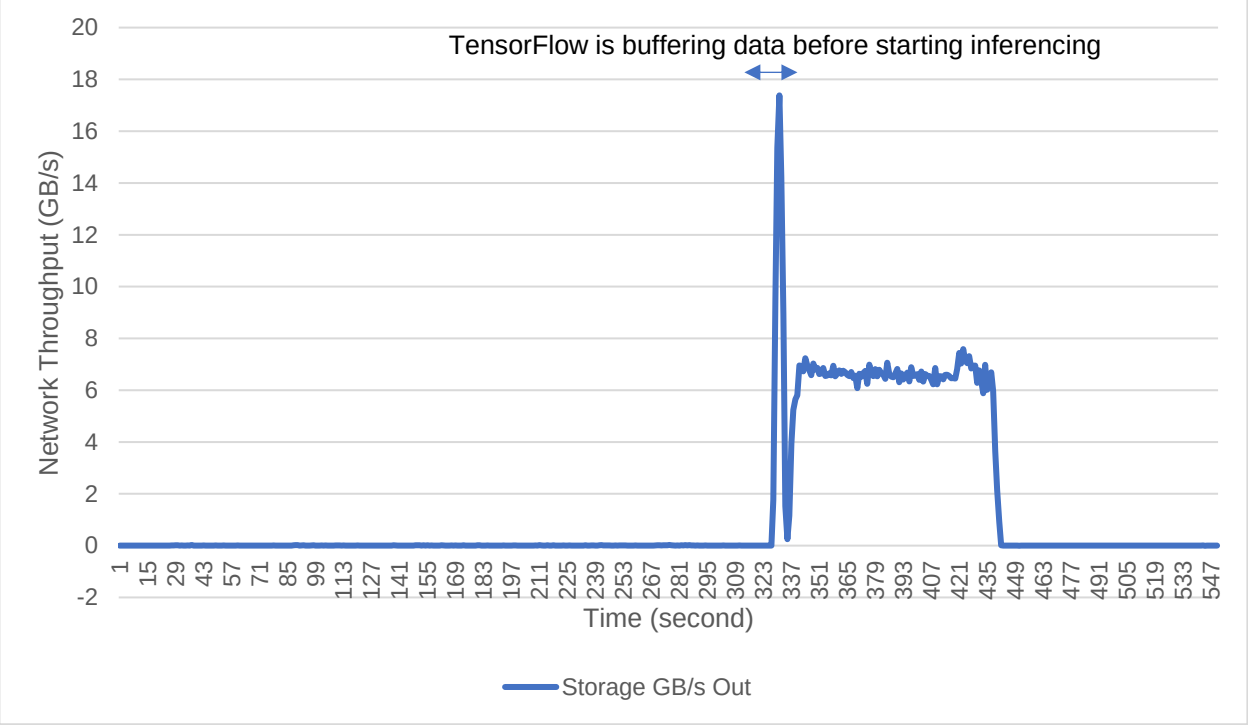
In contrast to the training workload, inferencing uses CPU for preprocessing of image data and moves the bottleneck from the GPU to the CPU. Again using ResNet-50 as an example, Figure 14 shows that CPU utilization is higher than GPU utilization. For models such as ResNet-50 and InceptionV3, throughput is limited by the ability of the CPU to prepare and feed data to the GPUs.

Figure 14) DGX-2 server CPU and GPU utilization during ResNet-50 inferencing.



Inferencing produces a higher storage workload than training, but still well within the capabilities of the AFF A800 storage system. Figure 15 shows the storage system throughput for inferencing with ResNet-50. Note that the storage system is effectively idle while the CPU is preparing the training; the bandwidth spike signals the loading of data into GPU memory and the beginning of actual training.

Figure 15) AFF A800 network throughput during ResNet-50 inferencing.



7.2 Scaled ImageNet Testing

To further demonstrate the capabilities of the AFF A800, the images in the ImageNet dataset were each scaled up 9x to approximately 1MB in size. Increasing the image size places a larger load on the CPU, because the images are normalized for training. The most complicated model, Inception-v4, was used to minimize the model dependency on the CPU to put more stress on the storage system and measure the maximum throughput that can be achieved at the cost of accuracy of the training run. Figure 16 shows the throughput of Inception-v4 with this scaled-up dataset. Again, the throughput scaled linearly with the number of DGX-2 servers.

Figure 16) Training throughput of Inception-v4 with scaled ImageNet dataset.

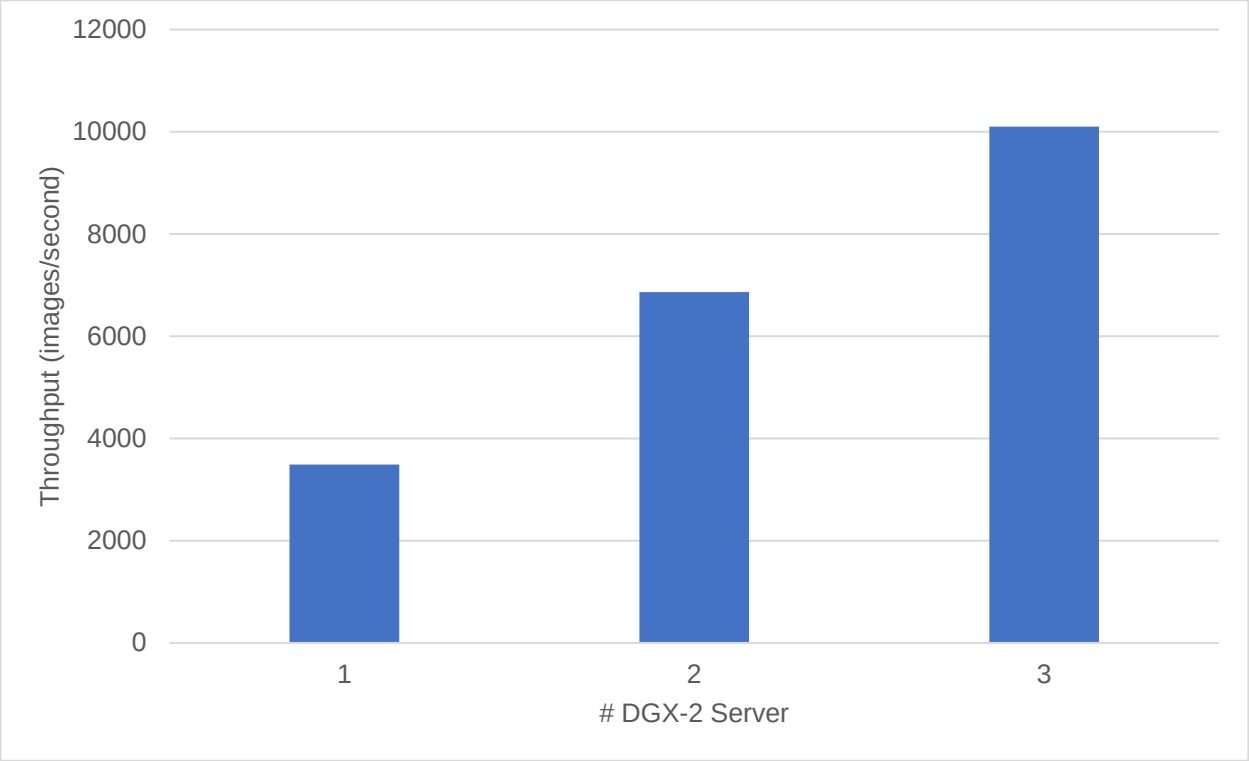
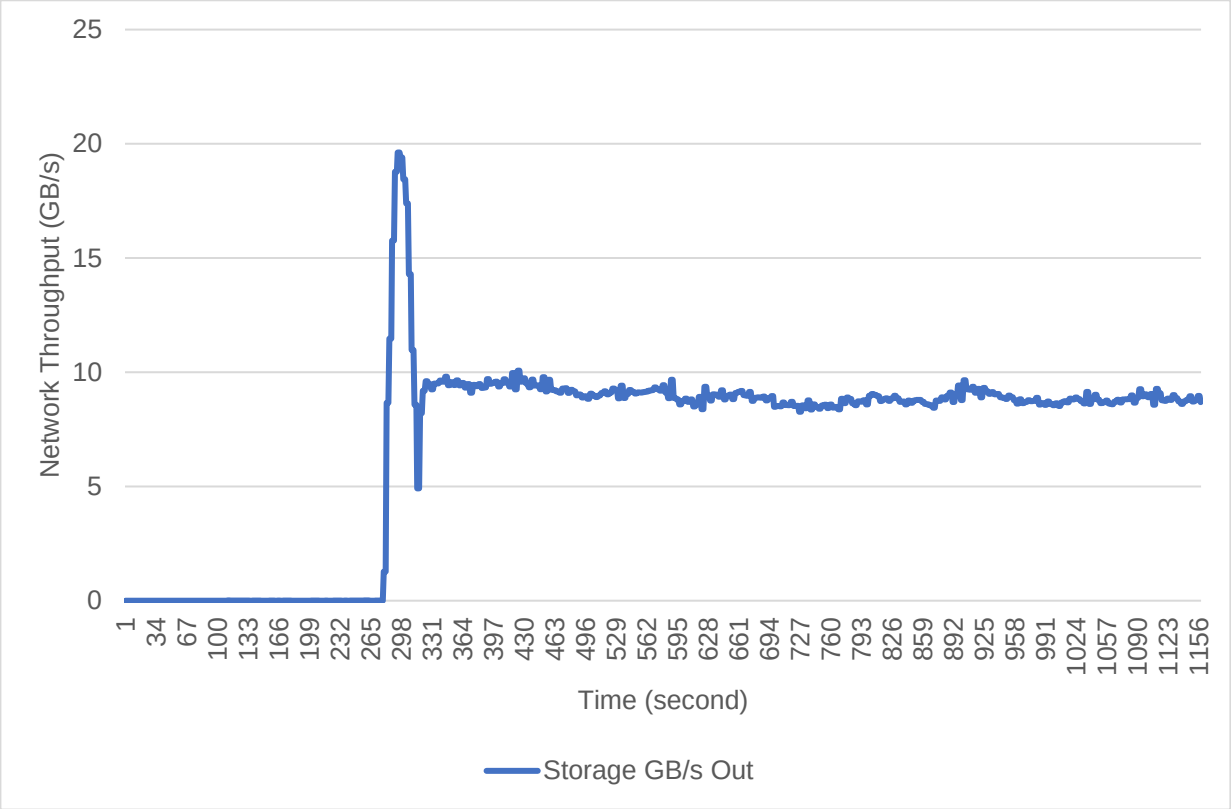


Figure 17 shows the network throughput from the AFF A800, which reaches a peak of around 20GB/s and sustained bandwidth of 9GB/s. In all of these tests, sustained bandwidth is lower than peak bandwidth because the CPUs and GPUs become saturated and cannot sustain the same I/O rate.

Figure 17) AFF A800 network throughput with scaled ImageNet dataset.

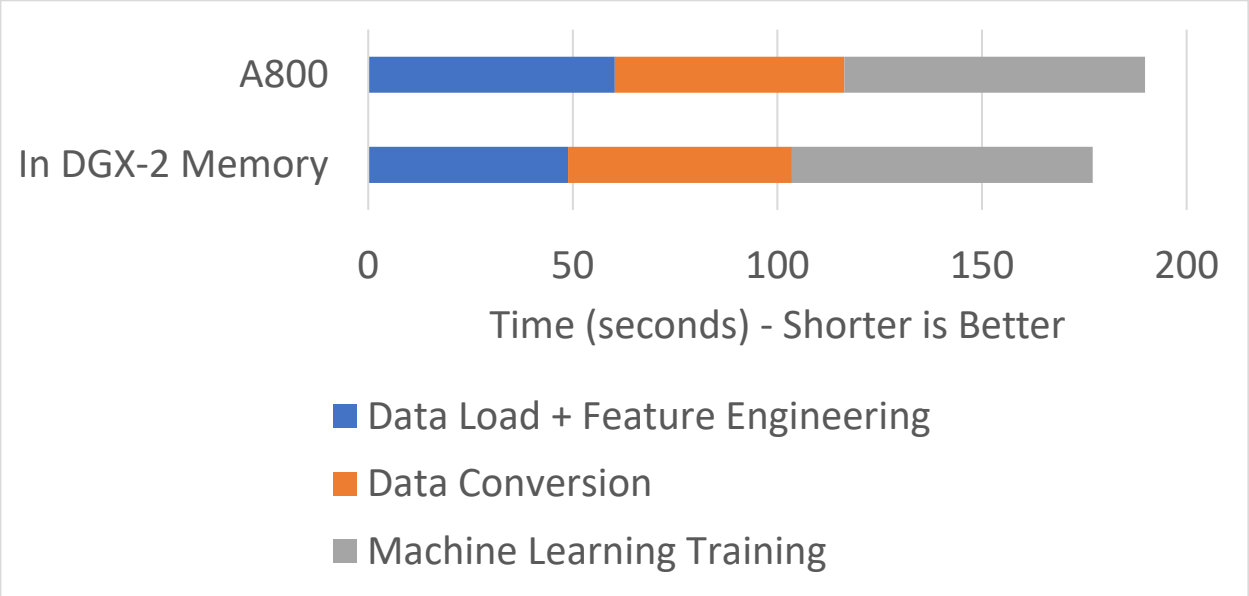


7.3 RAPIDS

RAPIDS is a set of libraries designed to integrate data preparation and training into a single GPU-accelerated workflow. RAPIDS uses common data structures and programming interfaces to enable developers to accelerate analytics and deep learning data preparation and model training. To validate the performance of a RAPIDS workflow, mortgage data from https://rapidsai.github.io/demos/datasets/mortgage-data was loaded into GPU memory via the RAPIDS CSV reader. The loaded data was then converted to train a gradient-boosted decision tree model on the GPU using XGBoost, which is one of the RAPIDS libraries. For detailed information about RAPIDS, see the RAPIDS website.

Because the goal of RAPIDS is to process the data in the GPU, storage system performance only contributes to the time spent initially loading data. Figure 18 compares the performance of RAPIDS using data from the AFF A800 with performance from DGX-2 server system memory. The AFF A800 produces similar results in overall processing time.

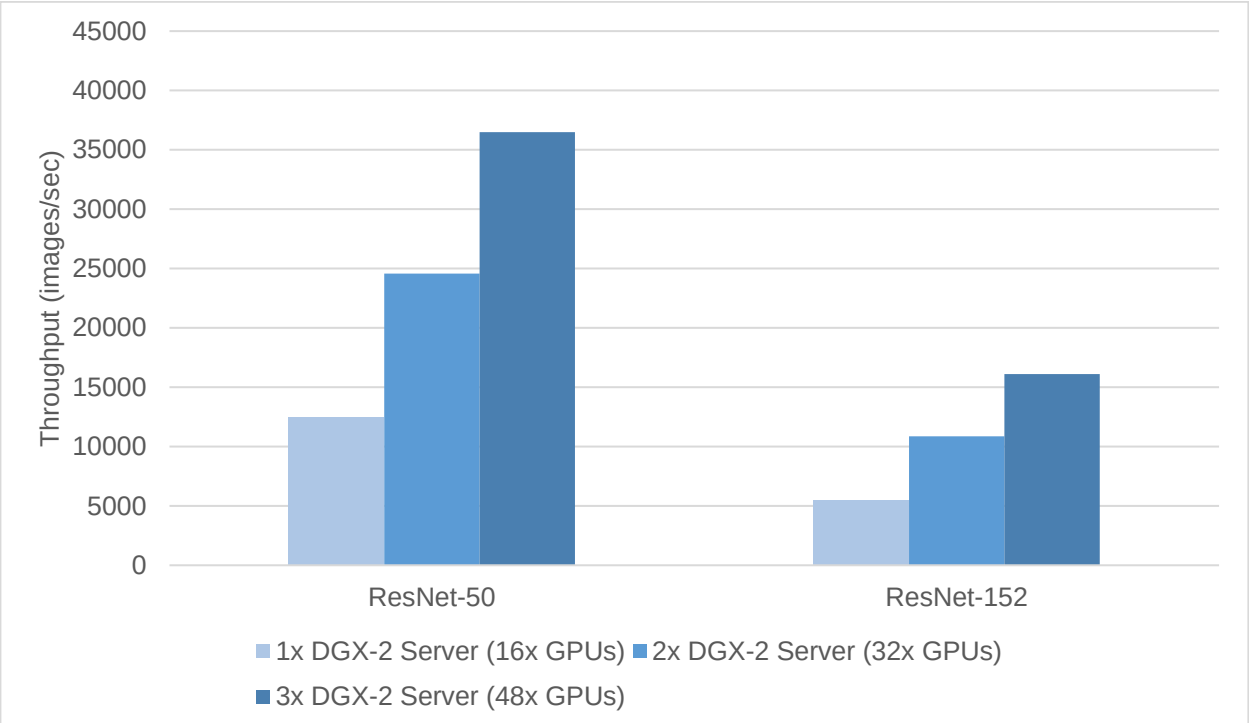
Figure 18) Performance comparison between AFF A800 and DGX-2 server memory.



7.4 DALI

DALI is another project integrated with TensorFlow to move data preprocessing to GPU. This experiment used ResNet-50 and ResNet-152, which have different model complexities, to validate that GPU training performance is not degraded by moving the data preparation pipeline to GPU. Both workloads rely on the GPUs; however, although DALI uses GPU resources for data preprocessing, it does not significantly affect training throughput. Figure 19 shows that training with DALI offers performance that is similar to the basic TensorFlow benchmarks.

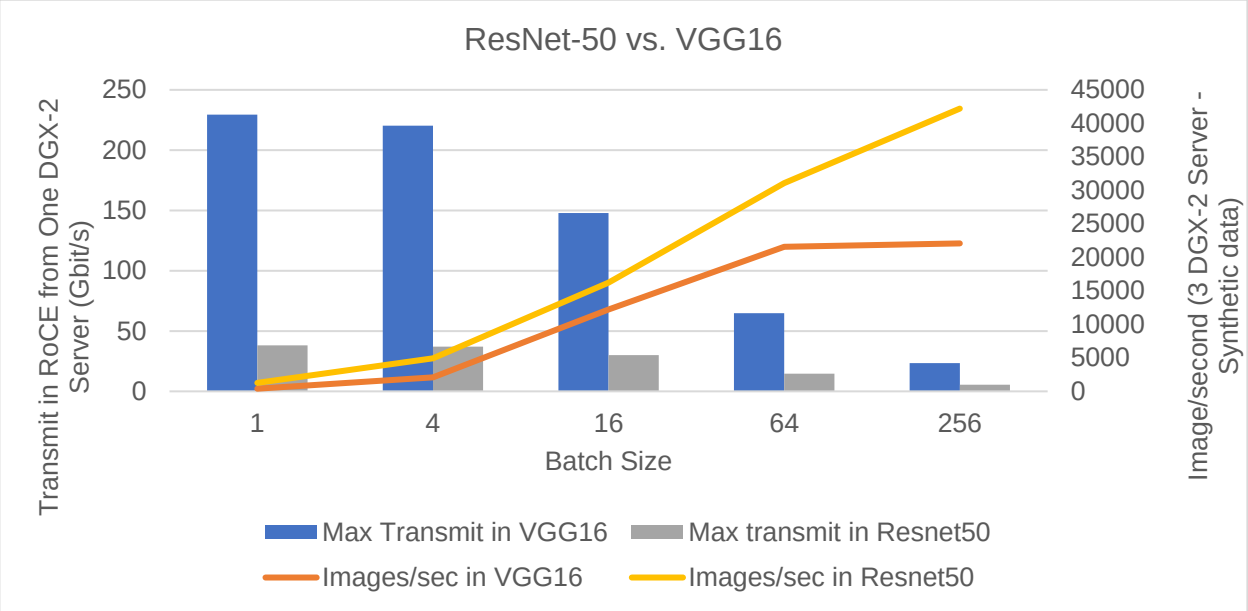
Figure 19) Training throughput with DALI.



7.5 RoCE Traffic

To better understand the network bandwidth required for the exchange of updates between DGX-2 servers during training, the bandwidth was measured using ResNet-50 and VGG16. VGG16 was chosen because it has the largest model to exchange between servers, and ResNet-50 was chosen because it has the most frequent updates. Several batch sizes were tested to illustrate the bandwidth requirements imposed by different exchange frequencies, because smaller batch sizes require more frequent updates. Figure 20 shows the traffic transmitted by a single DGX-2 server during training with all three servers, and it shows that with realistic batch sizes the network bandwidth per server is less than 50Gb/s.

Figure 20) RoCE bandwidth and training throughput for ResNet-50 and Inception-v3.



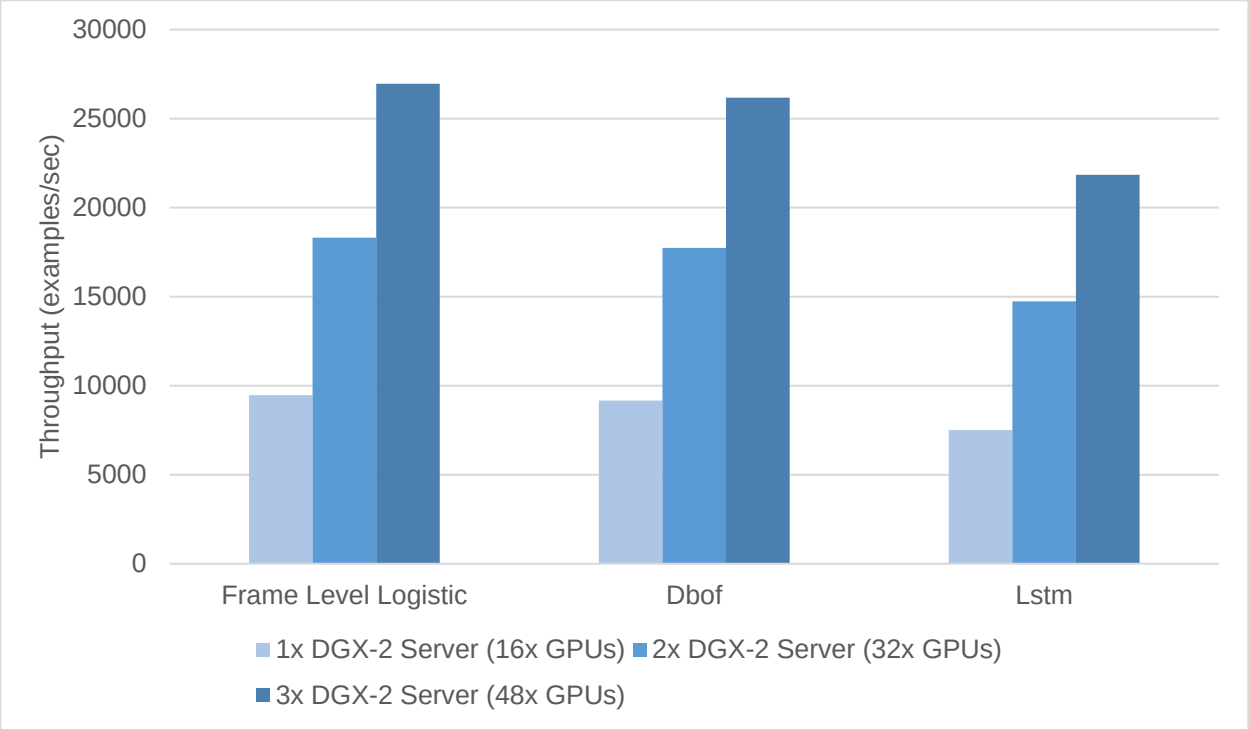
8 Testing with Additional Datasets

8.1 YouTube-8M Dataset

For large-scale video understanding, we tested the YouTube-8M dataset released by Google. This dataset includes 6.1 million YouTube video IDs, which have overall 2.6 billion audio/video features and 3,862 categories. We modified the starter TensorFlow code and enabled the communication among GPUs via Horovod, which allows training to be easily scaled across three DGX-2 servers. This testing was performed using the frame-level features dataset, which is 1.53TB in size.

Figure 21 shows the training throughput of different starter models provided by the YouTube-8M challenge. The throughput scaled linearly with the number of DGX-2 servers.

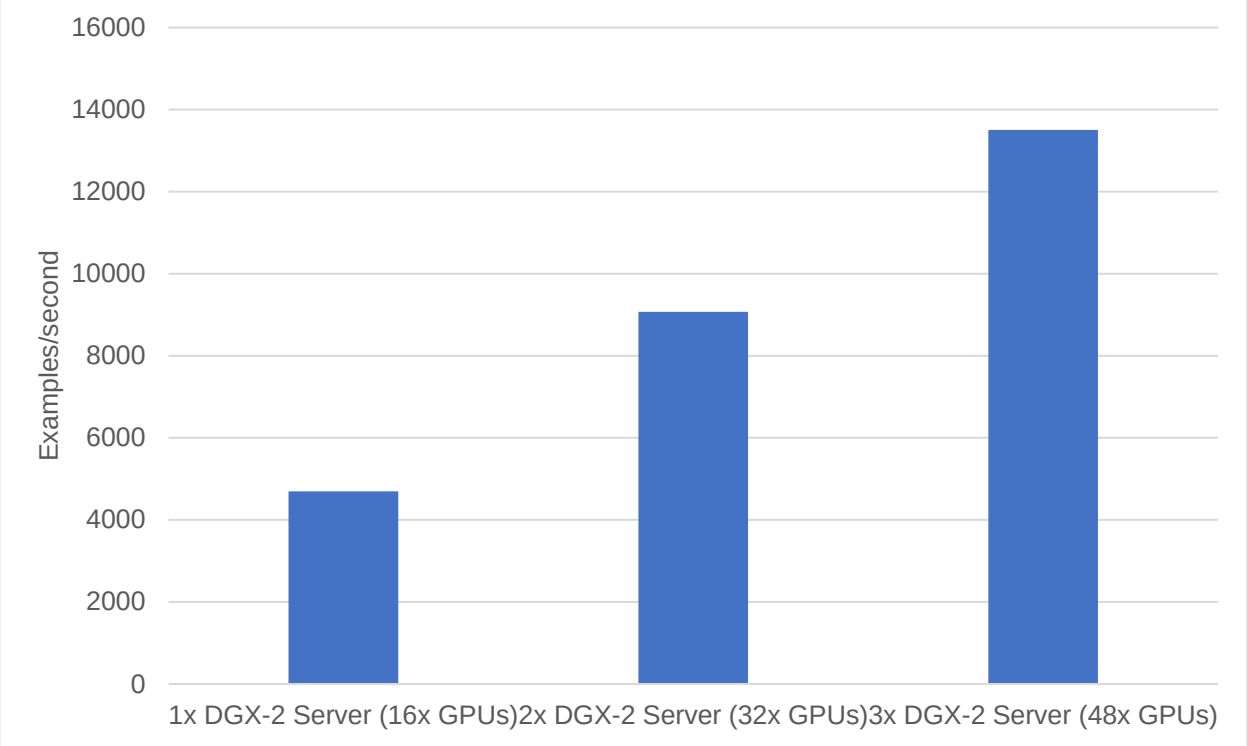
Figure 21) Training throughput with YouTube-8M dataset.



8.2 Generative Adversarial Networks (GANs)

To simulate and test for more write-intensive workloads, we tested the implementation of a VAE/GAN model [1], which combines the Variational Autoencoder (VAE) and Generative Adversarial Networks (GAN), by using the discriminator of GAN as the perceptual loss instead of the pixel-wise loss in the original VAE. We trained the VAE/GAN model with the large-scale CelebFaces Attributes (CelebA) dataset. This dataset includes the faces of 10,177 celebrities, with a total of 202,599 images. However, the whole dataset size is only around 2GB, which can all be cached in the physical memory. To demonstrate the performance of the AFF A800 storage system, the data was copied 1,000 times to make the overall dataset size 2TB. We used Horovod to scale out the training procedure across three DGX-2 servers. Figure 22 shows that the throughput scales linearly with the number of DGX-2 servers. This test was performed with default TensorFlow settings, while single-server tests with optimized TensorFlow code resulted in almost 7,100 examples per second.

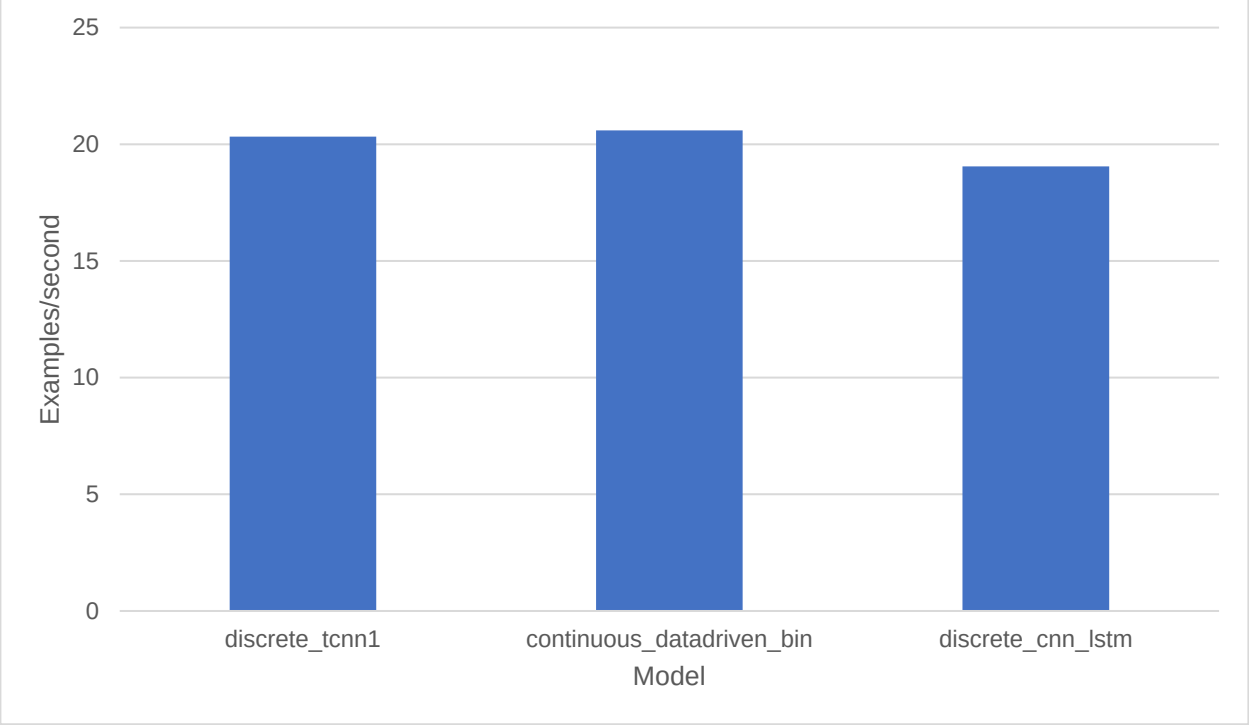
Figure 22) GAN training throughput.



8.3 Berkeley DeepDrive

This solution was also tested with the model and dataset released by the Berkeley DeepDrive (BDD) Driving Project, which focuses on advancing computer vision capabilities for automotive applications. This dataset includes 100,000 HD video sequences across many different times of the day, weather conditions, and driving scenarios. We tested the BDD_driving_model released on GitHub, which is the model demonstrated in “Autoencoding beyond pixels using a learned similarity metric” [2]. At this time, the model has only been tested on one DGX-2 server. Integration with Horovod for scaling out to more than one DGX server will be completed in the future.

Figure 23) Training throughput with Berkeley DeepDrive dataset.



9 Conclusion

The DGX-2 server is an extremely powerful deep learning platform that benefits from equally powerful storage and network infrastructure to deliver maximum value. By combining NetApp AFF systems with Cisco Nexus switches, this verified architecture can be implemented at almost any scale, from a single DGX-2 server paired to an AFF A220 system up to potentially 36 DGX-2 servers on a 24-node AFF A800 cluster. Combined with the superior cloud integration and software-defined capabilities of NetApp ONTAP, AFF enables a full range of data pipelines that spans the edge, the core, and the cloud for successful DL projects.

Acknowledgments

We gratefully acknowledge the contributions to this NetApp Verified Architecture by our esteemed colleagues from NVIDIA, Jacci Cenci, Satinder Nijjar, Darrin Johnson, Robert Sohigian, Tony Paikeday, and James Mauro. We would also like to thank the team at Arrow Electronics. We could not have completed this study without the support and guidance of our key NetApp team members Sundar Ranganathan, Robert Franz, and Kesari Mishra.

Our sincere appreciation and thanks to all these individuals, who provided insight and expertise that greatly assisted in the research for this paper.

Where to Find Additional Information

To learn more about the information that is described in this document, see the following resources:

- NVIDIA DGX-1 and DGX-2 servers
  - NVIDIA DGX-2 servers
    https://www.nvidia.com/en-us/data-center/dgx-2/
  - NVIDIA Tesla V100 Tensor core GPU
    https://www.nvidia.com/en-us/data-center/tesla-v100/
  - NVIDIA GPU Cloud
    https://www.nvidia.com/en-us/gpu-cloud/
- NetApp AFF systems
  - AFF datasheet
    https://www.netapp.com/us/media/ds-3582.pdf
  - NetApp Flash Advantage for AFF
    https://www.netapp.com/us/media/ds-3733.pdf
  - ONTAP 9.x documentation
    http://mysupport.netapp.com/documentation/productlibrary/index.html?productID=62286
  - NetApp FlexGroup technical report
    https://www.netapp.com/us/media/tr-4557.pdf
- NetApp Interoperability Matrix Tool
  http://support.netapp.com/matrix
- Cisco Nexus networking

  The following links provide more information about Cisco Nexus 3232C series switches:
  - Cisco Nexus 3232C series switches
    https://www.cisco.com/c/en/us/products/switches/nexus-3232c-switch/index.html
  - Cisco Nexus 3232C configuration guide
    https://www.cisco.com/c/en/us/support/switches/nexus-3000-series-switches/products-installation-and-configuration-guides-list.html
  - Cisco Nexus 3232C command line reference
    https://www.cisco.com/c/en/us/support/switches/nexus-3000-series-switches/products-command-reference-list.html
- Ansible Automation
  - Ansible website – https://www.ansible.com/
  - NetApp DevOps website – https://netapp.io
  - ONTAP AI Ansible blog and demo
    https://blog.netapp.com/how-to-configure-ontap-ai-in-20-minutes-with-ansible-automation/
- Machine learning framework
  - TensorFlow: An Open-Source Machine Learning Framework for Everyone
    https://www.tensorflow.org/
  - Horovod: Uber's Open-Source Distributed Deep Learning Framework for TensorFlow
    https://eng.uber.com/horovod/
  - Enabling GPUs in the Container Runtime Ecosystem
    https://devblogs.nvidia.com/gpu-containers-runtime/
- Datasets and benchmarks:
  - ImageNet – http://www.image-net.org/
  - TensorFlow benchmarks – https://www.tensorflow.org/performance/benchmarks
  - Large-scale CelebFaces Attributes (CelebA) dataset
    http://mmlab.ie.cuhk.edu.hk/projects/CelebA.html

- Berkeley DeepDrive Project – https://deepdrive.berkeley.edu/
- The Berkeley DeepDrive Video Dataset(BDD-V) – https://bdd-data.berkeley.edu/
- BDD_Driving_Mode – https://github.com/gy20073/BDD_Driving_Model
- YouTube-8M Dataset – https://research.google.com/youtube8m/
- YouTube-8M TensorFlow Starter Code – https://github.com/google/youtube-8m

References

[1] Anders Boesen Lindbo Larsen, Søren Kaae Sønderby, Hugo Larochelle, and Ole Winther. Autoencoding beyond pixels using a learned similarity metric. Cornell University, 2016.

[2] Huazhe Xu, Yang Gao, Fisher Yu, and Trevor Darrell. End-to-end learning of driving models from large-scale video datasets. Cornell University, 2016.

Appendix

This section contains additional results for the tests that were performed using this architecture.

Scaled ImageNet Testing

Figures 24 and 25 show the GPU and CPU utilization, respectively, during training with the scaled ImageNet dataset. As noted earlier, these graphs show that the DGX-2 server was almost 100% utilized in both CPU and GPU during this testing.

Figure 24) DGX-2 server GPU utilization with scaled ImageNet dataset.

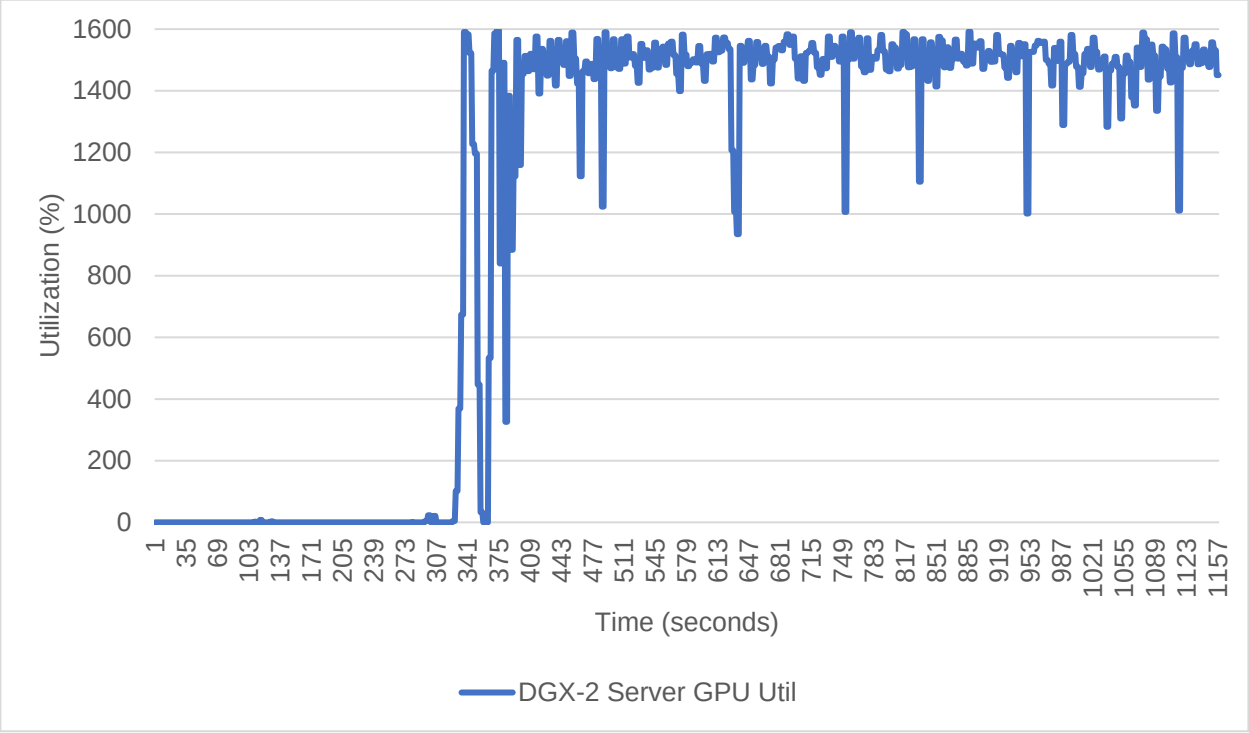
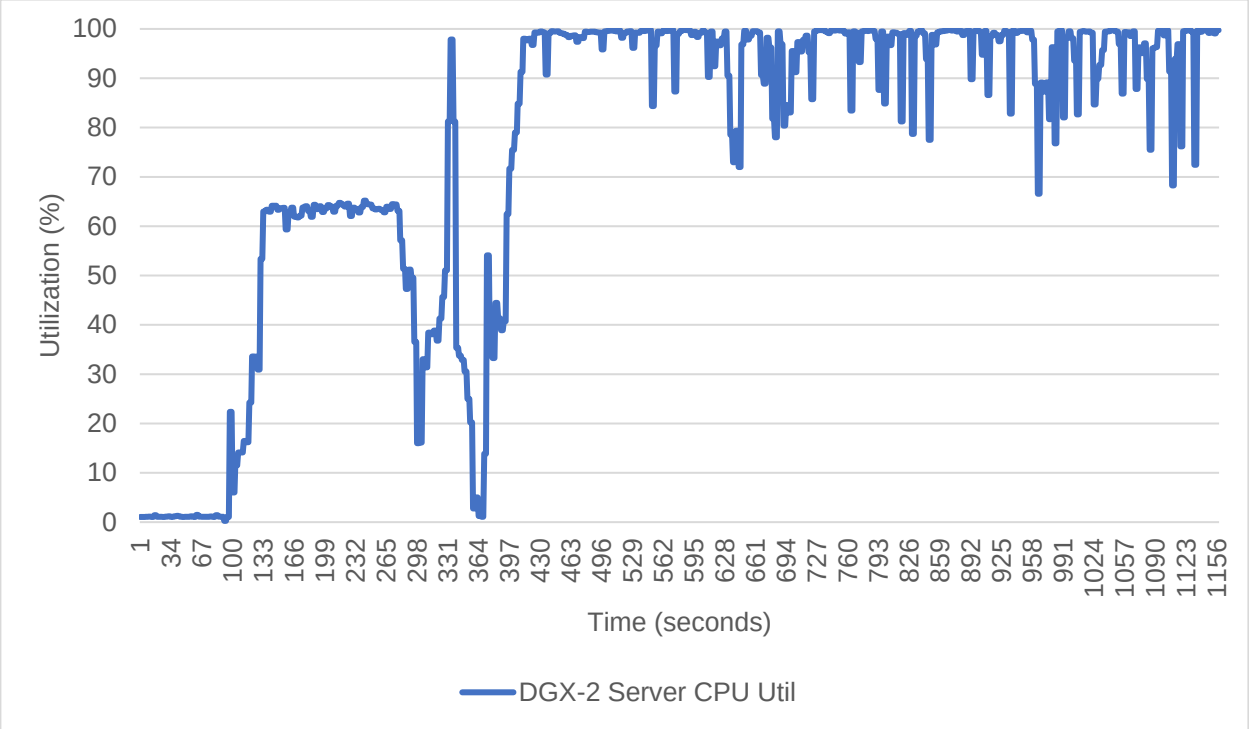


Figure 25) DGX-2 server CPU utilization with scaled ImageNet dataset.



YouTube-8M Dataset

Figures 26 and 27 show the CPU and GPU utilization of the DGX-2 servers while testing the YouTube-8M dataset and model. However, even though this dataset is composed of images, the bottleneck of this training is on the CPU of the DGX-2 server. Figure 26 shows the CPU and GPU utilization of running the frame-level logistic model, which uses CPU to correlate the features shown in the series of frames before they are fed to the GPU. The graph shows that CPU utilization is at or above 90%, while GPU utilization is 75% or less.

Figure 26) DGX-2 server CPU and GPU utilization for YouTube-8M frame-level logistic model.

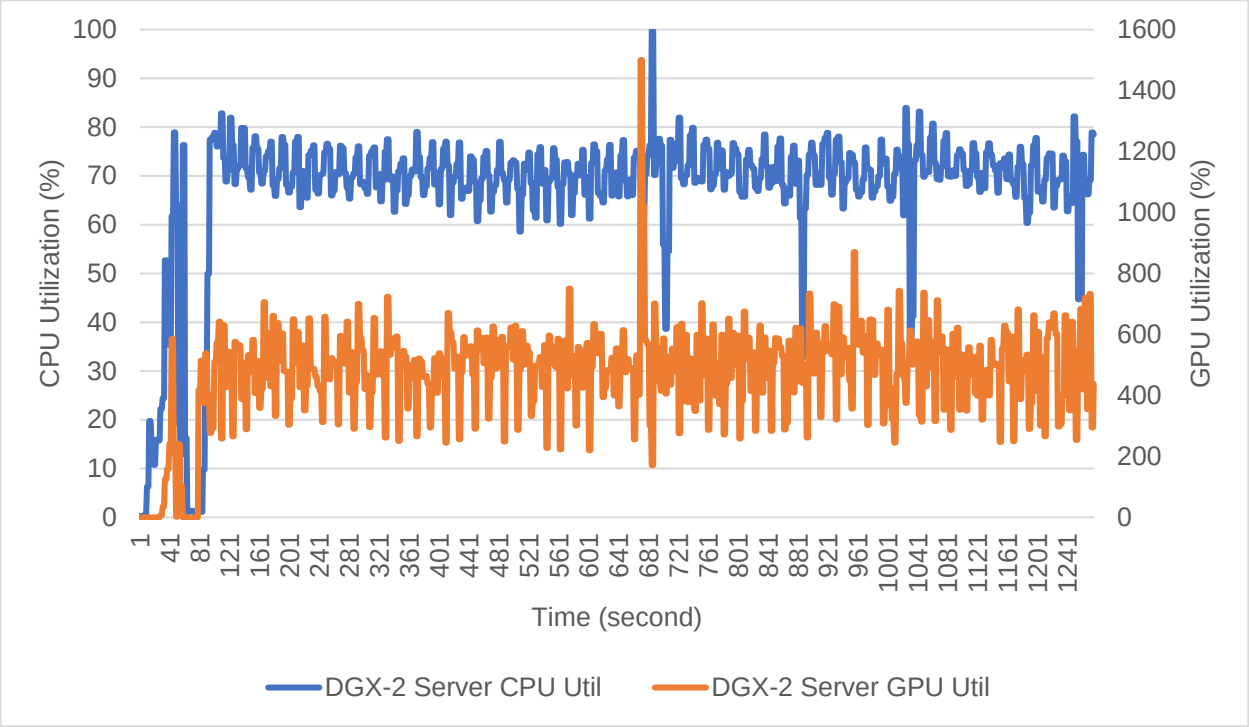
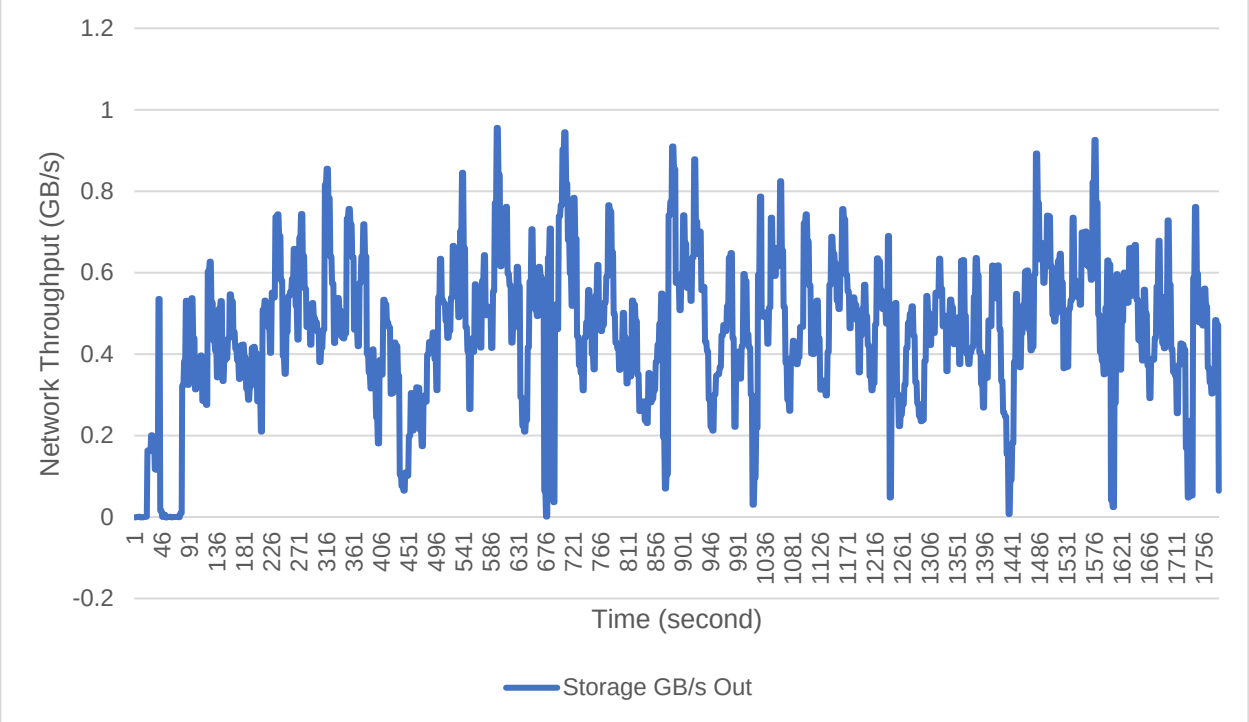


Figure 27 shows the storage network throughput during training of the YouTube-8M dataset and model, which is significantly lower than that observed with ImageNet training.

Figure 27) AFF A800 network throughput for YouTube-8M frame-level logistic model.



Refer to the Interoperability Matrix Tool (IMT) on the NetApp Support site to validate that the exact product and feature versions described in this document are supported for your specific environment. The NetApp IMT defines the product components and versions that can be used to construct configurations that are supported by NetApp. Specific results depend on each customer's installation in accordance with published specifications.

Copyright Information

Trademark Information