v1.3 | February 2022 | RA-2114

REFERENCE ARCHITECTURE

Nutanix for AI

NUTANIX
YOUR ENTERPRISE CLOUD

Copyright

Contents

1. Executive Summary

Artificial intelligence (AI) models human intelligence in a computer system to drive advanced data processing capabilities without needing human intervention. Computer systems equipped with AI can be trained to sense, infer, and in other ways act "human," using silicon technology to create artificial neural networks (ANNs) that replicate the inner workings of the human brain. Like the human brain, an ANN is composed of millions of configurable network layers, and the granularity at which these units operate can quickly translate to highly complex models with large data sets.

There are many forms of AI in development. The most popular is machine learning—the idea that when machines encounter a series of data sets, they can learn, think, and act for themselves based on input data. AI has already unlocked many advanced processing capabilities, including:

- Visual perception (for example, autonomous vehicles, robots).
- Speech recognition systems (Siri, Alexa).
- Decision-making systems.
- Predictive healthcare.
- Language-translation applications.

Data scientists use data derived from a large collection of data sets to predict outcomes through a statistical approach to creating complex algorithms known as the machine learning model. There is a clear correlation between the accuracy of the predicted outcome and the size of the data set; however, larger data sets require more computational power to process, which places stress on related infrastructure components (particularly storage and network).

To support highly complex AI use cases, organizations need an agile IT infrastructure that is quick to start, simple to scale, and built with the data services software developers and data scientists need. Reducing complexity, improving data security, and eliminating bottlenecks must be top priorities. Traditional IT infrastructure is ill-suited to address the serious problems that come with scaling AI, making it the ideal time to explore the Nutanix platform's web-scale approach.

Nutanix has partnered with NVIDIA and Mellanox to build and validate this reference architecture to assist customers with planning and deploying their AI infrastructures.

Table 1: Solution Details

Product Name	Product Version	Nutanix AOS Version	Hypervisor	Hypervisor Version
NVIDIA DGX-1 OS	3.7.1	N/A	N/A	N/A
Mellanox NEO	2.2.0.4	5.9	AHV	2017830166
Mellanox Onyx	3.6.8130	N/A	N/A	N/A
Nutanix Files	3.1	5.9	AHV	2017830166
Nutanix Prism Central	5.9	5.9	AHV	2017830166

2. Introduction

Audience

This reference architecture is part of the Nutanix Solutions Library. We wrote it for data scientists, architects, and deep-learning experts responsible for designing a scalable platform to run AI workloads. Readers should already be familiar with enterprise architectures, virtualization, storage, and convolutional neural networks (CNNs) and models.

Purpose

In this document, we cover the following topics:

- Overview of the Nutanix solution.
- Overview of the solution architecture.
- The benefits of running AI workloads on Nutanix.
- Solution verification and testing results.

Unless otherwise stated, the solution described in this document is valid on AOS 5.9 and later AOS versions.

Table 2: Document Version History

Version Number	Published	Notes
1.0	November 2018	Original publication.
1.1	January 2020	Content refresh.
1.2	February 2021	Refreshed content.
1.3	February 2022	Refreshed content.

3. Solution Overview

The solution architecture we present here is composed of hardware and software elements integrated to form a complete solution for onboarding AI workloads into an environment. We have created and validated our design using an enterprise design methodology and technology components from Nutanix, NVIDIA, and Mellanox.

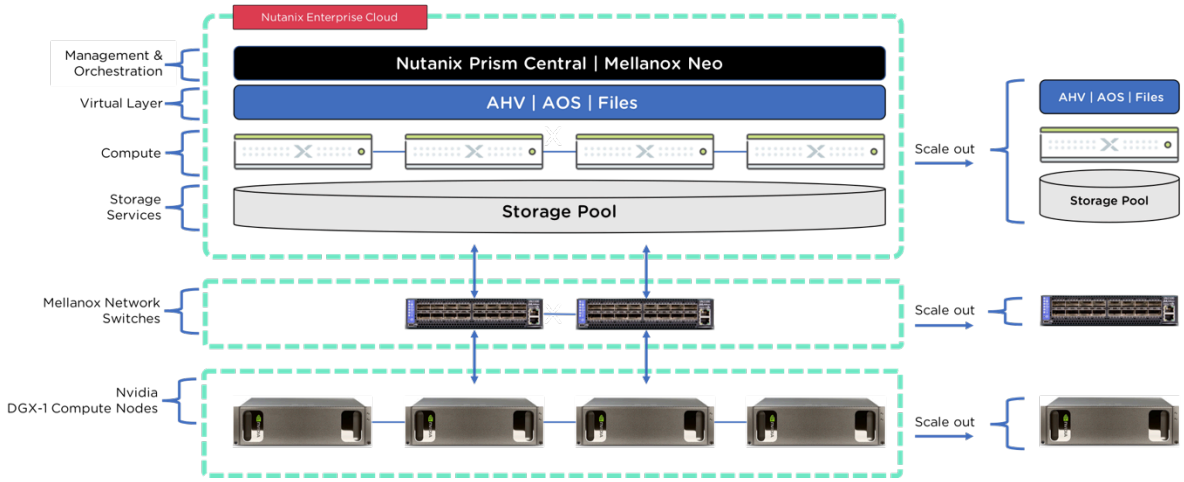

Figure 1: Logical Design of the Nutanix AI Solution

Hardware Components

We used the following hardware components for the test described in this reference architecture.

Nutanix 3060-G6 All-Flash Appliance

The Nutanix 3060-G6 all-flash hardware appliance offers the best blend of compute performance and storage capacity for AI workloads. Each consolidated 2RU Nutanix 3060 block has four independent compute nodes with up to six direct-attached SSDs or all-flash devices with NVMe, representing a total of up to 24 storage devices per block. Since each appliance is built with hyperconverged technology and supported by a virtualization layer, each appliance delivers:

- Compute performance: CPU and memory to run VMs.
- Storage performance: A software-defined storage processor (referred to as a CVM, or Controller Virtual Machine).
- Storage capacity: Locally attached storage.

Nutanix 3060 all-flash appliances are configured to order and feature the latest Intel Skylake processors.

Table 3: Nutanix Hardware Configuration

Item	Configuration (Per Node/Appliance)
Server compute	Dual Intel Skylake Gold 5120 (14 cores / 2.2 GHz)
Storage capacity (all flash)	6x 960 GB SSD
Memory	512 GB
Network connections	1 Dual-port 25 GbE SFP+

When our nodes were clustered together, we provided a usable capacity of 6.05 TiB (excluding data reduction) and planned for n + 1 redundancy.

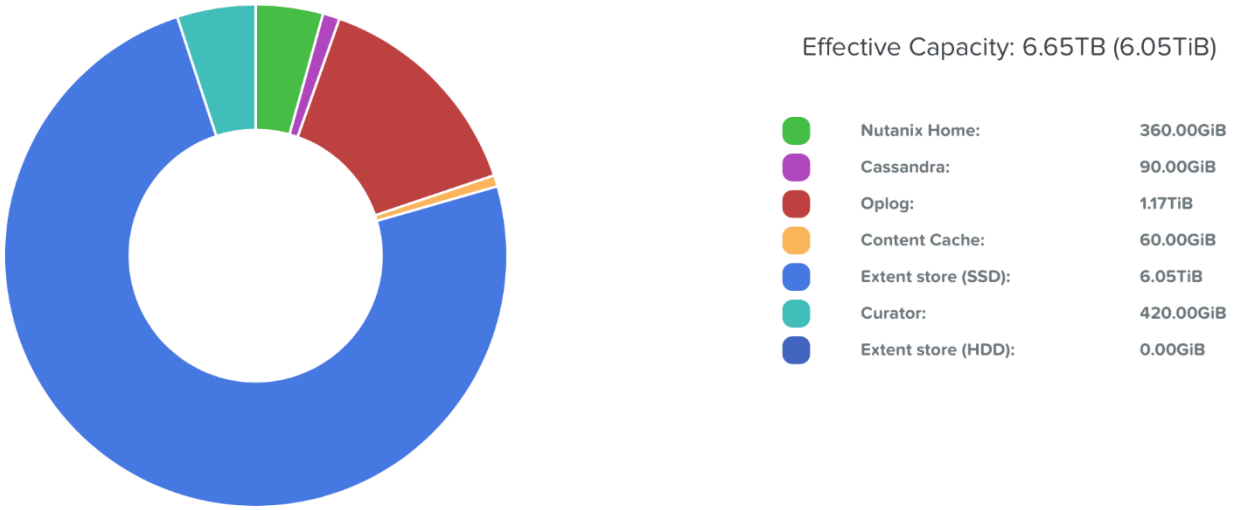

Figure 2: Cluster Capacity Planning and Overheads

For further information regarding the Nutanix hardware specifications, visit the Nutanix website.

NVIDIA DGX-1

The NVIDIA DGX-1 is an integrated system that provides a flexible, maximum-performance platform for the development and deployment of deep-learning applications in both production and research settings. A cluster of DGX-1 servers provides performance for deep-learning training using 100 Gbps RDMA from Mellanox ConnectX NICs. For more information, see the NVIDIA DGX-1 With Tesla V100 System Architecture white paper.

Table 4: NVIDIA Hardware Configuration

Item	Configuration
GPUs	8 Tesla V100s
Performance (mixed precision)	1 petaFLOPS
GPU memory	256 GB total system
CPU	Dual 20-core Intel Xeon E5-2698 v4 2.2 GHz
NVIDIA CUDA cores	40,960
NVIDIA tensor cores	5,120
System memory	512 GB 2,133 MHz DDR4 RDIMM
Storage	4x 1.92 TB SSD RAID 0
Network	Dual 10 GbE, quad 100 GbE

For more information about the NVIDIA DGX-1, visit the NVIDIA website.

Mellanox Spectrum Switches: SN2100

The Mellanox SN2100 is powered by the Mellanox Spectrum application-specific integrated circuit (ASIC) and contains 16 ports that you can configure to operate at 25, 40, 50, or 100 GbE. The SN2100 is capable of 3.2 Tbps throughput with a landmark processing capacity of 2.38 billion packets per second in a compact 1RU form factor.

The SN2100 is part of Mellanox's end-to-end RDMA over Converged Ethernet (RoCE) solution, which provides high performance and ultra-low-latency interconnectivity ranging from 10 GbE to 100 GbE in the datacenter. Other devices we used in this solution include ConnectX-4 NICs and LinkX copper or fiber cabling. The switch design (two SN2100s mounted in 1RU) provides a high-density, side-by-side 100 GbE switching solution that can scale up to 128 ports while maintaining low power consumption.

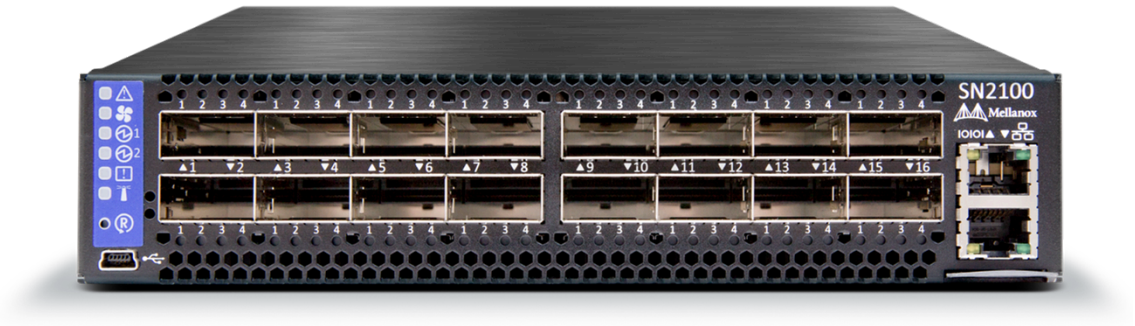

Figure 3: Mellanox SN2100 Switch

Mellanox Spectrum Switches: SN2700

Mellanox SN2700 switches provide a predictable, high-density 100 GbE switching platform for the growing demands of today's datacenters. You can mount a variety of operating systems on the SN2700 switch and take advantage of open networking and the capabilities of the Mellanox Spectrum ASIC. The SN2700 switch provides speeds from 10 Gbps to 100 Gbps per port and port density that enables full rack connectivity to any server at any speed. The uplink ports allow a variety of blocking ratios that suit various application requirements. The SN2700 switch is enhanced for deep learning, storage, and virtualized architectures and meets the growing need for wire-speed performance, zero packet loss, and consistently low latency. Dynamically shared buffers offer congestion management for nonblocking data throughput.

Figure 4: Mellanox SN2700 Switch

Software Components

We tested all the previously mentioned hardware with the following software components for this reference architecture.

Table 5: Validation Software Components

Product Name	Product Version
Nutanix AHV	2017830166
Nutanix Cloud Platform OS	5.9
Nutanix Prism Central	5.9
Nutanix Files	3.1
NVIDIA DGX-1 OS	3.7.1
Mellanox Onyx	3.6.8130
Mellanox NEO	2.2.0.4

Nutanix Files

Nutanix Files is a software-defined, scale-out file storage solution that provides a repository for unstructured data, such as home directories, user profiles, departmental shares, application logs, backups, and archives. Flexible and responsive to workload requirements, Files is a fully integrated core component of Nutanix.

You can deploy Nutanix Files on an existing cluster or a standalone cluster. Unlike standalone NAS appliances, Files consolidates VM and file storage, eliminating the need for an infrastructure silo. Administrators can manage Files with Nutanix Prism, unifying and simplifying management. Integration with Active Directory enables quotas and

access-based enumeration, as well as self-service restores with the Windows previous version feature. Nutanix Files also supports file server cloning, which lets you back up Files off-site, as well as run antivirus scans and machine learning workloads without affecting production.

Nutanix Files can run on a dedicated cluster or be collocated on a cluster running user VMs. Beginning with AOS 5.0, Nutanix supports Files with both ESXi and AHV. Files includes native high availability and uses AOS storage for intracluster data resiliency and intercluster asynchronous disaster recovery. Distributed storage also provides data efficiency techniques such as erasure coding (EC-X), compression, and deduplication.

> **Note:** We validated this solution on Nutanix Files version 3.1.

Nutanix Prism

Nutanix Prism provides central access for administrators to configure, monitor, and manage virtual environments in an efficient and elegant way. Powered by advanced data analytics, heuristics, and rich automation, Prism offers unprecedented simplicity by combining several aspects of datacenter management into a single consumer-grade solution. Using innovative machine learning technology, Prism can mine large volumes of system data quickly and easily and generate actionable insights for optimizing all aspects of virtual infrastructure management. Prism is a part of every Nutanix deployment and has two core components:

- Prism Element

  Prism Element is a service built into the platform for every Nutanix cluster deployed. Prism Element provides the ability to fully configure, manage, and monitor Nutanix cloud clusters running any hypervisor. Because Prism Element only manages the cluster it is part of, each Nutanix cluster in a deployment has a unique Prism Element instance for management.

- Prism Central

  Prism Central offers an organizational view into a distributed Nutanix deployment, with the ability to attach all remote and local Nutanix cloud clusters to a single Prism Central deployment. This global management experience offers a single place to monitor performance, health, and inventory for all Nutanix cloud clusters. Prism Central is available in a standard version included with every Nutanix deployment and as a Pro version that is licensed separately and enables several advanced features.

Note: We validated this solution on Prism Central version 5.9.

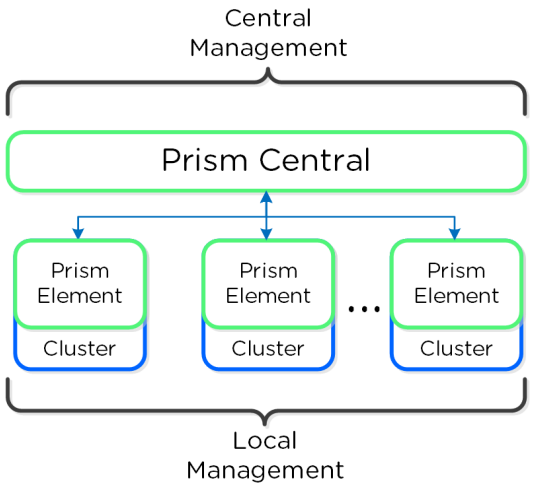

Figure 5: Relationship Between Prism Element and Prism Central

NVIDIA Server OS

The NVIDIA Server OS (formerly Ubuntu OS) is the base OS image installed on the DGX-1 system. It contains all drivers, utilities, and Docker binaries necessary to successfully run GPU-accelerated applications in Docker containers maintained by NVIDIA as well as in customized Docker containers. NVIDIA maintains the OS and provides future updates over the network.

Note: We validated this solution on DGX OS Server version 3.1.7, which features Ubuntu 16.04.5 LTS.

NVIDIA GPU Cloud

The NVIDIA GPU Cloud (NGC) offers access to a comprehensive catalogue of GPU-accelerated containers for deep-learning software, high performance computing (HPC) applications, and HPC visualization and empowers AI researchers with performance-engineered AI containers.

Mellanox Onyx

Mellanox Onyx is a switch OS designed to meet the demands of next-generation datacenters. With built-in workflow automation, monitoring and visibility tools, and enhanced high-availability mechanisms, among other features, Mellanox Onyx simplifies

network processes, increases efficiency, and reduces operating expenses and time to service.

> **Note:** We validated this solution on Mellanox Onyx version 3.6.8130.

Mellanox NEO

Mellanox NEO is a platform for datacenter network orchestration, designed to simplify the network provisioning, monitoring, and operations of the modern datacenter. NEO offers robust automation capabilities that extend existing tools, from network staging and bring-up to day-to-day operations. NEO serves as a network API for Mellanox Ethernet solutions.

> **Note:** We validated this solution on Mellanox NEO version 2.2.0.4.

4. Solution Architecture

When architecting this solution, we took into consideration the following AI-specific requirements:

- Availability: The solution must have high availability and maintain availability during upgrades and failure scenarios.
- Recoverability: The solution must have a strategy for recovering AI workloads and restoring data in case of a disaster, while also minimizing recovery point objectives (RPOs) and recovery time objectives (RTOs).
- Manageability: The solution must reduce administrative effort for day-one and day-two operations.
- Performance and scalability: The solution must increase resilience, performance, and capacity by scaling without impacting performance.
- Security: The solution must implement security policies across the full stack.
- Storage: The solution must reduce administrative effort and maximize storage performance regardless of the application workload.
- Network: The solution must provide a network architecture that maximizes bandwidth and decreases latency.

Availability

In this section, we discuss the common failure scenarios in the solution as well as how the various hardware and software components increase infrastructure availability.

Table 6: Summary of Availability Design

Configuration Item	Parameter
Nutanix CVM design	1 per host (4 total) 12 vCPUs, 64 GB of RAM
Nutanix file server VM design	3 VMs (initial configuration) 4 vCPUs, 12 GB of RAM
Nutanix file server export protocol	NFS v4
Nutanix file server export type	Sharded directories
Nutanix file server DNS settings	Records automatically created in AD during provisioning (round robin)
Cluster redundancy factor	2
Cluster high availability reservation	Enabled
Cluster virtual IP address	Set
Cluster iSCSI data services IP	Set

Nutanix Availability Features

Nutanix can operate as either a single node or a cluster of nodes in which three or more nodes share resources and distributed data, which increases application and storage availability. As represented in the following figure, as AOS ingests data from the NVIDIA DGX-1 system or application, it creates a local copy on the home node and distributes a secondary copy to another node in the cluster. Then the system sends an acknowledgment back to the ingesting application that the write operation is complete. Consequently, as the application writes data, the system always stores a secondary copy stored on another node. This process is the replication factor, which by default is set to 2. An administrator can set the replication factor to 3, which requires a minimum of five Nutanix nodes but dramatically increases availability.

Note: We selected replication factor 2 because it provides an acceptable level of availability for this architecture. However, if customers are evaluating larger clusters, it may be useful to increase the replication factor to 3.

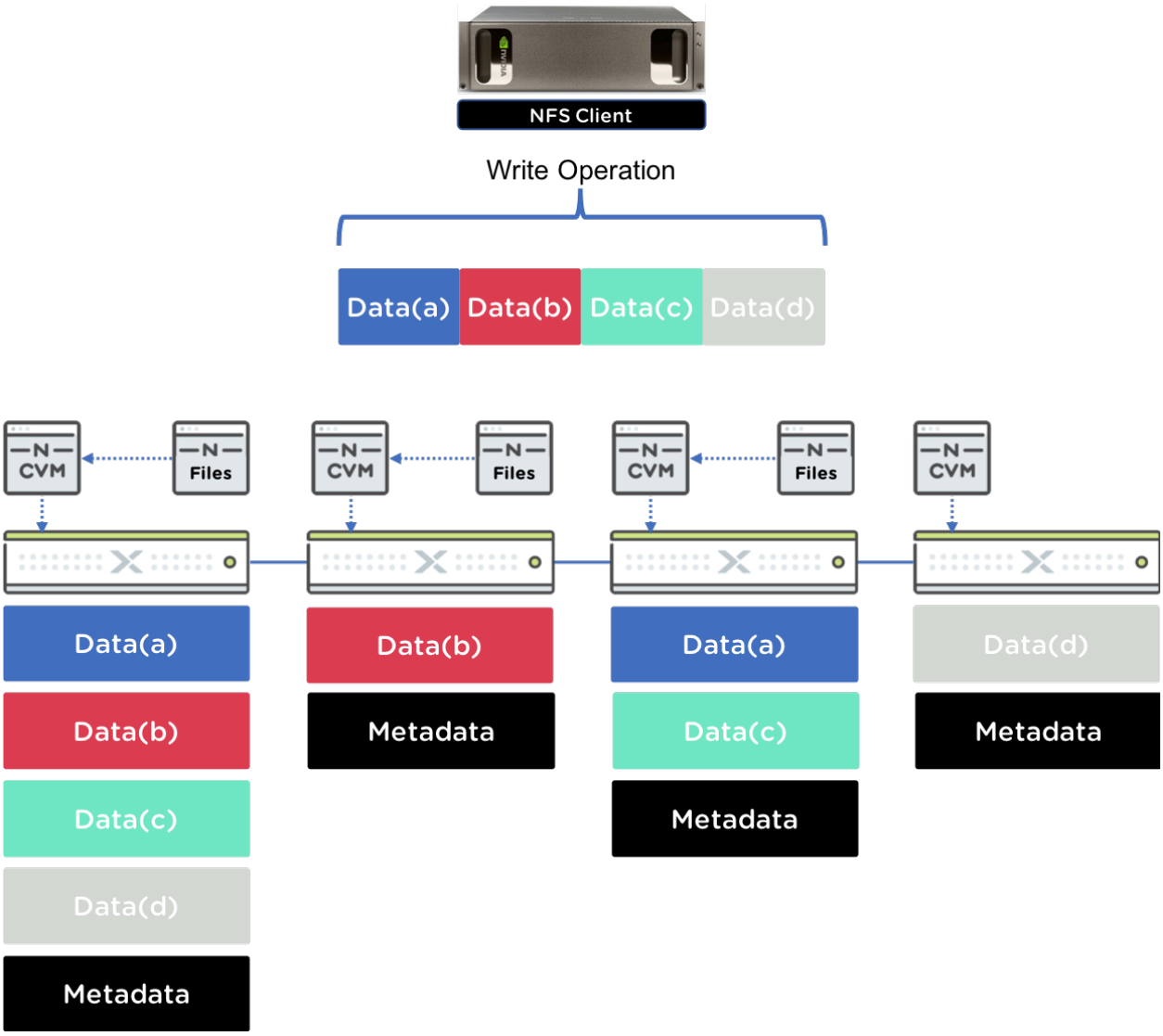

Figure 6: Nutanix Data Availability

Block Awareness

A block is a rack-mountable enclosure that contains one to four Nutanix nodes. In multinode blocks, the power supplies and the fans are the only components shared by nodes in a block. When certain conditions are met, Nutanix cloud clusters are block aware, which means that redundant copies of any data needed to serve I/O are placed on nodes that aren't in the same block, which maximizes the solution's availability. When

you scale your AI infrastructure, it's important to note that block awareness is applied automatically when all the following conditions are met:

- The cluster is three or more blocks (unless the cluster was created with replication factor 3, in which case the cluster is five or more blocks).
- Every storage tier in the cluster contains at least one drive on each block.
- Every container in the cluster has a replication factor of at least 2.
- The storage tiers on each block in the cluster are of comparable size.
- The size of cluster SSD tiers with replication factor 2 isn't more than 33 percent different across blocks.
- The size of cluster SSD tiers with replication factor 3 isn't more than 25 percent different across blocks.

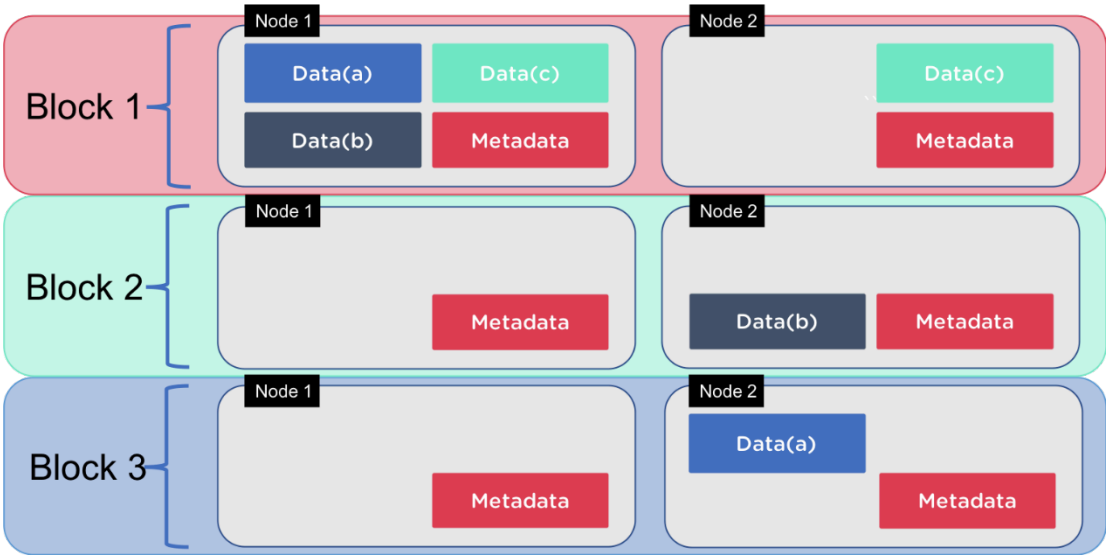

Figure 7: Nutanix Data Availability with Block Awareness

Tip: When you scale your AI infrastructure with Nutanix blocks, consider physically distributing blocks across racks to further increase availability.

Node Availability

Because all Nutanix cloud clusters have at least three nodes, we used an additional node in our solution to meet the requirement for n + 1 availability. This extra node allows us to handle planned or unplanned events without operating in a degraded state. When you

enable **Cluster High Availability** in Prism, the system maintains this level of availability automatically.

Controller VM

Nutanix is a 100 percent software-defined solution that places a software storage controller (the CVM) on each node in the cluster. The storage controller actively accepts I/O from applications running locally on that node and participates in cluster-wide operations such as replicating data, self-healing, and rebalancing data.

Nutanix Files Load Balancing

As a complement to the CVM, Nutanix Files also serves NFS and SMB requests from clients and internal and external systems. Nutanix Files use the CVM for reads and writes to distributed storage, providing resilience (replication factor 2), data integrity, and scalability, as detailed in the Nutanix Data Availability figure. Nutanix Files doesn't need to be present on every node; rather, it starts off with a minimum of three file server VMs (FSVMs) and automatically scales out when needed. The following figure shows a high-level representation of the relationship between the Nutanix CVM and FSVMs—specifically the distribution of NFS exports and directories across multiple FSVMs.

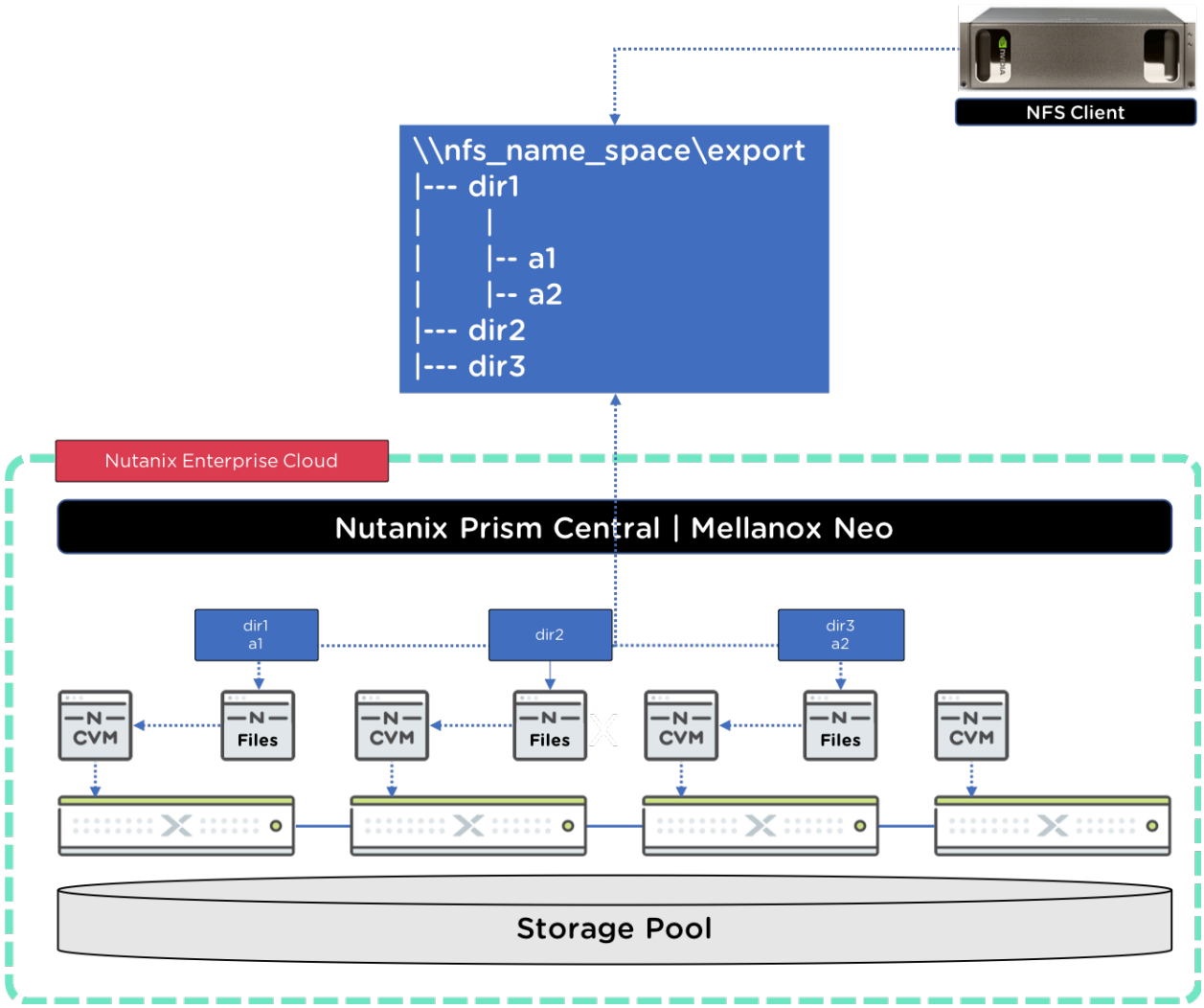

Figure 8: High-Level Nutanix Files Architecture

Refer to the Nutanix Files tech note and the Nutanix Volumes best practices guide for more detailed information on load balancing. For AI architects, a DGX system is equivalent to the NFS client.

> **Note:** In our testing, we implemented Nutanix Files with the Sharded Directory option and configured an iSCSI data services IP.

Mellanox Availability Features

Mellanox SN2100 series switches are designed for high availability from both a software and hardware perspective. Key high availability features include:

- Color-coded PSUs and fans.
- Up to 64x 10 or 25 GbE ports, 32x 50 GbE ports, or 16x 100 GbE ports.
- MLAG for active-active L2 multipathing.
- 64-way equal-cost multipath (ECMP) routing for load balancing and redundancy.
- 1 + 1 power supplies.

The following table provides a summary of how Mellanox SN2100 switches maintain availability during certain network failures.

Table 7: Network Failures Summary

Event	Detection	Action	Effect on Network
Leader down	Three continuous keepalives were lost and the leader isn't visible on the management network.	Subordinate role changed to standalone. Flush all MLAG MACs. Flush all IPL MACs.	No traffic loss.
Leader up	IPL up and received leader keepalive.	Standalone role changed to subordinate.	No traffic loss.
Subordinate down	Three continuous keepalives are lost and the subordinate isn't visible on the management network.	Flush any MACs the subordinate has learned. Flush all IPL MACs.	No traffic loss.
Subordinate up	IPL up and received subordinate keepalive.	Sync subordinate with leader tables.	No traffic loss.

Event	Detection	Action	Effect on Network
MLAG port failure	Port-down notification received from l2 protocol on leader or subordinate.	Local switch: Redirect to IPL and send failure notification. Peer switch: Open isolation.	Less than 1 second of traffic loss.
IPL failure	Three continuous keepalives were lost or the IPL port is down and the peer is visible on the management network.	Subordinate: MLAG operational state moved to Down. Leader: No change.	Fail back to two standalone switches.

Recoverability

Nutanix Recoverability Features

Following modern data-protection methodologies, Nutanix provides administrators and users quick-restore access using self-service restore (SSR) and site recovery with Nutanix-based snapshots. The Nutanix disaster recovery (DR) strategy is simple and can protect AI workloads. In the following figure, Cluster 1 and Cluster 2 could be geographically separate and serve as independent failure domains, but they are still available to provide storage services and compute for VMs at their location.

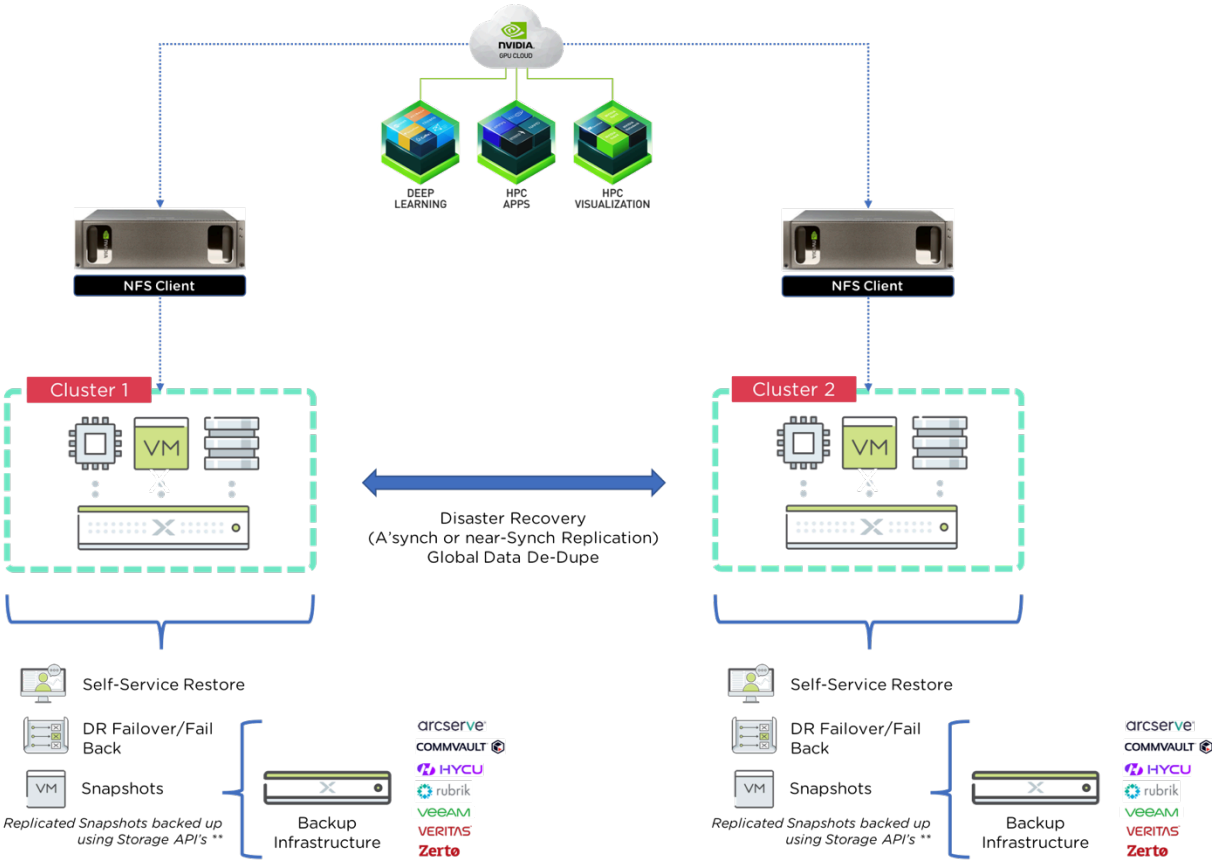

Figure 9: Replication Between Two Clusters

In each failure domain, a DGX or cluster of DGXs serves as a platform for running AI workloads and Docker-based containers pulled from the NGC repository. Although the Docker instances are stateless and you can easily pull them from or push them to the NGC repository, you must protect all training data and models and their respective data and be able to restore the data within the RPOs and RTOs defined in your organization's service-level agreements (SLAs).

The data protection and recovery strategy in this architecture uses local snapshots (stored in the cluster) to capture the FSVMs as restore points for the administrator or users to quickly restore their data. Local snapshots provide the lowest RPO and RTO; however, local snapshots aren't considered a backup or a means of disaster recovery because they don't protect against corruption, human error, environmental conditions, or other disasters.

When you create a file server, Prism automatically sets up a corresponding protection domain, which it annotates with the Nutanix Files cluster UUID and file server name. Prism also creates multiple consistency groups in a protection domain, including a group that includes all FSVMs.

Once Nutanix Files is protected, all future operations on it (such as adding or removing FSVMs or adding or deleting volume groups) automatically update the corresponding consistency group in the protection domain.

In our reference architecture, the administrator can configure asynchronous replication (replication intervals greater than 15 minutes) between the two protection domains (Cluster 1 and Cluster 2) for FSVMs that provide NFS storage to our DGX-1s. This configuration supports a minimum RPO of 60 minutes.

However, if you also run VMs in your environment, it's beneficial to use the near-synchronous option (RPO between 1 and 15 minutes) since this protection strategy is supported with AHV iSCSI-backed storage.

RPOs depend on considerations such as the size of the data set, change rate, and available network bandwidth between sites. Nutanix AOS features an intelligent method to automatically switch between replication strategies if the WAN is not capable of meeting the desired RPO.

We have provided a list of documents that you can view for additional information on recoverability topics:

- For additional information on protection domains and consistency groups, see the Protection Domains section of the Nutanix Data Protection and Disaster Recovery best practices guide.
- For further details regarding our partner ecosystem, solutions, and documentation for backup, refer to the Nutanix website.
- For more information on self-service restore features, see the Nutanix Data Protection and Disaster Recovery best practices guide.
- For additional information on how to restore Nutanix Files after a cluster failure, see the cluster failure and restoration section of the Nutanix Files tech note.
- For further details on how Nutanix Files uses cloning, see the cloning section of the Nutanix Files tech note.

Table 8: Summary of Recoverability Design

Configuration Item	Parameter
Protection domain for Nutanix Files	Enabled, hourly snapshots, application consistent, retention policy 2 (local/remote)
Remote site type	Nutanix cluster
Replication strategy	Asynchronous replication with AOS
Remote site capabilities	Disaster recovery
RPO	60 minutes

Mellanox Recoverability Features

Mellanox NEO automates the backup and restoration of switch configuration and images. Once the virtual appliance has been downloaded and successfully provisioned, administrators can automate the backups (snapshots) of their switch configurations and restore them using the NEO GUI, removing the need for advanced switch CLI knowledge. Because NEO creates backups of the network switches, reports you generate, and configuration data, we recommend the NEO appliance as part of an enterprise backup strategy. Consider the following options:

- Back up the entire Mellanox NEO VM.
- Use the NEO tool to redirect datafiles to a network share and subsequently back up the individual datafiles. Datafiles include switch backups, configuration data, users and groups, reports, logs, and templates.

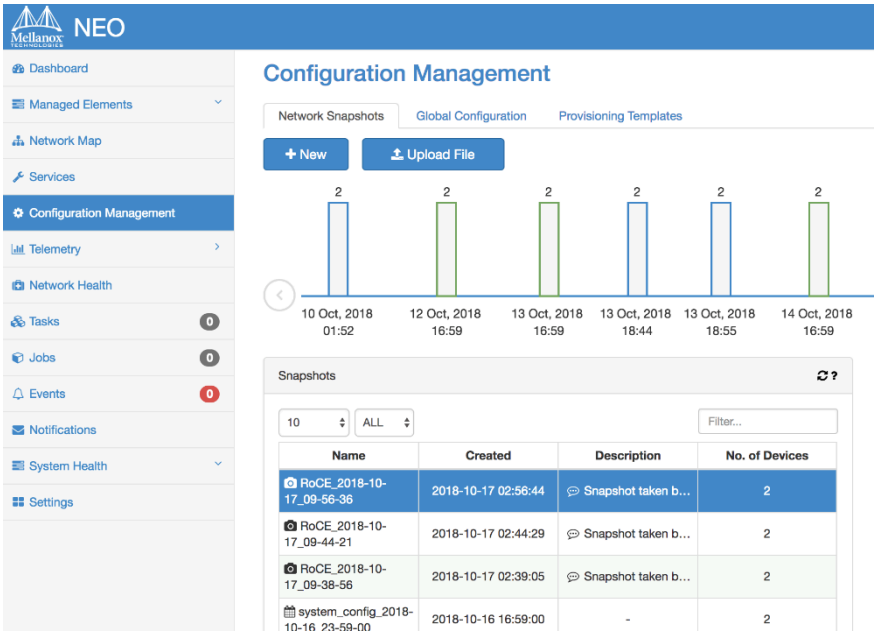

Figure 10: Mellanox NEO Switch Backups

Manageability

Nutanix Prism Element and Mellanox NEO are value-added management tools used in conjunction to simplify day-one and day-two management operations when maintaining an AI infrastructure.

Table 9: Summary of Management Design

Configuration Item	Parameter
Management	Prism Element
Multiple cluster management	Prism Central version 5.9
Prism Central configuration	Single VM (small configuration)
Name servers	Active Directory domain
Network switch configuration	Mellanox Switches via snmpv2
Pulse	Enabled
Acropolis software edition	Ultimate

Configuration Item	Parameter
Prism software edition	Pro
Mellanox NEO configuration	Single VM; 2 vCPUs, 4 GB of vRAM

Nutanix Prism Manageability Features

Prism is the user interface for the Nutanix solution, and it must communicate with and collect data from each of the underlying services. Prism interacts with a distributed configuration management database for cluster configuration data and with a distributed NoSQL key-value store for statistics to present to the user. It also liaises with the hypervisor hosts for VM status, configuration, performance data, and related information. To give you more visibility into your VMs, Prism delivers a VM-focused management view. Refer to the Nutanix Prism and Nutanix Prism Operation Tiers tech notes for more information on Prism.

Mellanox NEO Manageability Features

Deploy NEO as a VM either from the Nutanix Self-Service Marketplace or from the Mellanox Support Portal. Once it's set up and configured with an IP address, you can access the tool using a conventional web browser.

An administrator begins the process of discovering the Mellanox SN2100 switches using the Discover functionality. Once that workflow is complete, the administrator then automatically constructs the protocols required to support the network configuration, including multichassis link aggregation groups (MLAGs) and RoCE. You don't need to understand advanced switch commands or settings; we kept all set to their respective defaults for this design.

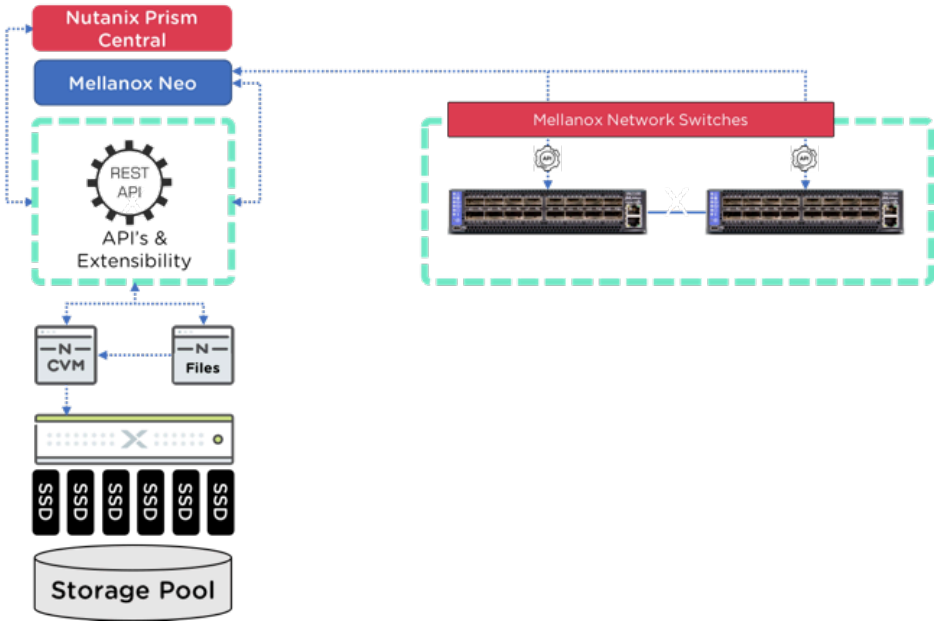

Figure 11: Mellanox and Prism APIs

Nutanix Prism offers enhanced network capabilities, including a new set of APIs to notify the network of guest VM life cycle events that Mellanox NEO uses specifically to automatically provision VLANs on the physical switches during VM creation, migration, and deletion.

NEO also provides day-two functionality such as firmware updates, configuration backups, and analyses of network utilization.

Performance and Scalability

We designed Nutanix to satisfy the performance, availability, flexibility, and consumption requirements of the current generation of applications, including big data and distributed databases. The Nutanix cloud platform OS is optimized to use local and remote performance resources efficiently, so you get an excellent application experience on a highly available distributed architecture. Great performance comes from more than just providing faster flash media; it's rooted in platform architecture, data locality, and the continuous innovation that stems from being software-defined.

Node Scalability

As noted previously, Nutanix is a scale-out solution that not only scales out linearly but fosters the ability to mix different appliances and hardware generations. For example, users adopting this reference architecture can use the recommended NX 3060-G6 appliances; however, if their organization grows, they can adopt larger appliances with higher storage densities or appliances with greater compute power. They could also mix different storage mediums in the cluster (for example, all-flash with hybrid (SSD and HDD) storage devices).

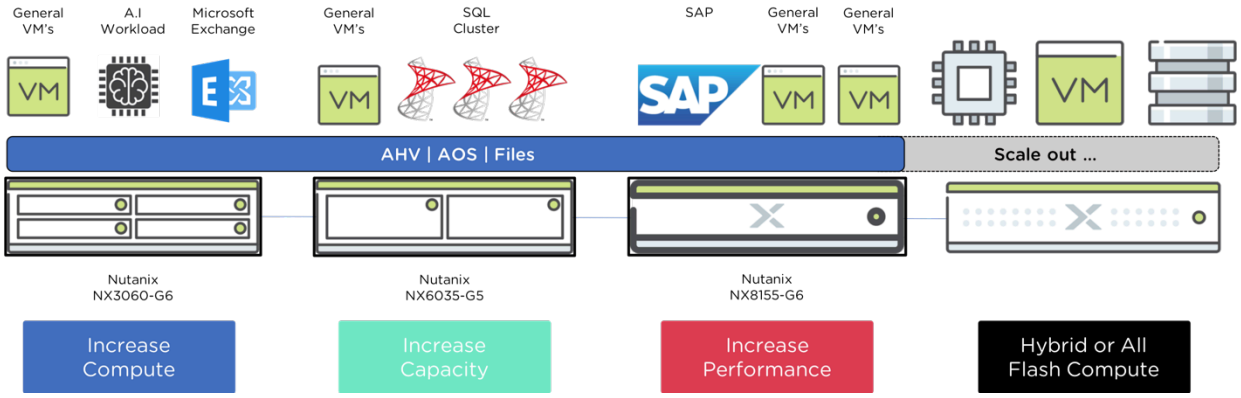

Figure 12: Nutanix Node Scalability

Data Locality

Nutanix has always been committed to data locality as a fundamental requirement in building modern application and enterprise cloud platforms, given that the rapid application deployment and fractional consumption characteristics of the enterprise cloud align perfectly with data locality. Nutanix designed the AOS storage fabric with data locality as a core principle for this reason. Data locality reduces network traffic, removing the network fabric as the choke point in both centralized and remotely accessed storage architectures. It also improves performance—providing lower latencies, higher throughput, and better use of performance resources—by taking advantage of local flash and memory capabilities. The Nutanix hyperconverged architecture can handle far more IOPS than most enterprise workloads require. Each Nutanix CVM performs comparably to a dedicated storage controller in an all-flash storage array, while also enabling linear scale-out.

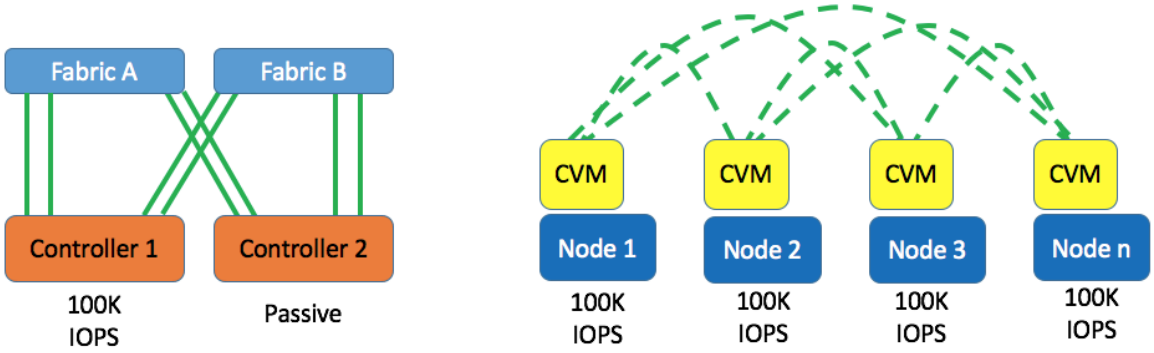

Figure 13: Dual Controller vs. Nutanix CVM

Clustered Performance Resources

Nutanix is a high-performing system for storage I/O, rebuilding from failure, and provisioning operations, snapshots, and upgrade processes. In each cluster, the SSD tier is globally distributed, offering the advantages of data locality and the ability to access remote resources when required. Distributed storage keeps hot data local; when the local tier is full or busy, distributed storage uses the cluster's SSD tier to fulfill requests. Distributed storage ensures that data remains available when a drive or node fails by evenly distributing the rebuild operations across all the CVMs in the cluster. Distributed storage also ensures that Nutanix cloud clusters maintain consistent performance during rebuild operations.

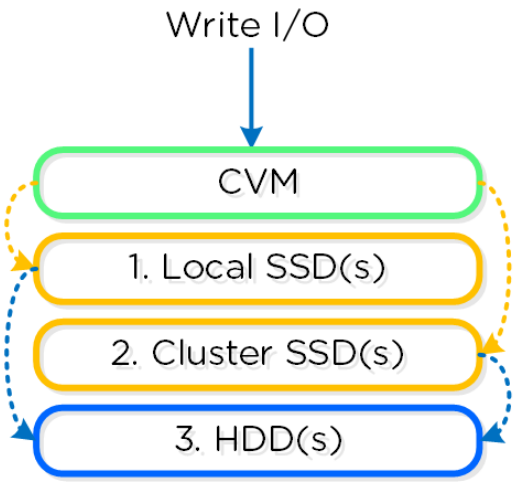

Figure 14: Information Life Cycle Management

Software-Defined Solution

Nutanix software does away with expensive and frequently disruptive upgrades by continually tuning the platform to optimize performance and application-specific requirements. A software-defined approach abstracts the Nutanix platform from the physical infrastructure, enabling flexibility as well as rapid adoption of hardware updates, such as updates to Intel processors that provide additional CPU cores and faster memory speeds. Performance improvements result from:

- Increasing CVM memory.
- Increasing CVM CPU resources.
- Improving storage and software efficiencies.
- Adding nodes to the cluster.

Administrators can configure policies that govern an application's resiliency, capacity reduction, and disaster protection at the application level rather than the platform level. You can also enable or disable features according to the application's requirements; disabling unnecessary features prevents wasted resources and improves user experience.

VSA Architecture

A virtual storage appliance (VSA) architecture offers numerous advantages:

- Supports multiple hypervisors (AHV, ESXi, Hyper-V).
- Upgrades software-defined storage independently from hypervisor software.
- Converts hypervisors simply and without affecting the storage layer.
- Scales storage performance by adjusting resources provided to Nutanix CVM.
- Provides a clear boundary for fault isolation, so storage doesn't take down compute.
- Abstracts security from hypervisor and creates a higher standard of self-healing compliance across hypervisors.

Storage Performance

A core strength of the software-defined Nutanix platform is that it brings the advantages of Moore's Law to storage performance, continually shrinking cost per I/O. Customers also immediately benefit from storage performance improvements with each nondisruptive one-click upgrade to the latest software release.

Security

Cybersecurity threats grow and change every day, demanding perpetual vigilance and adaptation to the shifting security landscape. However, upgrading security in a traditional three-tier architecture is so time consuming and expensive, often involving multiple separate vendors, that some enterprises put off innovation. In light of competing concerns—the need to reclaim resources for innovation versus the need to keep costs down—corporate and government environments demand a simpler approach: one vendor, with a technology platform secured by design, and automated security compliance and reporting. Nutanix takes a holistic approach to security, with an inherently secure platform, extensive automation, and a robust partner ecosystem. For greater detail on the Nutanix approach to cybersecurity, see the Information Security tech note.

Native Local Key Manager

To reduce cost and complexity, we used the Nutanix native local key manager (LKM) for this reference architecture. The LKM runs as a service distributed among all the nodes. You activate it with one click in Prism Element, so anyone can enable encryption without another silo to manage.

Usually, external key managers (EKMs) must be purchased separately for both software and hardware. Because the Nutanix LKM runs natively in the CVM, it's highly available, and the key management services are also upgraded with new software releases. Upgrading both the infrastructure and management services in lockstep ensures that you can maintain your security posture and availability by staying in line with the support matrix.

AOS places keys in a distributed key-value store, where they are available for use by a service called Mantle. A highly available service, Mantle runs on all nodes in the cluster and acts as a proxy for the EKM and internal AOS services. Mantle also allows you to rotate the keys easily across the entire cluster. When you use Prism to create a new container with encryption turned on, Prism generates the new 512-bit data encryption key (DEK) and tells Mantle to store it. Mantle then wraps the DEK with a 256-bit key encryption key (KEK) and stores it in the EKMs when using self-encrypting drives (SEDs). The native LKM service uses the FIPS 140 Crypto module to keep all the DEKs safe. When a service needs to be decrypted, Mantle fetches the KEK to decrypt the DEK. When a service like Stargate starts, the system fetches the key from Mantle and caches it on the client side. You don't need any separate VMs to support the native LKM.

Because the media encryption key (MEK) is shared, each node can read what other nodes have written. A majority of the nodes need to be present to reconstruct the keys. We use the equation K = ceiling (n / 2) to determine how many nodes are required for the majority. For example, in an 11-node cluster (n = 11), we need 6 nodes online to decrypt the data.

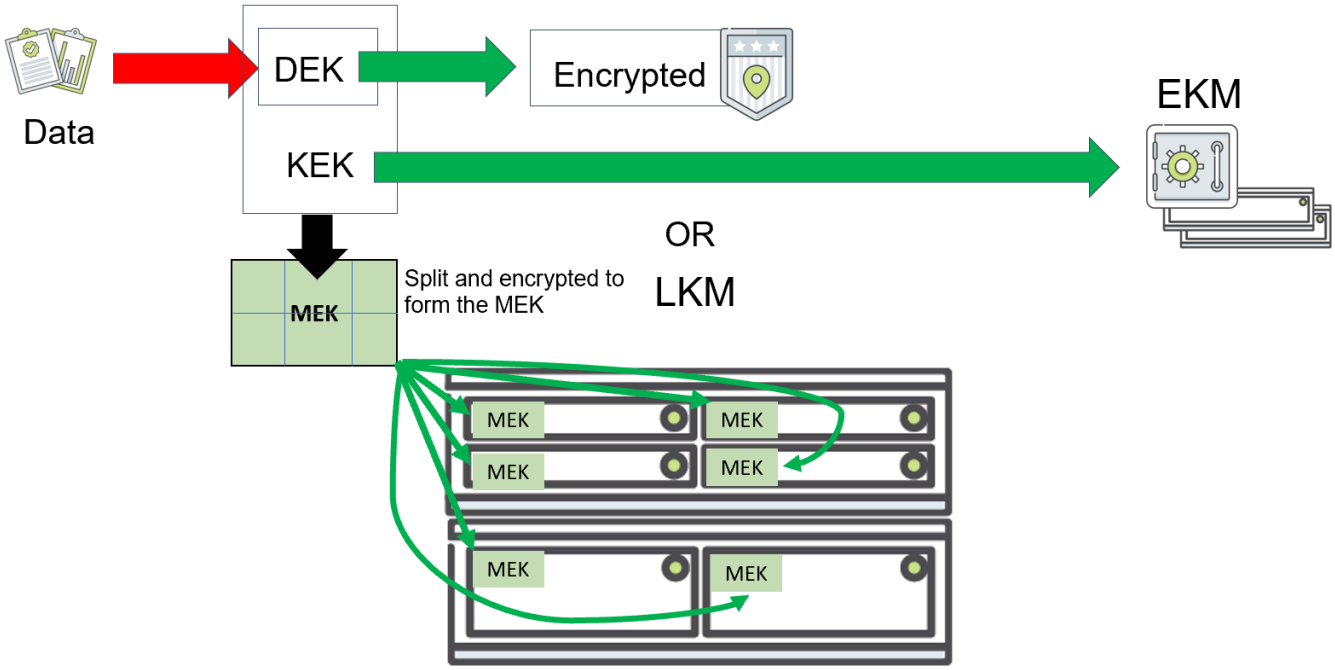

Figure 15: EKM and LKM Workflows

Nutanix also provides an easy way to back up your DEKs from Prism. Each storage container has a DEK, so when a new storage container is created, an alert encourages administrators to make a backup. The backup is password protected and should be securely stored. With the backup in hand, if a catastrophic event happens in your datacenter, you can replicate the data and reimport the backup keys to get your environment running again.

> **Tip:** Back up DEKs as part of your organization's standard operational procedures.

Software-Based Data-at-Rest Encryption

Because AI data can include terabytes of sensitive data and proprietary information across various industry verticals including government organizations, it's important that the storage of such data is encrypted on the disk to mitigate the risk of unauthorized physical access or theft. It's also important that you don't affect performance or management simplicity when you implement encryption.

> **Note:** For this reference architecture, we implemented our native software-based data-at-rest encryption (DARE) feature set to maintain physical security and test the impact of security features on performance.

Software-based encryption uses the Intel Advanced Encryption Standard (AES) New Instructions (NI), an encryption instruction set that improves on the AES algorithm and accelerates data encryption. Supporting AES NI in software gives customers flexibility across hardware models while reducing CPU overhead. The default encryption setting is AES-256.

NFS Data Security

To protect access to the NFS export, which is accessible by each DGX-1 system, we used the NFS V4 framework security options, including authentication against directory services (Open LDAP, Active Directory) via the Keberos protocol (5, 5i, and 5P) and support for Netgroups in NFS, which can limit access to hosts.

Other Security Features and Considerations

In addition to SecDL and the RHEL 7 STIG, Nutanix also provides risk management features that have become requirements for most organizations. Depending on your organization's security needs and compliance requirements, we suggest that you consider configuring the following features.

Table 10: Other Nutanix Security Features

Nutanix Feature	Description
Log shipping	The Nutanix CVM provides a simple method for preserving CVM log integrity across a cluster. The cluster-wide log shipping setting forwards all logs to a central log host. Local logs lack integrity because someone can escalate privilege and delete logs to cover their tracks.
Two-factor authentication	Useful for system administrators in environments with stringent security needs. When implemented, sign ins require both a client certificate and a username and password. Administrators can use local accounts or Active Directory for usernames and passwords. Nutanix also supports Common Access Cards (CACs).
Cluster lockdown	Allows administrators to restrict access to a Nutanix cluster in security-conscious environments, such as government facilities and healthcare provider systems. Cluster lockdown disables interactive shell sign ins automatically.

Nutanix Feature	Description
Password complexity support	Enables additional password rules to meet regulatory requirements.
Banner support	Adds warnings and custom prompts when users sign in to the CVM or Prism in order to meet federal and compliance regulations.

Storage

Table 11: Summary of Storage Design

Configuration Item	Parameter
Storage pool	Single storage pool
Controller Virtual Machine	12 vCPU, 64 GB of RAM
Nutanix Files container settings	Inline compression, replication factor 2, EC-X
General purpose VM container settings	Inline compression, replication factor 2

Distributed storage simplifies storage by abstracting the advanced storage functionality from the underlying hardware and creating a truly software-defined storage solution that is highly available and automated and meets or exceeds the storage requirements of today's enterprise workloads. Bringing data closer to compute minimizes expensive data movement, resulting in significantly better performance at scale. With intelligent software, you can scale out infrastructure at the granularity of a single x86 server, enabling fractional consumption and incremental growth, making the process of buying infrastructure less painful, and decreasing space and power requirements. By simplifying storage, hyperconvergence also helps break down organizational barriers and streamline operations.

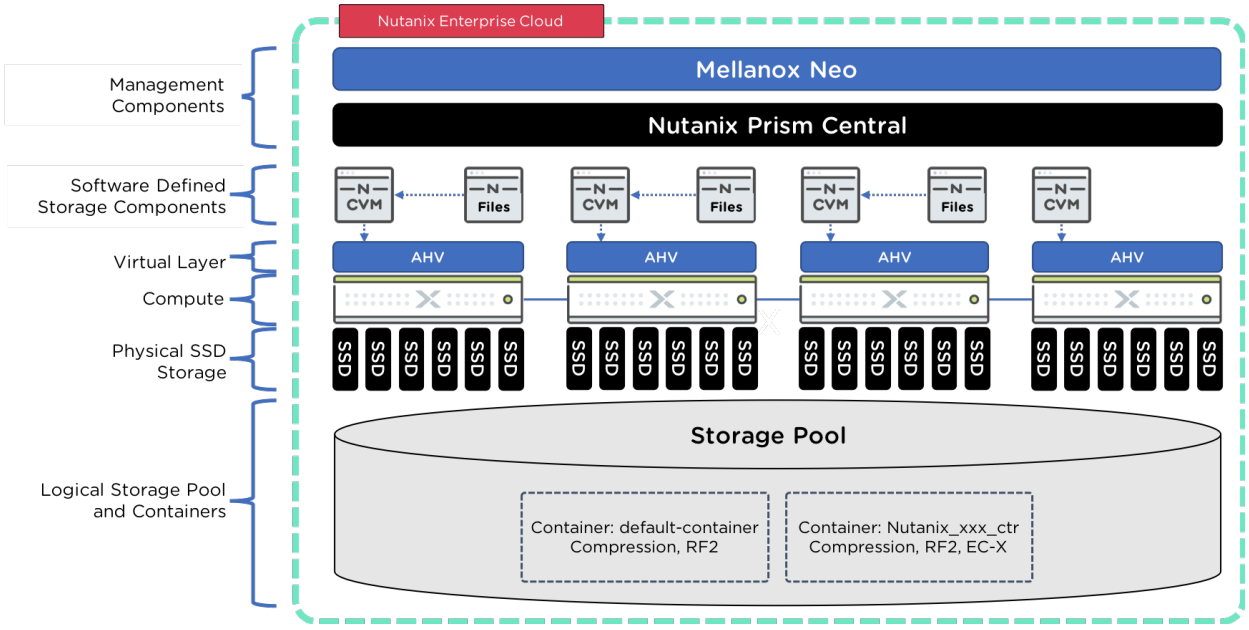

Figure 16: Relationship Between Hosts and Physical and Logical Storage Components

Note: For this reference architecture, we used Nutanix 3060-G6 appliances provisioned with six SSD devices per node for local attached storage as the basis of the AOS storage fabric. This setup provides a balance of compute, storage, and performance in a converged form factor for AI workloads; however, you can mix and match appliances to support higher storage densities and additional performance requirements.

Nutanix Compression

You can perform compression in-line or post-process and enable it at the container level. The choice to enable compression depends on workload; therefore, it's important to understand the characteristics of the application workload and whether compression is advantageous. For more information about the Nutanix compression process, see the Compression section of the Prism Web Console Guide.

Nutanix Erasure Coding

As noted previously, the Nutanix platform uses a replication factor for data protection and availability. This method provides the highest degree of availability, but at the cost of storage resources because it requires full copies. To provide a balance between availability while reducing the amount of storage required, distributed storage allows users to encode data with Nutanix Erasure Coding (EC-X). As with RAID levels, EC-X encodes a strip of data blocks on different nodes and calculates parity. In the event of a

host or disk failure, the parity can be decoded to calculate any missing data blocks. With distributed storage, the data block is an extent group and each data block must be on a different node and belong to a different vDisk. To learn more about EC-X, refer to the Nutanix Bible.

> **Note:** In this reference architecture, we used four NX 3060-G6 appliances, so the strip size was automatically set to 2:1 (two data blocks to one parity block). When you increase the number of nodes in a cluster, the strip size readjusts to a maximum of 4:1 for replication factor 2 or 4:2 for replication factor 3 workloads.

Nutanix Storage Tier Design

AOS storage contains two tiers of data storage: a performance or hot tier backed by flash devices and memory and cold tier storage backed by HDDs (or SSDs if you use NVMe devices in the hot tier). Data access patterns automatically designate the data tier and are processed in real time with no tuning requirements from the administrator.

> **Note:** For this reference architecture, we used NX3060-G6 appliances that had an all-flash configuration so tiering wasn't required; however, it's important to note that Nutanix supports combining different hardware platforms in the same clusters (for example, mixing hybrid appliances with all-flash appliances).

For more information on hardware models, visit the Nutanix website.

Network

The way you design and build your virtual and physical networks plays a crucial role in operating and scaling any AI solution. The physical network must provide predictability and low-latency switching as well as maximum throughput and linear scalability. Maximum throughput and scalability are especially crucial considering the recent introduction of NVMe-based SSDs and Intel's Optane Technology because the network must supply sufficient bandwidth for these devices to perform optimally without impacting existing applications or services. Because our solution features hyperconverged technology, there is also a virtual element associated with the technology and how traffic moves between the virtual layer and the physical.

For detailed information on how to configure and support a leaf-spine network (the network we used for this architecture), refer to the Mellanox Networking with Nutanix tech note.

Table 12: Summary of Nutanix Network Design

Configuration Item	Parameter
Storage traffic segmentation	VLAN based
VM traffic segmentation	VLAN based
Number of bridges	1
Number of bonds	1
NIC teaming adapters	Eth2, eth3
Local balancing algorithm	Balance-TCP
LACP failback to active-backup	Set: LACP-Failback-AB=True
OVS LACP timer configuration	Set: LACP-Timer=fast (3 seconds)
Provisioned VLANs	100, 101, 102

Remote Direct Memory Access over Converged Ethernet (RoCE)

Distributed machine learning frameworks, such as Uber's Horovod, use multiple GPUs across multiple DGX systems when distributing their machine learning tasks, which means they require a technology medium to provide high bandwidth coupled with low-latency connectivity between DGX systems. RoCE fulfills this requirement by introducing a lossless network protocol over an Ethernet fabric, which allows communication and direct memory access from one host to another without involving the remote OS and CPU.

RDMA is a key capability natively used by the InfiniBand interconnect technology. It has different physical and link layers when used with RoCE. To implement the end-to-end RoCE in this Nutanix AI solution, enable RDMA offload on the Mellanox Connect-X NICs in each DGX-1 system and configure RoCE v2 (PFC + ECN) on the Mellanox SN2100 switches for best flow control. At this point, you can append service classes and their communication VLANs.

Network Scalability and Performance

The following figure is a high-level diagram of our AI deployment, which scales up as well as out across multiple racks. Each rack in the solution uses two Mellanox SN2100 switches as leaf switches and two Mellanox SN2700 switches as the spine. A Mellanox

AS4610 switch for out-of-band management connectivity is also situated on each rack to provide access to remote server consoles (IPMI) for Nutanix and DGX-1 nodes.

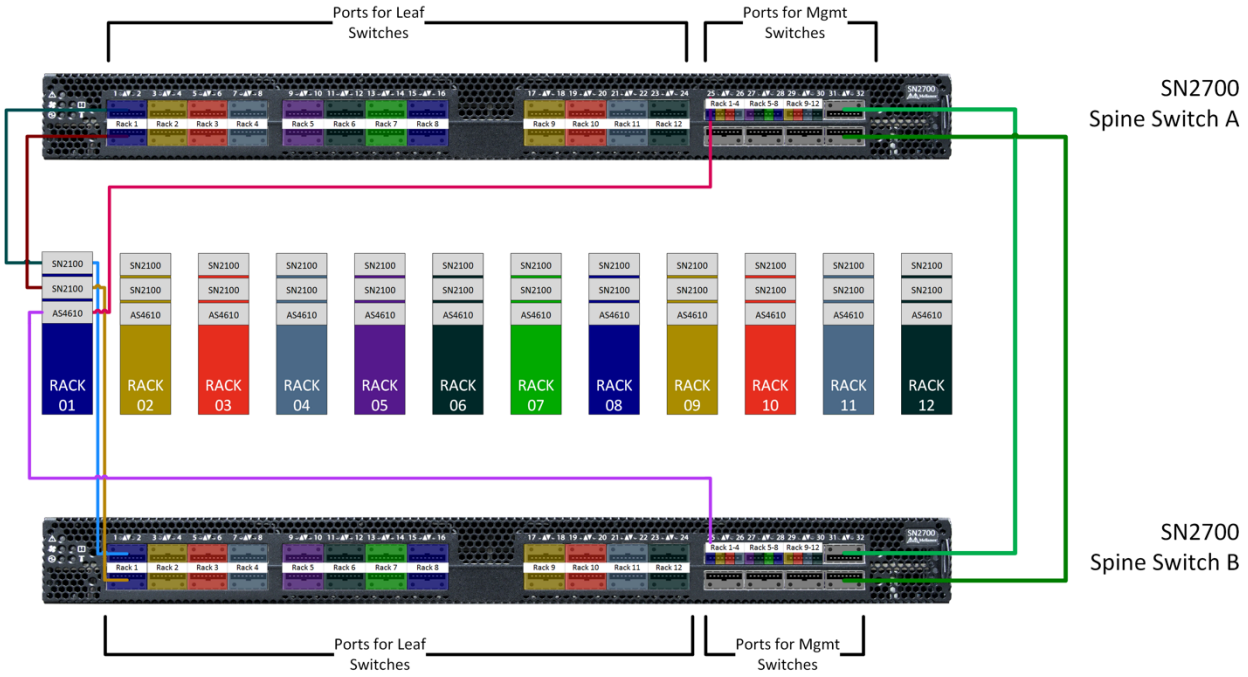

Figure 17: Leaf-Spine Scalability

This solution starts with a single rack (containing at least four hyperconverged appliances or nodes and up to four DGX-1s) and can scale to 12 racks. Each rack has full 100 GbE redundant connectivity, with 1 GbE connections for out-of-band management.

Nutanix and NVIDIA DGX-1 nodes connect to their respective leaf switches and to the AS4610, which provides 1 GbE out-of-band management connectivity.

For our Nutanix hosts, we used 25 GbE QSFP-to-4xSFP+ splitter cables to maximize our port density, as shown in the following figure.

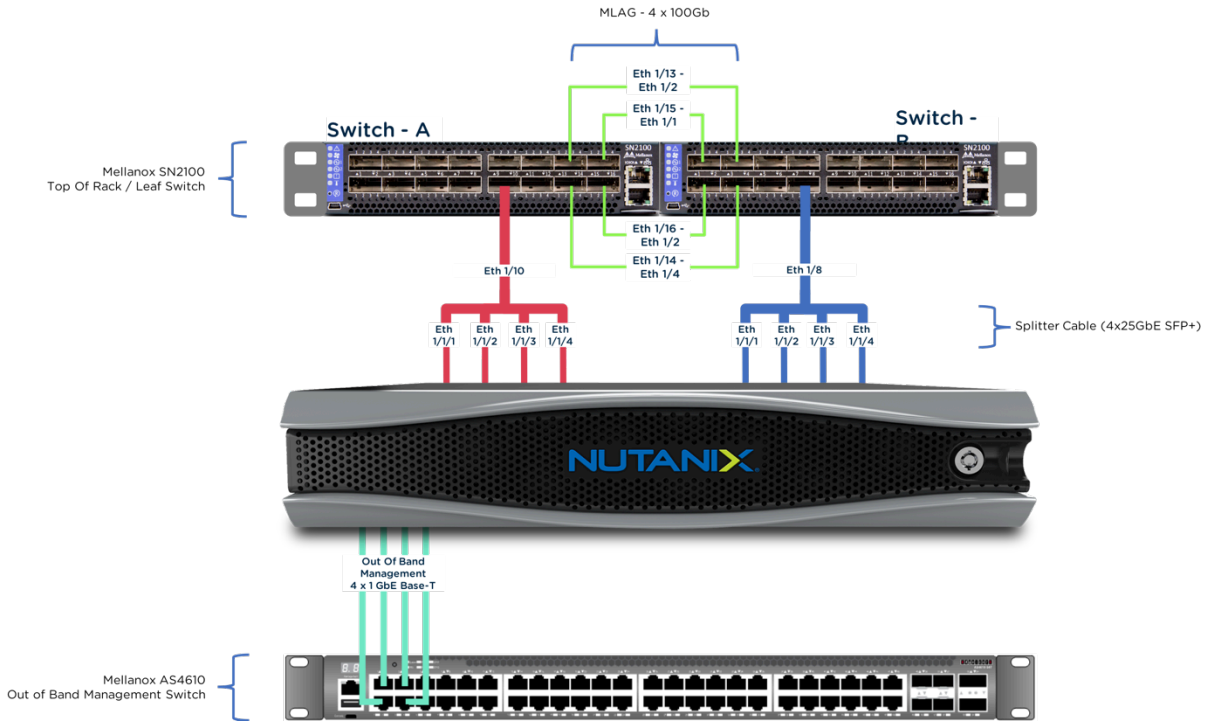

Figure 18: Nutanix Host Connectivity with Splitter Cables

The Mellanox SN2100 leaf switches connect to the SN2700 spine via QSFP cables. These connections provide 100 Gbps throughput per uplink back to the spine. Each SN2100 features two 100 GbE uplinks back to the spine; however, oversubscription ratios may vary depending on the number of hosts connected.

Using the Nutanix 3060 series in combination with the NVIDIA DGX-1s, our reference architecture can support:

- A total of eight Nutanix nodes or two hyperconverged blocks per rack (this deployment supports up to 96 servers across 12 racks).
- A total of four NVIDIA DGX-1 nodes per rack (this deployment supports up to 48 servers across 12 racks).

> **Tip:** While you can gain greater density through additional Nutanix or NVIDIA DGX-1 nodes per rack by using the Mellanox SN2700 (which offers additional network ports) instead of the SN2100, power and cooling factors are often a constraint in datacenters and should be taken into consideration when scaling up.

Leaf Switch Density Calculations

- Two Nutanix 3060 G6 blocks in each rack with four nodes per block (eight nodes per rack).
- Because each node contains 2x 25 GbE ports, we needed a total of 8x 25 GbE ports per four-node block, a total of 32 ports per rack. Because each SN2100 switch contains 16 ports, when we used two of them as leaf switches, we met our connectivity requirement by using a Mellanox QSFP-to-4xSFP+ cable to convert a single 100 GbE to 4x 25 GbE ports. We used four of these cables to fulfill the requirement for 32 ports. With QSFP-to-4xSFP+ cables from each leaf switch, 2 ports provide 16x 25 GbE uplinks per switch, or 32 uplinks total between both switches. This configuration leaves 14 ports on each leaf switch.
- Four additional ports from the leaf switches form an MLAG peering between the pair, while two more ports uplink to their spine switch.
  - Each leaf switch had eight spare ports available, which we used to connect our NVIDIA DGX-1 nodes. Each DGX-1 node requires 2x 100 GbE connections to each leaf switch.
- We needed 8x 1 GbE ports to satisfy the Nutanix out-of-band connectivity requirements and another 4x 1 GbE ports for the NVIDIA DGX-1 hosts. One AS4610 switch provides 48x 1 GbE connectivity per switch and 4x 10 GbE ports per switch. We used two of these 4x 10 GbE ports to establish uplinks to the respective spine switches.

Spine Density Calculations

- The spine, which consists of two SN2700 switches, contains 32x 100 GbE ports. We could convert a 100 GbE port into two 50 GbE ports or four (or fewer) 25 GbE ports using the appropriate QSFP+ Optic breakout cable.
- To satisfy the connectivity requirements for each rack, we needed 4x 100 GbE ports per rack (for the two leaf switches) and 2x 10 GbE ports (for the one out-of-band AS4610 switch), all of which were trunked to our Mellanox SN2700 spine switches.
- Subtracting the two ports required for the MLAG between our SN2700 switches (2x 100 GbE), each switch has 30x 100 GbE ports available for leaf connectivity.
- Therefore, to scale the solution to 12 racks with 2 ports per rack, we needed 24 switch ports per spine for leaf connectivity, which left 6 ports available per spine. Of these 6,

we needed to use 3 ports to provide out-of-band switch connectivity using the QSFP-to-4xSFP+ breakout cables (3 x 4 ports = 12 ports), so we had 3 ports remaining per spine.

Logical Connectivity

In the following figure, we illustrate the logical network topology in the Nutanix and NVIDIA DGX-1 nodes and the Mellanox network switches that connect the environment.

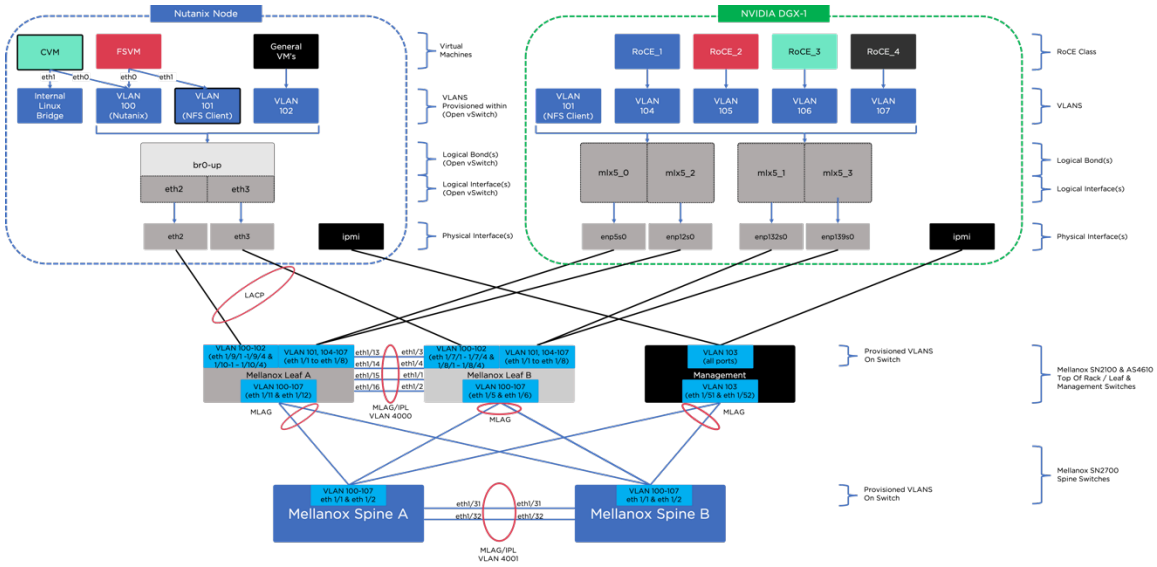

Figure 19: Logical Network Connectivity for Nutanix, NVIDIA, and Mellanox

NVIDIA DGX-1 Connectivity

Each NVIDIA DGX-1 system comes preconfigured from the factory with a base OS that consists of an Ubuntu OS, Docker, Docker Engine Utility for NVIDIA GPUs, and NVIDIA drivers. The system is designed to run NVIDIA-optimized deep-learning framework applications packaged in Docker containers.

Although each DGX-1 is preconfigured, an administrator must still run through an initial setup process, configure the appropriate host networking, and use Docker-based containers with NGC. For information on the initial setup process and interacting with Docker, see the DGX-1 User Guide.

Note: The base OS version we tested used Ubuntu 16.04.5 LTS (GNU/Linux 4.4.0-131-generic x86_64) and Docker Client/Server version 8.03.1-ce. This reference architecture also needed to switch the NVIDIA CX4 NICs from InfiniBand mode to Ethernet mode.

For network connectivity, each DGX-1 has four Mellanox ConnectX-4 single-port NICs responsible for NFS and RoCE connectivity. You achieve RoCE connectivity by provisioning four RoCE VLAN interfaces on the leaf switches (A and B), which are trunked to all four Mellanox CX4 adapters. The system assigns the NICs an IP address from each of the four RoCE VLANs.

On the DGX-1 system, the Mellanox drivers apply a network class-of-service (CoS) value of 4 to each of the RoCE VLANs and collectively work with the Mellanox switches to guarantee lossless service to the RoCE class. We outline the steps required to configure RoCE over a lossless network in the appendix.

RoCE doesn't support aggregating multiple links into a single logical connection; the NVIDIA Collective Communications Library (NCCL) software alleviates this constraint by using multiple links for bandwidth aggregation and fault tolerance. To provide optimal performance for the RoCE connections, all NFS traffic is assigned to the default best-effort quality-of-service (QoS) class.

All physical interfaces and the bond interfaces are configured with a maximum transmission unit (MTU) of 9,000 bytes.

Nutanix Hosts

AHV uses Open vSwitch (OVS) to connect the CVM, Nutanix Files, the hypervisor, and guest VMs to one another and to the physical network. The OVS service, which starts automatically, runs on each AHV node. OVS is an open-source software switch implemented in the Linux kernel and designed to work in a multiserver virtualization environment. By default, OVS behaves like a layer 2 learning switch that maintains a MAC address table. The hypervisor host and VMs connect to virtual ports on the switch.

OVS supports many popular switch features, including VLAN tagging, Link Aggregation Control Protocol (LACP), port mirroring, and QoS, to name a few. Each AHV server maintains an OVS instance, and all OVS instances combine to form a single logical switch. Constructs called bridges manage the switch instances residing on the AHV hosts. For more detailed information on Nutanix hosts, see the AHV Networking best practices guide.

Mellanox Switches

MLAGs are formed in the Mellanox switch pairs (A and B) for leaf and spine access to support the leaf-spine network topology and allow the Nutanix hosts to use dynamic

LACP for the purposes of load balancing VM ingress and egress traffic. MLAGs don't disable links to prevent network loops like a spanning tree policy (STP) does. Although STP is still enabled to prevent loops during switch startup, once the switch is initialized and in a forwarding state, the MLAG disables STP and ensures that all the links are available to pass traffic and benefit from the aggregated bandwidth.

Switch pairs are configured with an interpeer link (IPL)—a link between switches that maintains state information—over an MLAG (port channel) with ID 1.

For high availability and throughput, we provisioned 4x 100 Gbps links between our leaf switches (A and B) and 2x 100 Gbps links between our spine switches. All respective VLANs are open on these ports; however, our architecture also used VLAN 4000 and 4001 to configure the IP addresses of the virtual interfaces between switches and maintain switch state information.

To support our RoCE configuration, the NVIDIA DGX-1 hosts assign a CoS value of 3 to ingress and egress traffic on each of the four RoCE VLANs. We configured the Mellanox switches to apply a QoS policy to traffic tagged with a CoS value of 3, using a link-layer protocol (flow control and explicit congestion notification (ECN)) to ensure that the traffic doesn't drop. Enabling link-level flow control or priority-based flow control (PFC) in the network creates a lossless network, ensuring that no packets are dropped. PFC pauses traffic by priority, while link-level flow control pauses traffic by port. When there is congestion in the network element, the link pauses the traffic until congestion is released.

Table 13: Summary of Mellanox Network Design

Configuration Item	Parameter
Storage traffic segmentation	VLAN based
VM traffic segmentation	VLAN based
Number of bridges	1
Number of bonds	1

Physical Connectivity

The following figure shows the physical topology and connections between each host and the physical switch port as well as between spine, leaf, and management

switches. In this topology, you can add an additional Nutanix block using the splitter cable attached to eth1/9 on leaf A and eth1/7 on leaf B. Splitter cables are useful because they minimize cabling and remove the requirement for additional physical ports on the network switches. The Nutanix hosts take a 100 Gbps port on the Mellanox switches and split it into four 25 Gbps logical ports.

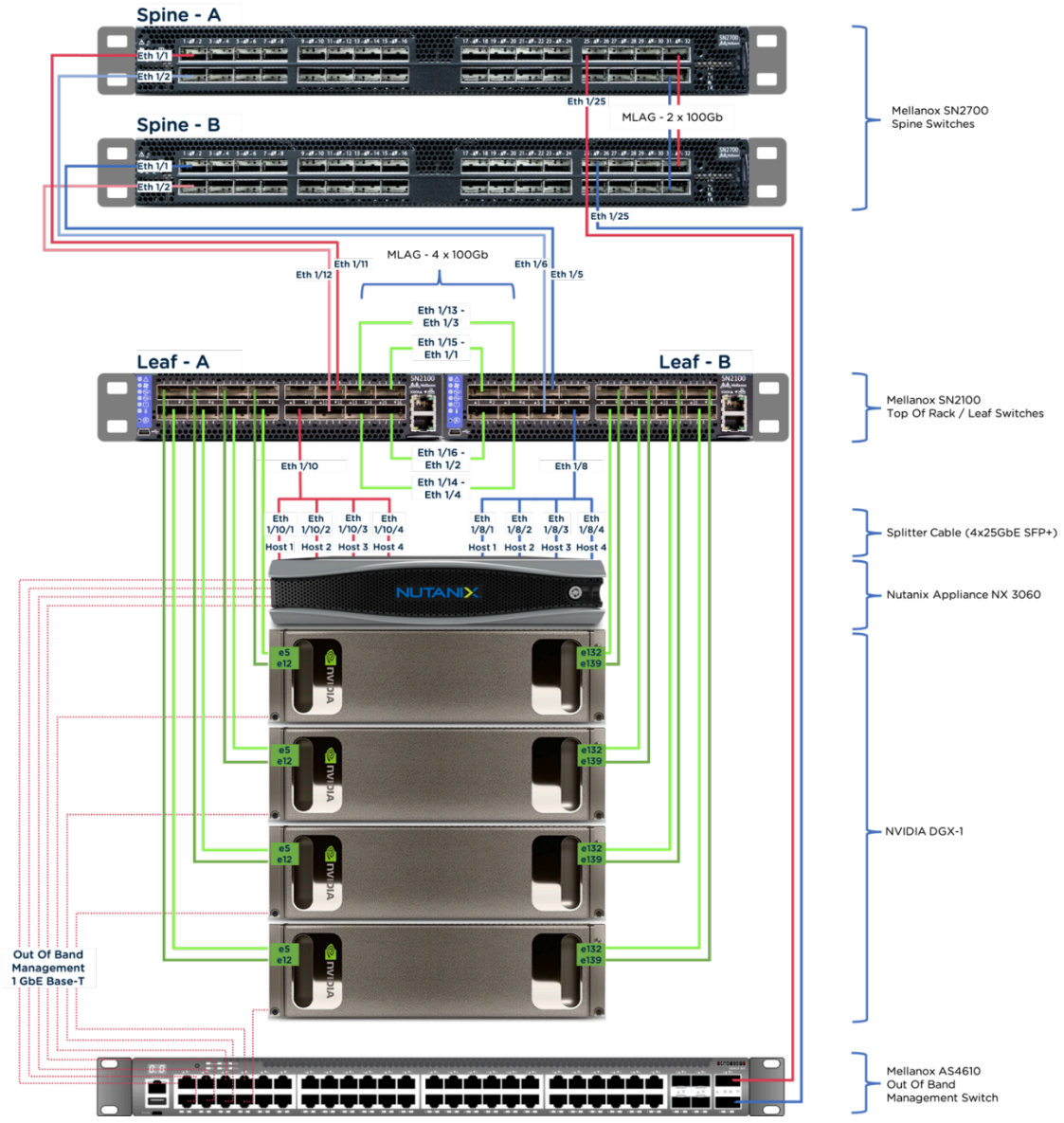

Figure 20: Physical Connectivity Between Infrastructure Components

To support additional hosts (Nutanix or DGX-1), you can substitute two SN2700 leaf switches for the two SN2100 leaf switches, which offers double the switch port density (32x 100 GbE ports); however, you must first account for power and cooling requirements. See the Environmental Power and Cooling Study section for our examination of these requirements.

5. Environmental Power and Cooling Study

When you scale an AI solution vertically or deploy additional NVIDIA DGXs, Nutanix nodes, or Mellanox switches, you must factor in the power and cooling requirements for the solution, as these are constraints in most datacenters.

Our test datacenter was equipped with two power distribution units (PDUs) delivering 60 amps of power each and 17.3 kW (59,030 BTU/hour) of cooling capacity per rack. Because power was the limiting factor, it was important that we didn't exceed 80 percent of the total delivery, as we needed to account for headroom and additional bursts. In our example, this limitation meant we couldn't exceed 48 amps per PDU or 96 amps per rack unit.

Table 14: Solution Power Requirements Study

Quantity	Component	Watts (Watts x Qty)	Operating Voltage	Amps
4	NVIDIA DGX-1	3,200 (12,800)	200–240 V	15 amps (60 total)
1	Nutanix 3060-G6	2,800 (2,800)	100–240 V	13 amps
2	Mellanox SN2100	94.3 (188.6)	100–240 V	0.42 amps (0.84 total)
2	Mellanox SN2700	150 (300)	100–240 V	0.68 amps (1.36 total)
1	Mellanox AS4610	99.1	100–240 V	0.45 amps
Total		16,187.7		75.65 amps

The previous table demonstrates that although we met our power requirements, scaling our rack vertically could be a constraint.

6. Solution Verification and Testing

Our aim was to derive performance-related data and test the functionality between various components in our solution architecture. To validate our solution, we needed a sizable and comprehensive data set we could use for both inferencing and training techniques. The ImageNet database is a large-scale prebuilt model that features a comprehensive collection of images organized according to the WordNet hierarchy for nouns, which made it an excellent database to use for testing our solution.

After we downloaded the ImageNet database and stored it on the Nutanix Files NFS export, we used Google's TensorFlow machine learning framework coupled with various convolutional neural networks (CNNs) on the two DGX-1 systems in our test lab to compare the performance of inferencing versus training workloads.

CNNs are multilayer neural networks, designed solely for the purpose of recognizing visual patterns directly from pixel images. The CNNs we used in our validation included the following:

- AlexNet
- Googlenet
- InceptionV3
- Resnet152
- Resnet50

Note: We obtained TensorFlow version 1.10 from the NVIDIA GPU Cloud as part of the container nvcr.io/nvidia/tensorflow:18.10-py3.

Testing Methodology

- We conducted tests with varying batch sizes (number of training examples used per iteration) to demonstrate the connection between processing time and accuracy and impact on performance.
  - We specifically chose our batches, which ranged from 128 to 1,024 training examples, based on the CNN, in line with NVIDIA's testing methodology.
- We tested a single DGX-1 system initially, then scaled the tests out to two DGX-1 systems, which was when we saw the performance benefits of a distributed system and GPU.
  - We used Horovod, Uber's open source distributed deep-learning framework, as part of our scalability and distributed system tests, which also enabled us to benchmark the performance of our RoCE-configured environment.
  - Horovod allows you to schedule machine learning tasks by using multiple GPUs across multiple DGX-1 systems.
  - We used the Synthetic Gradients for PyTorch and TensorFlow MNIST data sets to build a CNN and perform synthetic testing with Horovod, while also testing its accuracy.
- During each test, sequential reads were operating at 100 percent, which means the DGX-1 system read our entire model then cached the model to its local NVMe drives and memory. For this reason, we cleared the SSD cache and memory cache on the DGX-1 before we performed each subsequent test in order to test across a common baseline.
- We used the following script to run our tests and specified additional parameters to enable the best possible performance. The **-np 8** specification allows us to run the neural networks with parallel processes while using all 8 GPUs. You can use different layers across the different models and change them using **--layers=**. For example, to run resnet50, set **layers=50**, and for resnet152, set **layers=152**. Use **--precision=** to change the precision. For mixed precision, we specify **--precision=fp16**.

```
mpiexec --allow-run-as-root --bind-to socket -np 8 python resnet.py --layers=50
 --data_dir=/dgx_export1/image_net_data --precision=fp16
```

Following are the CNN-specific scripts and parameters we used to run our tests.

- Alexnet:

```
mpiexec --allow-run-as-root --bind-to socket -np 8 python alexnet.py --data_dir=/
dgx_export1/image_net_data --precision=fp16
```

- Googlenet:

```
mpiexec --allow-run-as-root --bind-to socket -np 8 python googlenet.py --
data_dir=/dgx_export1/image_net_data --precision=fp16
```

- Inceptionv3:

```
mpiexec --allow-run-as-root --bind-to socket -np 8 python inception_v3.py  --
data_dir=/dgx_export1/image_net_data --precision=fp16
```

- Vgg16:

```
mpiexec --allow-run-as-root --bind-to socket -np 8 python vgg.py --layers=16 --
data_dir=/dgx_export1/image_net_data --precision=fp16
```

- Resnet50:

```
mpiexec --allow-run-as-root --bind-to socket -np 8 python resnet.py --layers=50
 --data_dir=/dgx_export1/image_net_data --precision=fp16
```

Results and Observations

The following table records our test results for deep-learning training across the various CNNs, noting batch sizes and quantifying performance in number of images analyzed per second.

Table 15: CNN Performance in Images Per Second

CNN	Batch Size	Images per Second
Resnet50	256	6,300
Resnet152	256	2,645
AlexNet	1,024	11,929
Googlenet	128	11,211
InceptionV3	128	3,336
VGG16	512	3,331

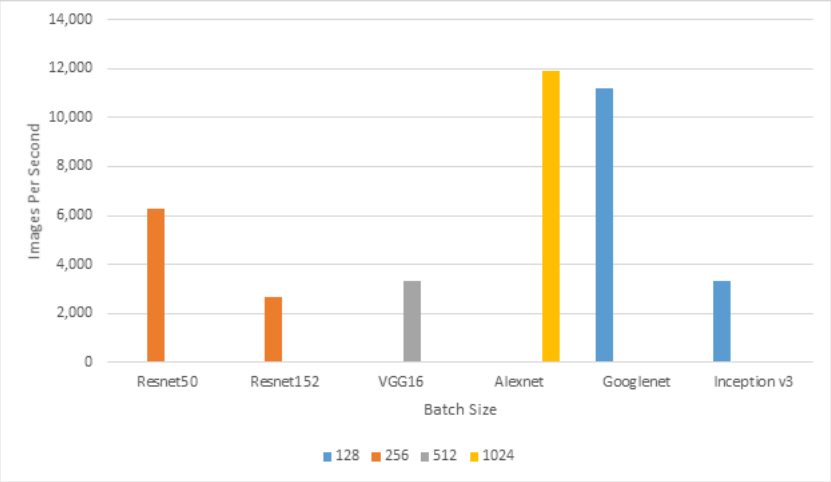

Figure 21: Graph of CNN Performance in Images per Second

Distributed Training with Horovod

When we ran a distributed training job with Horovod, we used the Message Passing Interface (MPI) for worker discovery and reduction coordination across the DGX nodes. MPI launched the process and scheduled the training job across the GPUs in all DGX nodes. We used NVIDIA Collective Communication Library (NCCL) extensively for communicating over multiple GPUs both within and across nodes. NCCL also supports a variety of interconnect technologies including PCIe, NVLINK, InfiniBand Verbs, and IP sockets.

When we tested Horovod with our synthetic data set, we ran each of the following three scripts nine times to obtain our performance data. We tested a single DGX-1 server, first with a single GPU, then with all eight GPUs. We then scaled the test across the two DGX-1 servers in our lab.

- Single DGX-1
  - In the ninth iteration, we obtained 279.0 image/second/GPU performance.
  - The average number of images per second per GPU across the nine test iterations was 278.4 plus or minus 1.5.
  - The average total number of images per second on eight GPUs was 2,226.9 plus or minus 12.3.
  - pytorch_synthetic_benchmark:

```
$ mpirun -np 8 \-H localhost:8 \-bind-to none -map-by slot \-x NCCL_DEBUG=INFO -x
 LD_LIBRARY_PATH -x PATH \-mca pml ob1 -mca btl ^openib \

python pytorch_synthetic_benchmark.py
```

- Two DGX-1s
  - In the ninth iteration, we obtained 530.4 image/second/GPU performance.
  - The average number of images per second per GPU across the nine test iterations was 529.8 plus or minus 0.9.
  - The average total number of images per second on 16 GPUs was 4,231.11 plus or minus 7.3.
  - pytorch_synthetic_benchmark:

```
$ mpirun -np 16 \-H 10.33.252.70:8,10.33.252.38:8 \-bind-to none -map-by slot \-x
 NCCL_DEBUG=INFO -x LD_LIBRARY_PATH -x PATH \-mca pml ob1 -mca btl ^openib \

python pytorch_synthetic_benchmark.py
```

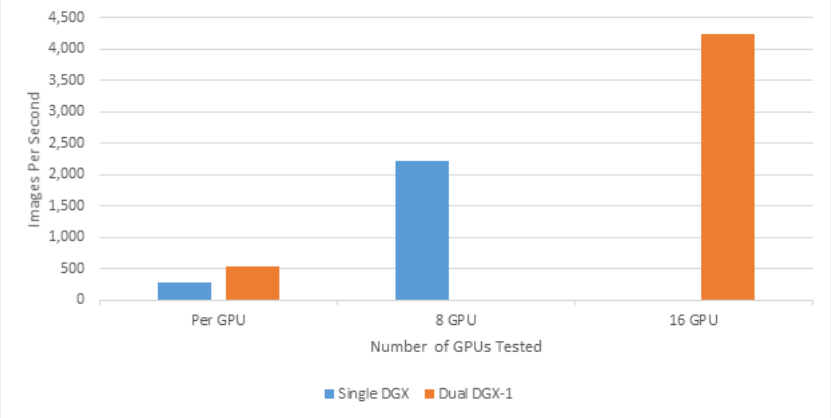

Figure 22: Distributed Training Results

As part of running our tests with Horovod, we also recorded the accuracy of the TensorFlow MNIST results, which was very high.

Table 16: TensorFlow MNIST Accuracy Results

Model	Number of DGX-1s	Result
TensorFlow MNIST	1	tensorflow:loss = 0.035151124, step = 5001
	2	tensorflow:loss = 0.0009622343, step = 5000

RoCE Performance Testing

Because it uses a low-latency, high-bandwidth Ethernet communication medium, RDMA is instrumental to achieving the highest number of images per second, which is an important factor when you're distributing jobs across multiple DGX-1 nodes like we were. We tested our RDMA write performance by running the following command on our DGX-1 server (10.33.252.38 represents the second DGX-1 node in our lab).

```
nutanix@nvidiadgx:~$ ib_write_bw -d mlx5_0 -x 3 10.33.252.38 --report_gbits --
run_infinitely
```


7. Conclusion

Architecting an AI infrastructure solution, deploying the necessary hardware and software, and identifying and eliminating bottlenecks is often an expensive, time-consuming, and error-prone process. To enable your team to focus on its AI goals rather than infrastructure complexities, Nutanix partnered with NVIDIA and Mellanox to design, test, and validate a reference architecture capable of taking on the world's toughest deep-learning problems.

The Nutanix for AI solution combines the proven Nutanix hyperconverged infrastructure technology with NVIDIA DGX-1 computing and low-latency 100 GbE switching from Mellanox. The result is a balanced best-of-class solution designed to eliminate bottlenecks, support large and diverse data sets, efficiently deliver data to NVIDIA GPUs, and scale out as your needs grow.

The key advantages of the Nutanix solution include:

- Ease of deployment and management

  Comprehensive tools streamline deployment and ensure that firmware, patching, and software upgrades are seamless. Easy-to-manage data protection and disaster recovery protect your AI operations.

- Superior performance and scaling

  Nutanix enables you to start small and scale out as your needs grow, without concerns about bottlenecks or having to rearchitect.

- Built-in data security

  Nutanix software builds in security features including two-factor authentication and data-at-rest encryption in a hardened security framework that has been certified to ensure compliance with the strictest standards.

For feedback or questions, contact us using the Nutanix NEXT Community forums.

Appendix

Configure Link Aggregation with LACP on Nutanix

Configure link aggregation with LACP and balance-TCP on all Nutanix CVMs in the cluster using the following commands. For more information, see the load balancing section in the Nutanix best practices guide for AHV networking.

> **Tip:** Configure upstream switches for link aggregation with LACP before you configure the AHV host from the CVM. Upstream LACP settings such as timers should match the AHV host settings for configuration consistency.

- If upstream LACP negotiation fails, the default AHV host configuration disables the bond, blocking all traffic. The following command allows fallback to active-backup bond mode in the AHV host in the event of LACP negotiation failure:

```
nutanix@CVM$ ssh root@192.168.5.1 "ovs-vsctl set port br0-up other_config:lacp-fallback-ab=true"
```

- In the AHV host and on most switches, the default OVS LACP timer configuration is slow (30 seconds). This value—which is independent of the switch timer setting—determines how frequently the AHV host requests LACPDUs from the connected physical switch. The fast setting (1 second) requests LACPDUs from the connected physical switch every second, detecting interface failures more quickly. Failure to receive three LACPDUs—in other words, after 3 seconds with the fast setting—shuts down the link with the bond. Nutanix recommends setting lacp-time to **fast** to decrease link failure detection time from 90 seconds to 3 seconds. Only use the slow lacp-time setting if the physical switch requires it for interoperability.

```
nutanix@CVM$ ssh root@192.168.5.1 "ovs-vsctl set port br0-up other_config:lacp-time=fast"
```

- Enable LACP negotiation and set the hash algorithm to **balance- tcp**:

```
nutanix@CVM$ ssh root@192.168.5.1 "ovs-vsctl set port br0-up lacp=active"

nutanix@CVM$ ssh root@192.168.5.1 "ovs-vsctl set port br0-up bond_mode=balance-tcp"
```

- Confirm the LACP negotiation with the upstream switch or switches using **ovs-appctl** and look for the word “negotiated” in the status lines.


```
nutanix@CVM$ ssh root@192.168.5.1 "ovs-appctl bond/show br0-up"
nutanix@CVM$ ssh root@192.168.5.1 "ovs-appctl lacp/show br0-up"
```

Configure RoCE over a Lossless Network

- Run the following command to check the flow control settings for Mellanox network adapters:

  ```
  # ethtool -a <mlnx interface name>
  ```

- If RX and TX settings are turned on, disable them as shown in the following figure.

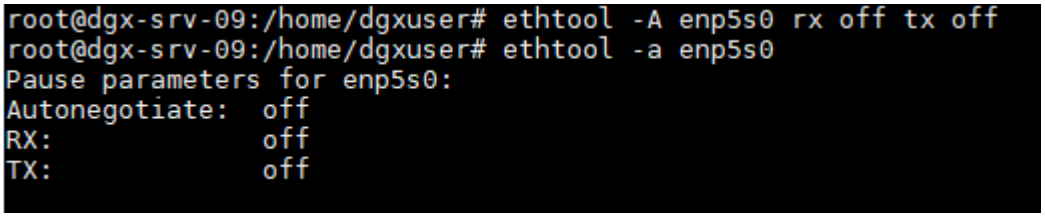

Figure 23: RX and TX Disabled

- Set up the QoS parameters:

  ```
  mlx_qos -I enp5s0 - - trust dscp
  ```



```
root@dgx-srv-09:/home/dgxuser# mlnx_qos -i enp5s0 --trust dscp
DCBX mode: OS controlled
Priority trust state: dscp
dscp2prio mapping:
        prio:0 dscp:07,06,05,04,03,02,01,00,
        prio:1 dscp:15,14,13,12,11,10,09,08,
        prio:2 dscp:23,22,21,20,19,18,17,16,
        prio:3 dscp:31,30,29,28,27,26,25,24,
        prio:4 dscp:39,38,37,36,35,34,33,32,
        prio:5 dscp:47,46,45,44,43,42,41,40,
        prio:6 dscp:55,54,53,52,51,50,49,48,
        prio:7 dscp:63,62,61,60,59,58,57,56,
Cable len: 7
PFC configuration:
        priority    0   1   2   3   4   5   6   7
        enabled     0   0   0   0   0   0   0   0
tc: 0 ratelimit: unlimited, tsa: vendor
         priority:  1
tc: 1 ratelimit: unlimited, tsa: vendor
         priority:  0
tc: 2 ratelimit: unlimited, tsa: vendor
         priority:  2
tc: 3 ratelimit: unlimited, tsa: vendor
         priority:  3
tc: 4 ratelimit: unlimited, tsa: vendor
         priority:  4
tc: 5 ratelimit: unlimited, tsa: vendor
         priority:  5
tc: 6 ratelimit: unlimited, tsa: vendor
         priority:  6
tc: 7 ratelimit: unlimited, tsa: vendor
         priority:  7
```

Figure 24: QoS Paramaters

- Enable PFC for RoCE:

```
mlnx_qos -i enp5s0 –pfc 0,0,0,1,0,0,0,0
```



```
root@dgx-srv-09:/home/dgxuser# mlnx_qos -i enp5s0 --pfc 0,0,0,1,0,0,0,0
DCBX mode: OS controlled
Priority trust state: dscp
dscp2prio mapping:
        prio:0 dscp:07,06,05,04,03,02,01,00,
        prio:1 dscp:15,14,13,12,11,10,09,08,
        prio:2 dscp:23,22,21,20,19,18,17,16,
        prio:3 dscp:31,30,29,28,27,26,25,24,
        prio:4 dscp:39,38,37,36,35,34,33,32,
        prio:5 dscp:47,46,45,44,43,42,41,40,
        prio:6 dscp:55,54,53,52,51,50,49,48,
        prio:7 dscp:63,62,61,60,59,58,57,56,
Cable len: 7
PFC configuration:
        priority    0   1   2   3   4   5   6   7
        enabled     0   0   0   1   0   0   0   0
tc: 0 ratelimit: unlimited, tsa: vendor
         priority:  1
tc: 1 ratelimit: unlimited, tsa: vendor
         priority:  0
tc: 2 ratelimit: unlimited, tsa: vendor
         priority:  2
tc: 3 ratelimit: unlimited, tsa: vendor
         priority:  3
tc: 4 ratelimit: unlimited, tsa: vendor
         priority:  4
tc: 5 ratelimit: unlimited, tsa: vendor
         priority:  5
tc: 6 ratelimit: unlimited, tsa: vendor
         priority:  6
tc: 7 ratelimit: unlimited, tsa: vendor
         priority:  7
```

Figure 25: Enable PFC

Mellanox NEO MLAG Setup

Using NEO automation and visualization simplifies network configuration and eliminates manual errors. To install NEO, download a VM for your hypervisor, import it, and start discovering the Mellanox infrastructure. For additional information on how to set up and install NEO, read Mellanox Doc-2784.

MLAG is one of the services you can provision with NEO, using the Add button to create a new service for MLAG. In a new window, select your MLAG switch pair from a dropdown menu of the managed devices, choose the network OS (Mellanox Onyx or

Cumulus Linux), and enter the name and description for your MLAG pair. You also have the option to choose interfaces.

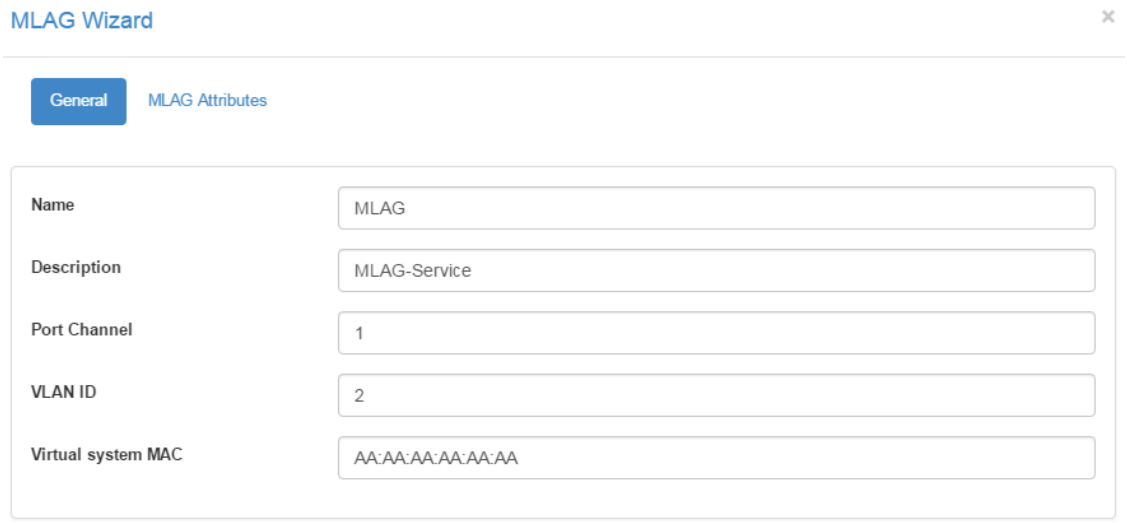

Figure 26: Initial MLAG Setup Wizard

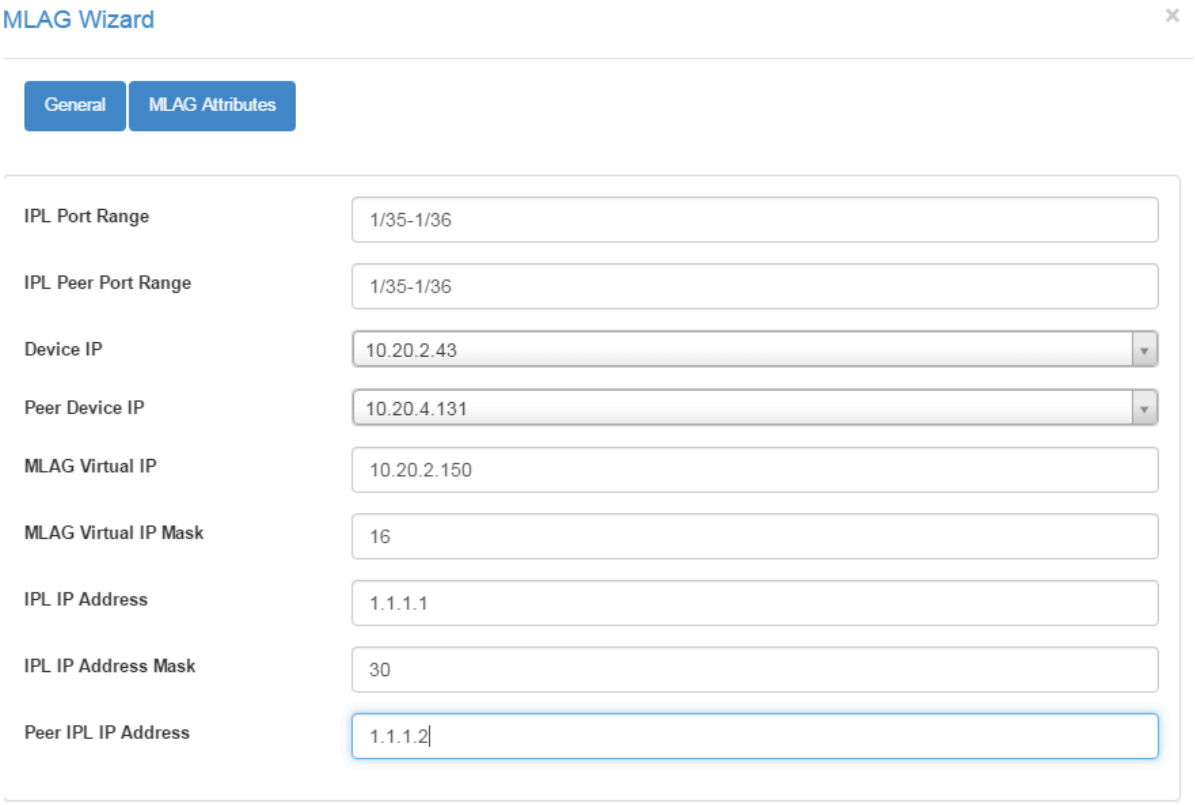

Figure 27: Set Up IPL and MLAG IP Address Details

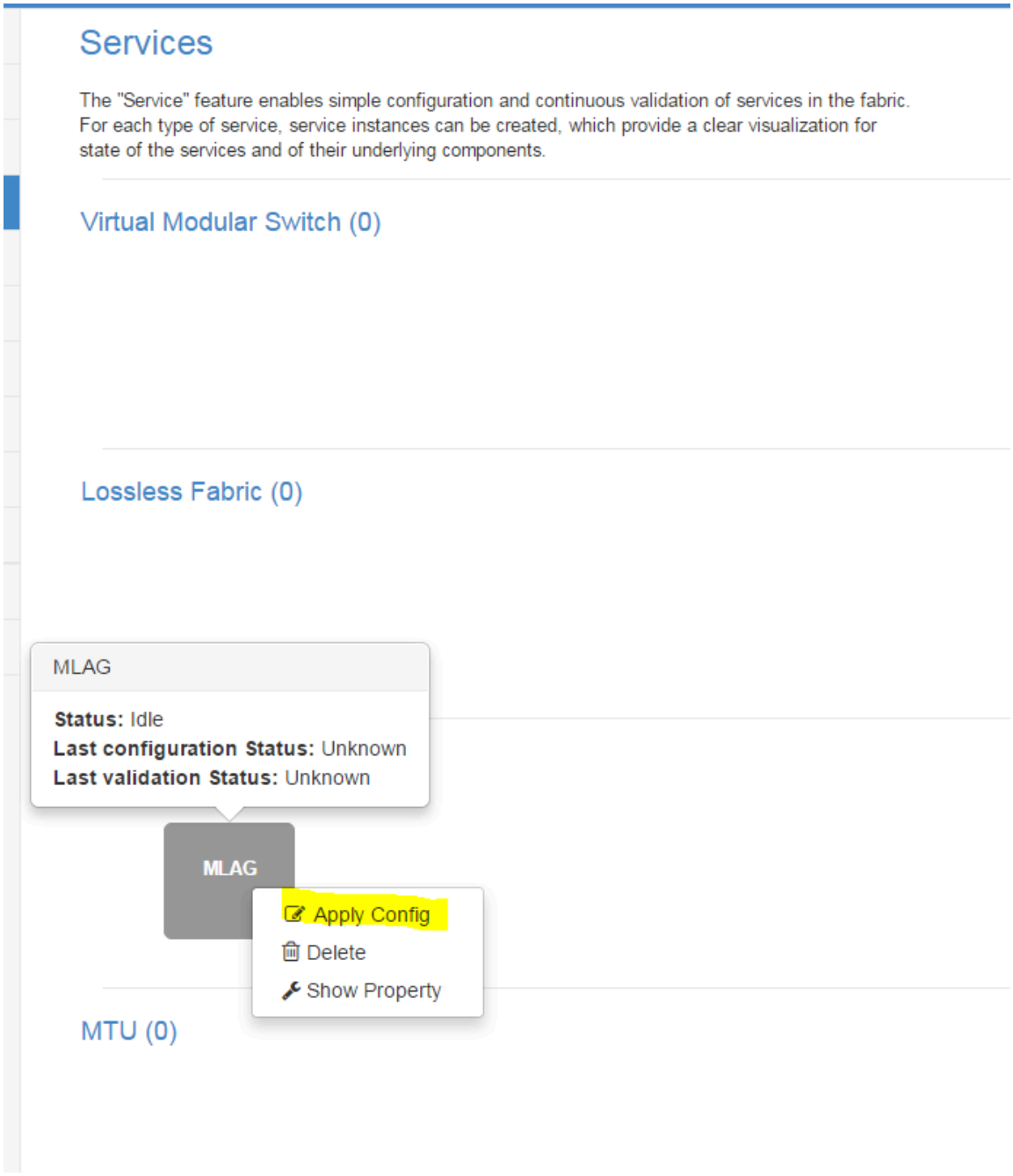

Figure 28: Applying Configuration

Mellanox Bill of Materials

We used the following Mellanox parts in this reference architecture.

Table 17: Mellanox Parts

Item	Mellanox Part No.	Description
Leaf switch (SN2100)	MSN2100-CB2F	Spectrum-based 100 GbE, 1RU open Ethernet switch with Mellanox Onyx, 16 QSFP28 ports, 2 power supplies (AC), short depth, Rangeley CPU, P2C airflow, rail kit, RoHS6
Spine switch (SN2700)	MSN2700-BS2F	Spectrum-based 100 GbE, 1RU open Ethernet switch with Mellanox Onyx, 32 QSFP28 ports, 2 power supplies (AC), x86 CPU, standard depth, P2C airflow, rail kit, RoHS6
Rail kit	MTEF-KIT-D	Rack installation kit for SN2100 and SN2010 series 1RU switches; allows one or two switches to be installed side by side in standard-depth racks
Cables for ISLs	MCP1600-C00AE30N	Mellanox passive copper cable, 100 GbE, 100 Gbps, QSFP28, 0.5 m, black, 30AWG, CA-N
Cables for DGXs	MCP1600-C003E30L	Mellanox passive copper cable, 100 GbE, 100 Gbps, QSFP28, 3 m, black, 30AWG, CA-L
Cables for Nutanix nodes	MCP7F00-A003R30L	Mellanox passive copper hybrid cable, 100 GbE to 4x 25 GbE, QSFP28-to-4xSFP28, 3 m, colored, 30AWG, CA-

References

1. AHV Networking best practices guide
2. Compression section of the Prism Web Console Guide
3. Data Protection and Disaster Recovery best practices guide
4. DGX OS Server version 3.1.7
5. ImageNet

6. Information Security tech note
7. Mellanox NEO
8. Mellanox Networking with Nutanix tech note
9. Mellanox Onyx
10. Mellanox SN2100
11. Mellanox SN2700
12. NEO Plugin for Nutanix Appliance: Installation and Usage
13. NEO Software Download
14. Nutanix Data Protection
15. Nutanix Bible
16. Nutanix Developer
17. Nutanix Files
18. Nutanix Files tech note
19. Nutanix Hardware Platforms
20. Nutanix Prism
21. Nutanix Prism Operation Tiers tech note
22. Nutanix Prism tech note
23. Self-Service Restore
24. Nutanix Volumes best practices guide
25. NVIDIA DGX-1
26. NVIDIA DGX-1 With Tesla V100 System Architecture white paper
27. NVIDIA GPU Cloud
28. Open MPI: Open Source High Performance Computing
29. Overview of NCCL
30. Switch Infiniband and Ethernet in DGX-1
31. TensorFlow
32. Uber's Horovod
33. WordNet

About the Authors

Richard Arsenian is a Senior Staff Solution Architect on the Solutions and Performance R&D Engineering team at Nutanix.

Boris Kovalev is a Solution Architect at Mellanox Technologies. Follow Boris on Twitter @wandbor.

About Nutanix

Nutanix makes infrastructure invisible, elevating IT to focus on the applications and services that power their business. The Nutanix cloud platform software leverages web-scale engineering and consumer-grade design to natively converge compute, virtualization, and storage into a resilient, software-defined solution with rich machine intelligence. The result is predictable performance, cloud-like infrastructure consumption, robust security, and seamless application mobility for a broad range of enterprise applications. Learn more at www.nutanix.com or follow us on Twitter @nutanix.

List of Figures

List of Tables