Abstract

This document describes how NetApp® HCI can be designed to host AI inferencing workloads at edge data center locations. The design, based on NVIDIA T4 GPU powered NetApp HCI compute nodes, is a Kubernetes infrastructure built using NVIDIA DeepOps and TensorRT Inference Server. The design also establishes the data pipeline between the core and edge data centers and illustrates its implementation to complete the data lifecycle path.
TABLE OF CONTENTS

1 Executive Summary ........................................................................................................... 4

2 Program Summary .............................................................................................................. 4
   2.1 NetApp Verified Architecture .................................................................................... 4
   2.2 NetApp HCI AI Solution ............................................................................................. 4
   2.3 The Value Proposition and Differentiation for NetApp HCI for Edge Inferencing ......... 5

3 The Data Pipeline for Inferencing ...................................................................................... 5

4 Solution Overview ............................................................................................................. 6
   4.1 Solution Technology .................................................................................................... 7
   4.2 NetApp HCI Compute Nodes ....................................................................................... 9
   4.3 NVIDIA T4 GPUs ........................................................................................................ 9
   4.4 Element Software ........................................................................................................ 10
   4.5 ONTAP Select (Optional) ............................................................................................ 11
   4.6 NetApp Trident (Optional) .......................................................................................... 12
   4.7 Containers and Kubernetes ......................................................................................... 13
   4.8 NVIDIA DeepOps ......................................................................................................... 15
   4.9 NVIDIA TensorRT Inference Server ........................................................................... 15

5 Technology Requirements ................................................................................................ 16
   5.1 Hardware Requirements .............................................................................................. 16
   5.2 Software Requirements ............................................................................................... 17

6 Conclusion ......................................................................................................................... 17

Where to Find Additional Information .................................................................................. 18

Version History .................................................................................................................... 18

LIST OF TABLES
Table 1) Edge verticals and applications .............................................................................. 6
Table 2) Hardware requirements ......................................................................................... 16
Table 3) Software requirements ......................................................................................... 17

LIST OF FIGURES
Figure 1) Data pipeline ........................................................................................................ 6
Figure 2) Solution architecture ............................................................................................ 8
Figure 3) Virtual and physical components ......................................................................... 9
Figure 4) SnapMirror with Element software .................................................................... 11
Figure 5) ONTAP Select presence. ................................................................. 11
Figure 6) FlexCache operation................................................................. 12
Figure 7) Trident workflow. .................................................................. 13
Figure 8) Comparison of VMs and containers ....................................... 14
1 Executive Summary

NetApp® and NVIDIA have partnered to create the NetApp HCI AI inferencing solution specialized for edge datacenters. NetApp HCI exhibits all the features required for an edge data center: it has a low datacenter footprint; is easily deployable with the power of automation; provides cloud connectivity to build a hybrid cloud infrastructure; enables seamless connectivity to a core data center; comes with well-defined data flow channels in and out of the edge to the core and cloud; and provides all of this on a secure and reliable platform.

Modern day applications that are driven by artificial intelligence (AI) and machine learning (ML) have pushed the limits of the internet. End users and devices demand access to applications, data, and services at any place, any time, with minimal latency. In order to meet these demands, data centers are moving closer to their users to boost performance, reduce back and forth transfers of data, and provide a cost-effective way to meet user requirements.

For AI driven applications, edge locations act as a major source of data. The available data can be used for training when collected from multiple edge locations over a period to form a training dataset. The trained model can then be deployed back to the edge locations where the data was collected, enabling quicker inferencing without the need to repeatedly transfer production data to a dedicated inferencing platform.

The term edge corresponds to any device that generates and uses data, such as Internet of Things (IoT), mobile phones, autonomous cars, and so on. In the context of this document, the NetApp HCI solution is deployed in edge locations to host the mission critical applications of end users and deliver faster services with minimal latency. In the context of AI, the core data center is a platform that provides centralized services, such as machine learning and analytics, and the edge data centers are where the real-time production data is subject to inferencing. These edge data centers are usually connected to a core data center. They provide end-user services and serve as a staging layer for data generated by IoT devices that need additional processing that is too time sensitive to be transmitted back to a centralized core.

This document describes a reference architecture for AI inferencing using NetApp HCI as the base platform.

2 Program Summary

2.1 NetApp Verified Architecture

The NetApp Verified Architecture (NVA) program offers customers a verified, referenceable architecture for NetApp solutions. With an NVA solution, you get a NetApp solution architecture that provides the following advantages:

- Thoroughly tested
- Prescriptive in nature
- Minimized deployment risks
- Accelerated time to market

2.2 NetApp HCI AI Solution

The NetApp HCI AI inferencing solution powered by the NetApp H615c compute nodes with NVIDIA T4 GPUs and NetApp cloud-connected storage systems, was developed and verified by NetApp and NVIDIA. This solution provides IT organizations a prescriptive architecture that –

- Enables inferencing on edge data centers
- Optimizes consumption of GPU resources
• Provides a Kubernetes-based inferencing platform for flexibility and scalability
• Eliminates design complexities

NetApp HCI is an ideal edge platform to host multiple concurrent workloads or applications that can either be virtualized or non-virtualized in nature. Most applications today are not AI driven, but they are evolving to include AI capabilities to reap the immense benefits. To support the adoption of AI, applications need an infrastructure that can provide them with the necessary resources to function at their optimum level and continue to support their evolution.

NetApp HCI is built on a modular architecture that enables independent scaling of the compute and storage nodes. Customers can start small and grow non-disruptively while intelligently managing data from the edge to the cloud from the core to the cloud back.

In this solution, NetApp HCI integrates H410c compute nodes, H615c compute nodes equipped with NVIDIA T4 GPUs, and H410s storage nodes with a dedicated high-speed backend network. NetApp HCI simplifies the deployment of AI inferencing solutions at edge data centers by addressing areas of ambiguity, eliminating complexities in the design and guesswork.

2.3 The Value Proposition and Differentiation for NetApp HCI for Edge Inferencing

NetApp HCI provides differentiation in the hyperconverged market for this inferencing solution, including the following:

• A disaggregated architecture that allows for independent scaling of compute and storage and eliminates virtualization licensing costs and performance tax on independent NetApp HCI storage nodes
• NetApp Element® storage provides quality of service (QoS) per storage volume and allows for guaranteed storage performance for workloads on NetApp HCI, preventing adjacent workloads from negatively affecting inferencing performance
• Data fabric powered by NetApp allows data to be replicated from core to edge to cloud data centers to move the data closer to where the application needs the data
• With data fabric powered by NetApp and NetApp FlexCache®, AI deep learning models trained on NetApp ONTAP AI can be accessed from NetApp HCI without having to export the model
• NetApp HCI can host inference servers on the same infrastructure concurrently with multiple workloads, either virtual machine or container based, without sacrificing performance degradation
• NetApp HCI is NVIDIA NGC-ready certified for NVIDIA AI containerized applications
• An NGC-ready stack means that it is validated by NVIDIA, purpose built for AI, and enterprise support is available through NGC Support Services
• With the NetApp AI portfolio, NetApp can support the entire spectrum of AI use cases from edge to core to cloud, including ONTAP AI for training and inferencing, Cloud Volumes Service and Azure NetApp files for training in the cloud, and inferencing on the edge with NetApp HCI.

3 The Data Pipeline for Inferencing

The flow of data in the lifecycle of AI begins in the edge and ends in the edge. A lot of high-quality data is necessary to train a model that delivers high performance and accuracy, and most of this data is available in edge locations. Once trained, the model must be put into production and is returned to end-user applications running in the edge. The end-to-end flow of data can be categorized into three stages: data ingestion from endpoints; preparation and training; and tiering/analysis.
Figure 1 illustrates the three stages of data flow and the NetApp technologies that can be used to move data between the stages. In this document, the emphasis is on obtaining the trained model from the NetApp AI system, which is part of the core, and implementing it in the edge production environment for inferencing.

4 Solution Overview

The target audience for this solution includes, but is not limited to, data scientists, IT architects, field consultants, professional services, IT managers, and customers who want to take advantage of an infrastructure that has been built to deliver IT innovation and robust data and application services at edge locations.

Edge data centers manage and process data at locations that are very near to the generation point. This increases the efficiency and reduces the latency involved in handling data. Many vertical markets have realized the benefits of an edge data center and are heavily adopting this distributed approach to data processing.

The below table lists some of the key verticals and their areas of application that will immensely benefit from running on edge data centers.

Table 1) Edge verticals and applications

<table>
<thead>
<tr>
<th>Vertical</th>
<th>Applications</th>
</tr>
</thead>
<tbody>
<tr>
<td>Medical</td>
<td>Computer-aided diagnostics, assist medical staff in early disease detection</td>
</tr>
<tr>
<td>Oil &amp; Gas</td>
<td>Autonomous inspection of remote production facilities, video/image analytics</td>
</tr>
<tr>
<td>Aviation</td>
<td>Air traffic control assistance, real-time video feed analytics</td>
</tr>
</tbody>
</table>
### Vertical Applications

<table>
<thead>
<tr>
<th>Vertical</th>
<th>Applications</th>
</tr>
</thead>
<tbody>
<tr>
<td>Media &amp; Entertainment</td>
<td>Audio/video content filtering, deliver family-friendly content</td>
</tr>
<tr>
<td>Business Analytics</td>
<td>Brand recognition, analyze brand appearance in live-streamed televised events</td>
</tr>
<tr>
<td>eCommerce</td>
<td>Smart bundling of supplier offers, find ideal merchant/warehouse combination</td>
</tr>
<tr>
<td>Retail</td>
<td>Automated checkout, recognize items customer placed in cart, pay digitally</td>
</tr>
<tr>
<td>Smart City</td>
<td>Improve traffic flow, optimize parking, enhance pedestrian/cyclist safety</td>
</tr>
<tr>
<td>Manufacturing</td>
<td>Quality control, assembly line monitoring, defect identification</td>
</tr>
<tr>
<td>Customer Service</td>
<td>Customer service automation, analyze and triage inquiries (phone/email/social media)</td>
</tr>
<tr>
<td>Agriculture</td>
<td>Intelligent farm operation, activity planning, optimize fertilizer/herbicide application</td>
</tr>
</tbody>
</table>

### 4.1 Solution Technology

This solution is designed with a NetApp HCI system that comprises two H615c compute nodes with NVIDIA T4 GPUs, two H410c compute nodes, four H410s storage nodes, and two Mellanox SN2010 10GbE/25GbE switches.

Figure 2 illustrates the solution architecture for the NetApp HCI AI inferencing solution.
A VMware infrastructure is used to host the management services required by this inferencing solution. These services do not need to be deployed on a dedicated infrastructure; they can co-exist with any existing workloads. The NetApp Deployment Engine (NDE) uses the H410c and H410s nodes to deploy the VMware infrastructure.

Once the NDE has completed the configuration, the below components are deployed as virtual machines (VMs) in the virtual infrastructure:

- **Deployment Jump VM** – Used to automate the deployment of NVIDIA DeepOps and storage management using NetApp Trident.

  **Note:** NVIDIA DeepOps is discussed below.

- **ONTAP Select (optional)** – An instance of ONTAP Select is deployed to establish a connection with another ONTAP system that is part of the training environment

- **Kubernetes Masters** – Three VMs installed and configured with Ubuntu 18.04 LTS and configured as Kubernetes master nodes during deployment

After the management services have been setup, the two H615c compute nodes with NVIDIA T4 GPUs are installed with Ubuntu 18.04 LTS. These two nodes function as the Kubernetes worker nodes and provide the infrastructure for the inferencing platform.

Figure 3 illustrates the virtual and physical elements of this solution.
4.2 NetApp HCI Compute Nodes

The NetApp HCI compute nodes are available in two form factors - half-width and full-width - and in two rack unit sizes – 1 RU and 2 RU. The 410c nodes used in this solution are half-width and 1 RU and are housed in a chassis that can hold a maximum of four such nodes. The other compute node that is used in this solution is the H615c, which is a full-width node and 1 RU in size. While the H410c nodes are based on Intel Skylake processors, the H615c nodes are based on the second-generation Intel Cascade Lake processors. NVIDIA GPUs can be added to the H615c nodes and each node can host a maximum of three NVIDIA Tesla T4 16GB GPUs.

The H615c nodes are the latest series of compute nodes for NetApp HCI and the second series that can support GPUs. The first model to support GPUs is the H610c node (full width, 2RU) and can support two NVIDIA Tesla M10 GPUs.

In this solution, the H615c nodes are preferred over the H610c nodes due to the following advantages:

- Reduced datacenter footprint, critical for edge deployments
- Support for newer generation of GPUs designed for faster inferencing
- Reduced power consumption
- Reduced heat dissipation

4.3 NVIDIA T4 GPUs

The resource requirements of inferencing are nowhere close to that of training workloads. In fact, most modern-day hand-held devices are capable of handling small amounts of inferencing without powerful resources like GPUs. However, when it comes to mission critical applications and data centers that are dealing with a wide variety of applications that demand very low inferencing latencies when subject to extreme parallelization and massive input batch sizes, the GPUs play a key role in reducing the inference time and help to boost application performance.
The NVIDIA Tesla T4 is a x16 PCIe Gen3 single-slot low-profile GPU based on the Turing architecture. The T4 GPUs deliver universal inference acceleration that spans applications such as image classification and tagging, video analytics, natural language processing, automatic speech recognition, and intelligent search. The breadth of Tesla T4’s inferencing capabilities enables it to be used in enterprise solutions and edge devices.

These GPUs are ideal to be deployed in edge infrastructures due to their low-power consumption and small PCIe form-factor. The size of the T4 GPUs enables the installation of two T4 GPUs inside the same space as a double-slot full-sized GPU. Though small, with a memory of 16GB, the T4’s can support large ML models or run inference on multiple smaller models simultaneously.

The Turing based T4 GPUs include an enhanced version of Tensor Cores and support a full range of precisions for inferencing FP32, FP16, INT8 and INT4. The GPU includes 2,560 CUDA Cores and 320 Tensor Cores, delivering up to 130 Tera Operations per second (TOPS) of INT8 and up to 260 TOPS of INT4 inferencing performance. When compared to CPU-based inferencing, the Tesla T4, powered by the new Turing Tensor Cores, delivers up to 40 times higher inference performance.

The Turing Tensor Cores accelerate the matrix-matrix multiplication at the heart of neural network training and inferencing functions. They particularly excel at inference computations, in which useful and relevant information can be inferred and delivered by a trained deep neural network (DNN) based on a given input.

The Turing GPU architecture inherits the enhanced Multi-Process Service (MPS) feature first introduced in the Volta architecture. Compared to Pascal-based Tesla GPUs, MPS on Tesla T4 improves inference performance for small batch sizes, reduces launch latency, improves QoS, and enables the servicing of higher numbers of concurrent client requests.

The NVIDIA T4 GPU is a part of the NVIDIA AI Inference Platform that supports all AI frameworks and provides comprehensive tooling and integrations to drastically simplify the development and deployment of advanced AI.

4.4 Element Software

NetApp Element software powers the storage of the NetApp HCI systems. It delivers agile automation through scale-out flexibility and guaranteed application performance to accelerate new services.

Storage nodes can be added to the system non-disruptively in increments of one and the storage resources are made available to the applications instantaneously. With every new node added to the system, it delivers a precise amount of additional performance and capacity to a usable pool. The data is automatically load-balanced in the background across all nodes in the cluster, maintaining even utilization as the system grows.

NetApp Element software supports the HCI system to comfortably host multiple workloads by guaranteeing QoS to each workload. By providing fine-grained performance control with minimum, maximum, and burst settings for each workload, it allows for well-planned consolidations while protecting application performance. It decouples performance from capacity and allows each volume to be allocated a specific amount of capacity and performance. You can modify these specifications dynamically without any interruption to data access.

As illustrated in Figure 4, Element software integrates with NetApp ONTAP® for enabling data mobility between NetApp storage systems that are running different storage operating systems. Data can be moved from Element software to ONTAP or vice-versa using the NetApp SnapMirror® technology. Element leverages the same technology to provide cloud connectivity by integrating with NetApp Cloud Volumes ONTAP. This enables data mobility from the edge to the core and to multiple public cloud service providers.

In this solution, the Element backed storage provides the storage services that are required to run the workloads/ applications on the HCI system.
4.5 ONTAP Select (Optional)

NetApp ONTAP Select introduces a software-defined data storage service model on top of NetApp HCI. It builds on NetApp HCI capabilities, adding a rich set of file and data services to the HCI platform while extending the data fabric.

While ONTAP Select is an optional component for implementing this solution, it does provide a host of benefits, ranging from data gathering, protection, mobility and so on, that are extremely useful in the context of the overall AI data lifecycle. It helps to simplify several day-to-day challenges that are involved in data handling, starting from ingestion, collection, training, deployment and tiering.

ONTAP Select can run as a VM on VMware and still bring in most of ONTAP’s capabilities that are available when it is running on a dedicated FAS platform, such as the following:

- Support for NFS/ CIFS
- NetApp FlexClone® technology
- NetApp FlexCache® technology
- NetApp ONTAP FlexGroup volumes
- NetApp SnapMirror® software
In this solution, ONTAP Select can be used to leverage the FlexCache feature that helps in reducing the data read latencies by caching the frequently read data from a back-end origin volume. In case of high-end inferencing applications with a lot of parallelization, multiple instances of the same model are deployed across the inferencing platform leading to multiple reads of the same model. Newer versions of the trained model can be seamlessly introduced to the inferencing platform by verifying that the desired model is available in the origin/source volume.

Figure 6) FlexCache operation.

4.6 NetApp Trident (Optional)

NetApp Trident is an open-source dynamic storage orchestrator that enables managing storage resources across all major NetApp storage platforms. It integrates with Kubernetes natively by means of which Persistent Volumes (PVs) can be provisioned on demand by using native Kubernetes interfaces and constructs. It enables microservices and containerized applications to use enterprise-class storage services such as QoS, storage efficiencies, and cloning to meet the persistent storage demands of applications.

Containers have become one of the most popular methods of packaging and deploying applications and Kubernetes is one of the most popular platforms for hosting containerized applications. This solution falls within the same category and the inferencing platform is built on top of a Kubernetes based infrastructure.

Currently, Trident supports storage orchestration across the following platforms:

- ONTAP – NetApp AFF, FAS, Select
- Element software – HCI and NetApp SolidFire® all-flash storage
- NetApp SANtricity® software – E-series and EF-series
- Cloud Volumes ONTAP
- Azure NetApp Files
- NetApp Cloud Volumes Service – AWS and Google Cloud

Trident is a simple yet powerful tool to enable storage orchestration not just across multiple storage platforms, but also across the entire spectrum of the AI data lifecycle, ranging from the edge to the core to the cloud.

In this solution, Trident can be used to provision a PV from a Snapshot copy that comprises the trained model. Figure 7 illustrates the Trident workflow in which a Persistent Volume Claim (PVC) is created by referring to an existing Snapshot copy. Following this, Trident creates a volume by using the Snapshot copy.
This method of introducing trained models from a Snapshot copy helps in maintaining versioning of the models. It simplifies the process of introducing newer versions of models to the applications and switching inferencing between different versions of the model.

### 4.7 Containers and Kubernetes

Containers provide a layer of abstraction between the applications and the environment in which they run by providing a logical packaging mechanism for the applications. This decoupling allows container-based applications to be deployed easily and consistently across several platforms. With container-based application deployment, there is a clear separation of duty and responsibility. The developers deal with the application’s code and its dependent libraries, and the IT team focusses on the management and deployment.

Quite often, containers are compared with VMs because they offer many of the same application sandboxing benefits as VMs. The significant difference between them is that the containers have eliminated the hypervisor and guest operating system layers, which makes them much lighter. Figure 8 illustrates the difference between VMs and containers.
One of the most popular container runtime environments is Docker, which can be used to create and build software inside containers. It uses Docker images to deploy containerized applications or software across multiple environments. An application that has been containerized in the Docker format can run on any machine that can run Docker containers, including common operating systems such as Linux, Microsoft Windows, and other on-premises or cloud-based infrastructures.

On the flip side, managing containers at scale is a complex task. A single application that is containerized can be made up of several containers, and for the application to work without any issues, all the constituent containers must be running without any downtime.

For example, if a container goes down, another container must start and continue to deliver the service. Monitoring hundreds of containers and spinning up replacement containers manually is an arduous and inefficient task.

The solution to this problem is to use a container orchestrator. Kubernetes is one of the most popular choices that supports multiple container runtime environments including Docker. Kubernetes orchestrates the operation of multiple containers in harmony together. It takes care of scaling and failover for the applications, provides deployment patterns, and manages the use of underlying infrastructure resources, such as the amount of compute, network, and storage resources required. Kubernetes make it easier to automate and scale container-based workloads for live production environments.

Listed below are some of the key features of Kubernetes:

- **Service discovery and load balancing**
  - Containers can be exposed with their DNS names or IP address
  - Heavy traffic to a container is load balanced by distributing the network traffic
- **Storage orchestration**
  - Flexibility to mount a storage system of choice – Local, NFS, iSCSI, Cloud, and so on.
- **Automated rollouts and rollbacks**
  - Automate removal of existing containers and creation of new containers with the freed-up resources
- **Automatic bin packing**
  - Best use of Kubernetes cluster resources is guaranteed based on resource specifications at the container level
Container placement on nodes is managed by Kubernetes

• Self-healing
  − Failed containers are restarted, containers are replaced/ killed when they do not respond to user-defined health checks

• Secret and configuration management
  − Secure management of sensitive information – passwords, SSH keys, OAuth tokens

This solution leverages the features and benefits provided by Kubernetes and containers to build the inferencing platform discussed in the following sections.

4.8 NVIDIA DeepOps

NVIDIA DeepOps is a modular collection of ansible scripts that can be used to automate the deployment of a Kubernetes infrastructure. There are multiple deployment tools available today that can automate the deployment of a Kubernetes cluster. In this solution, DeepOps is the preferred choice as it does not just deploy a Kubernetes infrastructure, but also installs the necessary GPU drivers, NVIDIA Container Runtime for Docker (nvidia-docker2), and various other dependencies for GPU-accelerated work. It encapsulates the best practices for NVIDIA GPUs and can be customized or run as individual components as needed.

DeepOps internally uses Kubespray to deploy Kubernetes, and it is included as a submodule in DeepOps. Therefore, common Kubernetes cluster management operations such as adding nodes, removing nodes, and cluster upgrades should be performed using Kubespray.

In this solution, three Kubernetes master nodes are deployed as VMs, and the two H615c compute nodes with NVIDIA Tesla T4 GPUs are set up as Kubernetes worker nodes.

NVIDIA GPU Operator

The GPU operator deploys the NVIDIA k8s-device-plugin for GPU support and runs the NVIDIA drivers as containers. It is based on the Kubernetes operator framework, which helps to automate the management of all NVIDIA software components that are needed to provision GPUs. The components include NVIDIA drivers, Kubernetes device plug-in for GPUs, NVIDIA container runtime, and automatic node labeling, which is used in tandem with Kubernetes Node Feature Discovery.

The GPU operator is an important component of the NVIDIA EGX software-defined platform that is designed to make large-scale hybrid-cloud and edge operations possible and efficient. It is specifically useful when the Kubernetes cluster needs to scale quickly - for example, when provisioning additional GPU based worker nodes and managing the lifecycle of the underlying software components. Since the GPU operator runs everything as containers, including NVIDIA drivers, the administrators can easily swap various components simply by starting or stopping containers.

The GPU operator is enabled in DeepOps by invoking a prebuilt Ansible playbook egxstack-installation.yml that is available at https://github.com/NVIDIA/deepops

4.9 NVIDIA TensorRT Inference Server

NVIDIA TensorRT Inference Server (TRTIS) simplifies the deployment of AI inferencing solutions in production data centers. It is a specially designed microservice for inferencing in the production data centers. It maximizes GPU utilization and integrates into DevOps deployments with Docker and Kubernetes seamlessly.

With TRTIS, a common solution for AI inferencing is now available. This allows researchers to focus on creating high-quality trained models, DevOps engineers to focus on deployment, and developers to focus on applications without a need to redesign the platform for each AI-powered application.

Below are some of the key features of TRTIS:
• Support for multiple frameworks
  − TRTIS can handle a mix of models and the number of models is only limited by the system disk and memory resources. It can support TensorRT, TensorFlow GraphDef, TensorFlow SavedModel, ONNX, PyTorch, and Caffe2 NetDef model formats

• Concurrent model execution
  − Multiple models or multiple instances of the same model can be run simultaneously on a GPU

• Multi-GPU support
  − TRTIS can maximize GPU utilization by enabling inference for multiple models on one or more GPUs

• Support for batching
  − The server can accept requests for a batch of inputs and respond with the corresponding batch of outputs. The inference server supports multiple scheduling and batching algorithms that combine individual inference requests together to improve inference throughput. Batching algorithms are available for both stateless and stateful applications and need to be used appropriately. These scheduling and batching decisions are transparent to the client requesting inference.

• Ensemble support
  − A pipeline with multiple models with connections of input and output tensors between those models is an ensemble. An inference request can be made to an ensemble, which will result in the execution of the complete pipeline.

• Metrics
  − Details indicating GPU utilization, server throughput, server latency and health metrics for auto scaling and load balancing

In this solution, TRTIS is deployed on the Kubernetes cluster using a helm chart. With this method, the default configuration of TRTIS can be overridden and customized as per requirement. Once deployed, TRTIS provides an inference service using an HTTP or GRPC endpoint, allowing remote clients to request inferencing for any model being managed by the server.

NetApp HCI is a platform that can host multiple workloads and applications, and TRTIS is well equipped to support the inferencing requirements of multiple applications. In this solution, multiple sample applications are deployed on TRTIS to showcase its capabilities.

5 Technology Requirements

This section lists the hardware and software models or versions used during solution validation.

5.1 Hardware Requirements

Table 2 lists the hardware components that were used to implement this validated solution. The components that are used in any implementation of the solution might vary according to customer requirements.

<table>
<thead>
<tr>
<th>Layer</th>
<th>Product Family</th>
<th>Quantity</th>
<th>Details</th>
</tr>
</thead>
<tbody>
<tr>
<td>Compute</td>
<td>H615c</td>
<td>2</td>
<td>w/ 3 x NVIDIA Tesla T4 GPUs per node</td>
</tr>
<tr>
<td></td>
<td>H410c</td>
<td>2</td>
<td>Compute nodes for management infra</td>
</tr>
<tr>
<td>Storage</td>
<td>H410s</td>
<td>4</td>
<td>Storage for OS and workload</td>
</tr>
</tbody>
</table>
5.2 Software Requirements

Table 3 lists the software components that were used to build the base solution.

Table 3) Software requirements.

<table>
<thead>
<tr>
<th>Layer</th>
<th>Software</th>
<th>Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>Storage</td>
<td>NetApp Element OS</td>
<td>11.7.0.76</td>
</tr>
<tr>
<td></td>
<td>ONTAP Select Cluster</td>
<td>9.7</td>
</tr>
<tr>
<td></td>
<td>NetApp Trident</td>
<td>20.01</td>
</tr>
<tr>
<td>NetApp HCI engine</td>
<td>NetApp Deployment Engine</td>
<td>1.7P1</td>
</tr>
<tr>
<td>Hypervisor</td>
<td>Hypervisor</td>
<td>VMware vSphere ESXi 6.7U1</td>
</tr>
<tr>
<td></td>
<td>Hypervisor Management System</td>
<td>VMware vCenter Server 6.7U1</td>
</tr>
<tr>
<td>Inferencing Platform</td>
<td>NVIDIA DeepOps</td>
<td>20.02</td>
</tr>
<tr>
<td></td>
<td>NVIDIA GPU Operator</td>
<td>1.0.0</td>
</tr>
<tr>
<td></td>
<td>Ansible</td>
<td>2.7.11</td>
</tr>
<tr>
<td></td>
<td>Kubernetes</td>
<td>1.15.3</td>
</tr>
<tr>
<td></td>
<td>Docker</td>
<td>Docker CE 19.03.2</td>
</tr>
<tr>
<td></td>
<td>NVIDIA Container Toolkit</td>
<td>1.0.5</td>
</tr>
<tr>
<td></td>
<td>NVIDIA Kubernetes Device Plugin</td>
<td>1.0.0-beta4</td>
</tr>
<tr>
<td></td>
<td>NVIDIA Tesla Driver</td>
<td>418.87.01</td>
</tr>
<tr>
<td></td>
<td>TensorRT Inference Server</td>
<td>1.11.0</td>
</tr>
<tr>
<td>K8 Master VMs</td>
<td>Ubuntu</td>
<td>18.04 LTS</td>
</tr>
<tr>
<td>Host OS/ K8 Worker Nodes</td>
<td>Ubuntu</td>
<td>18.04 LTS</td>
</tr>
</tbody>
</table>

6 Conclusion

The adoption of AI is growing exponentially and is proving to be critical for the success of a business. To enable this adoption, specialized data center infrastructures are necessary to meet the requirements of the different stages of AI – Training and Inferencing.

In this solution, the focus is on designing an edge data center for AI inferencing using NetApp HCI.

While AI training has its own challenges and high resource requirements, inferencing is a challenging task in its own ways. Each application has its own inferencing requirements and prefer to access the trained model in a specific way. There is also a need for a well-defined data flow pipeline between the training
infrastructure and the edge inferencing platform to ensure that inferencing is always carried out with the latest or desired version of the model.

Mission critical workloads and applications on edge data centers are growing at rapid rates and so is the amount of data generated by them. On the other hand, IT administrators are always looking at ways to increase application density on these infrastructures for a higher return on investment (ROI).

NetApp HCI is a perfect solution for the edge data center. With support to host multiple containerized/ virtualized/ non-virtualized workloads concurrently, it helps in increasing application density and easy onboarding of new applications. The HCI infrastructure can be granularly scaled by adding compute or storage nodes to the system as required. In case of containerized applications, the compute nodes can be added as Kubernetes worker nodes to an existing Kubernetes cluster to enable onboarding of additional applications.

To meet AI inferencing requirements, the H615c compute node platform with NVIDIA Tesla T4 GPUs that are optimized for inferencing, is now available with NetApp HCI. When combined with NVIDIA TensorRT Inference Server this provides a dynamic and scalable platform that optimizes and enhances GPU utilization, which in turn boosts the inferencing performance of the applications onboard.

The data pipeline between training and inferencing can be established using multiple NetApp technologies. Specifically, NetApp Trident, a Kubernetes native tool, helps in providing seamless access to persistent data across nodes or regions quickly and easily.

With this offering, NetApp now provides a complete portfolio of solutions for AI training and inferencing across the core and edge with technologies to seamlessly move mission-critical data between them.

Where to Find Additional Information

To learn more about the information described in this document, refer to the following documents and/or websites:

- NetApp HCI Theory of Operations
- NetApp Product Documentation
  docs.netapp.com
- NetApp HCI Solution Catalog Documentation
- HCI Resources page
  https://mysupport.netapp.com/info/web/ECMLP2831412.html
- ONTAP Select
- NetApp Trident
- NVIDIA DeepOps
  https://github.com/NVIDIA/deepops
- NVIDIA TensorRT Inference Server

Version History

<table>
<thead>
<tr>
<th>Version</th>
<th>Date</th>
<th>Document Version History</th>
</tr>
</thead>
<tbody>
<tr>
<td>Version 1.0</td>
<td>March 2020</td>
<td>Initial release.</td>
</tr>
</tbody>
</table>
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

Copyright © 2020 NetApp, Inc. All Rights Reserved. Printed in the U.S. No part of this document covered by copyright may be reproduced in any form or by any means—graphic, electronic, or mechanical, including photocopying, recording, taping, or storage in an electronic retrieval system—without prior written permission of the copyright owner.

Software derived from copyrighted NetApp material is subject to the following license and disclaimer:

THIS SOFTWARE IS PROVIDED BY NETAPP “AS IS” AND WITHOUT ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE, WHICH ARE HEREBY DISCLAIMED. IN NO EVENT SHALL NETAPP BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

NetApp reserves the right to change any products described herein at any time, and without notice. NetApp assumes no responsibility or liability arising from the use of products described herein, except as expressly agreed to in writing by NetApp. The use or purchase of this product does not convey a license under any patent rights, trademark rights, or any other intellectual property rights of NetApp.

The product described in this manual may be protected by one or more U.S. patents, foreign patents, or pending applications.

Data contained herein pertains to a commercial item (as defined in FAR 2.101) and is proprietary to NetApp, Inc. The U.S. Government has a non-exclusive, non-transferrable, non-sublicensable, worldwide, limited irrevocable license to use the Data only in connection with and in support of the U.S. Government contract under which the Data was delivered. Except as provided herein, the Data may not be used, disclosed, reproduced, modified, performed, or displayed without the prior written approval of NetApp, Inc. United States Government license rights for the Department of Defense are limited to those rights identified in DFARS clause 252.227-7015(b).

Trademark Information

NETAPP, the NETAPP logo, and the marks listed at http://www.netapp.com/TM are trademarks of NetApp, Inc. Other company and product names may be trademarks of their respective owners.

NVA-1144-0320