



# **FlexPod for Genomics**

## **FlexPod**

NetApp  
January 21, 2025

This PDF was generated from <https://docs.netapp.com/us-en/flexpod/healthcare/flexpod-genomics-introduction.html> on January 21, 2025. Always check docs.netapp.com for the latest.

# Table of Contents

- FlexPod for Genomics . . . . . 1
  - TR-4911: FlexPod Genomics . . . . . 1
  - Benefits of deploying genomic workloads on FlexPod . . . . . 3
  - Solution infrastructure hardware and software components . . . . . 8
  - Genomics - GATK setup and execution . . . . . 11
  - Output for execution of GATK using the jar file . . . . . 21
  - Output for execution of GATK using the `./gatk` script . . . . . 24
  - Output for execution of GATK using the Cromwell engine . . . . . 26
  - GPU setup . . . . . 30
  - Conclusion . . . . . 39

# FlexPod for Genomics

## TR-4911: FlexPod Genomics

JayaKishore Esanakula, NetApp

There are few fields of medicine that are more important than genomics for healthcare and the life sciences, and genomics are fast becoming a key clinical tool for doctors and nurses. Genomics, when combined with medical imaging and digital pathology, help us understand how a patient's genes might be affected by treatment protocols. The success of genomics in healthcare increasingly depends on data interoperability at scale. The end goal is to make sense of the enormous volumes of genetic data and identify clinically relevant correlations and variants that improve diagnosis and make precision medicine a reality. Genomics help us understand the origin of disease outbreaks, how diseases evolve, and which treatments and strategies might be effective. Clearly, genomics has many benefits that span prevention, diagnosis, and treatment. Healthcare organizations are grappling with several challenges, including the following:

- Improved care quality
- Value-based care
- Data explosion
- Precision medicine
- Pandemics
- Wearables, remote monitoring, and care
- Cyber security

Standardized clinical pathways and clinical protocols are one of the critical components of modern medicine. One of the key aspects of standardization is interoperability between care providers, not just for medical records but also for genomic data. The big question is will healthcare organizations relinquish ownership of genomic data in lieu of patient ownership of their personal genomics data and related medical records?

Interoperable patient data is key for enabling precision medicine, one of the driving forces behind the recent explosion of data growth. The objective for precision medicine is to make health maintenance, disease prevention, diagnoses, and treatment solutions more effective and precise.

The rate of data growth has been exponential. In early February 2021, US laboratories sequenced approximately 8,000 COVID-19 strains per week. The number of genomes sequenced had increased to 29,000 per week by April 2021. Each fully sequenced human genome is around 125GB in size. Therefore, at a rate of 29,000 genomes sequenced per week, total genome storage at rest would be more than 180 petabytes per year. Various countries have committed resources to genomic epidemiology to improve genomic surveillance and prepare for the next wave of global health challenges.

The reduced cost of genomic research is driving genetic testing and research at an unprecedented rate. The three Ps are at an inflection point: computer power, privacy of data, and personalization of medicine. By 2025 researchers estimate that 100 million to as many as 2 billion human genomes will be sequenced. For genomics to be effective and a valuable proposition, genomics capabilities must be a seamless part of care workflows; it should be easy to access and be actionable during a patient's visit. It is also equally important that patient electronic medical-record data be integrated with patient genomics data. With the advent of state-of-the-art

converged infrastructure like FlexPod, organizations can bring their genomics capabilities into the everyday workflows of physicians, nurses, and clinic managers. For the latest FlexPod platform information, see this [FlexPod Datacenter with Cisco UCS X-Series White Paper](#).

For a physician, the true value of genomics includes precision medicine and personalized treatment plans based on the genomic data of a patient. There has never been such synergy between clinicians and data scientists in the past, and genomics is benefiting from the technological innovations in the recent past, and also real partnerships between healthcare organizations and technology leaders in the industry.

Academic medical centers and other healthcare and life science organizations are well on their way to establishing center of excellence (COE) in genome science. According to Dr. Charlie Gersbach, Dr. Greg Crawford, and Dr. Tim E Reddy from Duke University, “We know that genes aren’t turned on or off by a simple binary switch, but instead it’s a result of multiple gene regulatory switches that work together. ” They have also determined that “none of these parts of the genome work in isolation. The genome is a very complicated web that evolution has woven” ( [ref](#)).

NetApp and Cisco have been hard at work implementing incremental improvements into the FlexPod platform for over 10 years. All customer feedback is heard, evaluated, and tied into the value streams and feature sets in FlexPod. It is this continuous loop of feedback, collaboration, improvement, and celebration that sets FlexPod apart as a trusted converged infrastructure platform the world over. It has been simplified and designed from the ground up to be the most reliable, robust, versatile, and agile platform for healthcare organizations.

## Scope

The FlexPod converged infrastructure platform enables a healthcare organization to host one or more genomics workloads, along with other clinical and nonclinical healthcare applications. This technical report uses an open-source, industry-standard genomics tool called GATK during FlexPod platform validation. However, a deeper discussion of genomics or GATK is outside the scope of this document.

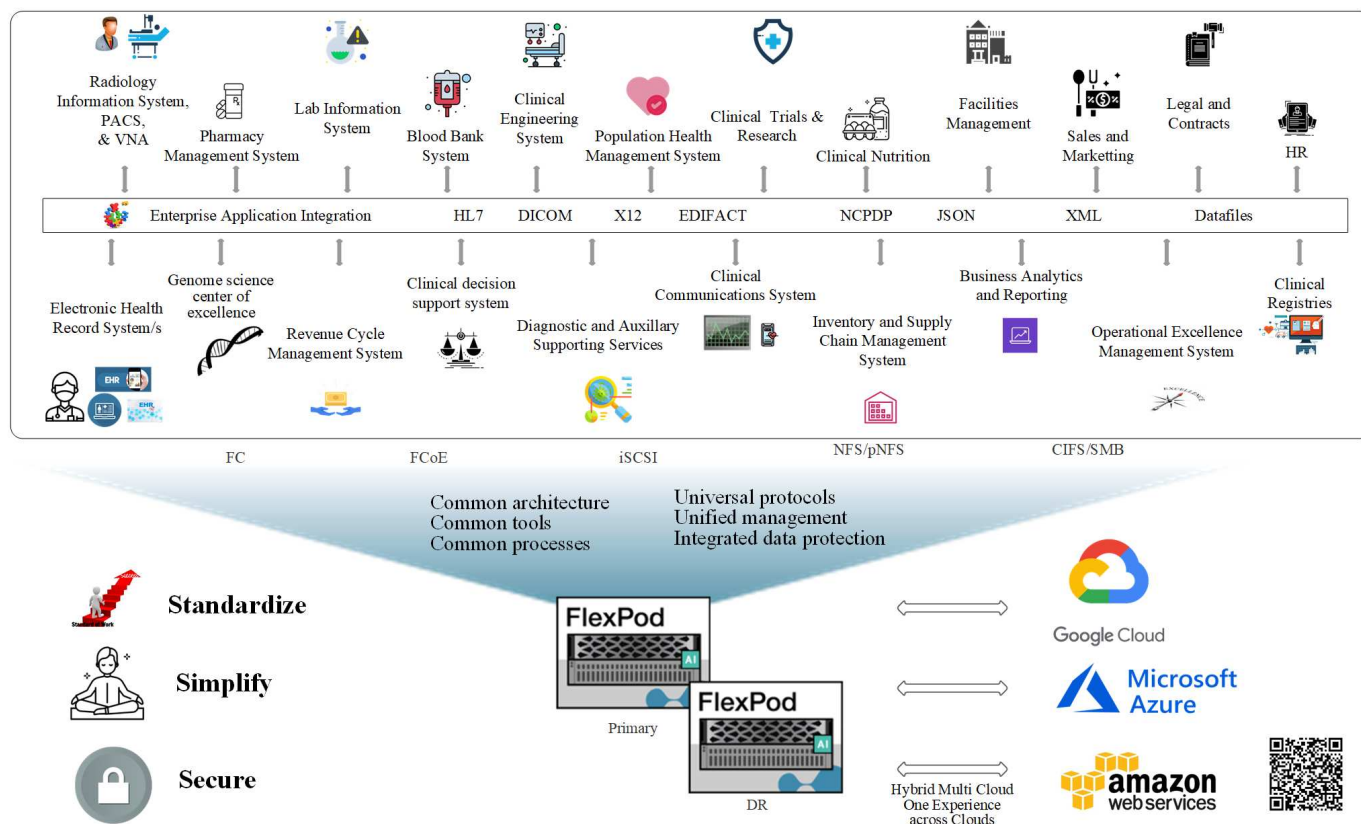
## Audience

This document is intended for technical leaders in the healthcare industry and for Cisco and NetApp partner solutions engineers and professional services personnel. NetApp assumes that the reader has a good understanding of compute and storage sizing concepts as well as a technical familiarity with healthcare threats, healthcare security, healthcare IT systems, Cisco UCS, and NetApp storage systems.

## Hospital capabilities deployed on FlexPod

A typical hospital has a diversified set of IT systems. The majority of such systems are purchased from a vendor, whereas very few are built by the hospital system in house. Therefore, the hospital system must manage a diverse infrastructure environment in their data centers. When hospitals unify their systems into a converged infrastructure platform like FlexPod, organizations can standardize their data center operations. With FlexPod, healthcare organizations can implement clinical and non-clinical systems on the same platform, thereby unifying data center operations.

## Hospital capabilities deployed on a FlexPod



Next: [Benefits of deploying genomic workloads on FlexPod.](#)

## Benefits of deploying genomic workloads on FlexPod

Previous: [Introduction.](#)

This section provides a brief list of benefits for running a genomics workload on a FlexPod converged infrastructure platform. Let's quickly describe the capabilities of a hospital. The following business architecture view shows a hospital's capabilities deployed on a hybrid-cloud-ready FlexPod converged infrastructure platform.

- **Avoid siloes in healthcare.** Silos in healthcare are a very real concern. Departments are often siloed into their own set of hardware and software not by choice but organically by evolution. For example, radiology, cardiology, EHR, genomics, analytics, revenue cycle, and other departments end up with their individual set of dedicated software and hardware. Healthcare organizations maintain a limited set of IT professionals to manage their hardware and software assets. The inflection point comes when this set of individuals are expected to manage a very diversified set of hardware and software. Heterogeneity is made worse by an incongruent set of processes brought to the healthcare organization by vendors.
- **Start small and grow.** The GATK tool kit is tuned for CPU execution, which best suites platforms like FlexPod. FlexPod enables independent scalability of network, compute, and storage. Start small and scale as your genomics capabilities and the environment grows. Healthcare organizations don't have to invest in specialized platforms to run genomic workloads. Instead, organizations can leverage versatile platforms like a FlexPod to run genomics and non-genomics workloads on the same platform. For example, if the pediatrics department wants to implement genomics capability, IT leadership can provision compute, storage, and networking on an existing FlexPod instance. As the genomics business unit grows, healthcare

organization can scale their FlexPod platform as needed.

- **Single control pane and unparalleled flexibility.** Cisco Intersight significantly simplifies IT operations by bridging applications with infrastructure, providing visibility and management from bare-metal servers and hypervisors to serverless applications, thereby reducing costs and mitigating risk. This unified SaaS platform uses a unified Open API design that natively integrates with third-party platforms and tools. Moreover, it allows management to occur from your data center operations team on site or from anywhere by using a mobile app.

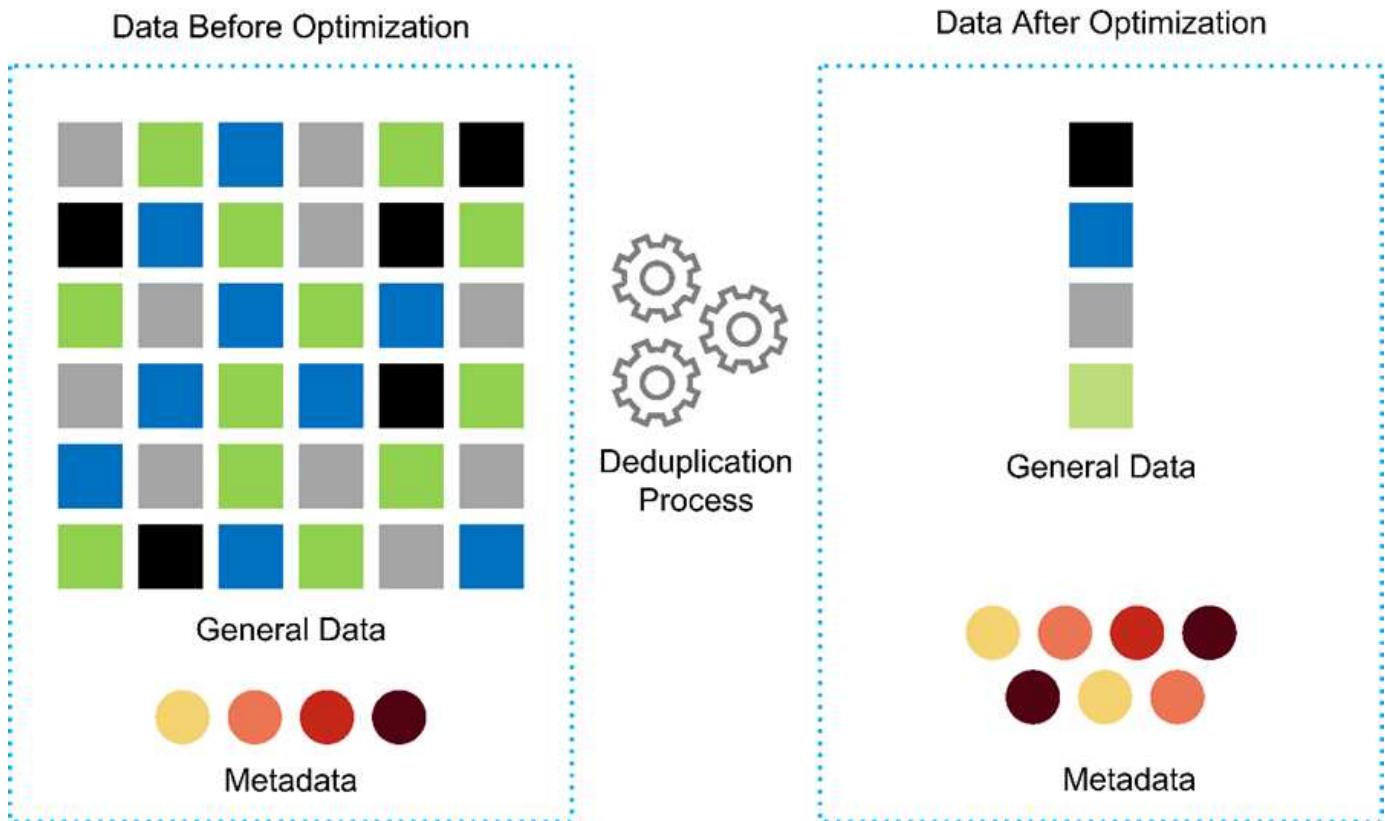
Users quickly unlock tangible value in their environment by leveraging Intersight as their management platform. Enabling automation for many daily manual tasks, Intersight removes errors and simplifies your daily operations. Moreover, advanced support capabilities facilitated by Intersight allow adopters to stay ahead of problems and accelerate issue resolution. Taken in combination, organizations spend far less time and money on their application infrastructure and more time on their core business development.

Leveraging Intersight management and FlexPod's easily scalable architecture enables organizations to run several genome workloads on a single FlexPod platform, increasing utilization and reducing total cost of ownership (TCO). FlexPod allows for flexible sizing, with choices starting with our small FlexPod Express and scaling into large FlexPod Datacenter implementations. With role-based access control capabilities built into Cisco Intersight, healthcare organizations can implement robust access control mechanisms, avoiding the need for separate infrastructure stacks. Multiple business units within the healthcare organization can leverage genomics as a key core competency.

Ultimately FlexPod helps simplify IT operations and lower operating costs, and it allows IT infrastructure admins to focus on tasks that help clinicians innovate rather than being relegated to keeping the lights on.

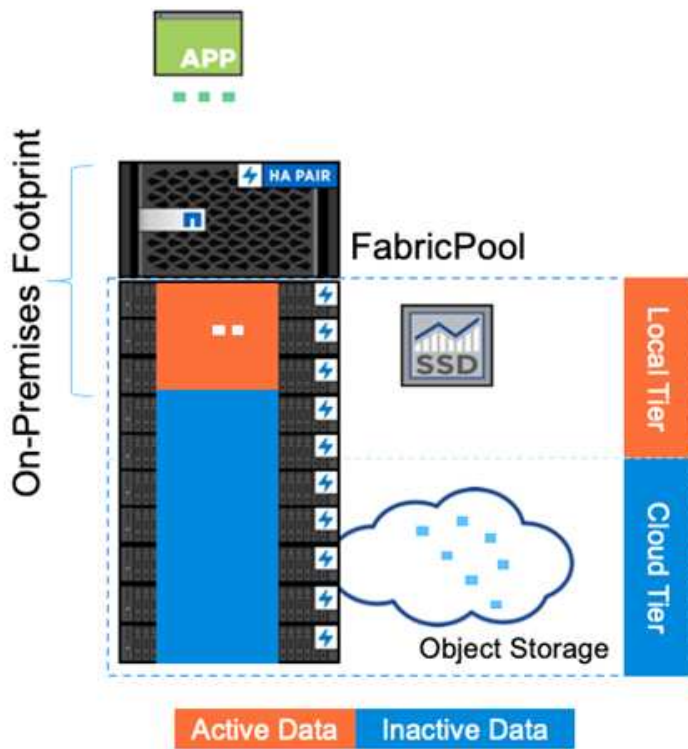
- **Validated design and guaranteed outcomes.** FlexPod design and deployment guides are validated to be repeatable, and they cover comprehensive configuration details and industry best practices that are needed to deploy a FlexPod with confidence. Cisco and NetApp validated design guides, deployment guides, and architectures help your healthcare or life science organization remove guesswork from the implementation of a validated and trusted platform from the beginning. With FlexPod, you can speed up deployment times and reduce cost, complexity, and risk. FlexPod validated designs and deployment guides establish FlexPod as the ideal platform for a variety of genomics workloads.
- **Innovation and agility.** FlexPod is recommended as an ideal platform by EHRs like Epic, Cerner, Meditech and imaging systems like Agfa, GE, Philips. For more information on [Epic honor roll](#) and target platform architecture, see the Epic userweb. Running genomics on [FlexPod](#) enables healthcare organizations to continue their journey of innovation with agility. With FlexPod, implementing organizational change comes naturally. When organizations standardize on a FlexPod platform, healthcare IT experts can provision their time, effort, and resources to innovate and thus be as agile as the ecosystem demands.
- **Data liberated.** With the FlexPod converged infrastructure platform and a NetApp ONTAP storage system, genomics data can be made available and accessible using a wide variety of protocols at scale from a single platform. FlexPod with NetApp ONTAP offers a simple, intuitive, and powerful hybrid cloud platform. Your data fabric powered by NetApp ONTAP weaves data together across sites, beyond physical boundaries, and across applications. Your data fabric is built for data-driven enterprises in a data-centric world. Data is created and used in multiple locations, and it often needs to be leveraged and shared with other locations, applications, and infrastructures. Therefore, you need a consistent and integrated way to manage it. FlexPod puts your IT team in control and simplifies ever-increasing IT complexity.
- **Secure multitenancy.** FlexPod uses FIPS 140-2 compliant cryptographic modules, hence enabling organizations to implement security as a foundational element, not an afterthought. FlexPod enables organizations implement secure multitenancy from a single converged infrastructure platform irrespective of the size of the platform. FlexPod with secured multitenancy and QoS help with workload separation and maximize utilization. This helps avoid capital being locked into specialized platforms that is potentially underutilized and requires a specialized skill set to manage.

- **Storage efficiency.** Genomics requires that the underlying storage have industry- leading storage efficiency capabilities. You can reduce storage costs with NetApp storage efficiency features such as deduplication (inline and on demand), data compression, and data compaction ( [ref](#)). NetApp deduplication provides block-level deduplication in a FlexVol volume. Essentially, deduplication removes duplicate blocks, storing only unique blocks in the FlexVol volume. Deduplication works with a high degree of granularity and operates on the active file system of the FlexVol volume. The following figure shows an overview of how NetApp deduplication works. Deduplication is application transparent. Therefore, it can be used to deduplicate data originating from any application that uses the NetApp system. You can run volume deduplication as an inline process and as a background process. You can configure it to run automatically, to be scheduled, or to run manually through the CLI, NetApp ONTAP System Manager, or NetApp Active IQ Unified Manager.



- **Enable genomics interoperability.** ONTAP FlexCache is a remote caching capability that simplifies file distribution, reduces WAN latency, and lowers WAN bandwidth costs, ( [ref](#)). One of the key activities during genomic variant identification and annotation is collaboration between clinicians. ONTAP FlexCache technology increases data throughput even when collaborating clinicians are in different geographic locations. Given the typical size of a \*.BAM file (1GB to 100s of GB), it is critical that the underlying platform can make files available to clinicians in different geographic locations. FlexPod with ONTAP FlexCache makes genomic data and applications truly multisite ready, which makes collaboration between researchers located around the world seamless with low latency and high throughput. Healthcare organizations running genomics applications in a multisite setting can scale-out using the data fabric to balance manageability with cost and speed.
- **Intelligent use of storage platform.** FlexPod with ONTAP auto-tiering and NetApp Fabric Pool technology simplifies data management. FabricPool helps reduce storage costs without compromising performance, efficiency, security, or protection. FabricPool is transparent to enterprise applications and capitalizes on cloud efficiencies by lowering storage TCO without the need to rearchitect the application infrastructure. FlexPod can benefit from the storage tiering capabilities of FabricPool to make more efficient use of ONTAP flash storage. For more information, see [FlexPod with FabricPool](#). The following diagram provides a high-level overview of FabricPool and its benefits.





Automatic tiering  
 Zero-touch management  
 Preserves file system  
 Lower cost of ownership  
 Choice of object tier locations



- **Faster variant analysis and annotation.** The FlexPod platform is faster to deploy and operationalize. The FlexPod platform enables clinician collaboration by making data available at scale with low latency and increased throughput. Increased interoperability enables innovation. Healthcare organizations can run their genomic and non-genomic workloads side by side, which means organizations do not need specialized platforms to start their genomics journey.

FlexPod ONTAP routinely adds cutting edge features to the storage platform. FlexPod Datacenter is the optimal shared infrastructure foundation for deploying FC- NVMe to allow high-performance storage access to applications that need it. As FC- NVMe evolves to include high availability, multipathing, and additional operating system support, FlexPod is well suited as the platform of choice, providing the scalability and reliability needed to support these capabilities. ONTAP with faster I/O with end-to-end NVMe allows genomics analyses to completed faster ( [ref](#)).

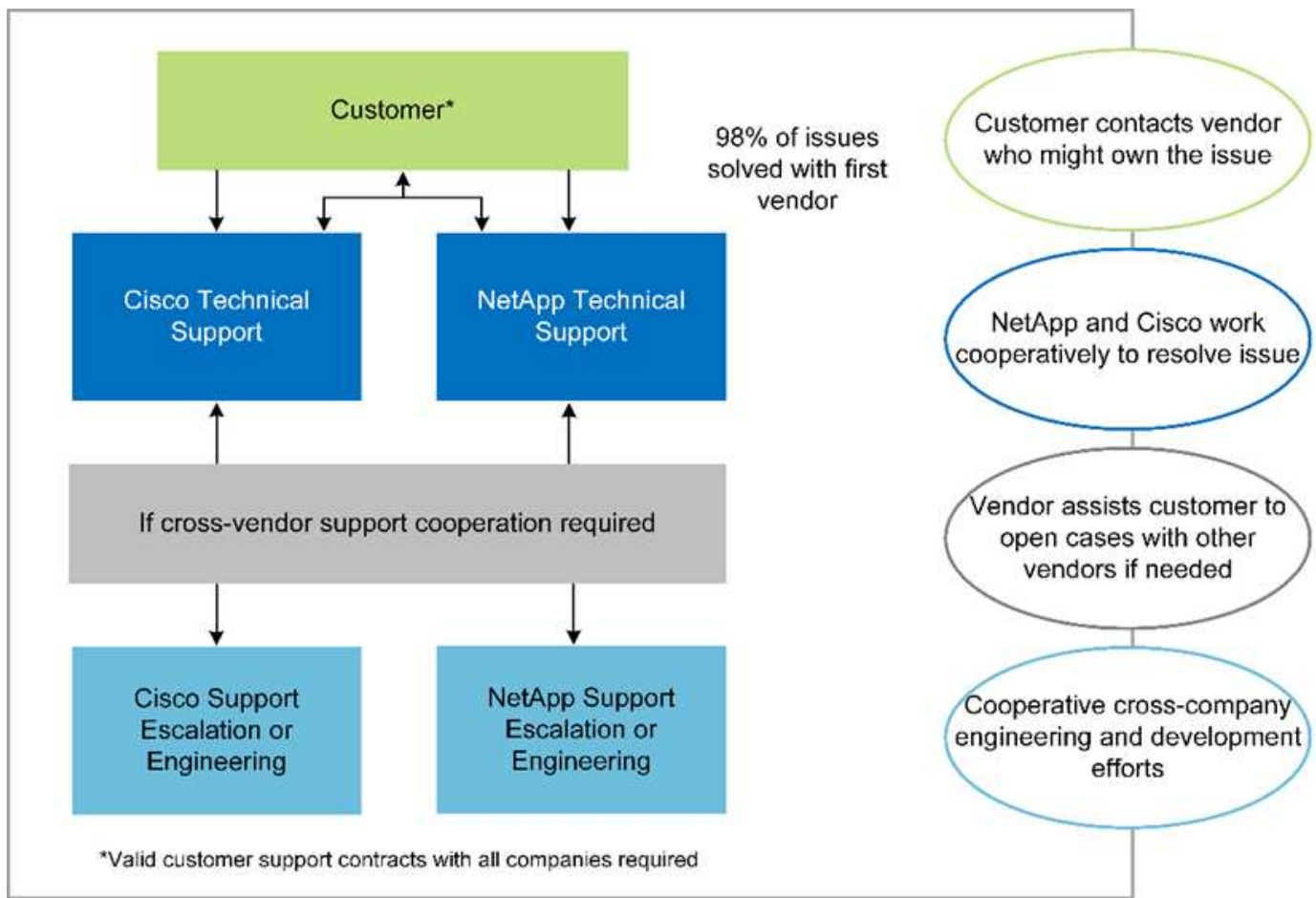
Sequenced raw genome data produces large file sizes, and it is important that these files are made available to the variant analyzers to reduce the total time it takes from sample collection to variant annotation. NVMe (nonvolatile memory express) when used as a storage access and data transport protocol provides unprecedented levels of throughput and the fastest response times. FlexPod deploys the NVMe protocol while accessing flash storage via the PCI express bus (PCIe). PCIe enables implementation of tens of thousands of command queues, increasing parallelization and throughput. One single protocol from storage to memory makes data access fast.

- **Agility for clinical research from the ground up.** Flexible, expandable storage capacity and performance allows the healthcare research organizations to optimize the environment in an elastic or just-in-time (JIT) manner. By decoupling storage from compute and network infrastructure, FlexPod platform can be scaled up and out without disruption. Using Cisco Intersight, the FlexPod platform can be managed with both built-in and custom automated workflows. Cisco Intersight workflows enable healthcare organizations to reduce application life-cycle management times. When an academic medical center requires that patient data be anonymized and made available to their center for research informatics and/or center for quality, their IT organization can leverage Cisco Intersight FlexPod workflows to take secure data backups, clone, and the



restore in a matter of seconds, not hours. With NetApp Trident and Kubernetes, IT organizations can provision new data scientists and make clinical data available for model development in a matter of minutes, sometimes even in seconds.

- **Protecting genome data.** NetApp SnapLock provides a special-purpose volume in which files can be stored and committed to a non-erasable, non-rewritable state. The user's production data residing in a FlexVol volume can be mirrored or vaulted to a SnapLock volume through NetApp SnapMirror or SnapVault technology. The files in the SnapLock volume, the volume itself, and its hosting aggregate cannot be deleted until the end of the retention period. Using ONTAP FPolicy software organizations can prevent ransomware attacks by disallowing operations on files with specific extensions. An FPolicy event can be triggered for specific file operations. The event is tied to a policy, which calls out the engine it needs to use. You might configure a policy with a set of file extensions that could potentially contain ransomware. When a file with a disallowed extension tries to perform an unauthorized operation, FPolicy prevents that operation from executing ([ref](#)).
- **FlexPod Cooperative Support.** NetApp and Cisco have established FlexPod Cooperative Support, a strong, scalable, and flexible support model to meet the unique support requirements of the FlexPod converged infrastructure. This model uses the combined experience, resources, and technical support expertise of NetApp and Cisco to offer a streamlined process for identifying and resolving FlexPod support issues, regardless of where the problem resides. The following figure provides an overview of the FlexPod Cooperative Support model. The customer contacts the vendor who might own the issue, and both Cisco and NetApp work cooperatively to resolve it. Cisco and NetApp have cross-company engineering and development teams that work hand in hand to resolve issues. This support model reduces loss of information during translation, enables trust, and reduces downtime.

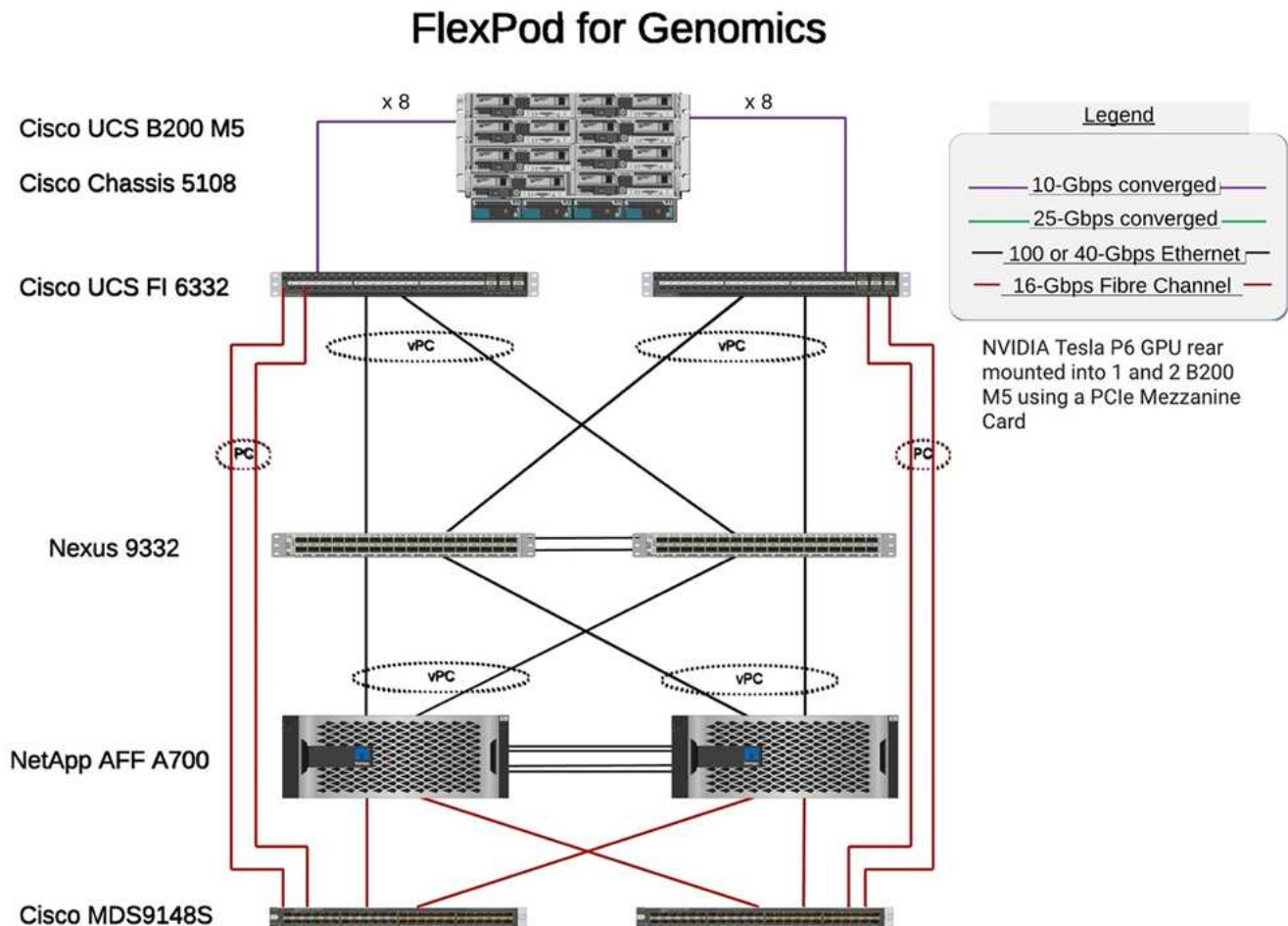


Next: Solution infrastructure hardware and software components.

# Solution infrastructure hardware and software components

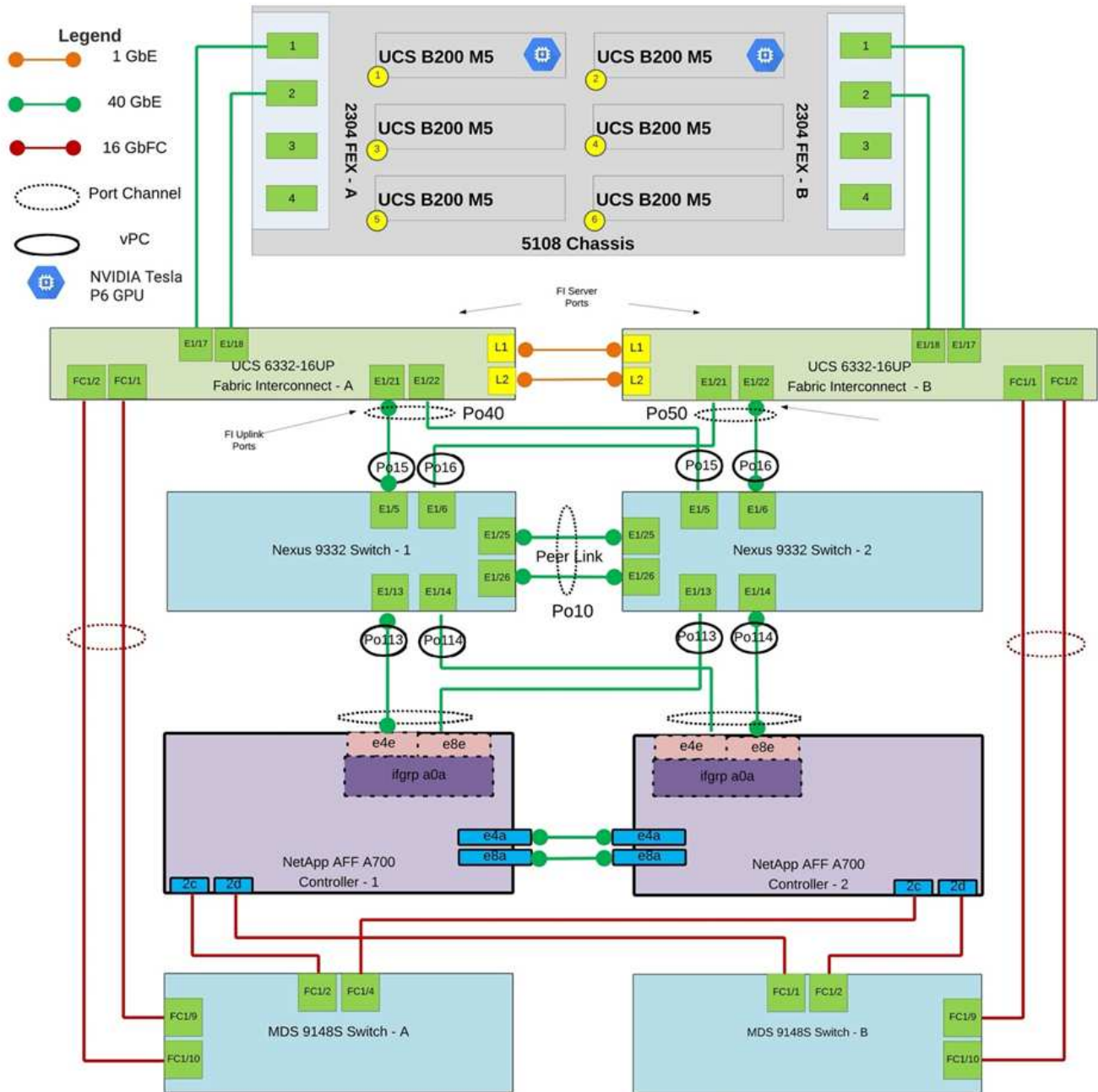
Previous: [Benefits of deploying genomic workloads on FlexPod.](#)

The following figure depicts the FlexPod system used for GATK setup and validation. We used [FlexPod Datacenter with VMware vSphere 7.0](#) and [NetApp ONTAP 9.7 Cisco Validated Design \(CVD\)](#) during the setup process.



The following diagram depicts the FlexPod cabling details.

# FlexPod for Genomics



The following table lists the hardware components used during the GATK testing enabling on a FlexPod. Here is the [NetApp Interoperability Matrix Tool \(IMT\)](#) and [Cisco Hardware Compatibility List \(HCL\)](#).

Layer	Product family	Quantity and model	Details
Compute	Cisco UCS 5108 chassis	1 or 2	
	Cisco UCS blade servers	6 B200 M5	Each with 2x 20 or more cores, 2.7GHz, and 128-384GB RAM

Layer	Product family	Quantity and model	Details
	Cisco UCS Virtual Interface Card (VIC)	Cisco UCS 1440	See the
	2x Cisco UCS Fabric Interconnects	6332	-
Network	Cisco Nexus switches	2x Cisco Nexus 9332	-
Storage network	IP network for storage access over SMB/CIFS, NFS, or iSCSI protocols	Same network switches as above	-
	Storage access over FC	2x Cisco MDS 9148S	-
Storage	NetApp AFF A700 all-flash storage system	1 Cluster	Cluster with two nodes
	Disk shelf	One DS224C or NS224 disk shelf	Fully populated with 24 drives
	SSD	24, 1.2TB or larger capacity	-

This table lists the infrastructure software.

Software	Product family	Version or release	Details
Various	Linux	RHEL 8.3	-
	Windows	Windows Server 2012 R2 (64 bit)	-
	NetApp ONTAP	ONTAP 9.8 or later	-
	Cisco UCS Fabric Interconnect	Cisco UCS Manager 4.1 or later	-
	Cisco Ethernet 3000 or 9000 series switches	For 9000 series, 7.0(3)I7(7) or later For 3000 series, 9.2(4) or later	-
	Cisco FC: Cisco MDS 9132T	8.4(1a) or later	-
	Hypervisor	VMware vSphere ESXi 7.0	-
Storage	Hypervisor management system	VMware vCenter Server 7.0 (vCSA) or later	-
Network	NetApp Virtual Storage Console (VSC)	VSC 9.7 or later	-
	NetApp SnapCenter	SnapCenter 4.3 or later	-
	Cisco UCS Manager	4.1(3c) or later	
Hypervisor	ESXi		

Software	Product family	Version or release	Details
Management	Hypervisor management systemVMware vCenter Server 7.0 (vCSA) or later		
	NetApp Virtual Storage Console (VSC)	VSC 9.7 or later	
	NetApp SnapCenter	SnapCenter 4.3 or later	
	Cisco UCS Manager	4.1(3c) or later	

Next: [Genomics - GATK setup and execution](#).

## Genomics - GATK setup and execution

Previous: [Solution infrastructure hardware and software components](#).

According to the National Human Genome Research Institute ( [NHGRI](#)), “Genomics is the study of all of a person’s genes (the genome), including interactions of these genes with each other and with a person’s environment. ”

According to the [NHGRI](#), “Deoxyribonucleic acid (DNA) is the chemical compound that contains the instructions needed to develop and direct the activities of nearly all living organisms. DNA molecules are made of two twisting, paired strands, often referred to as a double helix.” “An organism’s complete set of DNA is called its genome.”

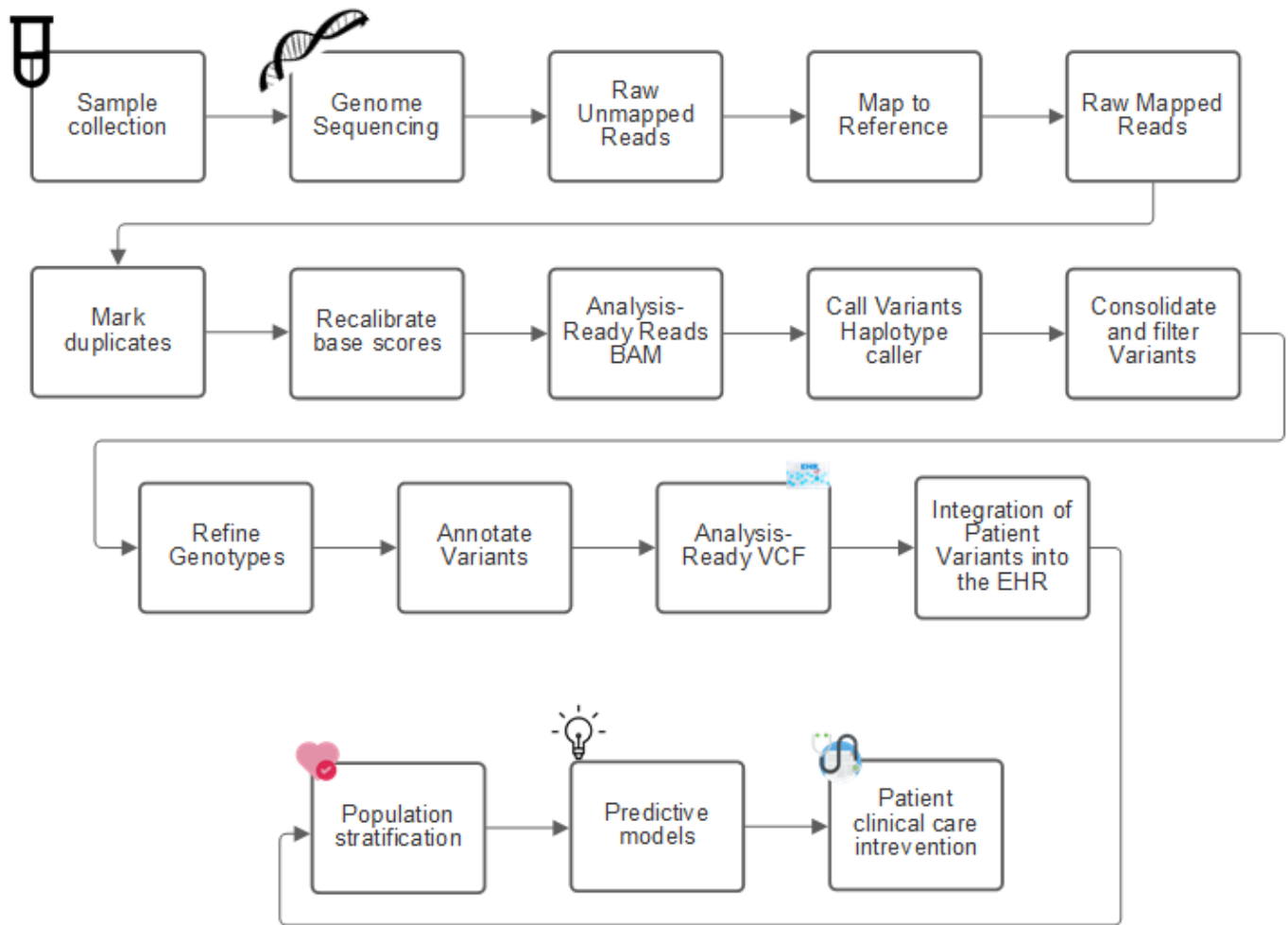
Sequencing is the process of determining the exact order of the bases in a strand of DNA. One of the most common types of sequencing used today is called sequencing by synthesis. This technique uses the emission of fluorescent signals to order the bases. Researchers can use DNA sequencing to search for genetic variations and any mutations that might play a role in the development or progression of a disease while a person is still in the embryonic stage.

### From sample to variant identification, annotation, and prediction

At a high level, genomics can be classified into the following steps. This is not an exhaustive list:

1. Sample collection.
2. [Genome sequencing](#) using a sequencer to generate the raw data.
3. Preprocessing. For example, [deduplication](#) using [Picard](#).
4. Genomic analysis.
  - a. Mapping to a reference genome.
  - b. [Variant](#) identification and annotation typically performed using GATK and similar tools.
5. Integration into the electronic health record (EHR) system.
6. [Population stratification](#) and identification of genetic variation across geographical location and ethnic background.
7. [Predictive models](#) using significant single- nucleotide polymorphism.
8. [Validation](#).

The following figure shows the process from sampling to variant identification, annotation, and prediction.



The Human Genome project was completed in April 2003, and the project made a very high-quality simulation of the human genome sequence available in the public domain. This reference genome initiated an explosion in research and development of genomics capabilities. Virtually every human ailment has a signature in that human's genes. Up until recently, physicians were leveraging genes to predict and determine birth defects like sickle cell anemia, which is caused by a certain inheritance pattern caused by a change in a single gene. The treasure trove of data being made available by the human genome project led to the advent of the current state of genomics capabilities.

Genomics has a broad set of benefits. Here is a small set of benefits in the healthcare and life sciences domains:

- Better diagnosis at point of care
- Better prognosis
- Precision medicine
- Personalized treatment plans
- Better disease monitoring
- Reduction in adverse events
- Improved access to therapies
- Improved disease monitoring



- Effective clinical trial participation and better selection of patients for clinical trials based on genotypes.

Genomics is a "four-headed beast," because of the computational demands across the lifecycle of a dataset: acquisition, storage, distribution, and analysis.

## Genome Analysis Toolkit (GATK)

GATK was developed as a data science platform at the [Broad Institute](#). GATK is a set of open-source tools that enable genome analysis, specifically variant discovery, identification, annotation, and genotyping. One of the benefits of GATK is that the set of tools and or commands can be chained to form a complete workflow. The primary challenges that Broad institute tackles are the following:

- Understand the root causes and biological mechanisms of diseases.
- Identify therapeutic interventions that act at the fundamental cause of a disease.
- Understand the line of sight from variants to function in human physiology.
- Create standards and policy [frameworks](#) for genome data representation, storage, analysis, security, and so on.
- Standardize and socialize interoperable genome aggregation databases (gnomAD).
- Genome-based monitoring, diagnosis, and treatment of patients with greater precision.
- Help implement tools that predict diseases well before symptoms appear.
- Create and empower a community of cross-disciplinary collaborators to help tackle the toughest and most important problems in biomedicine.

According to GATK and the Broad institute, genome sequencing should be treated as a protocol in a pathology lab; every task is well documented, optimized, reproducible, and consistent across samples and experiments. The following is a set of steps recommended by the Broad Institute, for more information, see the [GATK website](#).

## FlexPod setup

Genomics workload validation includes a from-scratch setup of a FlexPod infrastructure platform. The FlexPod platform is highly available and can be independently scaled; for example, network, storage and compute can be scaled independently. We used the following Cisco validated design guide as the reference architecture document to set up the FlexPod environment: [FlexPod Datacenter with VMware vSphere 7.0 and NetApp ONTAP 9.7](#). See the following FlexPod platform set up highlights:

To perform FlexPod lab setup, complete the following steps:

1. FlexPod lab setup and validation uses the following IP4 reservations and VLANs.

## IP Reservations

VLAN	IP Range	Subnet Mask	Purpose
3281	172.21.25 /24	255.255.255.0	IB-MGMT
3282	172.21.26 /24	255.255.255.0	vMotion
3283	172.21.27 /24	255.255.255.0	VM
3284	172.21.28 /24	255.255.255.0	NFS
3285	172.21.29 /24	255.255.255.0	iSCSI-A
3286	172.21.30 /24	255.255.255.0	iSCSI-B

2. Configure iSCSI-based boot LUNs on the ONTAP SVM.

**ONTAP System Manager**

**LUNs**

[+ Add](#)

	Name	Storage VM
▼	ESXi_Boot_Lun_1	Healthcare_SVM
▼	ESXi_Boot_Lun_2	Healthcare_SVM
▼	ESXi_Boot_Lun_3	Healthcare_SVM
▼	ESXi_Boot_Lun_4	Healthcare_SVM
▼	ESXi_Boot_Lun_5	Healthcare_SVM
▼	ESXi_Boot_Lun_6	Healthcare_SVM

3. Map LUNs to iSCSI initiator groups.

	Name	Storage VM	Volume	Size	IOPS	Latency (ms)	Throughput (MB/s)
▲	ESXi_Boot_Lun_1	Healthcare_SVM	ESXi_Boot_Vol	20 GB	3	0.16	0.01

STATUS	VOLUME	DESCRIPTION	SNAPSHOT COPIES (LOCAL)	SNAPMIRROR (LOCAL OR REMOTE)
Online	ESXi_Boot_Vol	-	Protected	Unprotected

SERIAL NUMBER	QOS POLICY GROUP	MAPPED TO INITIATORS	ID
80A4X+R8rAhP	-	GenomicsESXi_1 (1) iqn.1992-08.com.cisco:ucs-...	0

CAPACITY (AVAILABLE %   TOTAL)	LUN FORMAT
95%   20 GB	VMware

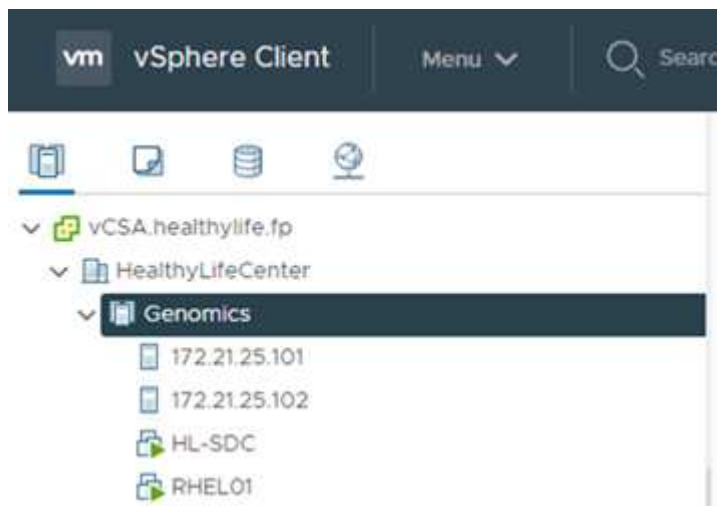
PATH  
/vol/ESXi\_Boot\_Vol/ESXi\_Boot\_Lun\_1

	Name	Storage VM	Volume	Size	IOPS	Latency (ms)	Throughput (MB/s)
▼	ESXi_Boot_Lun_1	Healthcare_SVM	ESXi_Boot_Vol	20 GB	1	0.25	0.01
▲	ESXi_Boot_Lun_2	Healthcare_SVM	ESXi_Boot_Vol	20 GB	4	0.18	0.02

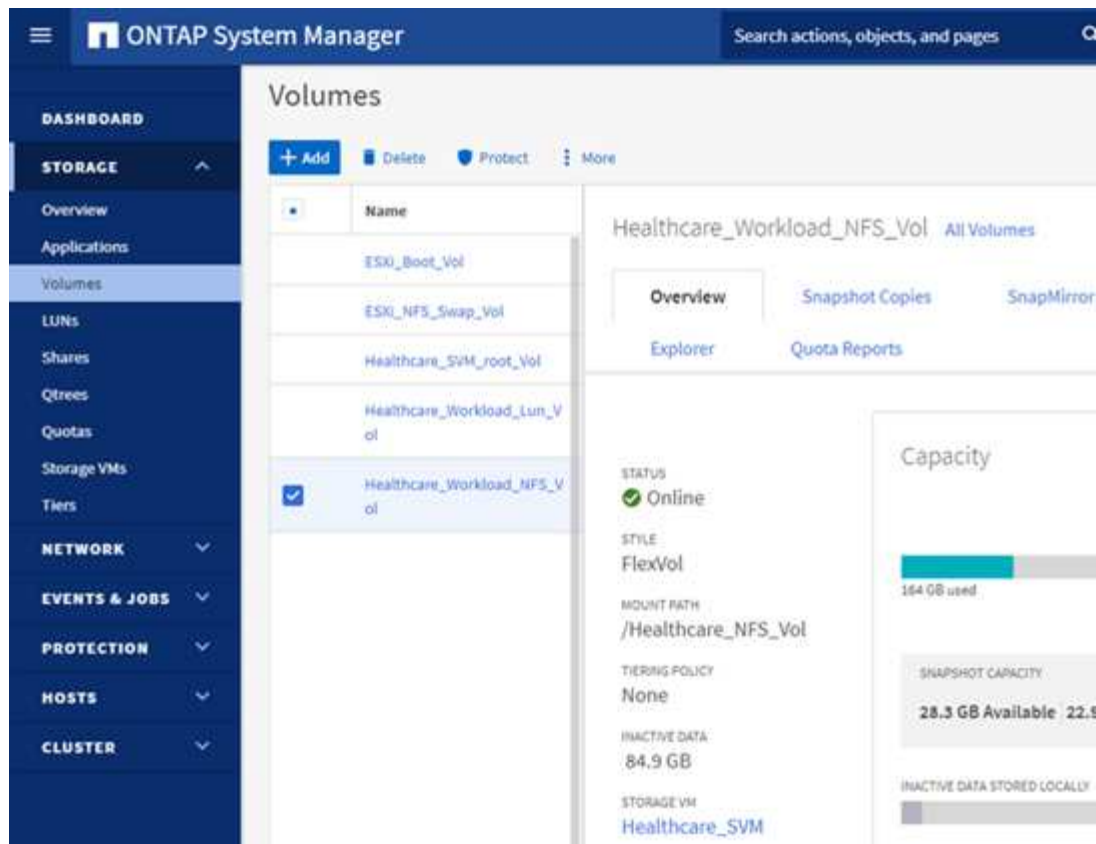
  

STATUS ✔ Online	VOLUME ESXi_Boot_Vol	DESCRIPTION -	SNAPSHOT COPIES (LOCAL) STATUS ✔ Protected	SNAPMIRROR (LOCAL OR REMOTE) STATUS ✔ Unprotected
SERIAL NUMBER 80A4X+R8rAhU	QOS POLICY GROUP -	MAPPED TO INITIATORS ^ GenomicsESXi_2 (1) iqn.1992-08.com.cisco:ucs-...	ID 0	SNAPSHOT POLICY default
CAPACITY (AVAILABLE %   TOTAL) 96%   20 GB	LUN FORMAT VMware			

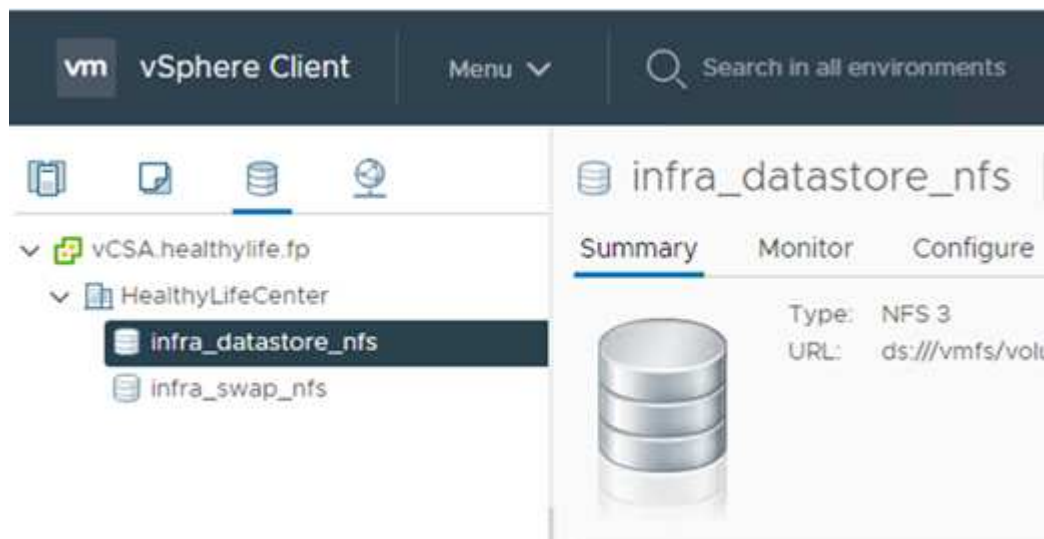
4. Install vSphere 7.0 with iSCSI boot.
5. Register ESXi hosts with the vCenter.



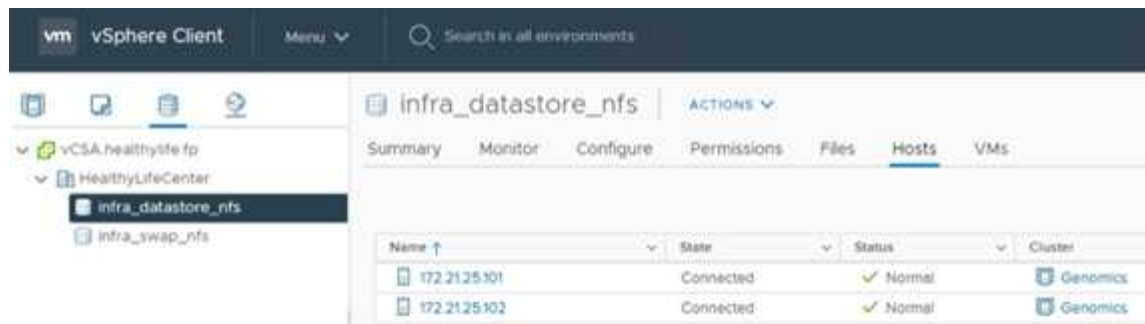
6. Provision an NFS datastore infra\_datastore\_nfs on the ONTAP storage.



7. Add the datastore to the vCenter.



8. Using vCenter, add an NFS datastore to the ESXi hosts.



9. Using the vCenter, create a Red Hat Enterprise Linux (RHEL) 8.3 VM to run GATK.
10. An NFS datastore is presented to the VM and mounted at `/mnt/genomics`, which is used to store GATK executables, scripts, Binary Alignment Map (BAM) files, reference files, index files, dictionary files, and out files for variant calling.

```
[root@genomics1 genomics]# df | grep genomics
/dev/sdb          308587328  5699492 287142812   2% /mnt/genomics
[root@genomics1 genomics]#
```

## GATK setup and execution

Install the following prerequisites on the RedHat Enterprise 8.3 Linux VM:

- Java 8 or SDK 1.8 or later
- Download GATK 4.2.0.0 from the Broad Institute [GitHub site](#). Genome sequence data is generally stored in the form of a series of tab-delimited ASCII columns. However, ASCII takes too much space to store. Therefore, a new standard evolved called a BAM (\*.bam) file. A BAM file stores the sequence data in a compressed, indexed, and binary form. We [downloaded](#) a set of publicly available BAM files for GATK execution from the [public domain](#). We also downloaded index files (\*.bai), dictionary files (\*.dict), and reference data files (\*.fasta) from the same public domain.

After downloading, the GATK tool kit has a jar file and a set of support scripts.

- `gatk-package-4.2.0.0-local.jar` executable
- `gatk` script file.

We downloaded the BAM files and the corresponding index, dictionary, and reference genome files for a family that consisted of father, mother, and son \*.bam files.

## Cromwell engine

Cromwell is an open-source engine geared towards scientific workflows that enables workflow management. The Cromwell engine can be run in two [modes](#), Server mode or a single- workflow Run mode. The behavior of the Cromwell engine can be controlled using the [Cromwell engine configuration file](#).

- **Server mode.** Enables [RESTful](#) execution of workflows in Cromwell engine.
- **Run mode.** Run mode is best suited for executing single workflows in Cromwell, [ref](#) for a complete set of available options in Run mode.

We use the Cromwell engine to execute the workflows and pipelines at scale. The Cromwell engine uses a user-friendly [workflow description language](#) (WDL)-based scripting language. Cromwell also supports a second

workflow scripting standard called the common workflow language (CWL). Throughout this technical report, we used WDL. WDL was originally developed by the Broad Institute for genome analysis pipelines. Using the WDL workflows can be implemented using several strategies, including the following:

- **Linear chaining.** As the name suggests, output from task#1 is sent to task #2 as input.
- **Multi-in/out.** This is similar to linear chaining in that each task can have multiple outputs being sent as input to subsequent tasks.
- **Scatter-gather.** This is one of the most powerful enterprise application integration (EAI) strategies available, especially when used in event-driven architecture. Each task executes in a decoupled fashion, and the output for each task is consolidated into the final output.

There are three steps when WDL is used to run GATK in a standalone mode:

1. Validate syntax using `womtool.jar`.

```
[root@genomics1 ~]# java -jar womtool.jar validate ghplo.wdl
```

2. Generate inputs JSON.

```
[root@genomics1 ~]# java -jar womtool.jar inputs ghplo.wdl > ghplo.json
```

3. Run the workflow using the Cromwell engine and `Cromwell.jar`.

```
[root@genomics1 ~]# java -jar cromwell.jar run ghplo.wdl --inputs  
ghplo.json
```

The GATK can be executed by using several methods; this document explores three of these methods.

### Execution of GATK using the jar file

Let's look at a single variant call pipeline execution using the Haplotype variant caller.

```
[root@genomics1 ~]# java -Dsamjdk.use_async_io_read_samtools=false \  
-Dsamjdk.use_async_io_write_samtools=true \  
-Dsamjdk.use_async_io_write_tribble=false \  
-Dsamjdk.compression_level=2 \  
-jar /mnt/genomics/GATK/gatk-4.2.0.0/gatk-package-4.2.0.0-local.jar \  
HaplotypeCaller \  
--input /mnt/genomics/GATK/TEST\ DATA/bam/workshop_1906_2-  
germline_bams_father.bam \  
--output workshop_1906_2-germline_bams_father.validation.vcf \  
--reference /mnt/genomics/GATK/TEST\ DATA/ref/workshop_1906_2-  
germline_ref_ref.fasta
```



In this method of execution, we use the GATK local execution jar file, we use a single java command to invoke the jar file, and we pass several parameters to the command.

1. This parameter indicates that we are invoking the HaplotypeCaller variant caller pipeline.
2. `--input` specifies the input BAM file.
3. `--output` specifies the variant output file in variant call format (\*.vcf) ([ref](#)).
4. With the `--reference` parameter, we are passing a reference genome.

Once executed, output details can be found in the section ["Output for execution of GATK using the jar file."](#)

### Execution of GATK using `./gatk` script

GATK tool kit can be executed using the `./gatk` script. Let's examine the following command:

```
[root@genomics1 execution]# ./gatk \
--java-options "-Xmx4G" \
HaplotypeCaller \
-I /mnt/genomics/GATK/TEST\ DATA/bam/workshop_1906_2-
germline_bams_father.bam \
-R /mnt/genomics/GATK/TEST\ DATA/ref/workshop_1906_2-
germline_ref_ref.fasta \
-O /mnt/genomics/GATK/TEST\ DATA/variants.vcf
```

We pass several parameters to the command.

- This parameter indicates that we are invoking the HaplotypeCaller variant caller pipeline.
- `-I` specifies the input BAM file.
- `-O` specifies the variant output file in variant call format (\*.vcf) ([ref](#)).
- With the `-R` parameter, we are passing a reference genome.

Once executed, output details can be found in the section ["Output for execution of GATK using the `./gatk` script."](#)

### Execution of GATK using Cromwell engine

We use the Cromwell engine to manage GATK execution. Let's examine the command line and its parameters.

```
[root@genomics1 genomics]# java -jar cromwell-65.jar \
run /mnt/genomics/GATK/seq/ghplo.wdl \
--inputs /mnt/genomics/GATK/seq/ghplo.json
```

Here, we invoke the Java command by passing the `-jar` parameter to indicate that we intend to execute a jar file, for example, `Cromwell-65.jar`. The next parameter passed (`run`) indicates that the Cromwell engine is running in Run mode, the other possible option is Server mode. The next parameter is `*.wdl` that the Run

mode should use to execute the pipelines. The next parameter is the set of input parameters to the workflows being executed.

Here's what the contents of the `ghplo.wdl` file look like:

```
[root@genomics1 seq]# cat ghplo.wdl
workflow helloHaplotypeCaller {
  call haplotypeCaller
}
task haplotypeCaller {
  File GATK
  File RefFasta
  File RefIndex
  File RefDict
  String sampleName
  File inputBAM
  File bamIndex
  command {
    java -jar ${GATK} \
      HaplotypeCaller \
      -R ${RefFasta} \
      -I ${inputBAM} \
      -O ${sampleName}.raw.indels.snps.vcf
  }
  output {
    File rawVCF = "${sampleName}.raw.indels.snps.vcf"
  }
}
[root@genomics1 seq]#
```

Here's the corresponding JSON file with the inputs to the Cromwell engine.

```
[root@genomics1 seq]# cat ghplo.json
{
  "helloHaplotypeCaller.haplotypeCaller.GATK": "/mnt/genomics/GATK/gatk-4.2.0.0/gatk-package-4.2.0.0-local.jar",
  "helloHaplotypeCaller.haplotypeCaller.RefFasta": "/mnt/genomics/GATK/TEST DATA/ref/workshop_1906_2-germline_ref_ref.fasta",
  "helloHaplotypeCaller.haplotypeCaller.RefIndex": "/mnt/genomics/GATK/TEST DATA/ref/workshop_1906_2-germline_ref_ref.fasta.fai",
  "helloHaplotypeCaller.haplotypeCaller.RefDict": "/mnt/genomics/GATK/TEST DATA/ref/workshop_1906_2-germline_ref_ref.dict",
  "helloHaplotypeCaller.haplotypeCaller.sampleName": "fatherbam",
  "helloHaplotypeCaller.haplotypeCaller.inputBAM": "/mnt/genomics/GATK/TEST DATA/bam/workshop_1906_2-germline_bams_father.bam",
  "helloHaplotypeCaller.haplotypeCaller.bamIndex": "/mnt/genomics/GATK/TEST DATA/bam/workshop_1906_2-germline_bams_father.bai"
}
[root@genomics1 seq]#
```

Please note that Cromwell uses an in-memory database for the execution. Once executed, the output log can be seen in the section ["Output for execution of GATK using the Cromwell engine."](#)

For a comprehensive set of steps on how to execute GATK, see the [GATK documentation](#).

Next: [Output for execution of GATK using the jar file](#).

## Output for execution of GATK using the jar file

Previous: [Genomics - GATK setup and execution](#).

Execution of GATK using the jar file produced the following sample output.

```
[root@genomics1 execution]# java -Dsamjdk.use_async_io_read_samtools=false \
-Dsamjdk.use_async_io_write_samtools=true \
-Dsamjdk.use_async_io_write_tribble=false \
-Dsamjdk.compression_level=2 \
-jar /mnt/genomics/GATK/gatk-4.2.0.0/gatk-package-4.2.0.0-local.jar \
HaplotypeCaller \
--input /mnt/genomics/GATK/TEST\ DATA/bam/workshop_1906_2-germline_bams_father.bam \
--output workshop_1906_2-germline_bams_father.validation.vcf \
--reference /mnt/genomics/GATK/TEST\ DATA/ref/workshop_1906_2-germline_ref_ref.fasta \
22:52:58.430 INFO NativeLibraryLoader - Loading libgkl_compression.so
from jar:file:/mnt/genomics/GATK/gatk-4.2.0.0/gatk-package-4.2.0.0-local.jar!/com/intel/gkl/native/libgkl_compression.so
```

```

Aug 17, 2021 10:52:58 PM
shaded.cloud_nio.com.google.auth.oauth2.ComputeEngineCredentials
runningOnComputeEngine
INFO: Failed to detect whether we are running on Google Compute Engine.
22:52:58.541 INFO HaplotypeCaller -
-----
22:52:58.542 INFO HaplotypeCaller - The Genome Analysis Toolkit (GATK)
v4.2.0.0
22:52:58.542 INFO HaplotypeCaller - For support and documentation go to
https://software.broadinstitute.org/gatk/
22:52:58.542 INFO HaplotypeCaller - Executing as
root@genomics1.healthylife.fp on Linux v4.18.0-305.3.1.el8_4.x86_64 amd64
22:52:58.542 INFO HaplotypeCaller - Java runtime: OpenJDK 64-Bit Server
VM v1.8.0_302-b08
22:52:58.542 INFO HaplotypeCaller - Start Date/Time: August 17, 2021
10:52:58 PM EDT
22:52:58.542 INFO HaplotypeCaller -
-----
22:52:58.542 INFO HaplotypeCaller -
-----
22:52:58.542 INFO HaplotypeCaller - HTSJDK Version: 2.24.0
22:52:58.542 INFO HaplotypeCaller - Picard Version: 2.25.0
22:52:58.542 INFO HaplotypeCaller - Built for Spark Version: 2.4.5
22:52:58.542 INFO HaplotypeCaller - HTSJDK Defaults.COMPRESSION_LEVEL : 2
22:52:58.543 INFO HaplotypeCaller - HTSJDK
Defaults.USE_ASYNC_IO_READ_FOR_SAMTOOLS : false
22:52:58.543 INFO HaplotypeCaller - HTSJDK
Defaults.USE_ASYNC_IO_WRITE_FOR_SAMTOOLS : true
22:52:58.543 INFO HaplotypeCaller - HTSJDK
Defaults.USE_ASYNC_IO_WRITE_FOR_TRIBBLE : false
22:52:58.543 INFO HaplotypeCaller - Deflater: IntelDeflater
22:52:58.543 INFO HaplotypeCaller - Inflater: IntelInflater
22:52:58.543 INFO HaplotypeCaller - GCS max retries/reopens: 20
22:52:58.543 INFO HaplotypeCaller - Requester pays: disabled
22:52:58.543 INFO HaplotypeCaller - Initializing engine
22:52:58.804 INFO HaplotypeCaller - Done initializing engine
22:52:58.809 INFO HaplotypeCallerEngine - Disabling physical phasing,
which is supported only for reference-model confidence output
22:52:58.820 INFO NativeLibraryLoader - Loading libgkl_utils.so from
jar:file:/mnt/genomics/GATK/gatk-4.2.0.0/gatk-package-4.2.0.0-
local.jar!/com/intel/gkl/native/libgkl_utils.so
22:52:58.821 INFO NativeLibraryLoader - Loading libgkl_pairhmm_omp.so
from jar:file:/mnt/genomics/GATK/gatk-4.2.0.0/gatk-package-4.2.0.0-
local.jar!/com/intel/gkl/native/libgkl_pairhmm_omp.so
22:52:58.854 INFO IntelPairHmm - Using CPU-supported AVX-512 instructions
22:52:58.854 INFO IntelPairHmm - Flush-to-zero (FTZ) is enabled when

```

```

running PairHMM
22:52:58.854 INFO   IntelPairHmm - Available threads: 16
22:52:58.854 INFO   IntelPairHmm - Requested threads: 4
22:52:58.854 INFO   PairHMM - Using the OpenMP multi-threaded AVX-
accelerated native PairHMM implementation
22:52:58.872 INFO   ProgressMeter - Starting traversal
22:52:58.873 INFO   ProgressMeter -           Current Locus   Elapsed Minutes
Regions Processed   Regions/Minute
22:53:00.733 WARN   InbreedingCoeff - InbreedingCoeff will not be
calculated at position 20:9999900 and possibly subsequent; at least 10
samples must have called genotypes
22:53:08.873 INFO   ProgressMeter -           20:17538652           0.2
58900              353400.0
22:53:17.681 INFO   HaplotypeCaller - 405 read(s) filtered by:
MappingQualityReadFilter
0 read(s) filtered by: MappingQualityAvailableReadFilter
0 read(s) filtered by: MappedReadFilter
0 read(s) filtered by: NotSecondaryAlignmentReadFilter
6628 read(s) filtered by: NotDuplicateReadFilter
0 read(s) filtered by: PassesVendorQualityCheckReadFilter
0 read(s) filtered by: NonZeroReferenceLengthAlignmentReadFilter
0 read(s) filtered by: GoodCigarReadFilter
0 read(s) filtered by: WellformedReadFilter
7033 total reads filtered
22:53:17.681 INFO   ProgressMeter -           20:63024652           0.3
210522             671592.9
22:53:17.681 INFO   ProgressMeter - Traversal complete. Processed 210522
total regions in 0.3 minutes.
22:53:17.687 INFO   VectorLoglessPairHMM - Time spent in setup for JNI call
: 0.010347438
22:53:17.687 INFO   PairHMM - Total compute time in PairHMM
computeLogLikelihoods() : 0.259172573
22:53:17.687 INFO   SmithWatermanAligner - Total compute time in java
Smith-Waterman : 1.27 sec
22:53:17.687 INFO   HaplotypeCaller - Shutting down engine
[August 17, 2021 10:53:17 PM EDT]
org.broadinstitute.hellbender.tools.walkers.haplotypecaller.HaplotypeCalle
r done. Elapsed time: 0.32 minutes.
Runtime.totalMemory()=5561122816
[root@genomics1 execution]#

```

Notice that the output file is located at the location specified after the execution.

Next: [Output for execution of GATK using the ./gatk script.](#)

# Output for execution of GATK using the `./gatk` script

Previous: [Output for execution of GATK using the jar file.](#)

The execution of GATK using the `./gatk` script produced the following sample output.

```
[root@genomics1 gatk-4.2.0.0]# ./gatk --java-options "-Xmx4G" \
HaplotypeCaller \
-I /mnt/genomics/GATK/TEST\ DATA/bam/workshop_1906_2-
germline_bams_father.bam \
-R /mnt/genomics/GATK/TEST\ DATA/ref/workshop_1906_2-
germline_ref_ref.fasta \
-O /mnt/genomics/GATK/TEST\ DATA/variants.vcf
Using GATK jar /mnt/genomics/GATK/gatk-4.2.0.0/gatk-package-4.2.0.0-
local.jar
Running:
    java -Dsamjdk.use_async_io_read_samtools=false
-Dsamjdk.use_async_io_write_samtools=true
-Dsamjdk.use_async_io_write_tribble=false -Dsamjdk.compression_level=2
-Xmx4G -jar /mnt/genomics/GATK/gatk-4.2.0.0/gatk-package-4.2.0.0-local.jar
HaplotypeCaller -I /mnt/genomics/GATK/TEST DATA/bam/workshop_1906_2-
germline_bams_father.bam -R /mnt/genomics/GATK/TEST
DATA/ref/workshop_1906_2-germline_ref_ref.fasta -O /mnt/genomics/GATK/TEST
DATA/variants.vcf
23:29:45.553 INFO  NativeLibraryLoader - Loading libgkl_compression.so
from jar:file:/mnt/genomics/GATK/gatk-4.2.0.0/gatk-package-4.2.0.0-
local.jar!/com/intel/gkl/native/libgkl_compression.so
Aug 17, 2021 11:29:45 PM
shaded.cloud_nio.com.google.auth.oauth2.ComputeEngineCredentials
runningOnComputeEngine
INFO: Failed to detect whether we are running on Google Compute Engine.
23:29:45.686 INFO  HaplotypeCaller -
-----
23:29:45.686 INFO  HaplotypeCaller - The Genome Analysis Toolkit (GATK)
v4.2.0.0
23:29:45.686 INFO  HaplotypeCaller - For support and documentation go to
https://software.broadinstitute.org/gatk/
23:29:45.687 INFO  HaplotypeCaller - Executing as
root@genomics1.healthyliife.fp on Linux v4.18.0-305.3.1.el8_4.x86_64 amd64
23:29:45.687 INFO  HaplotypeCaller - Java runtime: OpenJDK 64-Bit Server
VM v11.0.12+7-LTS
23:29:45.687 INFO  HaplotypeCaller - Start Date/Time: August 17, 2021 at
11:29:45 PM EDT
23:29:45.687 INFO  HaplotypeCaller -
-----
23:29:45.687 INFO  HaplotypeCaller -
```



```

-----
23:29:45.687 INFO HaplotypeCaller - HTSJDK Version: 2.24.0
23:29:45.687 INFO HaplotypeCaller - Picard Version: 2.25.0
23:29:45.687 INFO HaplotypeCaller - Built for Spark Version: 2.4.5
23:29:45.688 INFO HaplotypeCaller - HTSJDK Defaults.COMPRESSION_LEVEL : 2
23:29:45.688 INFO HaplotypeCaller - HTSJDK
Defaults.USE_ASYNC_IO_READ_FOR_SAMTOOLS : false
23:29:45.688 INFO HaplotypeCaller - HTSJDK
Defaults.USE_ASYNC_IO_WRITE_FOR_SAMTOOLS : true
23:29:45.688 INFO HaplotypeCaller - HTSJDK
Defaults.USE_ASYNC_IO_WRITE_FOR_TRIBBLE : false
23:29:45.688 INFO HaplotypeCaller - Deflater: IntelDeflater
23:29:45.688 INFO HaplotypeCaller - Inflater: IntelInflater
23:29:45.688 INFO HaplotypeCaller - GCS max retries/reopens: 20
23:29:45.688 INFO HaplotypeCaller - Requester pays: disabled
23:29:45.688 INFO HaplotypeCaller - Initializing engine
23:29:45.804 INFO HaplotypeCaller - Done initializing engine
23:29:45.809 INFO HaplotypeCallerEngine - Disabling physical phasing,
which is supported only for reference-model confidence output
23:29:45.818 INFO NativeLibraryLoader - Loading libgkl_utils.so from
jar:file:/mnt/genomics/GATK/gatk-4.2.0.0/gatk-package-4.2.0.0-
local.jar!/com/intel/gkl/native/libgkl_utils.so
23:29:45.819 INFO NativeLibraryLoader - Loading libgkl_pairhmm_omp.so
from jar:file:/mnt/genomics/GATK/gatk-4.2.0.0/gatk-package-4.2.0.0-
local.jar!/com/intel/gkl/native/libgkl_pairhmm_omp.so
23:29:45.852 INFO IntelPairHmm - Using CPU-supported AVX-512 instructions
23:29:45.852 INFO IntelPairHmm - Flush-to-zero (FTZ) is enabled when
running PairHMM
23:29:45.852 INFO IntelPairHmm - Available threads: 16
23:29:45.852 INFO IntelPairHmm - Requested threads: 4
23:29:45.852 INFO PairHMM - Using the OpenMP multi-threaded AVX-
accelerated native PairHMM implementation
23:29:45.868 INFO ProgressMeter - Starting traversal
23:29:45.868 INFO ProgressMeter -          Current Locus  Elapsed Minutes
Regions Processed  Regions/Minute
23:29:47.772 WARN InbreedingCoeff - InbreedingCoeff will not be
calculated at position 20:9999900 and possibly subsequent; at least 10
samples must have called genotypes
23:29:55.868 INFO ProgressMeter -          20:18885652          0.2
63390          380340.0
23:30:04.389 INFO HaplotypeCaller - 405 read(s) filtered by:
MappingQualityReadFilter
0 read(s) filtered by: MappingQualityAvailableReadFilter
0 read(s) filtered by: MappedReadFilter
0 read(s) filtered by: NotSecondaryAlignmentReadFilter
6628 read(s) filtered by: NotDuplicateReadFilter

```

```

0 read(s) filtered by: PassesVendorQualityCheckReadFilter
0 read(s) filtered by: NonZeroReferenceLengthAlignmentReadFilter
0 read(s) filtered by: GoodCigarReadFilter
0 read(s) filtered by: WellformedReadFilter
7033 total reads filtered
23:30:04.389 INFO ProgressMeter - 20:63024652 0.3
210522 681999.9
23:30:04.389 INFO ProgressMeter - Traversal complete. Processed 210522
total regions in 0.3 minutes.
23:30:04.395 INFO VectorLoglessPairHMM - Time spent in setup for JNI call
: 0.012129203000000002
23:30:04.395 INFO PairHMM - Total compute time in PairHMM
computeLogLikelihoods() : 0.267345217
23:30:04.395 INFO SmithWatermanAligner - Total compute time in java
Smith-Waterman : 1.23 sec
23:30:04.395 INFO HaplotypeCaller - Shutting down engine
[August 17, 2021 at 11:30:04 PM EDT]
org.broadinstitute.hellbender.tools.walkers.haplotypecaller.HaplotypeCalle
r done. Elapsed time: 0.31 minutes.
Runtime.totalMemory()=2111832064
[root@genomics1 gatk-4.2.0.0]#

```

Notice that the output file is located at the location specified after the execution.

[Next: Output for execution of GATK using the Cromwell engine.](#)

## Output for execution of GATK using the Cromwell engine

[Previous: Output for execution of GATK using the ./gatk script.](#)

The execution of GATK using the Cromwell engine produced the following sample output.

```

[root@genomics1 genomics]# java -jar cromwell-65.jar run
/mnt/genomics/GATK/seq/ghplo.wdl --inputs
/mnt/genomics/GATK/seq/ghplo.json
[2021-08-18 17:10:50,78] [info] Running with database db.url =
jdbc:hsqldb:mem:856a1f0d-9a0d-42e5-9199-
5e6c1d0f72dd;shutdown=false;hsqldb.tx=mvcc
[2021-08-18 17:10:57,74] [info] Running migration
RenameWorkflowOptionsInMetadata with a read batch size of 100000 and a
write batch size of 100000
[2021-08-18 17:10:57,75] [info] [RenameWorkflowOptionsInMetadata] 100%
[2021-08-18 17:10:57,83] [info] Running with database db.url =
jdbc:hsqldb:mem:6afe0252-2dc9-4e57-8674-
ce63c67aa142;shutdown=false;hsqldb.tx=mvcc
[2021-08-18 17:10:58,17] [info] Slf4jLogger started

```

```

[2021-08-18 17:10:58,33] [info] Workflow heartbeat configuration:
{
  "cromwellId" : "cromid-41b7e30",
  "heartbeatInterval" : "2 minutes",
  "ttl" : "10 minutes",
  "failureShutdownDuration" : "5 minutes",
  "writeBatchSize" : 10000,
  "writeThreshold" : 10000
}
[2021-08-18 17:10:58,38] [info] Metadata summary refreshing every 1
second.
[2021-08-18 17:10:58,38] [info] No metadata archiver defined in config
[2021-08-18 17:10:58,38] [info] No metadata deleter defined in config
[2021-08-18 17:10:58,40] [info] KvWriteActor configured to flush with
batch size 200 and process rate 5 seconds.
[2021-08-18 17:10:58,40] [info] WriteMetadataActor configured to flush
with batch size 200 and process rate 5 seconds.
[2021-08-18 17:10:58,44] [info] CallCacheWriteActor configured to flush
with batch size 100 and process rate 3 seconds.
[2021-08-18 17:10:58,44] [warn] 'docker.hash-lookup.gcr-api-queries-per-
100-seconds' is being deprecated, use 'docker.hash-lookup.gcr.throttle'
instead (see reference.conf)
[2021-08-18 17:10:58,54] [info] JobExecutionTokenDispenser - Distribution
rate: 50 per 1 seconds.
[2021-08-18 17:10:58,58] [info] SingleWorkflowRunnerActor: Version 65
[2021-08-18 17:10:58,58] [info] SingleWorkflowRunnerActor: Submitting
workflow
[2021-08-18 17:10:58,64] [info] Unspecified type (Unspecified version)
workflow 3e246147-b1a9-41dc-8679-319f81b7701e submitted
[2021-08-18 17:10:58,66] [info] SingleWorkflowRunnerActor: Workflow
submitted 3e246147-b1a9-41dc-8679-319f81b7701e
[2021-08-18 17:10:58,66] [info] 1 new workflows fetched by cromid-41b7e30:
3e246147-b1a9-41dc-8679-319f81b7701e
[2021-08-18 17:10:58,67] [info] WorkflowManagerActor: Starting workflow
3e246147-b1a9-41dc-8679-319f81b7701e
[2021-08-18 17:10:58,68] [info] WorkflowManagerActor: Successfully started
WorkflowActor-3e246147-b1a9-41dc-8679-319f81b7701e
[2021-08-18 17:10:58,68] [info] Retrieved 1 workflows from the
WorkflowStoreActor
[2021-08-18 17:10:58,70] [info] WorkflowStoreHeartbeatWriteActor
configured to flush with batch size 10000 and process rate 2 minutes.
[2021-08-18 17:10:58,76] [info] MaterializeWorkflowDescriptorActor
[3e246147]: Parsing workflow as WDL draft-2
[2021-08-18 17:10:59,34] [info] MaterializeWorkflowDescriptorActor
[3e246147]: Call-to-Backend assignments:
helloHaplotypeCaller.haplotypeCaller -> Local

```

```

[2021-08-18 17:11:00,54] [info] WorkflowExecutionActor-3e246147-b1a9-41dc-8679-319f81b7701e [3e246147]: Starting
helloHaplotypeCaller.haplotypeCaller
[2021-08-18 17:11:01,56] [info] Assigned new job execution tokens to the
following groups: 3e246147: 1
[2021-08-18 17:11:01,70] [info] BackgroundConfigAsyncJobExecutionActor
[3e246147helloHaplotypeCaller.haplotypeCaller:NA:1]: java -jar
/mnt/genomics/cromwell-executions/helloHaplotypeCaller/3e246147-b1a9-41dc-8679-319f81b7701e/call-haplotypeCaller/inputs/-179397211/gatk-package-4.2.0.0-local.jar \
    HaplotypeCaller \
    -R /mnt/genomics/cromwell-executions/helloHaplotypeCaller/3e246147-b1a9-41dc-8679-319f81b7701e/call-haplotypeCaller/inputs/604632695/workshop_1906_2-germline_ref_ref.fasta \
    -I /mnt/genomics/cromwell-executions/helloHaplotypeCaller/3e246147-b1a9-41dc-8679-319f81b7701e/call-haplotypeCaller/inputs/604617202/workshop_1906_2-germline_bams_father.bam \
    -O fatherbam.raw.indels.snps.vcf
[2021-08-18 17:11:01,72] [info] BackgroundConfigAsyncJobExecutionActor
[3e246147helloHaplotypeCaller.haplotypeCaller:NA:1]: executing: /bin/bash
/mnt/genomics/cromwell-executions/helloHaplotypeCaller/3e246147-b1a9-41dc-8679-319f81b7701e/call-haplotypeCaller/execution/script
[2021-08-18 17:11:03,49] [info] BackgroundConfigAsyncJobExecutionActor
[3e246147helloHaplotypeCaller.haplotypeCaller:NA:1]: job id: 26867
[2021-08-18 17:11:03,53] [info] BackgroundConfigAsyncJobExecutionActor
[3e246147helloHaplotypeCaller.haplotypeCaller:NA:1]: Status change from -
to WaitingForReturnCode
[2021-08-18 17:11:03,54] [info] Not triggering log of token queue status.
Effective log interval = None
[2021-08-18 17:11:23,65] [info] BackgroundConfigAsyncJobExecutionActor
[3e246147helloHaplotypeCaller.haplotypeCaller:NA:1]: Status change from
WaitingForReturnCode to Done
[2021-08-18 17:11:25,04] [info] WorkflowExecutionActor-3e246147-b1a9-41dc-8679-319f81b7701e [3e246147]: Workflow helloHaplotypeCaller complete.
Final Outputs:
{
  "helloHaplotypeCaller.haplotypeCaller.rawVCF": "/mnt/genomics/cromwell-executions/helloHaplotypeCaller/3e246147-b1a9-41dc-8679-319f81b7701e/call-haplotypeCaller/execution/fatherbam.raw.indels.snps.vcf"
}
[2021-08-18 17:11:28,43] [info] WorkflowManagerActor: Workflow actor for
3e246147-b1a9-41dc-8679-319f81b7701e completed with status 'Succeeded'.
The workflow will be removed from the workflow store.
[2021-08-18 17:11:32,24] [info] SingleWorkflowRunnerActor workflow
finished with status 'Succeeded'.

```

```

{
  "outputs": {
    "helloHaplotypeCaller.haplotypeCaller.rawVCF":
"/mnt/genomics/cromwell-executions/helloHaplotypeCaller/3e246147-b1a9-
41dc-8679-319f81b7701e/call-
haplotypeCaller/execution/fatherbam.raw.indels.snps.vcf"
  },
  "id": "3e246147-b1a9-41dc-8679-319f81b7701e"
}
[2021-08-18 17:11:33,45] [info] Workflow polling stopped
[2021-08-18 17:11:33,46] [info] 0 workflows released by cromid-41b7e30
[2021-08-18 17:11:33,46] [info] Shutting down WorkflowStoreActor - Timeout
= 5 seconds
[2021-08-18 17:11:33,46] [info] Shutting down WorkflowLogCopyRouter -
Timeout = 5 seconds
[2021-08-18 17:11:33,46] [info] Shutting down JobExecutionTokenDispenser -
Timeout = 5 seconds
[2021-08-18 17:11:33,46] [info] Aborting all running workflows.
[2021-08-18 17:11:33,46] [info] JobExecutionTokenDispenser stopped
[2021-08-18 17:11:33,46] [info] WorkflowStoreActor stopped
[2021-08-18 17:11:33,47] [info] WorkflowLogCopyRouter stopped
[2021-08-18 17:11:33,47] [info] Shutting down WorkflowManagerActor -
Timeout = 3600 seconds
[2021-08-18 17:11:33,47] [info] WorkflowManagerActor: All workflows
finished
[2021-08-18 17:11:33,47] [info] WorkflowManagerActor stopped
[2021-08-18 17:11:33,64] [info] Connection pools shut down
[2021-08-18 17:11:33,64] [info] Shutting down SubWorkflowStoreActor -
Timeout = 1800 seconds
[2021-08-18 17:11:33,64] [info] Shutting down JobStoreActor - Timeout =
1800 seconds
[2021-08-18 17:11:33,64] [info] Shutting down CallCacheWriteActor -
Timeout = 1800 seconds
[2021-08-18 17:11:33,64] [info] SubWorkflowStoreActor stopped
[2021-08-18 17:11:33,64] [info] Shutting down ServiceRegistryActor -
Timeout = 1800 seconds
[2021-08-18 17:11:33,64] [info] Shutting down DockerHashActor - Timeout =
1800 seconds
[2021-08-18 17:11:33,64] [info] Shutting down IoProxy - Timeout = 1800
seconds
[2021-08-18 17:11:33,64] [info] CallCacheWriteActor Shutting down: 0
queued messages to process
[2021-08-18 17:11:33,64] [info] JobStoreActor stopped
[2021-08-18 17:11:33,64] [info] CallCacheWriteActor stopped
[2021-08-18 17:11:33,64] [info] KvWriteActor Shutting down: 0 queued
messages to process

```

```
[2021-08-18 17:11:33,64] [info] IoProxy stopped
[2021-08-18 17:11:33,64] [info] WriteMetadataActor Shutting down: 0 queued
messages to process
[2021-08-18 17:11:33,65] [info] ServiceRegistryActor stopped
[2021-08-18 17:11:33,65] [info] DockerHashActor stopped
[2021-08-18 17:11:33,67] [info] Database closed
[2021-08-18 17:11:33,67] [info] Stream materializer shut down
[2021-08-18 17:11:33,67] [info] WDL HTTP import resolver closed
[root@genomics1 genomics]#
```

[Next: GPU setup.](#)

## GPU setup

[Previous: Output for execution of GATK using the Cromwell engine.](#)

At the time of publication, the GATK tool does not have native support for GPU-based execution on premises. The following setup and guidance is provided to enable the readers understand how simple it is to use FlexPod with a rear-mounted NVIDIA Tesla P6 GPU using a PCIe mezzanine card for GATK.

We used the following Cisco-Validated Design (CVD) as the reference architecture and best-practice guide to set up the FlexPod environment so that we can run applications that use GPUs.

- [FlexPod Datacenter for AI/ML with Cisco UCS 480 ML for Deep Learning](#)

Here is a set of key takeaways during this setup:

1. We used a PCIe NVIDIA Tesla P6 GPU in a mezzanine slot in the UCS B200 M5 servers.

Equipment / Chassis / Chassis 1 / Servers / Server 1

<

General

Inventory

Virtual Machines

Installed Firmware

CIMC Sessions

SEL Logs

VIF Paths

Health >

<

Motherboard

CIMC

CPUs

GPUs

Memory

Adapters

HBAs

NICs

iSCSI vNICs

Security >

⌵ Advanced Filter

⬆ Export

🖨 Print

⚙

Name	ID	Model	Serial	Mode
Graphics Card 2	2	UCSB-GPU-P6-R	FCH212373V7	Compute



Equipment / Chassis / Chassis 1 / Servers / Server 2

< General **Inventory** Virtual Machines Installed Firmware CIMC Sessions SEL Logs VIF Paths Health >

< Motherboard CIMC CPUs **GPUs** Memory Adapters HBAs NICs iSCSI vNICs Security >

Advanced Filter Export Print

Name	ID	Model	Serial	Mode
Graphics Card 2	2	UCSB-GPU-P6-R	FCH212373Y1	Compute

- For this setup, we registered on the NVIDIA partner portal and obtained an evaluation license (also known as an entitlement) to be able to use the GPUs in compute mode.
- We downloaded the NVIDIA vGPU software required from the NVIDIA partner website.
- We downloaded the entitlement \*.bin file from the NVIDIA partner website.
- We installed an NVIDIA vGPU license server and added the entitlements to the license server using the \*.bin file downloaded from the NVIDIA partner site.
- Make sure to choose the correct NVIDIA vGPU software version for your deployment on the NVIDIA partner portal. For this setup we used driver version 460.73.02.
- This command installs the [NVIDIA vGPU Manager](#) in ESXi.

```
[root@localhost:~] esxcli software vib install -v
/vmfs/volumes/infra_datastore_nfs/nvidia/vib/NVIDIA_bootbank_NVIDIA-
VMware_ESXi_7.0_Host_Driver_460.73.02-10EM.700.0.0.15525992.vib
Installation Result
Message: Operation finished successfully.
Reboot Required: false
VIBs Installed: NVIDIA_bootbank_NVIDIA-
VMware_ESXi_7.0_Host_Driver_460.73.02-10EM.700.0.0.15525992
VIBs Removed:
VIBs Skipped:
```

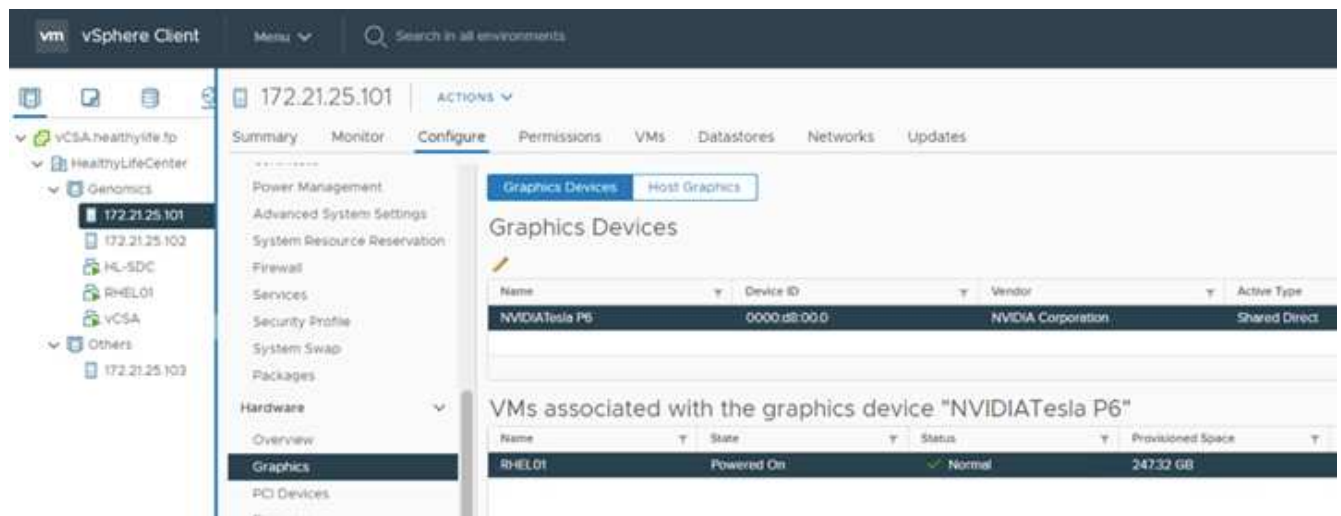
- After rebooting the ESXi server, run the following command to validate the installation and check the health of the GPUs.

```

[root@localhost:~] nvidia-smi
Wed Aug 18 21:37:19 2021
+-----+
+-----+
| NVIDIA-SMI 460.73.02      Driver Version: 460.73.02      CUDA Version: N/A
|
|-----+-----+
+-----+
| GPU  Name           Persistence-M| Bus-Id        Disp.A | Volatile
Uncorr. ECC |
| Fan  Temp  Perf  Pwr:Usage/Cap|      Memory-Usage | GPU-Util
Compute M. |
|
|
MIG M. |
|=====+=====+=====
=====|
|   0  Tesla P6             On   | 00000000:D8:00.0 Off |
0 |
| N/A   35C    P8      9W /  90W | 15208MiB / 15359MiB |      0%
Default |
|
|
N/A |
+-----+-----+
+-----+
+-----+
+-----+
+-----+
| Processes:
|
| GPU    GI    CI          PID    Type    Process name                  GPU
Memory |
|      ID    ID                 |          | Usage
|
|=====+=====+=====
=====|
|   0   N/A   N/A     2812553      C+G     RHEL01
15168MiB |
+-----+-----+
+-----+
[root@localhost:~]

```

9. Using vCenter, [configure](#) the graphics device settings to “Shared Direct.”



10. Make sure that secure boot is disabled for the RedHat VM.
11. Make sure that the VM Boot Options firmware is set to EFI ([ref](#)).

Edit Settings
RHEL01

Virtual Hardware
VM Options

> General Options	VM Name: RHEL01
> VMware Remote Console Options	<input type="checkbox"/> Lock the guest operating system when the last remote user disconnects
> Encryption	Expand for encryption settings
> Power management	Expand for power management settings
> VMware Tools	Expand for VMware Tools settings
> Boot Options	
Firmware	EFI (recommended) ▼
Secure Boot	<input type="checkbox"/> Enabled
Boot Delay	When powering on or resetting, delay boot order by 0 milliseconds
Force EFI setup	<input type="checkbox"/> During the next boot, force entry into the EFI setup screen
Failed Boot Recovery	<input type="checkbox"/> If the VM fails to find boot device, automatically retry after 10 seconds
> Advanced	Expand for advanced settings
> Fibre Channel NPIV	Expand for Fibre Channel NPIV settings

CANCEL
OK

12. Make sure that the following PARAMS are added to the VM Options advanced Edit Configuration. The value of the `pciPassthru.64bitMMIOSizeGB` parameter depends on the GPU memory and number of GPUs assigned to the VM. For example:
  - a. If a VM is assigned 4 x 32GB V100 GPUs, then this value should be 128.
  - b. If a VM is assigned 4 x 16GB P6 GPUs, then this value should be 64.

Edit Settings
RHEL01

> Boot Options
Expand for boot options

Advanced

Settings

☐ Disable acceleration
☒ Enable logging

Debugging and statistics

Run normally

Swap file location

☒ Default

Use the settings of the cluster or host containing the virtual machine.

☐ Virtual machine directory

Store the swap files in the same directory as the virtual machine.

☐ Datastore specified by host

Store the swap files in the datastore specified by the host to be used for swap files. If not possible, store the swap files in the same directory as the virtual machine. Using a datastore that is not visible to both hosts during vMotion might affect the vMotion performance for the affected virtual machines.

Configuration Parameters

EDIT CONFIGURATION...

Latency Sensitivity

Normal

> Fibre Channel NPIV
Expand for Fibre Channel NPIV settings

Configuration Parameters

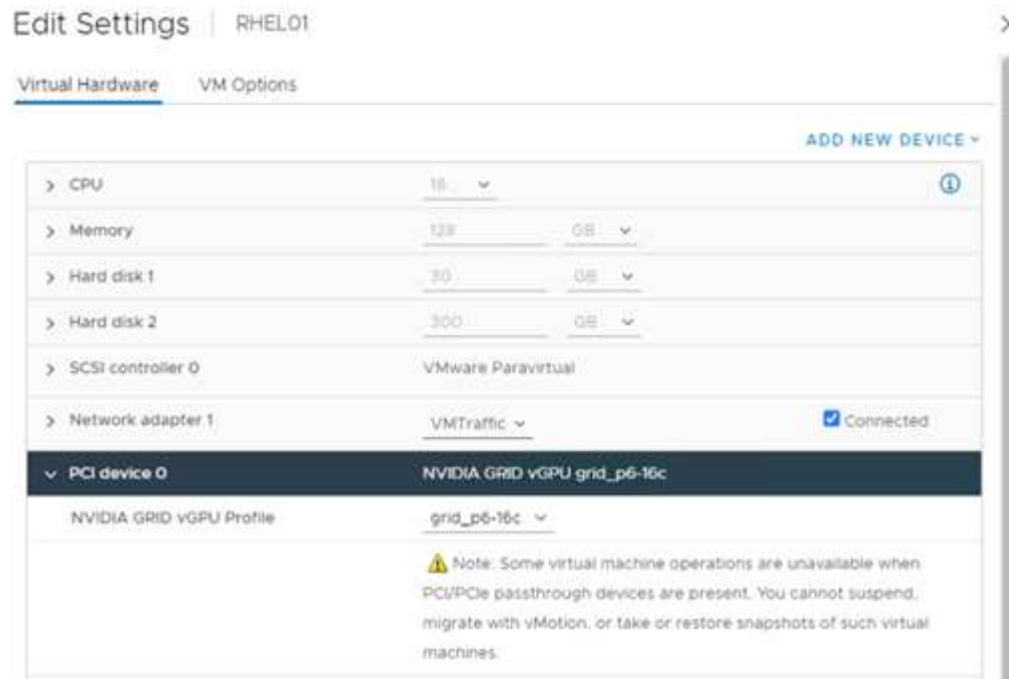
⚠

Modify or add configuration parameters as needed for experimental features or as instructed by technical support. Empty values will be removed (supported on ESXi 6.0 and later).

Name	Value
pciPassthru.64bitMMIOSizeGB	64
pciPassthru.use64bitMMIO	TRUE

- When adding vGPUs as a new PCI Device to the virtual machine in vCenter, make sure to select NVIDIA GRID vGPU as the PCI Device type.
- Choose the correct GPU profile that suites the GPU being used, the GPU memory, and the usage purpose: for example, graphics versus compute.

35



15. On the RedHat Linux VM, NVIDIA drivers can be installed by running the following command:

```
[root@genomics1 genomics]# sh NVIDIA-Linux-x86_64-460.73.01-grid.run
```

16. Verify that the correct vGPU profile is being reported by running the following command:

```
[root@genomics1 genomics]# nvidia-smi -query-gpu=gpu_name  
-format=csv,noheader -id=0 | sed -e 's/ /-/g'  
GRID-P6-16C  
[root@genomics1 genomics]#
```

17. After reboot, verify that the correct NVIDIA vGPU are reported along with the driver versions.

```

[root@genomics1 genomics]# nvidia-smi
Wed Aug 18 20:30:56 2021
+-----+
+-----+
| NVIDIA-SMI 460.73.01      Driver Version: 460.73.01      CUDA Version:
11.2      |
|-----+-----+
+-----+
| GPU  Name                Persistence-M| Bus-Id        Disp.A | Volatile
Uncorr. ECC |
| Fan  Temp  Perf  Pwr:Usage/Cap|      Memory-Usage | GPU-Util
Compute M. |
|
MIG M. |
|=====+=====+=====
=====|
|   0  GRID P6-16C          On    | 00000000:02:02.0 Off |
N/A |
| N/A   N/A    P8      N/A /  N/A |   2205MiB / 16384MiB |      0%
Default |
|
N/A |
+-----+-----+
+-----+
+-----+
+-----+
+-----+
| Processes:
|
| GPU    GI    CI          PID    Type    Process name                  GPU
Memory |
|        ID    ID                                   Usage
|
|=====+=====+=====
=====|
|   0    N/A   N/A         8604      G    /usr/libexec/Xorg
13MiB |
+-----+-----+
+-----+
[root@genomics1 genomics]#

```

18. Make sure that the license server IP is configured on the VM in the vGPU grid configuration file.

a. Copy the template.



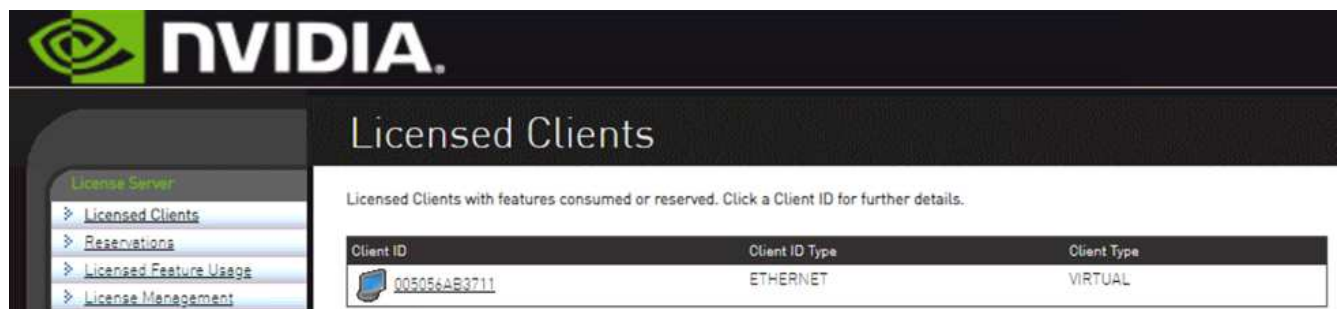
```
[root@genomics1 genomics]# cp /etc/nvidia/gridd.conf.template  
/etc/nvidia/gridd.conf
```

- b. Edit the file `/etc/nvidia/rid.conf`, add the license server IP address, and set the feature type to 1.

```
ServerAddress=192.168.169.10
```

```
FeatureType=1
```

19. After restarting the VM, you should see an entry under Licensed Clients in the license server as shown below.



20. Refer to the Solutions Setup section for more information on downloading the GATK and Cromwell software.
21. After GATK can use GPUs on premises, the workflow description language \*.wdl has the runtime attributes as shown below.

```

task ValidateBAM {
  input {
    # Command parameters
    File input_bam
    String output_basename
    String? validation_mode
    String gatk_path
    # Runtime parameters
    String docker
    Int machine_mem_gb = 4
    Int additional_disk_space_gb = 50
  }
  Int disk_size = ceil(size(input_bam, "GB")) + additional_disk_space_gb
  String output_name = "${output_basename}_${validation_mode}.txt"
  command {
    ${gatk_path} \
      ValidateSamFile \
      --INPUT ${input_bam} \
      --OUTPUT ${output_name} \
      --MODE ${default="SUMMARY" validation_mode}
  }
  runtime {
    gpuCount: 1
    gpuType: "nvidia-tesla-p6"
    docker: docker
    memory: machine_mem_gb + " GB"
    disks: "local-disk " + disk_size + " HDD"
  }
  output {
    File validation_report = "${output_name}"
  }
}

```

[Next: Conclusion.](#)

## Conclusion

[Previous: GPU setup.](#)

Many healthcare organizations around the world have standardized on FlexPod as a common platform. With FlexPod, you can deploy healthcare capabilities with confidence. FlexPod with NetApp ONTAP comes standard with the ability to implement an industry leading set of protocols out of the box. Irrespective of the origin of the request to run genomics of a given patient, interoperability, accessibility, availability, and scalability come standard with a FlexPod platform. When standardized on a FlexPod platform, the culture

of innovation becomes contagious.

## Where to find additional information

To learn more about the information that is described in this document, review the following documents and websites:

- FlexPod Datacenter for AI/ML with Cisco UCS 480 ML for Deep Learning

[https://www.cisco.com/c/en/us/td/docs/unified\\_computing/ucs/UCS\\_CVDs/flexpod\\_480ml\\_aiml\\_deployment.pdf](https://www.cisco.com/c/en/us/td/docs/unified_computing/ucs/UCS_CVDs/flexpod_480ml_aiml_deployment.pdf)

- FlexPod Datacenter with VMware vSphere 7.0 and NetApp ONTAP 9.7

[https://www.cisco.com/c/en/us/td/docs/unified\\_computing/ucs/UCS\\_CVDs/fp\\_vmware\\_vsphere\\_7\\_0\\_ontap\\_9\\_7.html](https://www.cisco.com/c/en/us/td/docs/unified_computing/ucs/UCS_CVDs/fp_vmware_vsphere_7_0_ontap_9_7.html)

- ONTAP 9 Documentation Center

<http://docs.netapp.com>

- Agile and efficient—how FlexPod drives data center modernization

<https://www.flexpod.com/idc-white-paper/>

- AI in healthcare

<https://www.netapp.com/us/media/na-369.pdf>

- FlexPod for healthcare Ease Your Transformation

<https://flexpod.com/solutions/verticals/healthcare/>

- FlexPod from Cisco and NetApp

<https://flexpod.com/>

- AI and Analytics for healthcare (NetApp)

<https://www.netapp.com/us/artificial-intelligence/healthcare-ai-analytics/index.aspx>

- AI in healthcare Smart infrastructure Choices Increase Success

<https://www.netapp.com/pdf.html?item=/media/7410-wp-7314.pdf>

- FlexPod Datacenter with ONTAP 9.8, ONTAP Storage Connector for Cisco Intersight, and Cisco Intersight Managed Mode.

<https://www.netapp.com/pdf.html?item=/media/25001-tr-4883.pdf>

- FlexPod Datacenter with Red Hat Enterprise Linux OpenStack Platform

[https://www.cisco.com/c/en/us/td/docs/unified\\_computing/ucs/UCS\\_CVDs/flexpod\\_openstack\\_osp6.html](https://www.cisco.com/c/en/us/td/docs/unified_computing/ucs/UCS_CVDs/flexpod_openstack_osp6.html)

## Version history

Version	Date	Document version history
Version 1.0	November 2021	Initial release.

## Copyright information

Copyright © 2025 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.

LIMITED RIGHTS LEGEND: Use, duplication, or disclosure by the government is subject to restrictions as set forth in subparagraph (b)(3) of the Rights in Technical Data -Noncommercial Items at DFARS 252.227-7013 (FEB 2014) and FAR 52.227-19 (DEC 2007).

Data contained herein pertains to a commercial product and/or commercial service (as defined in FAR 2.101) and is proprietary to NetApp, Inc. All NetApp technical data and computer software provided under this Agreement is commercial in nature and developed solely at private expense. The U.S. Government has a non-exclusive, non-transferrable, nonsublicensable, 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) (FEB 2014).

## 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.