



適用於基因組學FlexPod FlexPod

NetApp
October 30, 2025

目錄

適用於基因組學FlexPod	1
TR-4911：FlexPod 《基因組學》	1
範圍	2
目標對象	2
部署FlexPod 在NetApp上的醫院功能	2
將基因工作負載部署在FlexPod 架構上的優點	3
解決方案基礎架構硬體與軟體元件	7
基因組學：網關設定與執行	11
從樣本到變體識別、註釋和預測	11
基因組分析工具套件（網關）	13
系統設定FlexPod	13
網關設定與執行	16
使用Jar檔案執行網關的輸出	20
使用./gatk指令碼執行網關的輸出	23
使用Cromwell引擎執行網關的輸出	25
GPU設定	29
結論	38
何處可找到其他資訊	39
版本歷程記錄	40

適用於基因組學FlexPod

TR-4911：FlexPod 《基因組學》

NetApp的JayaKishore Esanakula

在醫療和生命科學方面、很少有領域比基因組學更重要、而基因組學正迅速成為醫生和護士的重要臨床工具。基因組學結合醫療成像和數位化的疾病、有助於我們瞭解病患的基因可能會受到治療規範的影響。基因組學在醫療領域的成功、越來越仰賴大規模的資料互通性。最終目標是瞭解大量的基因資料、並找出臨床相關的關聯和變異、以改善診斷並實現精準醫學。基因組學可協助我們瞭解疾病爆發的來源、疾病演變的方式、以及哪些治療和策略可能有效。顯然、基因組學有許多好處、包括預防、診斷和治療。醫療組織正努力克服多項挑戰、包括：

- 改善照護品質
- 以價值為基礎的照護
- 資料爆炸性增長
- 精密醫學
- 傳染病
- 穿戴式裝置、遠端監控及保養
- 網路安全

標準化的臨床途徑和臨床協定是現代醫學的關鍵要素之一。標準化的其中一個關鍵層面是醫療服務供應商之間的互通性、不僅是醫療記錄、也包括基因資料。最大的問題是、醫療組織是否會放棄基因資料的所有權、而不讓病患擁有其個人基因資料和相關病歷？

互通的病患資料是實現精密醫學的關鍵、這是資料成長最近爆炸的驅動力之一。精準藥物的目標是讓醫療維護、疾病預防、診斷和治療解決方案更有效且更精確。

資料成長率呈指數級。在2021年2月初、美國實驗室每週約有8、000次COVID-19應變。到2021年4月為止、已排序的基因組數量已增加至每週29、000個。每個完整排列的人類基因組大小約為125GB。因此、以每週29、000個基因組排序的速度、靜止的基因組總儲存量每年將超過180 PB。各國已投入資源進行基因流行病研究、以改善基因體監控、並為下一波全球的健康挑戰做好準備。

基因研究的成本降低、正以前所未有的速度推動基因測試與研究。這三個PS是一個轉折點：電腦能力、資料隱私及醫療個人化。到2025年、研究人員預估將有1億至20億個人類基因組進行排序。基因組學要想發揮效用並提出寶貴的主張、基因組學功能必須是照護工作流程的無縫環節；在病患造訪期間、應能輕鬆存取並採取行動。同樣重要的是、病患電子病歷資料必須與病患基因組學資料整合。隨著FlexPod 先進的融合式基礎架構（例如：）問世、組織可以將其基因組學功能融入醫師、護士和診所經理的日常工作流程。如需FlexPod 最新的版次資訊、請參閱 "[《採用Cisco UCS X系列的資料中心》白皮書FlexPod](#)"。

對醫師而言、基因組學的真正價值包括精準醫學和根據病患基因組資料的個人化治療計畫。過去、臨床醫師與資料科學家之間從來沒有這樣的協同作用、基因組學也從近期的技術創新中獲益、同時也從醫療組織與產業技術領導者之間建立真正的合作關係中獲益。

學術醫療中心和其他醫療與生命科學組織正準時建立基因組科學卓越中心（COE）。根據Dr.查理·格斯巴赫博士Greg Crawford和杜克大學的Tim E Reddy表示：「我們知道基因並非透過簡單的二進位交換器來開啟或關

閉、而是因為多個基因管制交換器共同運作。」他們也認定「基因組的這些部分都無法獨立運作。基因組是一種複雜的網路、進化過程已經交織在一起」（["參考資料"](#)）。

NetApp與Cisco+FlexPod 多年來一直努力在整個過程中、逐步將改良功能導入到這個解決方案平台。所有客戶的意見都會被傾聽、評估、並與FlexPod 資訊流和功能集緊密連結在一起。正是這種持續循環的意見回饋、協同作業、改善和慶祝活動、FlexPod 讓整個世界都能將其視為值得信賴的融合式基礎架構平台。它經過全新的簡化設計、是醫療組織最可靠、最強大、功能豐富、最靈活的平台。

範圍

利用融合式基礎架構平台、醫療機構可以託管一或多個基因組學工作負載、以及其他臨床和非臨床醫療應用程式。FlexPod本技術報告使用開放原始碼的產業標準基因組學工具、稱為網關FlexPod 驗證平台期間的網關。然而、更深入討論基因組學或網關不屬於本文件的範圍。

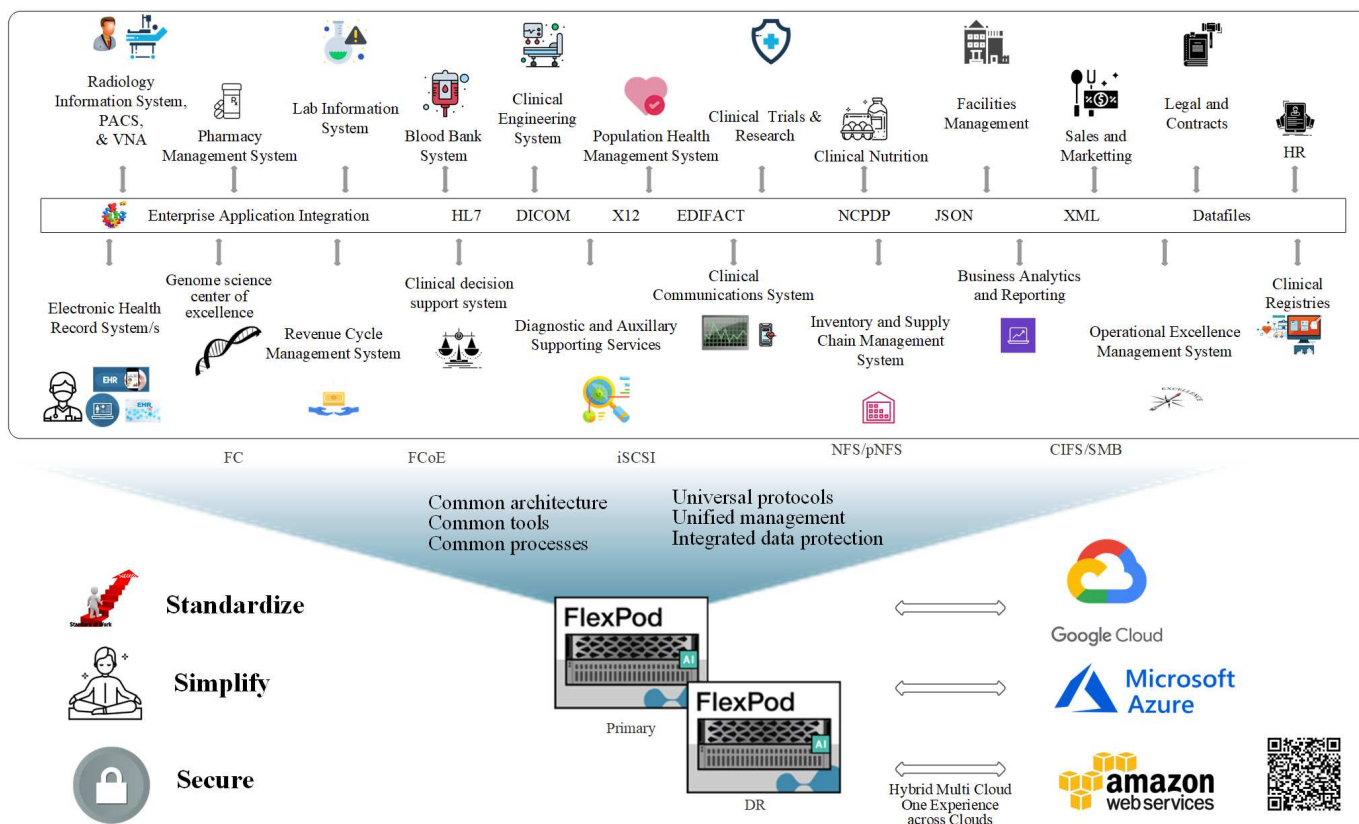
目標對象

本文適用於醫療產業的技術領導者、以及Cisco與NetApp合作夥伴解決方案工程師與專業服務人員。NetApp假設讀者已充分瞭解運算與儲存規模的概念、以及對醫療威脅、醫療安全、醫療IT系統、Cisco UCS及NetApp儲存系統的技術熟悉度。

部署FlexPod 在NetApp上的醫院功能

典型的醫院擁有多元化的IT系統。這類系統大多是向廠商購買、但很少是由內部的醫院系統所建置。因此、醫院系統必須管理資料中心內的多元化基礎架構環境。當醫院將系統統一化為FlexPod 融合式基礎架構平台（例如、Solid Infrastructure）時、組織就能將其資料中心作業標準化。有了這個功能、醫療組織就能在同一個平台上實作臨床和非臨床系統、進而統一化資料中心的營運。FlexPod

Hospital capabilities deployed on a FlexPod



"下一步：將基因工作負載部署在FlexPod 架構上的優點。"

將基因工作負載部署在FlexPod 架構上的優點

"上一篇：簡介。"

本節提供在FlexPod 融合式基礎架構平台上執行基因組學工作負載的簡短優點清單。讓我們快速說明一家醫院的功能。下列企業架構檢視顯示醫院部署在混合雲就緒FlexPod 的支援融合式基礎架構平台上的功能。

- *避免醫療領域的孤立現象。*醫療產業的封閉環境確實令人擔憂。各部門通常會獨立於自己的硬體和軟體組合、而非選擇、而是透過演進來組織化。例如放射學、心臟科、EHR、基因組學、分析、營收週期及其他部門最終會以各自專屬的軟體和硬體組合為基礎。醫療組織維持有限的IT專業人員來管理其硬體和軟體資產。這組人員必須管理一套非常多樣化的硬體與軟體、才是轉折點。廠商對醫療組織提出的一系列程序不一致、使得異質性更為嚴重。
- 「小而小」、「大」。「網關」工具套件專為CPU執行而設計、是FlexPod 最佳的套裝平台、例如：支援獨立擴充網路、運算和儲存設備。FlexPod從小規模開始、隨著基因組學功能和環境的成長而擴充。醫療機構不需要投資專門的平台來執行基因工作負載。相反地、企業組織可以運用FlexPod 諸如功能性的多元平台、在同一個平台上執行基因組學和非基因組學工作負載。例如、如果兒童部門想要實作基因組學功能、IT領導階層可以在現有FlexPod 的實例上配置運算、儲存和網路功能。隨著基因組學業務單位的成長、醫療機構可以FlexPod 視需要擴充其的階平台。
- 單一控制面板與無與倫比的靈活度。Cisco Intersight將應用程式與基礎架構連結、提供從裸機伺服器與Hypervisor到無伺服器應用程式的可見度與管理、大幅簡化IT作業、進而降低成本並降低風險。這款統一化SaaS平台採用統一化的開放式API設計、原生可與協力廠商平台和工具整合。此外、它還能讓您使用行動

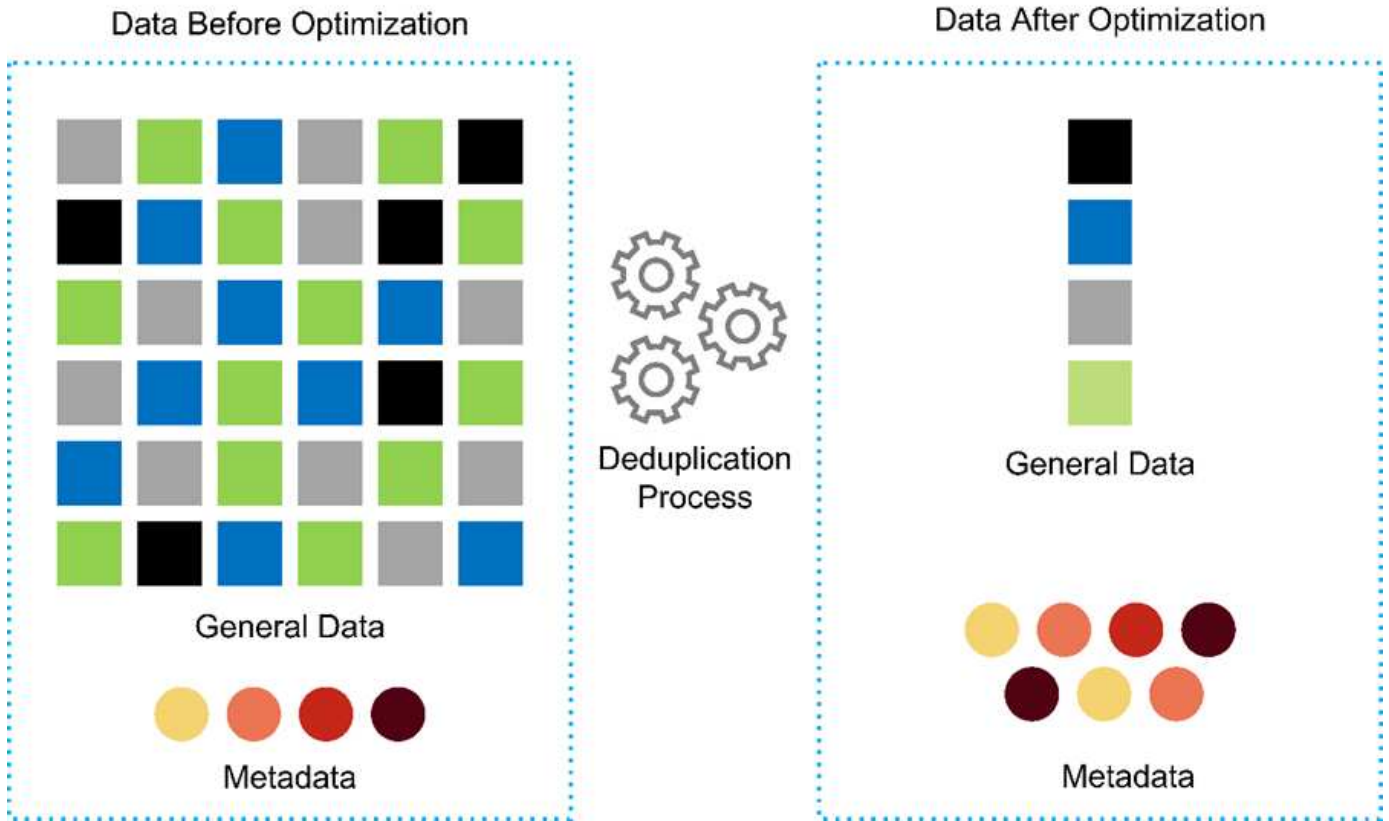
應用程式、從現場或任何地方的資料中心營運團隊進行管理。

使用者可利用Intersight做為其管理平台、快速釋放環境中的實際價值。Intersight可針對許多日常手動工作實現自動化、消除錯誤並簡化日常作業。此外、Intersight所提供的進階支援功能可讓採用者在問題發生之前保持領先地位、並加速解決問題。結合使用、組織在應用程式基礎架構上的時間與金錢都會大減、而在核心業務開發上的時間也會更多。

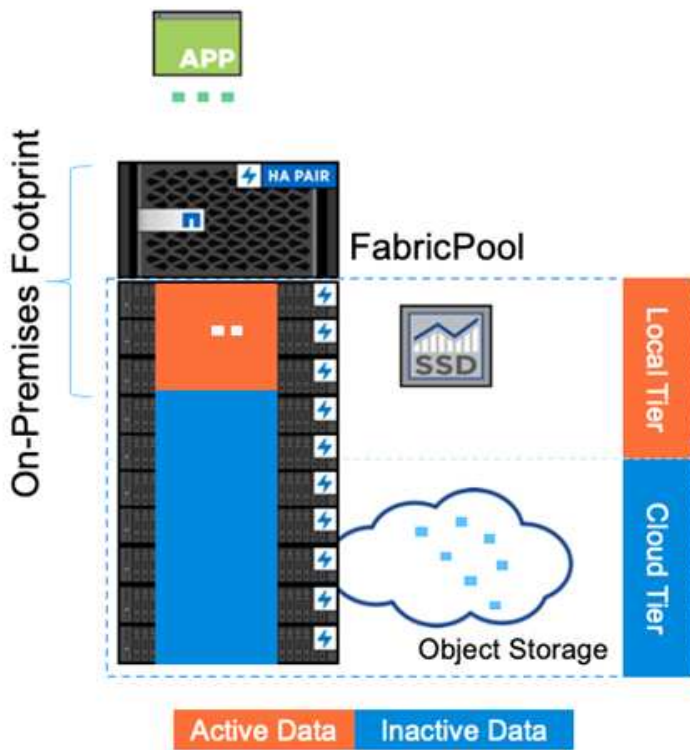
善用Intersight管理和FlexPod易於擴充的架構、讓組織能夠在單FlexPod 一的支援平台上執行數個基因組工作負載、藉此提高使用率並降低總體擁有成本 (TCO)。支援靈活調整規模、從我們的小型支援中心開始、即可擴充至大型的支援中心實作。FlexPod FlexPod FlexPod藉由Cisco Intersight內建的角色型存取控制功能、醫療機構可實作健全的存取控制機制、避免需要個別的基礎架構堆疊。醫療組織內的多個業務單位可以將基因組學視為關鍵核心能力。

最終的功能是協助簡化IT作業並降低營運成本、讓IT基礎架構管理員能夠專注於有助於臨床醫師創新的工作、而非只是為了維持營運不中斷。FlexPod

- 通過驗證的設計與保證成果 FlexPod 。*《整套設計與部署指南》已通過驗證、可重複使用、其中涵蓋完整的組態詳細資料、以及部署FlexPod 可靠的《整套解決方案》所需的業界最佳實務做法。Cisco與NetApp驗證的設計指南、部署指南與架構、可協助您的醫療或生命科學組織從一開始就不再猜測實作已驗證且值得信賴的平台。有了NetApp、您可以加快部署時間、並降低成本、複雜度和風險。FlexPod通過驗證的設計與部署指南、將此平台建立為各種基因組工作負載的理想平台。FlexPod FlexPod
- 創新與敏捷度 FlexPod 推薦使用Epic、Cerner、Meditech等EHRs、以及Agfa、GE、飛利浦等影像系統、作為理想的平台。如需詳細資訊、請參閱 "致勝榮譽" 以及目標平台架構、請參閱Epic使用者網路。在上執行基因組學 "FlexPod" 讓醫療組織能夠靈活地繼續創新之路。利用此功能、組織架構的改變自然就能實現。FlexPod當企業組織採用FlexPod 統一化的解決平台時、醫療IT專家就能撥備時間、精力和資源來進行創新、因此能夠靈活因應生態系統的需求。
- 資料解放 FlexPod 利用融合式基礎架構平台與NetApp ONTAP 支援儲存系統、您可以從單一平台、透過各種大規模的傳輸協定、提供並存取基因組學資料。NetApp支援的解決方案提供簡單、直覺且功能強大的混合雲平台。FlexPod ONTAP您的Data Fabric採用NetApp ONTAP 技術、可跨越實體界限、跨應用程式、將資料一起編織在各個站點之間。您的資料架構是專為資料導向企業打造、專為資料導向企業打造。資料是在多個位置建立和使用、通常需要運用資料、並與其他位置、應用程式和基礎架構共用。因此、您需要一致且整合的方式來管理IT。此功能可讓您的IT團隊掌控一切、並簡化不斷增加的IT複雜度。FlexPod
- 安全的多租戶共享 FlexPod 。*此功能使用FIPS 140-2相容的密碼編譯模組、因此組織能夠將安全性實作為基礎要素、而非事後考量。無論平台的規模為何、透過單一整合式基礎架構平台、均可讓組織實作安全的多租戶共享。FlexPod利用安全的多租戶共享和QoS、有助於將工作負載區隔、並將使用率最大化。FlexPod這有助於避免資金被鎖定在可能未充分利用的專業平台上、並需要一套專門的技能來管理。
- *儲存效率。*基因組需要基礎儲存設備具備領先業界的儲存效率功能。您可以利用NetApp儲存效率功能來降低儲存成本、例如重複資料刪除 (即時和隨需)、資料壓縮和資料壓縮 ("參考資料")。NetApp重複資料刪除技術可在FlexVol 支援重複資料刪除的功能區塊層級進行重複資料刪除。基本上、重複資料刪除技術會移除重複的區塊、只會將獨特的區塊儲存在FlexVol 整個過程中。重複資料刪除技術的運作精細度極高、可在FlexVol 使用中的檔案系統上運作。下圖概述NetApp重複資料刪除技術的運作方式。重複資料刪除技術是應用程式透明的。因此、它可用於刪除任何使用NetApp系統的應用程式所產生的重複資料。您可以將磁碟區重複資料刪除當作內嵌程序執行、也可以當作背景程序執行。您可以將其設定為自動執行、排程執行、或透過CLI、NetApp ONTAP 還原系統管理程式或NetApp Active IQ Unified Manager 還原手動執行。



- 實現基因組互通性 ONTAP FlexCache。效能不只是一種遠端快取功能、可簡化檔案發佈、減少WAN延遲、並降低WAN頻寬成本（"參考資料"）。基因變異辨識與註釋期間的重要活動之一、就是臨床醫師之間的協同作業。即使協同作業的臨床醫師位於不同的地理區域、也能提升資料處理量。ONTAP FlexCache根據。BAM檔案的一般大小（1GB至100s GB）、基礎平台必須能為不同地理位置的臨床醫師提供檔案。利用此技術、基因資料與應用程式可真正做好多站台的準備、讓全球各地研究人員之間的協同作業無縫銜接、並具備低延遲和高處理量。FlexPod ONTAP FlexCache在多站台環境中執行基因組學應用程式的醫療組織、可以使用資料架構進行橫向擴充、以平衡管理與成本與速度。
- * 儲存平台的智慧運用 *採用 ONTAP 自動分層和 NetApp Fabric Pool 技術的 FlexPod 可簡化資料管理。可協助降低儲存成本、而不會影響效能、效率、安全性或保護。FabricPool不需重新建構應用程式基礎架構、即可降低儲存TCO、使企業應用程式透明化、並善用雲端效率。FabricPool利用NetApp的儲存分層功能、更有效率地使用介紹性快閃儲存設備、讓您受益匪淺。FlexPod FabricPool ONTAP如需更多資訊、請參閱 "包含此功能的FlexPod FabricPool"。下圖提供FabricPool 關於支援及其效益的高階概述。



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



- 更快的變體分析和註釋。FlexPod 此功能可更快部署及操作。此平台可讓臨床工作者以低延遲和提高處理量的規模提供資料、進而實現協同作業。FlexPod提升互通性、實現創新。醫療組織可以同時執行基因組和非基因組工作負載、這表示組織不需要使用專門的平台來開始基因組學之旅。

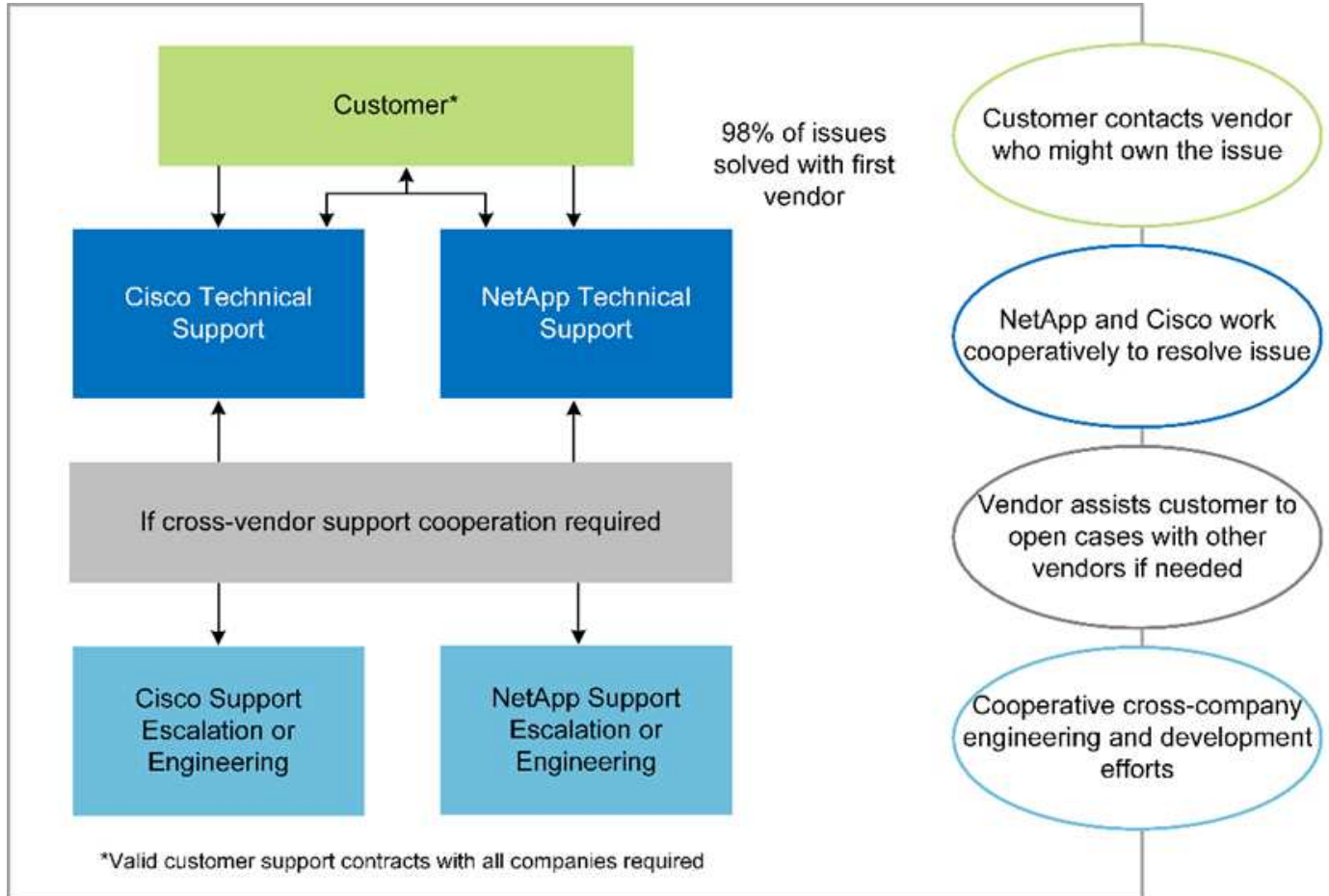
例行地將尖端功能加入儲存平台。FlexPod ONTAP支援支援以高效能儲存設備存取所需的應用程式、而VMware資料中心是部署FC-NVMe的最佳共享基礎架構基礎。FlexPod由於FC-NVMe的演進包括高可用度、多重路徑和額外的作業系統支援、FlexPod 因此很適合做為首選平台、提供支援這些功能所需的擴充性和可靠性。採用端點對端點NVMe的加速I/O技術、可讓基因組分析更快完成 (ONTAP "參考資料")。

循序原始基因組資料會產生較大的檔案大小、因此這些檔案必須提供給變體分析程式、以減少從取樣收集到變體註釋所需的總時間。NVMe (非揮發性記憶體Express) 可做為儲存存取和資料傳輸傳輸傳輸傳輸傳輸協定、提供前所未有的處理量等級和最快的回應時間。透過PCI Express匯流排 (PCIe) 存取Flash儲存設備時、可部署NVMe傳輸協定。FlexPodPCIe可實作數萬個命令佇列、增加平行處理和處理量。從儲存設備到記憶體單一傳輸協定、讓資料存取更快速。

- *從一開始就能靈活地進行臨床研究。*靈活、可擴充的儲存容量與效能、可讓醫療研究機構以靈活或即時 (JIT) 的方式最佳化環境。藉由將儲存設備與運算和網路基礎架構分離、FlexPod 可在不中斷營運的情況下、將其橫向擴充。利用Cisco Intersight FlexPod、可透過內建和自訂的自動化工作流程來管理此支援平台。Cisco Intersight工作流程可讓醫療組織縮短應用程式生命週期管理時間。當學術醫療中心要求將病患資料匿名、並提供給其中心以供研究資訊和/或中心品質之用時、其IT組織可利用Cisco Intersight FlexPod 流程、在數秒內 (而非數小時) 完成安全的資料備份、複製及還原。有了NetApp Trident和Kubernetes、IT組織就能在幾分鐘內配置新的資料科學家、並將臨床資料提供給模型開發、有時甚至只需幾秒鐘。
- 保護基因組資料。NetApp SnapLock 供應特殊用途的磁碟區、可將檔案儲存並承諾為不可擦除、不可重寫的狀態。使用者的資料駐留FlexVol 在一個SnapMirror Volume中、可透過SnapLock NetApp SnapMirror 或SnapVault SnapMirror技術鏡射或將其保存至一個SnapMirror Volume。在保留期間結束之前、無法刪除位於現象磁碟區中的檔案SnapLock、磁碟區本身及其託管Aggregate。使用VMware FPolicy軟體組織可以禁止對具有特定副檔名的檔案執行作業、藉此防止勒索軟體攻擊。ONTAP可針對特定檔案作業觸發FPolicy事件。此事件與原則有關、該原則會呼叫IT需要使用的引擎。您可以使用一組可能含有勒索軟體的副檔名來設

定原則。當副檔名不允許的檔案嘗試執行未獲授權的作業時、FPolicy會防止該作業執行 ("參考資料")。

- 《合作支援》 NetApp與Cisco已建立一套強大、可擴充且靈活的支援模式《支援》、以符合獨特的支援需求、滿足融合式基礎架構的需求。FlexPod FlexPod FlexPod此模式結合了NetApp與Cisco的經驗、資源和技術支援專業、提供簡化的程序來識別FlexPod 及解決不受問題影響的支援問題。下圖概述FlexPod 了《不合作支援》模式。客戶與可能擁有此問題的廠商聯絡、Cisco與NetApp合作解決此問題。Cisco與NetApp擁有跨公司的工程與開發團隊、可攜手解決問題。此支援模式可減少翻譯期間的資訊遺失、實現信任、並減少停機時間。



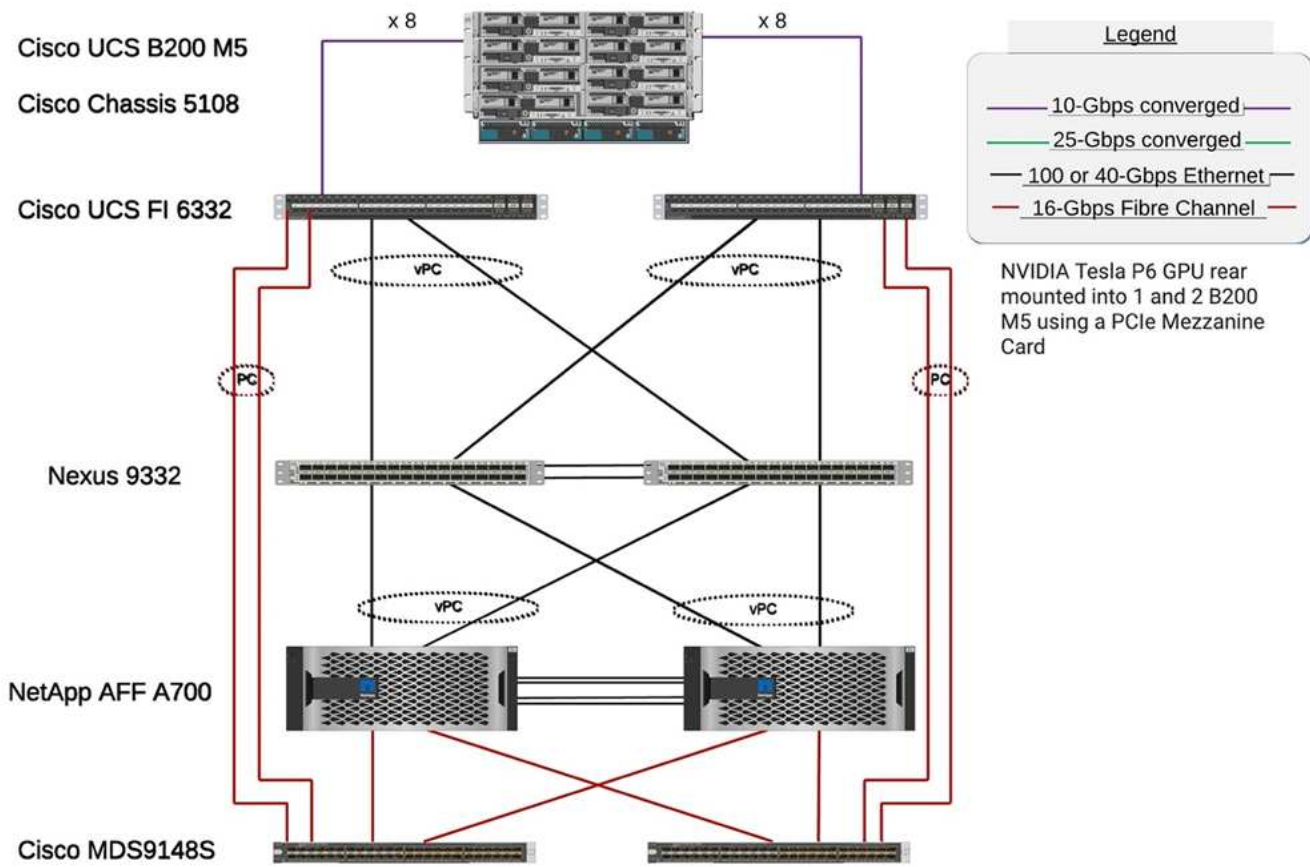
"下一步：解決方案基礎架構硬體與軟體元件。"

解決方案基礎架構硬體與軟體元件

"先前版本：將基因工作負載部署在FlexPod 架構上的優點。"

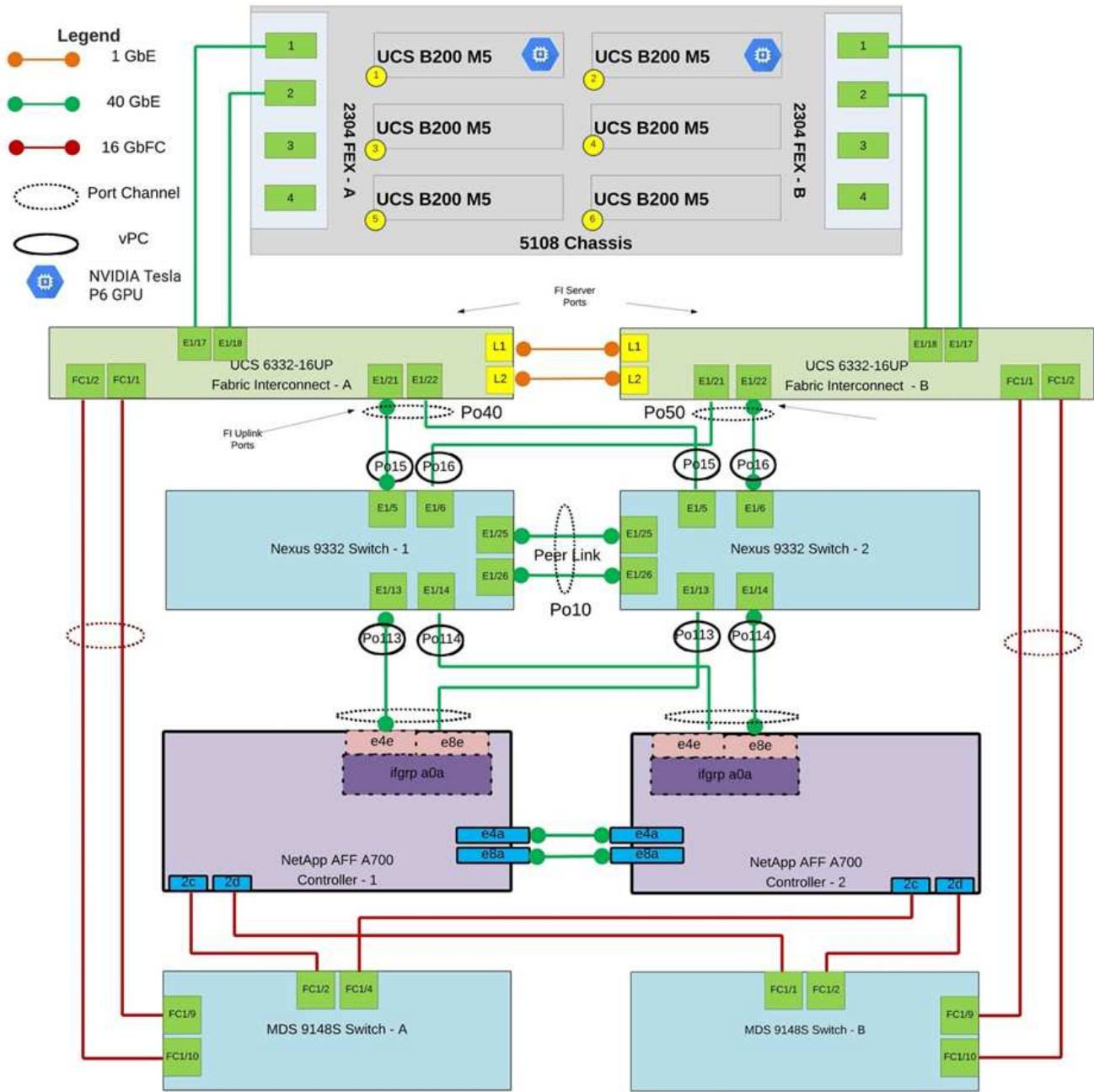
下圖說明FlexPod 用於網關設定與驗證的「不二」系統。我們使用 "採用VMware vSphere 7.0與NetApp的解決方案資料中心9.7 Cisco驗證設計 (CVD) FlexPod ONTAP" 在設定過程中。

FlexPod for Genomics



下圖說明FlexPod 了有關纜線的資訊。

FlexPod for Genomics



下表列出在FlexPod 啟用功能的網關測試期間所使用的硬體元件。以下是 "NetApp 互通性對照表工具" (部分) IMT 和 "Cisco硬體相容清單 (HCL)" 。

層級	產品系列	數量與模式	詳細資料
運算	Cisco UCS 5108機箱	1或2	
	Cisco UCS刀鋒伺服器	6個B200 M5	每個處理器具有2個20個以上核心、2.7GHz及128至384GB RAM

層級	產品系列	數量與模式	詳細資料
	Cisco UCS 虛擬介面卡 (VIC)	Cisco UCS 1440	請參閱
	2個Cisco UCS Fabric互連	6332	-
網路	Cisco Nexus 交換器	2倍Cisco Nexus 9332	-
儲存網路	透過SMB/CIFS、NFS 或iSCSI傳輸協定進行儲存 存取的IP網路	與上述相同的網路交換器	-
	透過FC存取儲存設備	2個Cisco MDS 9148S	-
儲存設備	NetApp AFF 產品豐富的NetApp解決方案：A700 All Flash儲存系統	1叢集	具有兩個節點的叢集
	磁碟櫃	一個DS224C或NS224磁 碟櫃	已裝滿24個磁碟機
	SSD	24、1.2TB或更大容量	-

下表列出基礎架構軟體。

軟體	產品系列	版本或版本	詳細資料
多種	Linux	RHEL 8.3	-
	Windows	Windows Server 2012 R2 (64位元)	-
	NetApp ONTAP	部分9.8或更新版本ONTAP	-
	Cisco UCS 光纖互連	Cisco UCS Manager 4.1或 更新版本	-
	Cisco 乙太網路3000 或9000系列交換器	9000系列、7.0 (3) i7 (7) 或更新版本適用於3000 系列、9.2 (4) 或更新版 本	-
	Cisco FC：Cisco MDS 9132T	8.4(1a)或更新版本	-
	Hypervisor	VMware vSphere ESXi 7.0	-
儲存設備	Hypervisor管理系統	VMware vCenter Server 7.0 (vCSA) 或更新版本	-
網路	NetApp 虛擬儲存主控台 (VSC)	VSC 9.7或更新版本	-
	NetApp SnapCenter	不含更新版本SnapCenter	-
	Cisco UCS Manager	4.1(3c)或更新版本	-
Hypervisor	ESXi		

軟體	產品系列	版本或版本	詳細資料
管理	Hypervisor管理系統VMware vCenter Server 7.0 (vCSA) 或更新版本		
	NetApp虛擬儲存主控台 (VSC)	VSC 9.7或更新版本	
	NetApp SnapCenter	不含更新版本SnapCenter	
	Cisco UCS Manager	4.1(3c)或更新版本	

"下一步：基因組學-網關設定與執行。"

基因組學：網關設定與執行

"先前版本：解決方案基礎架構硬體與軟體元件。"

根據國家人類基因組研究所（"NHGRI"《基因組學》（Genomics）、《基因組學》（Genomics）是一個人所有基因（基因組）的研究、包括這些基因彼此互動、以及與個人環境的互動。」

根據 "NHGRI" 「脫氧核糖DNA（DNA）是一種化學複合物、內含開發和引導幾乎所有生物體活動所需的指示。DNA分子是由兩股扭曲的配對股組成、通常稱為雙螺旋。」 「生物的完整DNA組稱為基因組。」

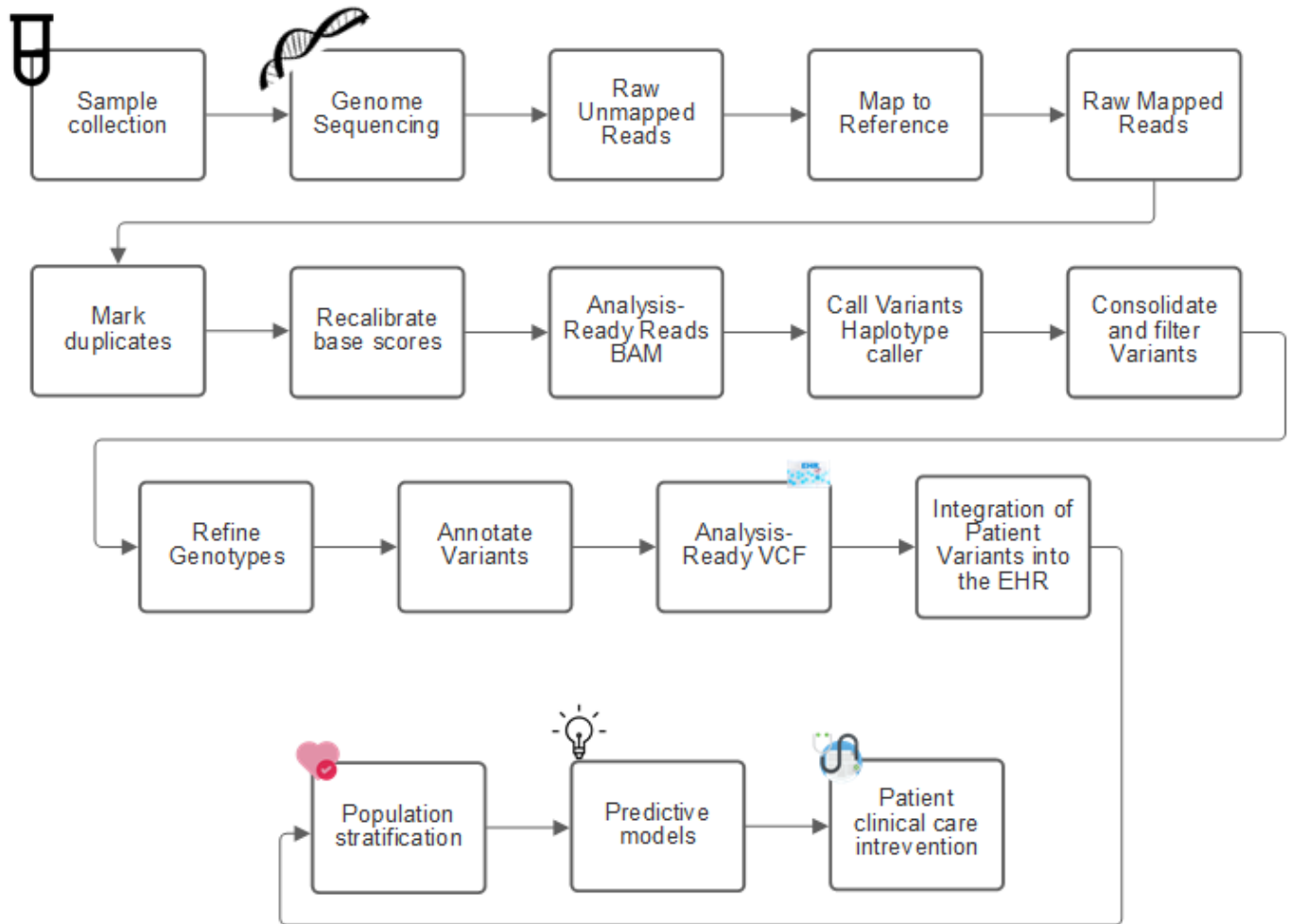
排序是決定DNA中基礎的確切順序的程序。當今最常見的排序類型之一、稱為「合成排序」。此技術使用螢光訊號的放射來訂購底座。研究人員可以使用DNA定序來搜尋基因變異、以及在病患仍處於初期階段時、可能在疾病的發展或發展過程中扮演重要角色的任何變異。

從樣本到變體識別、註釋和預測

高層級的基因組學可分為下列步驟。這份清單並非詳盡無遺：

1. 範例集合。
2. "基因組排序" 使用序列發生器來產生原始資料。
3. 預先處理：例如、"重複資料刪除" 使用 "皮卡爾"。
4. 基因分析。
 - a. 對應至參考基因組。
 - b. "變體" 通常使用網關和類似工具來進行識別和註釋。
5. 整合至電子醫療記錄（EHR）系統。
6. "人口分層" 並識別地理位置和族群背景之間的基因變異。
7. "預測模式" 使用顯著的單核多態性。
8. "驗證"。

下圖顯示從取樣到變體識別、註釋和預測的程序。



人類基因組項目於2003年4月完成、該項目對公共領域的人類基因組序列進行了非常高質量的模擬。這種參考基因組在基因組能力的研究與開發方面、開始爆炸性發展。幾乎每個人的基因都有其特徵。直到最近、醫師仍利用基因來預測和判斷生兒缺陷、例如單一基因改變所造成的特定繼承模式、例如：粗細胞性貧血。人類基因組專案所提供的寶貴資料、導致基因組功能的現況出現。

基因組學具有廣泛的優點。以下是醫療與生命科學領域的一小組優點：

- 在照護點進行更好的診斷
- 更好的預測
- 精密醫學
- 個人化的治療計畫
- 更佳的疾病監控
- 減少不良事件
- 改善治療方法
- 改善疾病監控
- 有效參與臨床試驗、並根據基因分型、更好地選擇病患進行臨床試驗。

基因組學是 "四頭獸、" 因為資料集生命週期內的運算需求：擷取、儲存、發佈及分析。

基因組分析工具套件（網關）

GAT是以資料科學平台的形式開發 "[廣泛研究所](#)"。GATE是一組開放原始碼工具、可進行基因組分析、特別是變種探索、識別、註釋和基因分型。網格連結的好處之一是工具和或命令集可以鏈結在一起、形成完整的工作流程。廣泛機構所面臨的主要挑戰如下：

- 瞭解疾病的根本原因和生物機制。
- 辨識出在疾病根本原因上採取行動的治療介入措施。
- 瞭解從變異到人類生理學的觀察線。
- 建立標準與原則 "[架構](#)" 適用於基因組資料呈現、儲存、分析、安全等。
- 標準化並推廣可互通的基因組集合體資料庫（gnomAD）。
- 以基因組為基礎的病患監控、診斷及治療、更精確。
- 協助實作可在症狀出現之前及早預測疾病的工具。
- 建立跨專業合作夥伴社群並賦予其能力、以協助處理生物醫學中最棘手且最重要的問題。

根據GATE和廣義研究所的資料、基因組排序應視為在實驗室中的一種傳輸協定；每項工作都有完整的記錄、最佳化、可重複執行、而且在各個樣本和實驗之間都一致。以下是廣泛研究所建議的一組步驟、如需詳細資訊、請參閱 "[網站](#)"。

系統設定FlexPod

基因組學工作負載驗證包括FlexPod 一套完整的功能、可從零開始設定一套完整的基礎架構平台。此平台具備高可用度、可獨立擴充、例如網路、儲存設備和運算都可獨立擴充。FlexPod我們使用下列Cisco驗證設計指南做為參考架構文件、來設定FlexPod 這個環境：["VMware vSphere 7.0與NetApp VMware vCenter 9.7的資料中心FlexPod ONTAP"](#)。請參閱FlexPod 下列的《不完整平台設定摘要》：

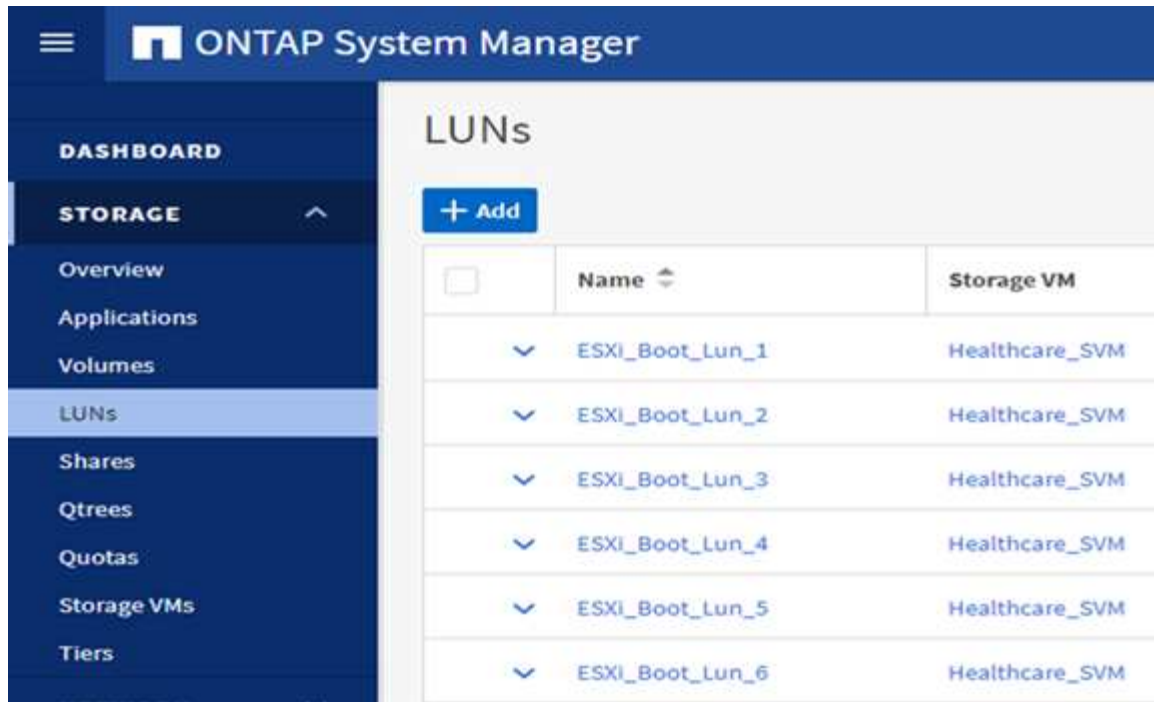
若要執行FlexPod 程式不實的設定、請完成下列步驟：

1. 支援下列IP4保留與VLAN、以進行內部測試設定與驗證。FlexPod

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. 在ONTAP SVM上設定iSCSI型開機LUN。



3. 將LUN對應至iSCSI啟動器群組。

	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

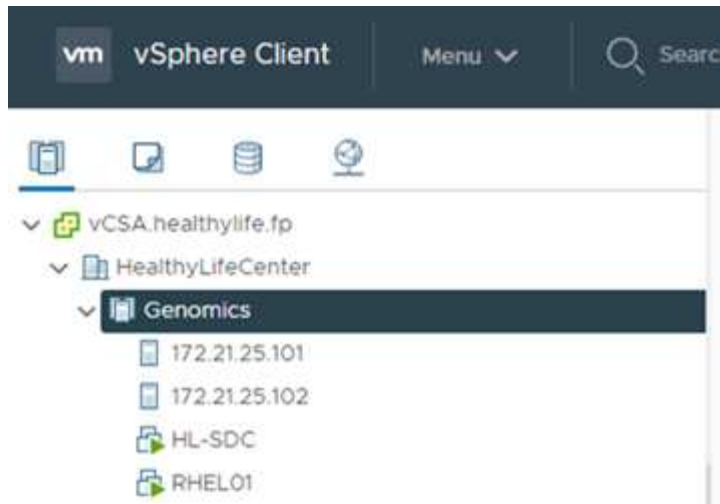
STATUS ✔ Online	VOLUME ESXi_Boot_Vol	DESCRIPTION -	SNAPSHOT COPIES (LOCAL) STATUS ✔ Protected	SNAPMIRROR (LOCAL OR REMOTE) STATUS ✔ Unprotected
SERIAL NUMBER 80A4X+R8rAhP	QOS POLICY GROUP -	MAPPED TO INITIATORS ▲ GenomicsESXi_1 (1) iqn.1992-08.com.cisco:ucs-...	ID 0	SNAPSHOT POLICY default
CAPACITY (AVAILABLE % TOTAL) 95% 20 GB	LUN FORMAT 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

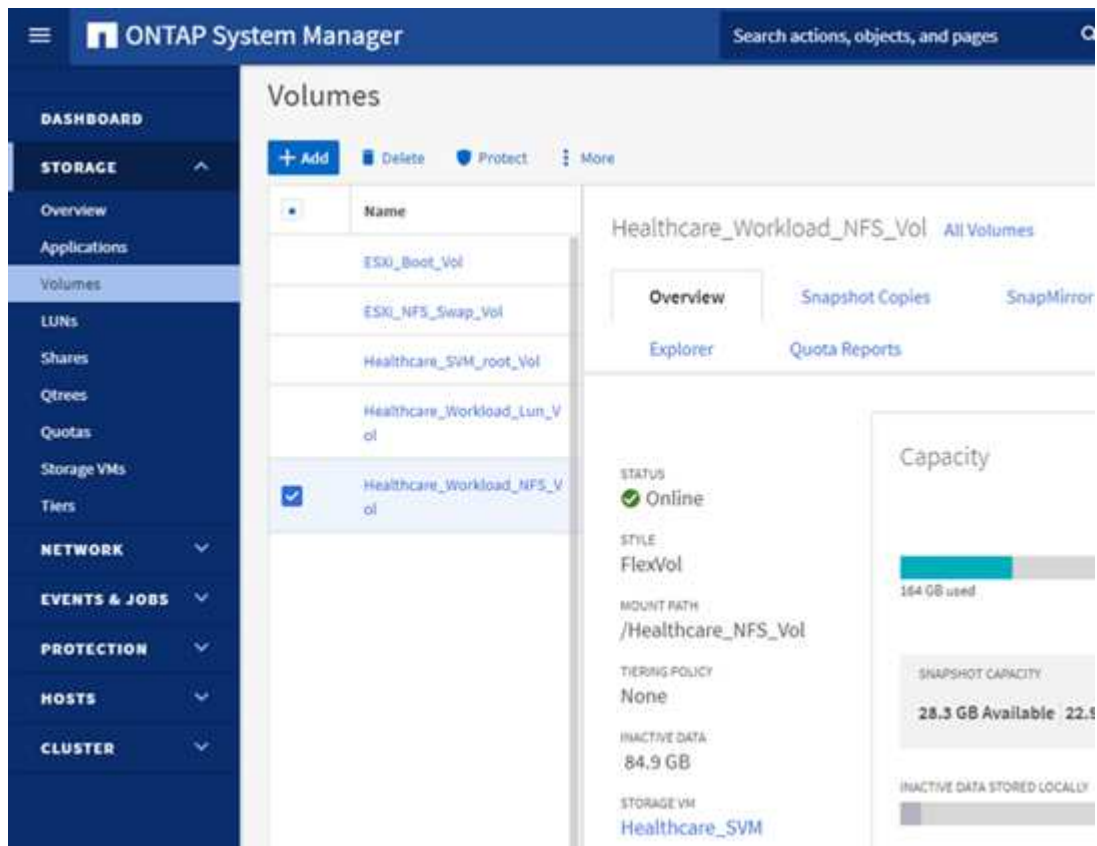
STATUS ✔ Online	VOLUME ESXi_Boot_Vol	DESCRIPTION -	SNAPSHOT COPIES (LOCAL) STATUS ✔ Protected	SNAPMIRROR (LOCAL OR REMOTE) STATUS ✔ Unprotected
SERIAL NUMBER 80A4X+R8rAhU	QOS POLICY GROUP -	MAPPED TO INITIATORS ▲ GenomicsESXi_2 (1) iqn.1992-08.com.cisco:ucs-...	ID 0	SNAPSHOT POLICY default
CAPACITY (AVAILABLE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

4. 安裝vSphere 7.0搭配iSCSI開機。

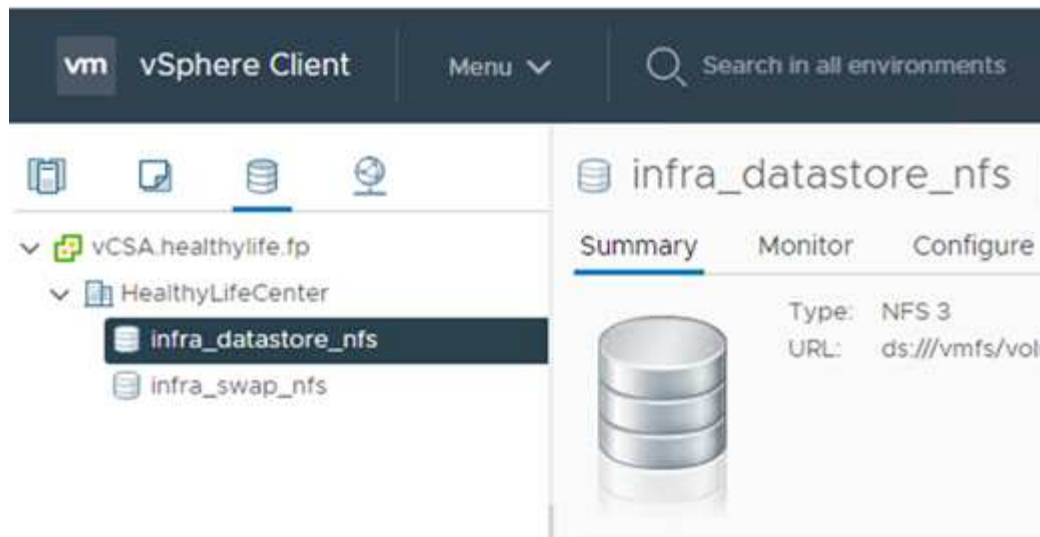
5. 向vCenter登錄ESXi主機。



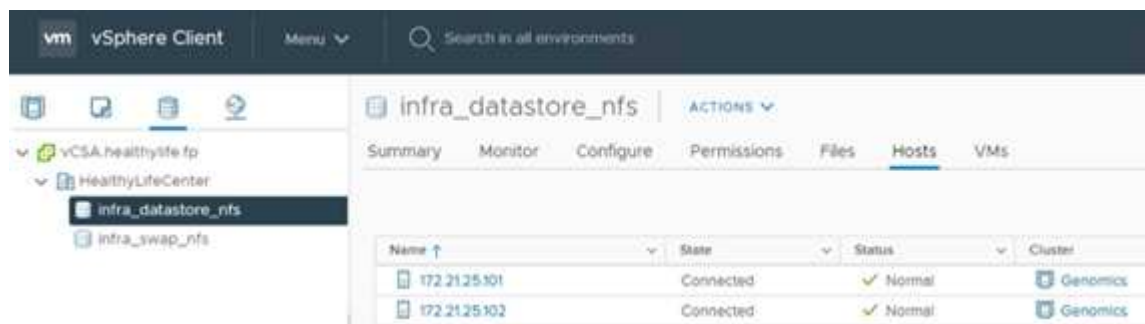
6. 將NFS資料存放區「infra_datastore_nfs」配置到ONTAP 整個儲存區。



7. 將資料存放區新增至vCenter。



8. 使用vCenter將NFS資料存放區新增至ESXi主機。



9. 使用vCenter建立Red Hat Enterprise Linux (RHEL) 8.3 VM來執行網關。

10. NFS資料存放區會顯示給VM、並掛載於「/mnt/genomics」、用於儲存網關執行檔、指令碼、二進位對齊對應 (BAM) 檔案、參考檔案、索引檔案、字典檔案及輸出檔案、以供進行變式呼叫。

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

網關設定與執行

在RedHat Enterprise 8.3 Linux VM上安裝下列先決條件：

- Java 8或SDK 1.8或更新版本
- 請從廣泛研究所下載網關4.2.0.0 "[GitHub網站](#)"。基因組序列資料通常以一系列以索引標籤分隔的Ascii欄的形式儲存。不過、要儲存的空間太多。因此、新的標準演變為BAM (*。bam) 檔案。BAM檔案會以壓縮、索引及二進位格式來儲存順序資料。我們 "[已下載](#)" 一組公開提供的BAM檔案、可從執行網頁網頁 "[公有網域](#)"。我們也下載了索引檔案 (*。Bai)、字典檔案 (*。dict)和參考資料檔案 (*。fasta)。

下載後、網關工具套件會有一個Jar檔案和一組支援指令碼。

- 「gatk-package-4.2.0.0-local.jar」執行檔
- "gatk"指令碼檔案。

我們下載了BAM檔案、以及父、母和子*。bam檔案所組成之系列的對應索引、字典和參考基因組檔案。

Cromwell引擎

Cromwell是開放原始碼引擎、專為科學工作流程所設計、可實現工作流程管理。克倫威爾引擎可在兩個引擎中執行 "模式"、伺服器模式或單一工作流程執行模式。您可以使用來控制Cromwell引擎的行為 "[Cromwell引擎組態檔](#)"。

- *伺服器模式。*啟用 "[休閒的](#)" 在Cromwell引擎中執行工作流程。
- *執行模式。*執行模式最適合在Cromwell中執行單一工作流程、"[參考資料](#)" 以取得執行模式的完整可用選項集。

我們使用Cromwell引擎大規模執行工作流程和管線。Cromwell引擎使用方便使用的產品 "[工作流程說明語言](#)" (WDL) 型指令碼語言。Cromwell也支援第二個工作流程指令碼標準、稱為通用工作流程語言 (CML)。在本技術報告中、我們使用WDL。WDL最初由廣義基因組分析管線研究所開發、使用WDL工作流程可以使用多種策略來實作、包括：

- *線性鏈結*顧名思義、工作#1的輸出會以輸入方式傳送至工作#2。
- *多重輸入/輸出。*這類似於線性鏈結、因為每個工作都能將多個輸出作為後續工作的輸入傳送。
- *Scater-GAI。*這是最強大的企業應用程式整合 (AI) 策略之一、尤其是在事件導向架構中使用時。每項工作都會以分離的方式執行、每項工作的輸出都會整合到最終輸出中。

使用WDL以獨立模式執行網關時、有三個步驟：

1. 使用「womtool.jar」驗證語法。

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

2. 產生輸入JSON.

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

3. 使用Cromwell引擎和「Cromwell.jar」來執行工作流程。

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

網關可以使用多種方法來執行、本文將探討其中三種方法。

使用Jar檔案執行GK

讓我們來看看使用Hplotype變體呼叫者執行的單一變體呼叫管線。

```
[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
```

在這種執行方法中、我們使用網關本機執行Jar檔案、使用單一Java命令來叫用Jar檔案、然後將數個參數傳遞給命令。

1. 此參數表示我們正在叫用「Haplotypefaller」變體呼叫者管道。
2. 「-INPUT」指定輸入BAM檔案。
3. 「-output」以變式呼叫格式 (*.vcf)指定變體輸出檔案。 ("參考資料")。
4. 利用「參考」參數、我們正在傳遞參考基因組。

執行後、可在區段中找到輸出詳細資料 ["使用Jar檔案執行網關的輸出。"](#)

使用./gatk指令碼執行網關

可以使用「/gatk」指令碼來執行GatK工具套件。讓我們來檢查下列命令：

```
[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
```

我們會將數個參數傳遞給命令。

- 此參數表示我們正在叫用「Haplotypefaller」變體呼叫者管道。
- 「-I」指定輸入BAM檔案。
- 「-o」以變式呼叫格式 (*.vcf)指定變體輸出檔案。 ("參考資料")。
- 使用「-R」參數、我們正在傳遞參考基因組。

執行後、可在區段中找到輸出詳細資料 ["016e203cf9beada735f224ab14d0b3af"](#)

使用Cromwell引擎執行網關

我們使用Cromwell引擎來管理網關執行。讓我們來檢查命令列及其參數。

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

在這裏，我們通過傳遞"-jar"參數來調用Java命令，表示我們打算執行一個Jar文件，例如"Cromwell -65.jar"。傳遞的下一個參數（「run」）表示Cromwell引擎以執行模式執行、其他可能的選項則是「Server mode」（伺服器模式）。下一個參數是「*」。wdl是「執行」模式用來執行管線的參數。下一個參數是要執行之工作流程的輸入參數集。

以下是「gplo.wdl」（ghplo.wdl）檔案內容的外觀：

```
[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"  
  }  
}
```

以下是對應的Json檔案、其中包含對Cromwell引擎的輸入。

```
[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]#
```

請注意、Cromwell使用記憶體內建資料庫來執行。執行後、可在一節中看到輸出記錄 ["使用Cromwell引擎執行網關的輸出。"](#)

如需如何執行網關的完整步驟集、請參閱 ["GK文件"](#)。

["下一步：使用Jar檔案執行GATE的輸出。"](#)

使用Jar檔案執行網關的輸出

["上一篇：基因組學-網頁設定與執行。"](#)

使用Jar檔案執行GK時產生下列範例輸出。

```
[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]#

```

請注意、輸出檔案位於執行後指定的位置。

["fb08e15744e912200b45cf04b5fce2ad"](#)

使用./gatk指令碼執行網關的輸出

"上一步：使用Jar檔案執行GATE的輸出。"

使用「/gatk」指令碼執行網關、產生下列範例輸出。

```
[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]#

```

請注意、輸出檔案位於執行後指定的位置。

"[下一步：使用Cromwell引擎執行網關的輸出。](#)"

使用Cromwell引擎執行網關的輸出

"[11fffe01d469840980d9b9a5f45bf9ed](#)"

使用Cromwell引擎執行GATE時、產生下列範例輸出。

```

[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]#
```

"下一步：GPU設定。"

GPU設定

"上一篇：使用Cromwell引擎執行網關的輸出。"

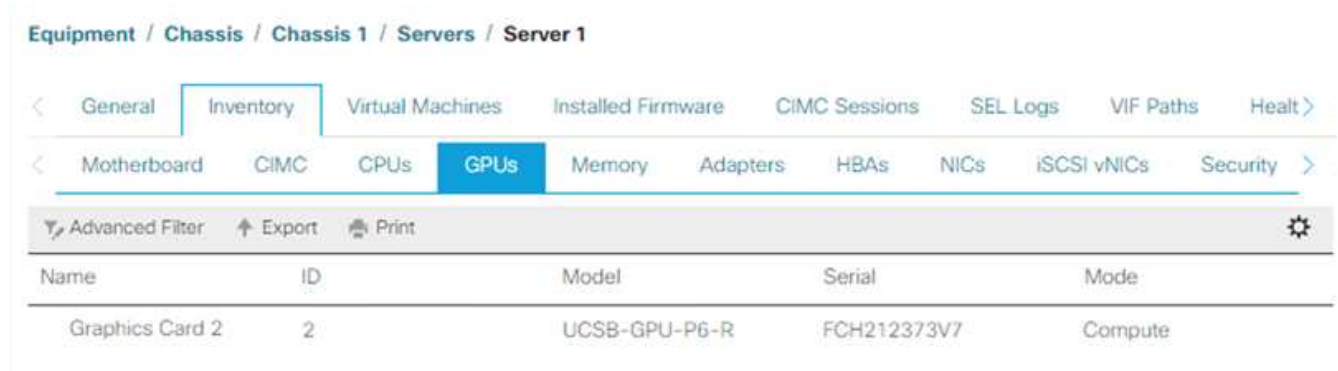
在發佈時、網關工具並未原生支援內部部署的GPU型執行。提供下列設定與指引、讓讀者瞭解FlexPod 使用GATE PCIe夾層卡的後置NVIDIA Tesla P6 GPU使用時、如何簡單易用。

我們使用下列Cisco驗證設計（CVD）做為參考架構和最佳實務指南、來設定FlexPod 「不支援」環境、以便執行使用GPU的應用程式。

- ["適用於AI/ML的Datacenter搭配Cisco UCS 480 ML、可進行深度學習FlexPod"](#)

以下是此設定的一組重點摘要：

1. 我們在UCS B200 M5伺服器的夾層插槽中使用PCIe NVIDIA Tesla P6 GPU。



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

- 在此設定中、我們已在NVIDIA合作夥伴入口網站上註冊、並取得評估授權（也稱為授權）、以便在運算模式下使用GPU。
- 我們已從NVIDIA合作夥伴網站下載所需的NVIDIA vGPU軟體。
- 我們從NVIDIA合作夥伴網站下載了「*.bin」檔案。
- 我們安裝了NVIDIA vGPU授權伺服器、並使用從NVIDIA合作夥伴網站下載的「*.bin」檔案、將授權新增至授權伺服器。
- 請務必在NVIDIA合作夥伴入口網站上、針對您的部署選擇正確的NVIDIA vGPU軟體版本。在此設定中、我們使用的是460.73.02版驅動程式。
- 此命令會安裝 **"NVIDIA vGPU管理程式"** 在ESXi中。

```
[root@localhost:~] esxcli software vib install -v
/vmfs/volumes/infra_datastore_nfs/nvidia/vib/NVIDIA_bootbank_NVIDIA-
VMware_ESXi_7.0_Host_Driver_460.73.02-1OEM.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_460.73.02-1OEM.700.0.0.15525992
VIBs Removed:
VIBs Skipped:
```

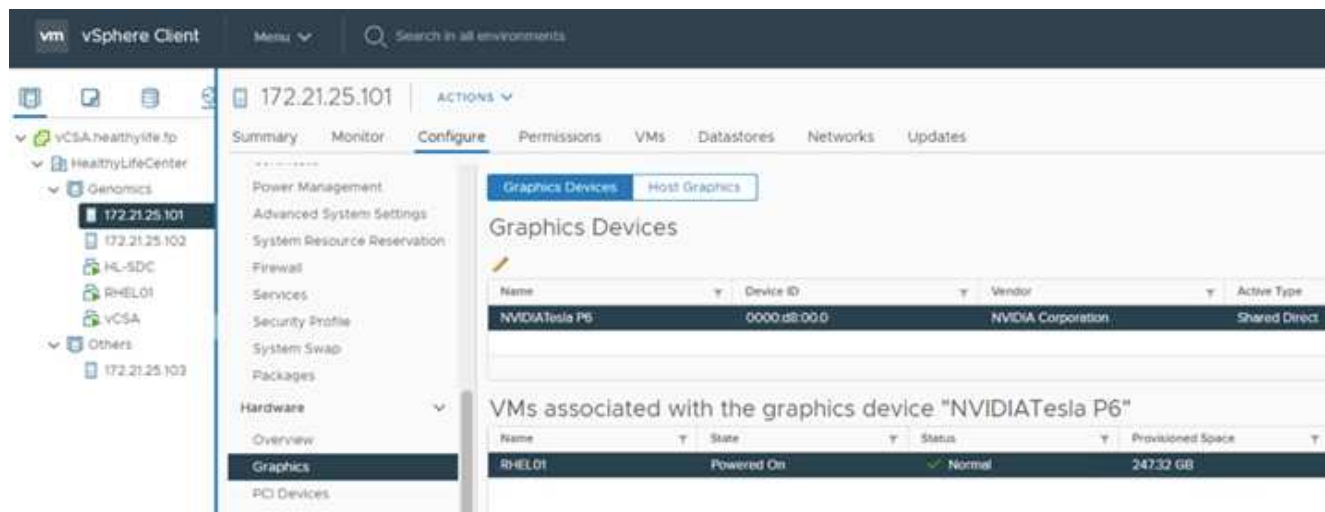
- 重新開機ESXi伺服器之後、請執行下列命令來驗證安裝並檢查GPU的健全狀況。


```

[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. 使用vCenter、"設定" 圖形裝置設定為「Shared Direct (共享直接)」。

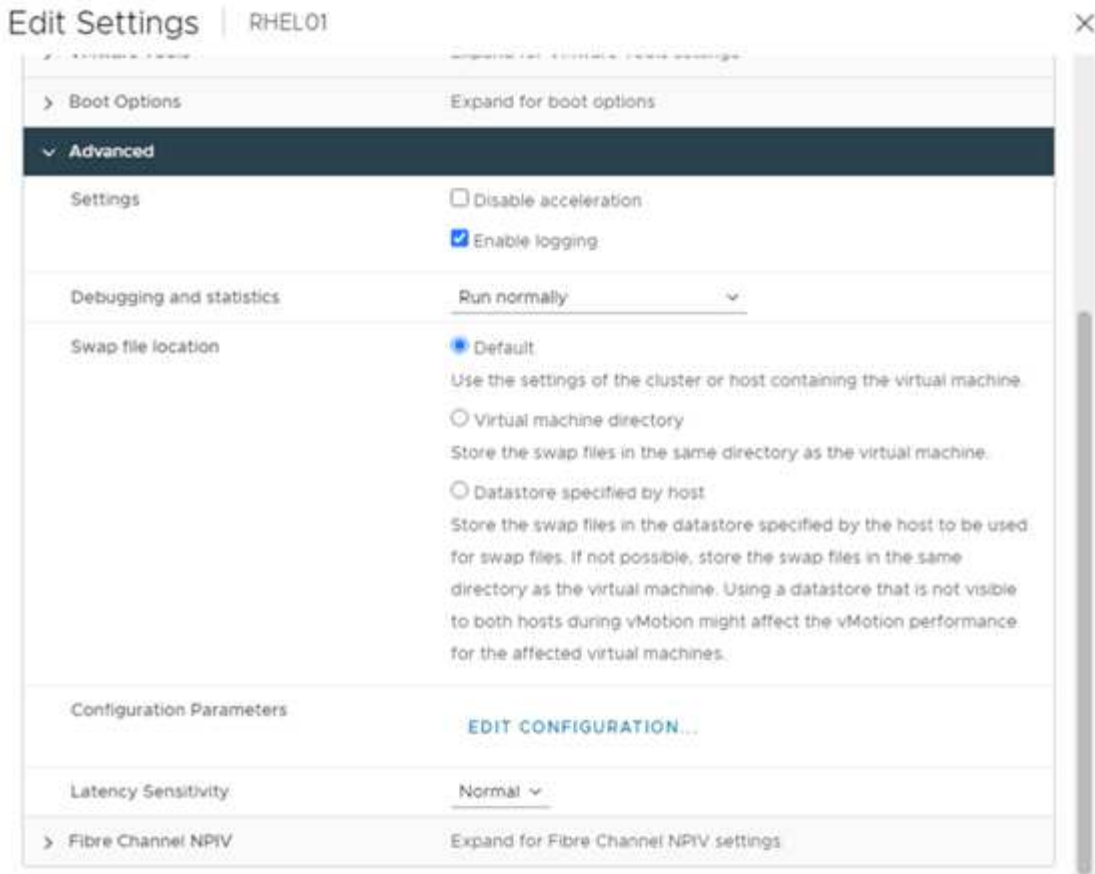


10. 確定RedHat VM已停用安全開機。
11. 確定VM Boot Options韌體已設定為EFI.. "[參考資料](#)")。

> 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 <input type="text" value="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 <input type="text" value="10"/> seconds
> Advanced	Expand for advanced settings
> Fibre Channel NPIV	Expand for Fibre Channel NPIV settings

CANCEL OK

12. 請確定下列「PARAMS」已新增至「VM選項」進階「編輯組態」。「pciPassThree.64bitMMIOSlezeGB」參數的值取決於GPU記憶體和指派給VM的GPU數量。例如：
- 如果VM被指派4 x 32GB V100 GPU、則此值應為128。
 - 如果VM被指派4 x 16GB p6 GPU、則此值應為64。

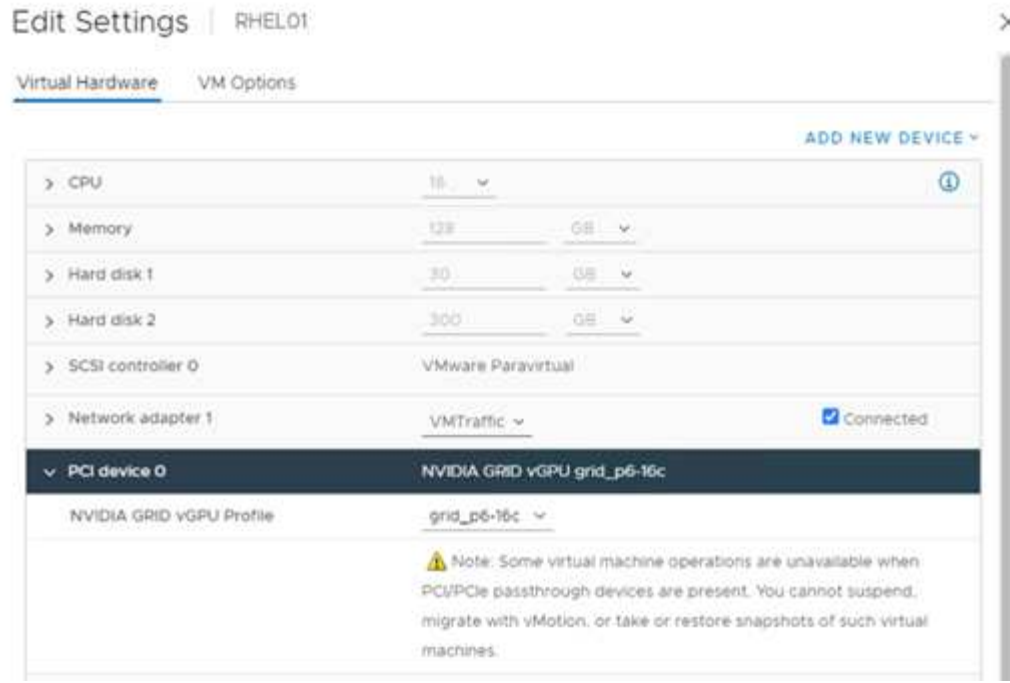


Configuration Parameters

⚠ 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

- 在vCenter中將vGPU新增為新的PCI裝置至虛擬機器時、請務必選取NVIDIA GRID vGPU作為PCI裝置類型。
- 選擇適合使用GPU、GPU記憶體及使用目的之正確GPU設定檔：例如、圖形與運算。



15. 在RedHat Linux VM上、您可以執行下列命令來安裝NVIDIA驅動程式：

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

16. 執行下列命令、確認報告的vGPU設定檔正確無誤：

```
[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. 重新開機後、請確認報告的NVIDIA vGPU與驅動程式版本是否正確。

```

[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. 請確定已在vGPU網絡組態檔的VM上設定授權伺服器IP。

a. 複製範本。

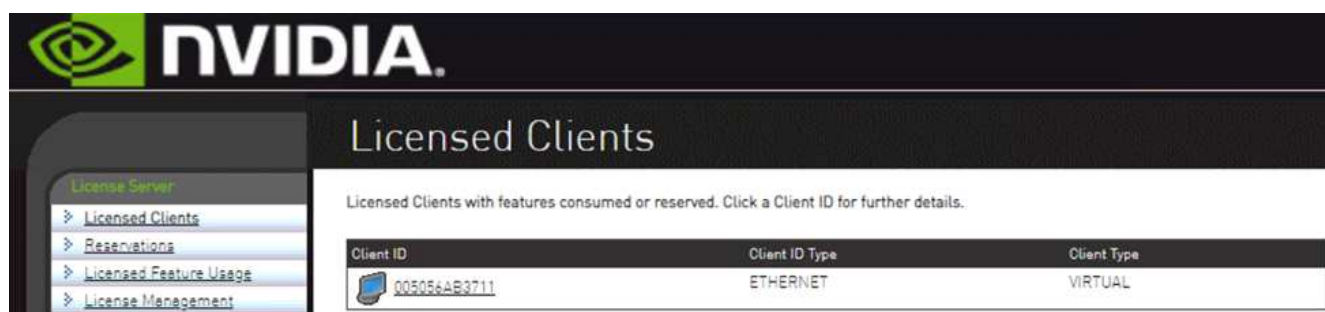
```
[root@genomics1 genomics]# cp /etc/nvidia/gridd.conf.template /etc/nvidia/gridd.conf
```

- b. 編輯檔案「/etc/nvidia/Rid.conf」、新增授權伺服器IP位址、並將功能類型設為1。

```
ServerAddress=192.168.169.10
```

```
FeatureType=1
```

19. 重新啟動VM之後、您應該會在授權伺服器的授權用戶端下看到一個項目、如下所示。



20. 如需下載網關軟體的詳細資訊、請參閱解決方案設定一節。
21. 在GK能夠在內部部署使用GPU之後、工作流程說明語言為*。WDL具有如下所示的執行時間屬性。

```

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}"
  }
}

```

["下一步：結論。"](#)

結論

["上一篇：GPU設定。"](#)

全球許多醫療組織都已將FlexPod 不僅僅是一個通用平台的不僅僅是功能、更是以功能為基礎。有了此功能、您就能安心部署醫療功能。FlexPodNetApp支援的解決方案標準配備能夠實作業界領先的一組現成通訊協定。FlexPod ONTAP無論申請執行特定病患基因組學的來源為何、FlexPod 互通性、存取能力、可用度及擴充性都是以一套完整的平台為標準。當我們在FlexPod 一個不完整的平台上標準化時、創新文化就變得具有感染力。

何處可找到其他資訊

若要深入瞭解本文所述資訊、請檢閱下列文件與網站：

- 適用於AI/ML的Datacenter搭配Cisco UCS 480 ML、可進行深度學習FlexPod

["https://www.cisco.com/c/en/us/td/docs/unified_computing/ucs/UCS_CVDs/flexpod_480ml_aiml_deployement.pdf"](https://www.cisco.com/c/en/us/td/docs/unified_computing/ucs/UCS_CVDs/flexpod_480ml_aiml_deployement.pdf)

- VMware vSphere 7.0與NetApp VMware vCenter 9.7的資料中心FlexPod ONTAP

["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)

- 供應說明文件中心 ONTAP

["http://docs.netapp.com"](http://docs.netapp.com)

- 敏捷又有效率：FlexPod 如何推動資料中心現代化

["https://www.flexpod.com/idc-white-paper/"](https://www.flexpod.com/idc-white-paper/)

- 醫療業的AI

["https://www.netapp.com/us/media/na-369.pdf"](https://www.netapp.com/us/media/na-369.pdf)

- 適用於醫療業的可讓您輕鬆轉型FlexPod

["https://flexpod.com/solutions/verticals/healthcare/"](https://flexpod.com/solutions/verticals/healthcare/)

- Cisco與NetApp提供的解決方案FlexPod

["https://flexpod.com/"](https://flexpod.com/)

- 醫療產業的AI與分析（NetApp）

["https://www.netapp.com/us/artificial-intelligence/healthcare-ai-analytics/index.aspx"](https://www.netapp.com/us/artificial-intelligence/healthcare-ai-analytics/index.aspx)

- 醫療業的AI智慧型基礎架構選項可提升成功率

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

- 包含《Cisco Intersight *》的《Datacenter with》、《Cisco Intersight Storage Connector for Cisco Intersight》和《Cisco Intersight託管模式》。FlexPod ONTAP ONTAP

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

- 採用Red Hat Enterprise Linux OpenStack平台的資料中心FlexPod

["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)

版本歷程記錄

版本	日期	文件版本歷程記錄
1.0版	2021年11月	初始版本。

版權資訊

Copyright © 2025 NetApp, Inc. 版權所有。台灣印製。非經版權所有人事先書面同意，不得將本受版權保護文件的任何部分以任何形式或任何方法（圖形、電子或機械）重製，包括影印、錄影、錄音或儲存至電子檢索系統中。

由 NetApp 版權資料衍伸之軟體必須遵守下列授權和免責聲明：

此軟體以 NETAPP「原樣」提供，不含任何明示或暗示的擔保，包括但不限於有關適售性或特定目的適用性之擔保，特此聲明。於任何情況下，就任何已造成或基於任何理論上責任之直接性、間接性、附隨性、特殊性、懲罰性或衍生性損害（包括但不限於替代商品或服務之採購；使用、資料或利潤上的損失；或企業營運中斷），無論是在使用此軟體時以任何方式所產生的契約、嚴格責任或侵權行為（包括疏忽或其他）等方面，NetApp 概不負責，即使已被告知有前述損害存在之可能性亦然。

NetApp 保留隨時變更本文所述之任何產品的權利，恕不另行通知。NetApp 不承擔因使用本文所述之產品而產生的責任或義務，除非明確經過 NetApp 書面同意。使用或購買此產品並不會在依據任何專利權、商標權或任何其他 NetApp 智慧財產權的情況下轉讓授權。

本手冊所述之產品受到一項（含）以上的美國專利、國外專利或申請中專利所保障。

有限權利說明：政府機關的使用、複製或公開揭露須受 DFARS 252.227-7013（2014 年 2 月）和 FAR 52.227-19（2007 年 12 月）中的「技術資料權利 - 非商業項目」條款 (b)(3) 小段所述之限制。

此處所含屬於商業產品和 / 或商業服務（如 FAR 2.101 所定義）的資料均為 NetApp, Inc. 所有。根據本協議提供的所有 NetApp 技術資料和電腦軟體皆屬於商業性質，並且完全由私人出資開發。美國政府對於該資料具有非專屬、非轉讓、非轉授權、全球性、有限且不可撤銷的使用權限，僅限於美國政府為傳輸此資料所訂合約所允許之範圍，並基於履行該合約之目的方可使用。除非本文另有規定，否則未經 NetApp Inc. 事前書面許可，不得逕行使用、揭露、重製、修改、履行或展示該資料。美國政府授予國防部之許可權利，僅適用於 DFARS 條款 252.227-7015(b)（2014 年 2 月）所述權利。

商標資訊

NETAPP、NETAPP 標誌及 <http://www.netapp.com/TM> 所列之標章均為 NetApp, Inc. 的商標。文中所涉及的所有其他公司或產品名稱，均為其各自所有者的商標，不得侵犯。