採用資料快取的混合雲AI作業系統

NetApp Solutions

NetApp
April 12, 2024

目錄

TR-4841：混合雲AI作業系統、含資料快取

Rick Huang、David Arnette、NetApp Yochay Ettun、cnvrg-io

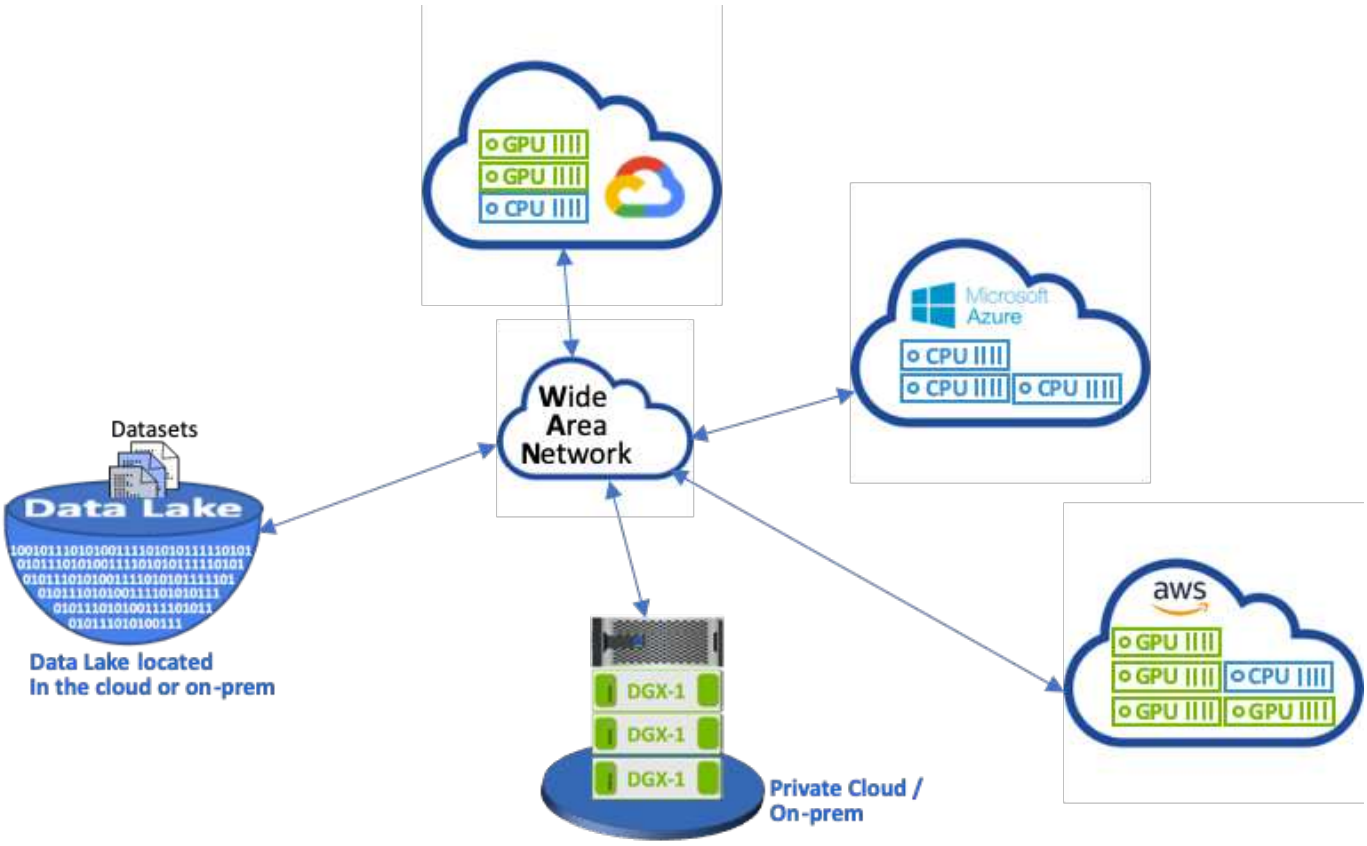
資料爆炸性成長、以及ML和AI的指數成長、已融合在一起、創造出一個具有獨特開發與實作挑戰的字節經濟。

雖然大家都知道、ML模型需要大量資料、而且需要近端的高效能資料儲存設備來處理運算資源、但實際上實作這種模式並不太直接、尤其是混合雲和彈性運算執行個體。大量資料通常儲存在低成本的資料湖中、因為GPU等高效能AI運算資源無法有效存取資料。在混合雲基礎架構中、有些工作負載會在雲端上運作、有些工作負載則位於內部部署環境或完全位於不同的HPC環境中、這種情況更形嚴重。

在本文件中、我們提供一款新穎的解決方案、讓IT專業人員和資料工程師能夠建立真正的混合雲AI平台、並具備拓撲感知資料中心、讓資料科學家能夠在運算資源附近、立即自動建立資料集快取、 無論位於何處。因此、不僅能完成高效能模式訓練、還能創造更多效益、包括多位AI從業人員的協同作業、他們可以立即存取資料集版本中樞內的資料集快取、版本和線路。

使用案例總覽與問題陳述

資料集與資料集版本通常位於資料湖中、例如NetApp StorageGRID 以物件為基礎的儲存設備、可降低成本及提供其他營運優勢。資料科學家會將這些資料集拉出、並以多個步驟來進行設計、以準備好使用特定模型進行訓練、通常會在過程中建立多個版本。下一步、資料科學家必須挑選最佳化的運算資源（GPU、高階CPU執行個體、內部部署叢集等）來執行模型。下圖說明ML運算環境中資料集的鄰近度不足。



然而、多項訓練實驗必須在不同的運算環境中平行執行、每項都需要從資料湖下載資料集、這是一項昂貴且耗時的程序。無法保證資料集與運算環境的距離（尤其是混合雲）。此外、在同一個資料集上執行自己實驗的其他團隊成員、也必須經歷同樣艱鉅的程序。除了明顯緩慢的資料存取速度之外、還有難以追蹤資料集版本、資料集共用、協同作業和可重複性等挑戰。

客戶需求

客戶的需求可能會有所不同、以便在有效率地使用資源的情況下執行高效能ML；例如、客戶可能需要下列項目：

- 從執行訓練模式的每個運算執行個體快速存取資料集、而不會產生昂貴的下載和資料存取複雜度
- 在雲端或內部部署中使用任何運算執行個體（GPU或CPU）、而不需擔心資料集的位置
- 在同一個資料集上同時執行多項訓練實驗、並使用不同的運算資源、而不會產生不必要的延遲和資料延遲、進而提升效率和生產力
- 將運算執行個體成本降至最低
- 利用工具來記錄資料集、其資料類型、版本及其他中繼資料詳細資料、藉此改善可重複性
- 增強共享與協同作業、讓團隊中的任何授權成員都能存取資料集並執行實驗

若要使用NetApp ONTAP 支援資料集快取管理軟體來實作資料集快取、客戶必須執行下列工作：

- 設定和設定最接近運算資源的NFS儲存設備。
- 判斷要快取的資料集和版本。
- 監控已認可給快取資料集的總記憶體、以及可用於其他快取認可的NFS儲存容量（例如快取管理）。
- 如果資料集在某段時間內未使用、則會在快取中逾時。預設值為一天、其他組態選項則可供使用。

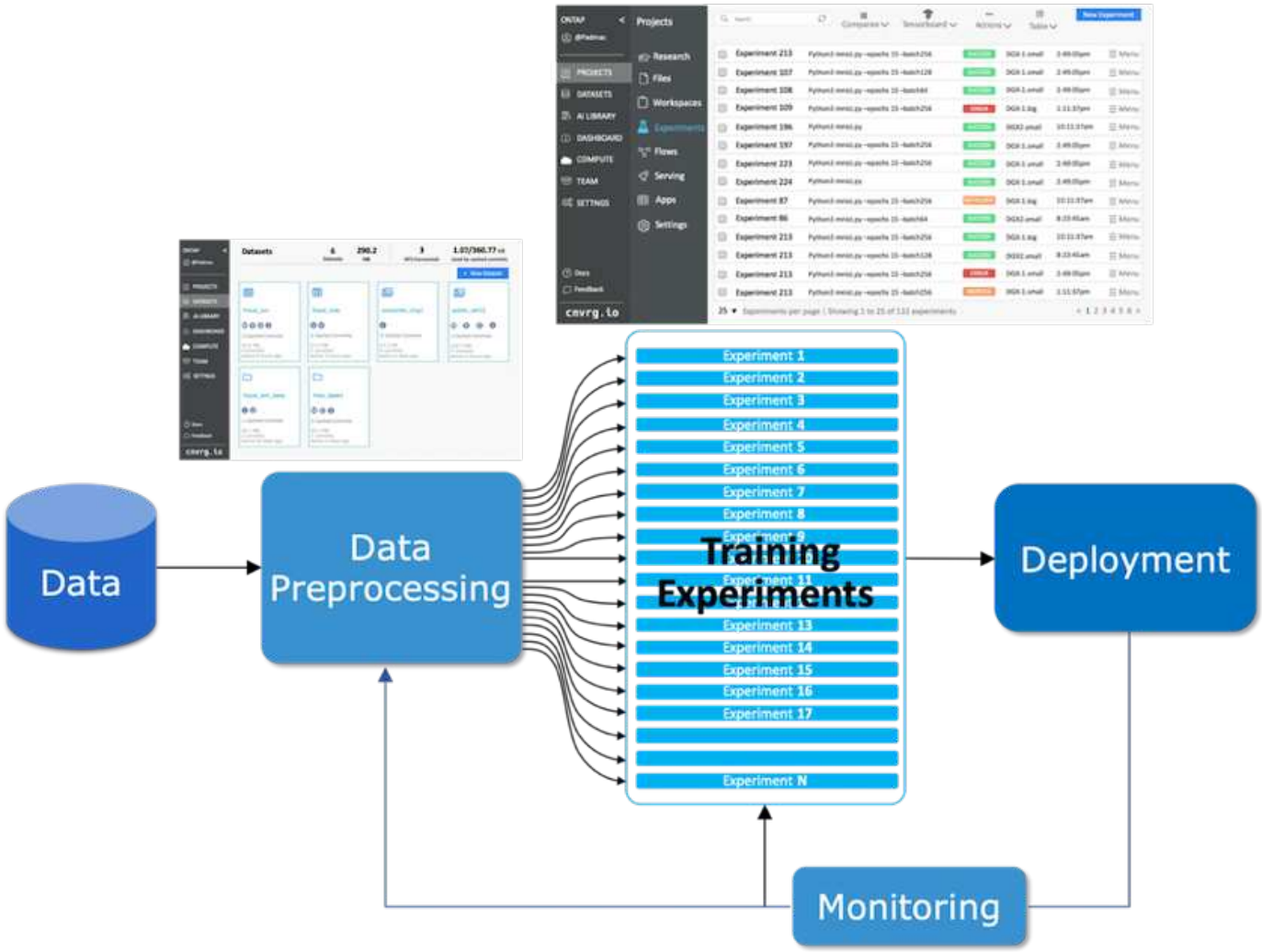
解決方案總覽

本節將回顧傳統的資料科學管道及其缺點。同時也介紹建議的資料集快取解決方案架構。

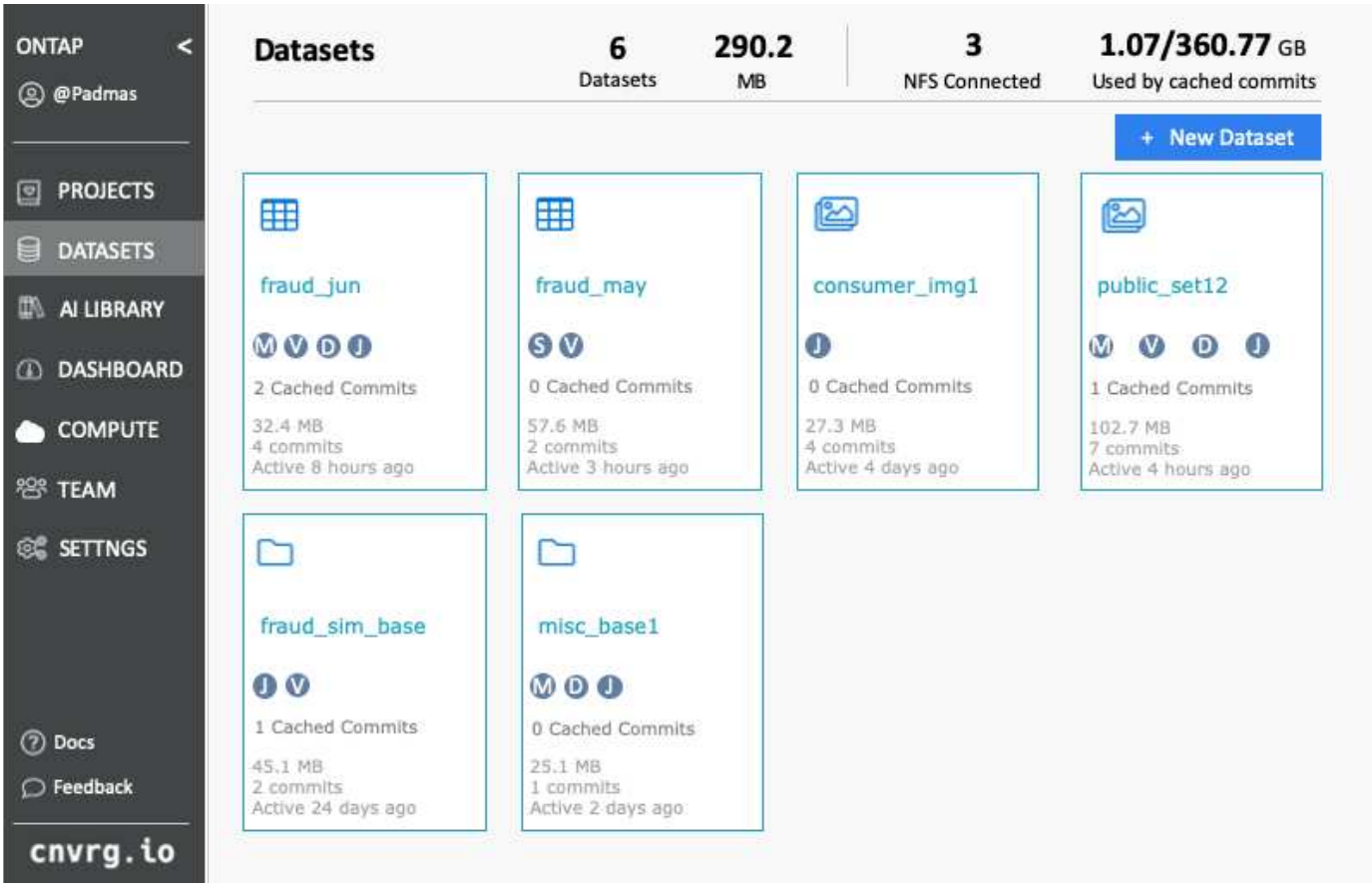
傳統的資料科學管道與缺點

典型的ML模型開發與部署順序涉及迭代步驟、包括下列步驟：

- 擷取資料
- 資料預先處理（建立多個版本的資料集）
- 執行多項涉及超參數最佳化、不同模型等的實驗
- 部署
- 監控cnvrg-IO已開發出全方位平台、可將研究到部署等所有工作自動化。下圖顯示與管線相關的儀表板快照範例。



在公有儲存庫和私有資料中、經常會有多個資料集在活動中。此外、每個資料集可能會因為資料集清理或功能工程而產生多個版本。需要提供資料集線器和版本集線器的儀表板、以確保團隊能夠使用協同作業和一致性工具、如下圖所示。



下一步是訓練、訓練模式需要多個平行執行個體、每個執行個體都與資料集和特定運算執行個體相關聯。將資料集繫結至特定運算執行個體的特定實驗、是一項挑戰、因為某些實驗可能是由Amazon Web Services（AWS）的GPU執行、而其他實驗則由內部部署的DGX-1或DGX-2執行個體執行。在GCP的CPU伺服器上執行其他實驗、而資料集位置與執行訓練的運算資源並不合理。合理的鄰近範圍、可將資料集儲存設備與運算執行個體之間的完整10GbE或更低延遲連線。

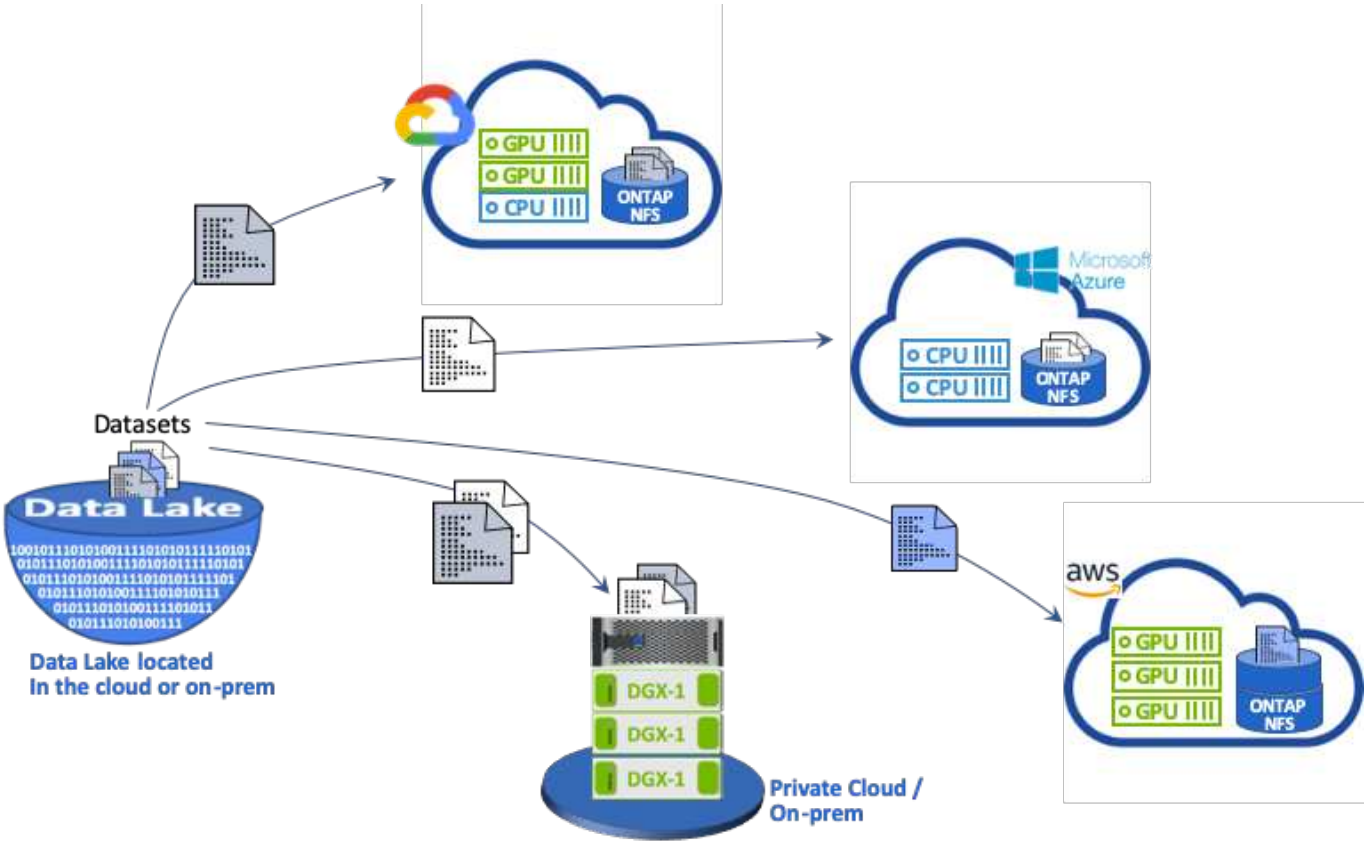
資料科學家通常會將資料集下載至執行訓練和執行實驗的運算執行個體。不過、這種方法可能有幾個問題：

- 當資料科學家將資料集下載至運算執行個體時、並不保證整合式運算儲存設備具備高效能（高效能系統的範例是ONTAP AFF 以支援A800 NVMe解決方案為例）。
- 當下載的資料集位於單一運算節點時、分散式模型在多個節點上執行時、儲存設備可能會成為瓶頸（與NetApp ONTAP 的高效能分散式儲存設備不同）。
- 由於佇列衝突或優先順序、訓練實驗的下一次迭代作業可能會在不同的運算執行個體中執行、同樣也會從資料集到運算位置建立相當長的網路距離。
- 在同一個運算叢集上執行訓練實驗的其他團隊成員無法共用此資料集；每個團隊成員都會從任意位置執行（昂貴）資料集下載。
- 如果後續訓練工作需要相同資料集的其他資料集或版本、資料科學家必須重新執行（昂貴）資料集下載、將資料集下載至執行training.NetApp的運算執行個體、並執行cnvrg-.IO建立新的資料集快取解決方案、以消除這些障礙。此解決方案可將Hot資料集快取至ONTAP 高效能的儲存系統、以加速執行ML管線。使用支援NetApp的Data Fabric（例如、Re A800）、資料集會在運算組合的資料架構中快取一次（而且只快取一次）ONTAP AFF 。由於NetApp ONTAP 不間斷NFS高速儲存設備可支援多個ML運算節點、因此訓練模式的效能已經過最佳化、可為組織帶來成本節約、生產力和營運效率。

解決方案架構

此解決方案來自NetApp和cnvrg-IO、提供資料集快取功能、如下圖所示。資料集快取可讓資料科學家挑選所需的資料集或資料集版本、並將其移至ONTAP 靠近ML運算叢集的支援NFS快取。資料科學家現在可以執行多項實驗、而不會產生延遲或下載。此外、所有協同作業的工程師都能將相同的資料集用於附加的運算叢集（可自由選擇任何節點）、而無需從資料湖下載額外資料。提供資料科學家儀表板、可追蹤及監控所有資料集和版本、並提供快取資料集的檢視。

cnvrg-IO平台會自動偵測未在特定時間內使用的老舊資料集、並從快取中予以移出、因為快取會為較常用的資料集保留可用的NFS快取空間。請務必注意ONTAP 、使用效益技術的資料集快取功能可在雲端和內部部署中運作、因此能提供最大的靈活度。



概念與元件

本節涵蓋與ML工作流程中的資料快取相關的概念與元件。

機器學習

對於全球許多企業和組織而言、ML正迅速成為不可或缺的一環。因此、IT與DevOps團隊現在面臨著將ML工作負載標準化、以及配置雲端、內部部署與混合式運算資源的挑戰、這些資源可支援ML工作與管線所需的動態密集工作流程。

以Container為基礎的機器學習與Kubernetes

容器是獨立的使用者空間執行個體、可在共享主機作業系統核心上執行。容器的採用率迅速增加。Container提供許多與虛擬機器（VM）相同的應用程式沙箱效益。不過、由於虛擬機器所仰賴的Hypervisor和客體作業系統

層已經被淘汰、因此容器的重量遠較輕。

容器也能直接透過應用程式、有效地封裝應用程式相依性、執行時間等項目。最常用的容器包裝格式是Docker容器。以Docker Container格式容器化的應用程式、可在任何能夠執行Docker Container的機器上執行。即使應用程式的相依性並不存在於機器上、這也是如此、因為所有相依性都會封裝在容器本身中。如需詳細資訊、請參閱 "Docker網站"。

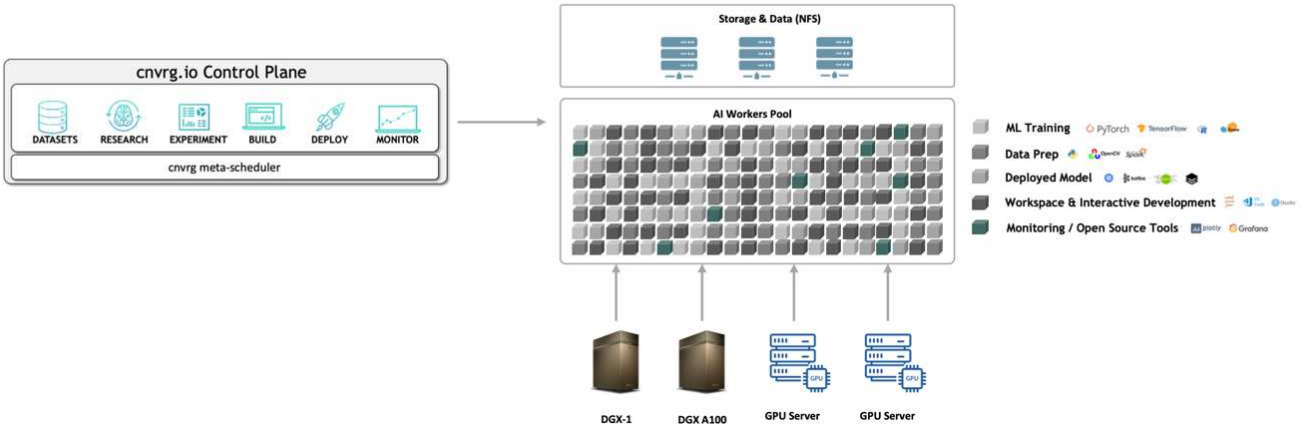
廣受歡迎的Container Orchestrator Kubernetes可讓資料科學家啟動靈活、以容器為基礎的工作和管線。此外、基礎架構團隊也能在單一託管與雲端原生環境中、管理及監控ML工作負載。如需詳細資訊、請參閱 "Kubernetes網站"。

cnvrg-io

Cnvrg-IO是一套AI作業系統、可將企業管理、擴充及加速AI與資料科學的開發、從研究到正式作業的方式轉變成全新的方式。程式碼優先平台是由資料科學家為資料科學家所建置、可靈活地在內部部署或雲端上執行。利用模型管理、MLOps和持續的ML解決方案、cnvrg-IO將頂尖技術引進資料科學團隊、讓他們能將更少的時間花在DevOps上、專注於真正的魔力演算法。自從使用cnvrg-IO之後、各產業的團隊已獲得更多的生產模式、進而提高商業價值。

Cnvrg-IO中繼排程器

Cnvrg：IO具有獨特的架構、可讓IT和工程師將不同的運算資源附加至同一個控制面板、並讓cnvrg-IO管理所有資源中的ML工作。這表示它可以附加多個內部部署的Kubernetes叢集、VM伺服器和雲端帳戶、並在所有資源上執行ML工作負載、如下圖所示。



Cnvrg-IO資料快取

Cnvrg-IO可讓資料科學家利用其資料快取技術來定義冷熱資料集版本。根據預設、資料集會儲存在集中式物件儲存資料庫中。然後、資料科學家可以快取所選運算資源上的特定資料版本、以節省下載時間、進而提高ML開發與生產力。快取且未使用數天的資料集會自動從選取的NFS清除。快取和清除快取只要按一下滑鼠、不需要編碼、IT或DevOps工作。

Cnvrg-IO流程和ML管路

Cnvrg-IO流程是建置正式作業ML管線的工具。流程中的每個元件都是在使用基礎泊塢視窗映像的選定運算上執行的指令碼/程式碼。這項設計可讓資料科學家和工程師建立單一管線、同時在內部部署和雲端上執行。Cnvrg-IO可確保資料、參數和成品在不同的元件之間移動。此外、系統會監控並追蹤每個流程、以確保100%可重現的資料科學。

Cnvrg-IO核心

Cnvrg-IO核心是資料科學社群的免費平台、可協助資料科學家更專注於資料科學、而非DevOps。核心的靈活基礎架構可讓資料科學家控制使用任何語言、AI架構或運算環境、無論是內部部署或雲端環境、讓他們能夠發揮最佳功能、建置演算法。在任何Kubernetes叢集上、只要使用一個命令、就能輕鬆安裝Cnvrg-IO核心。

NetApp ONTAP AI

支援ML和深度學習（DL）工作負載的資料中心參考架構、使用NetApp支援儲存系統、以及搭配Tesla V100 GPU的NVIDIA DGX系統。ONTAP AFFAI採用業界標準的NFS檔案傳輸協定、透過100Gb乙太網路、為客戶提供高效能的ML/DL基礎架構、使用標準資料中心技術來降低實作與管理成本。ONTAP使用標準化的網路和傳輸協定、ONTAP 讓AI能夠整合到混合雲環境、同時維持作業一致性和簡易性。作為預先驗證的基礎架構解決方案、ONTAP Realize AI可縮短部署時間與風險、大幅降低管理成本、讓客戶更快實現價值。

NVIDIA DeepOps

DeepOps是NVIDIA的開放原始碼專案、使用Ansible可根據最佳實務做法、自動部署GPU伺服器叢集。DeepOps是模組化的、可用於各種部署工作。本文件及其所說明的驗證作業中、DeepOps用於部署Kubernetes叢集、其中包含GPU伺服器工作節點。如需詳細資訊、請參閱 "DeepOps網站"。

NetApp Trident

Trident是NetApp開發與維護的開放原始碼儲存協調工具、可大幅簡化Kubernetes工作負載的持續儲存設備建立、管理與使用。Trident本身就是Kubernetes原生應用程式、直接在Kubernetes叢集內執行。Kubernetes使用者（開發人員、資料科學家、Kubernetes系統管理員等）可以使用他們已經熟悉的標準Kubernetes格式、建立、管理及與持續儲存磁碟區互動。同時、他們也能善用NetApp先進的資料管理功能、以及採用NetApp技術的資料架構。Trident將持續儲存設備的複雜度抽象化、使其易於使用。如需詳細資訊、請參閱 "Trident網站"。

NetApp StorageGRID

NetApp StorageGRID 功能區是軟體定義的物件儲存平台、可提供簡單、類似雲端的儲存設備、讓使用者使用S3傳輸協定來存取、以滿足這些需求。支援跨網際網路連線站台多個節點的橫向擴充系統、不受距離限制。StorageGRID有了NetApp的智慧型原則引擎StorageGRID 、使用者可以選擇跨站台的銷毀編碼物件、在遠端站台之間進行地理恢復或物件複寫、以將WAN存取延遲降至最低。本解決方案提供優異的私有雲主要物件儲存資料湖。StorageGRID

NetApp Cloud Volumes ONTAP

NetApp Cloud Volumes ONTAP 的資料管理軟體具備AWS、Google Cloud Platform和Microsoft Azure等公有雲供應商的靈活彈性、可為使用者資料提供控制、保護和效率。NetApp是以NetApp解決方案儲存軟體為基礎打造的雲端原生資料管理軟體、可為使用者提供卓越的通用儲存平台、滿足雲端資料需求。Cloud Volumes ONTAP ONTAP在雲端和內部部署使用相同的儲存軟體、讓使用者能夠享有Data Fabric的價值、而無需訓練IT人員採用全新的方法來管理資料。

對於對混合雲部署模式感興趣的客戶、Cloud Volumes ONTAP 在大多數公有雲中、可提供相同的功能和領先同級的效能、在任何環境中都能提供一致且無縫的使用者體驗。

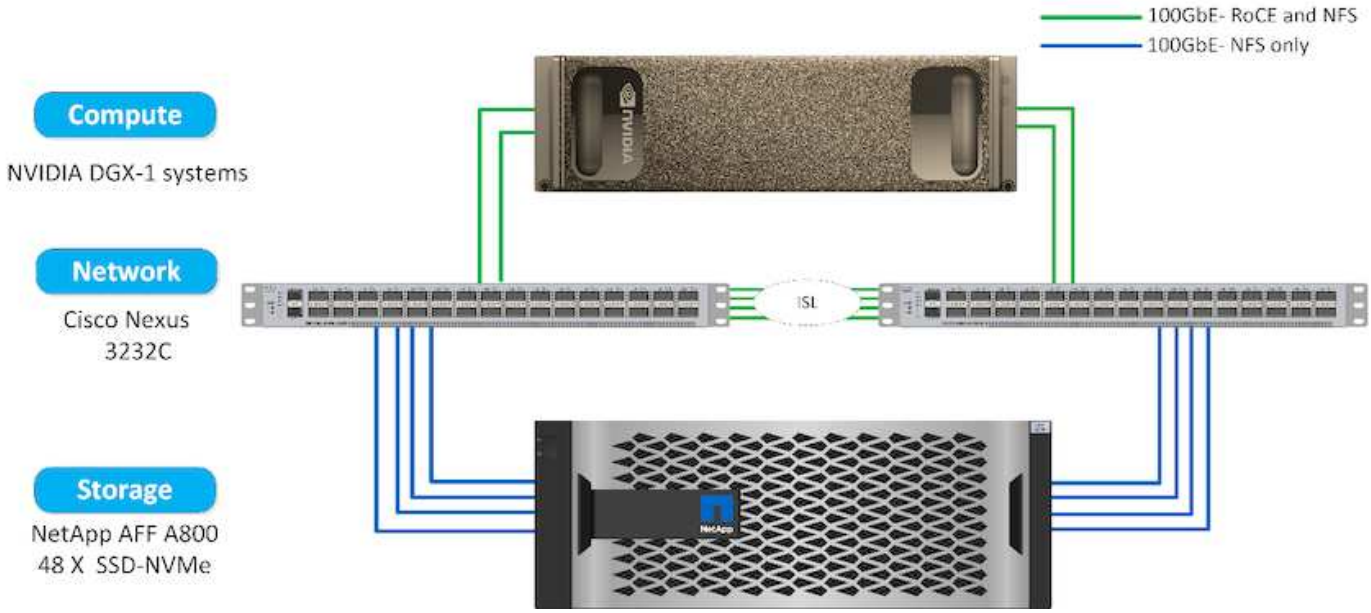
硬體與軟體需求

本節涵蓋ONTAP 有關整個解決方案的技術要求。

硬體需求

雖然硬體需求取決ONTAP 於特定的客戶工作負載、但從單一GPU到機架規模的組態、都能以任何規模部署、以進行資料工程、模型訓練及正式作業提示、以利大規模的ML/DL作業。如需ONTAP 更多關於AI的資訊、請參閱"AI網站ONTAP"。

此解決方案已通過DGX-1系統的運算驗證、NetApp AFF 支援A800儲存系統、以及Cisco Nexus 3232C的網路連線能力驗證。本驗證所使用的功能豐富、最多可支援10個DGX-1系統、以滿足大多數ML/DL工作負載的需求。AFF下圖顯示ONTAP 此驗證中用於模型訓練的AI解決方案。



為了將此解決方案延伸至公有雲、Cloud Volumes ONTAP 可將其與雲端GPU運算資源一起部署、並整合至混合雲資料架構中、讓客戶能夠使用適合任何特定工作負載的任何資源。

軟體需求

下表顯示本解決方案驗證所使用的特定軟體版本。

元件	版本
Ubuntu	18.04.4 LTS
NVIDIA DGX OS	4.4.0
NVIDIA DeepOps	20.02.1
Kubernetes	1.15
掌舵	3.1.0
cnvrg-io	3.0.00.0
NetApp ONTAP	9.6P4

在本解決方案驗證中、Kubernetes已部署為DGX-1系統上的單節點叢集。對於大規模部署、應部署獨立的Kubernetes主節點、以提供高可用度的管理服務、並為ML和DL工作負載保留寶貴的DGX資源。

解決方案部署與驗證詳細資料

下列各節將討論解決方案部署與驗證的詳細資料。

支援AI部署ONTAP

部署AI需要安裝和組態網路、運算和儲存硬體。ONTAP關於部署AI基礎架構的具體指示ONTAP 不在本文的討論範圍之內。如需詳細的部署資訊、請參閱 "NVA-1121-Deploy：採用ONTAP NVIDIA技術的NetApp支援"。

針對此解決方案驗證、已建立單一磁碟區並掛載至DGX-1系統。然後將該掛載點掛載到容器中、以便進行訓練時存取資料。對於大規模部署、NetApp Trident會自動建立及安裝磁碟區、以免除管理成本、並讓終端使用者能夠管理資源。

Kubernetes部署

若要使用NVIDIA DeepOps部署及設定Kubernetes叢集、請從部署跳接主機執行下列工作：

1. 依照上的指示下載NVIDIA DeepOps "入門頁面" 在NVIDIA DeepOps GitHub網站上。
2. 依照上的指示、在叢集中部署Kubernetes "Kubernetes部署指南" 在NVIDIA DeepOps GitHub網站上。

> 若要讓DeepOps Kubernetes部署正常運作、所有Kubernetes主節點和工作節點上都必須有相同的使用者。

如果部署失敗、請在「depops/config/group_vars/k8s-cluster・yml」中、將「kubecl_localhost」的值變更為「假」、然後重複步驟2。只有當值為「kubecl_localhost」時、「Copy kubectl二進位到Ansible host」工作才會執行、這項工作仰賴已知記憶體使用問題的擷取Ansible模組。這些記憶體使用量問題有時可能導致工作失敗。如果工作因為記憶體問題而失敗、則部署作業的其餘部分將無法成功完成。

如果在您將「kubecl_localhost」的值變更為「假」之後、成功完成部署、則必須手動將「kubecl二進位」從Kubernetes主節點複製到部署跳接主機。您可以在特定主節點上直接執行「that kubecll」命令、找到「kubecl二進位」的位置。

Cnvrg-IO部署

使用Helm部署cnvrg核心

使用任何叢集、內部部署、Minikube,或任何雲端叢集（例如、KS、EKS和GKE）、Helm是快速部署cnvrg的最簡單方法。本節說明如何在安裝Kubernetes的內部部署（DGX-1）執行個體上安裝cnvrg。

先決條件

在完成安裝之前、您必須先在本機機器上安裝並準備下列相依項目：

- Kubecl
- helm 3.x
- Kubernetes叢集1.15以上

1. 若要下載最新的cnvrg helm圖表、請執行下列命令：

```
helm repo add cnvrg https://helm.cnvrg.io
helm repo update
```

2. 部署cnvrg之前、您需要叢集的外部IP位址、以及要部署cnvrg的節點名稱。若要在內部部署的Kubernetes叢集上部署cnvrg、請執行下列命令：

```
helm install cnvrg cnvrg/cnvrg --timeout 1500s  --wait \ --set
global.external_ip=<ip_of_cluster> \ --set global.node=<name_of_node>
```

3. 執行「helm install」命令。所有服務和系統都會自動安裝在叢集上。此程序最多可能需要15分鐘。
4. 「helm install」命令最多可能需要10分鐘。部署完成後、請前往新部署的cnvrg的URL、或將新叢集新增為組織內部的資源。「helm」命令會通知您正確的URL。

```
Thank you for installing cnvrg.io!
Your installation of cnvrg.io is now available, and can be reached via:
Talk to our team via email at
```

5. 當所有容器的狀態都在執行或完成時、表示已成功部署cnvrg。其外觀應類似於下列輸出範例：

```
NAME                              READY   STATUS      RESTARTS   AGE
cnvrg-app-69fbb9df98-6xrgf                1/1     Running     0          2m
cnvrg-sidekiq-b9d54d889-5x4fc             1/1     Running     0          2m
controller-65895b47d4-s96v6               1/1     Running     0          2m
init-app-vs-config-wv9c4                  0/1     Completed   0          9m
init-gateway-vs-config-2zbpp              0/1     Completed   0          9m
init-minio-vs-config-cd2rg                0/1     Completed   0          9m
minio-0                                   1/1     Running     0          2m
postgres-0                                1/1     Running     0          2m
redis-695c49c986-kcbt9                    1/1     Running     0          2m
seeder-wh655                              0/1     Completed   0          2m
speaker-5sghr                             1/1     Running     0          2m
```

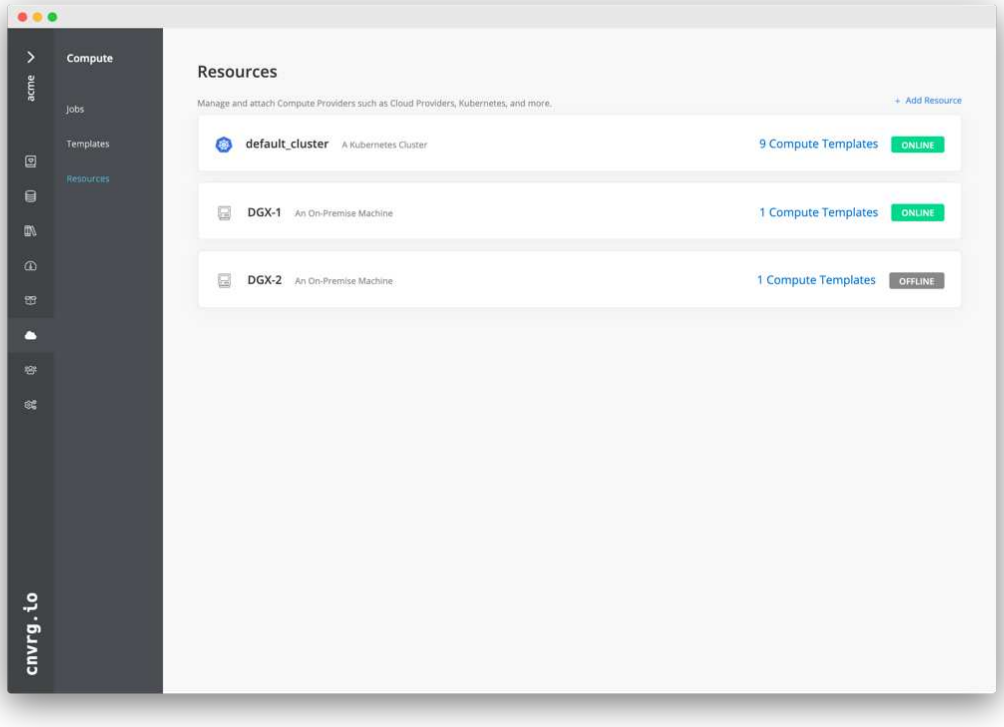
ResNet50和**Chest X**射線資料集的電腦願景模型訓練

Cnvrg-IO AI OS部署在Kubernetes設定上ONTAP 、採用NVIDIA DGX系統的NetApp AI架構上。為了進行驗證、我們使用NIH Chest X光資料集、其中包含已取消識別的胸前X光影像。影像採用的是PNG格式。資料由NIH臨床中心提供、可透過取得 "NIH下載網站"。我們使用250 GB的資料樣本、在15個類別中使用627、615個影像。

此資料集已上傳至cnvrg平台、並從NetApp AFF S16A800儲存系統快取至NFS匯出。

設定運算資源

Cnvrg架構和中繼排程功能可讓工程師和IT專業人員將不同的運算資源附加至單一平台。在我們的設定中、我們使用的叢集cnvrg與執行深度學習工作負載所部署的叢集cnvrg相同。如果您需要附加其他叢集、請使用GUI、如下面的快照所示。



負載資料

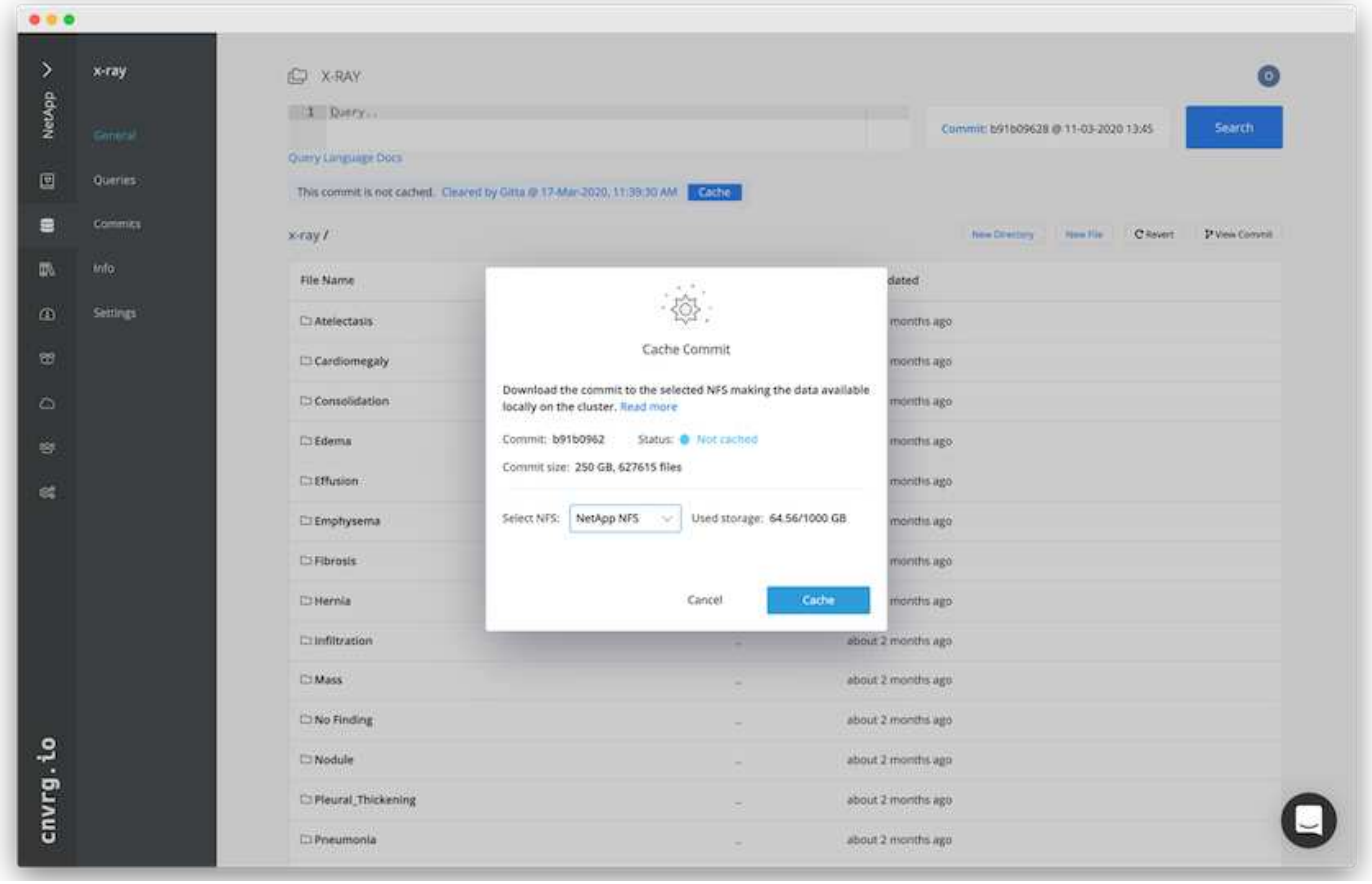
若要將資料上傳至cnvrg平台、您可以使用GUI或cnvrg CLI。對於大型資料集、NetApp建議使用CLI、因為它是強大、可擴充且可靠的工具、可處理大量檔案。

若要上傳資料、請完成下列步驟：

1. 下載 "Cnvrg CLI"。
2. 瀏覽至X光目錄。
3. 使用「cnvrg data init」命令、在平台中初始化資料集。
4. 使用「cnvrg data sync」命令、將目錄的所有內容上傳至中央資料湖。資料上傳至中央物件存放區StorageGRID （例如、S3或其他）之後、您就可以使用GUI瀏覽。下圖顯示已載入的胸前X光纖維化影像PNG檔案。此外、cnvrg會將資料版本轉換成資料版本、以便您建置的任何模型都能複製到資料版本。

Cach資料

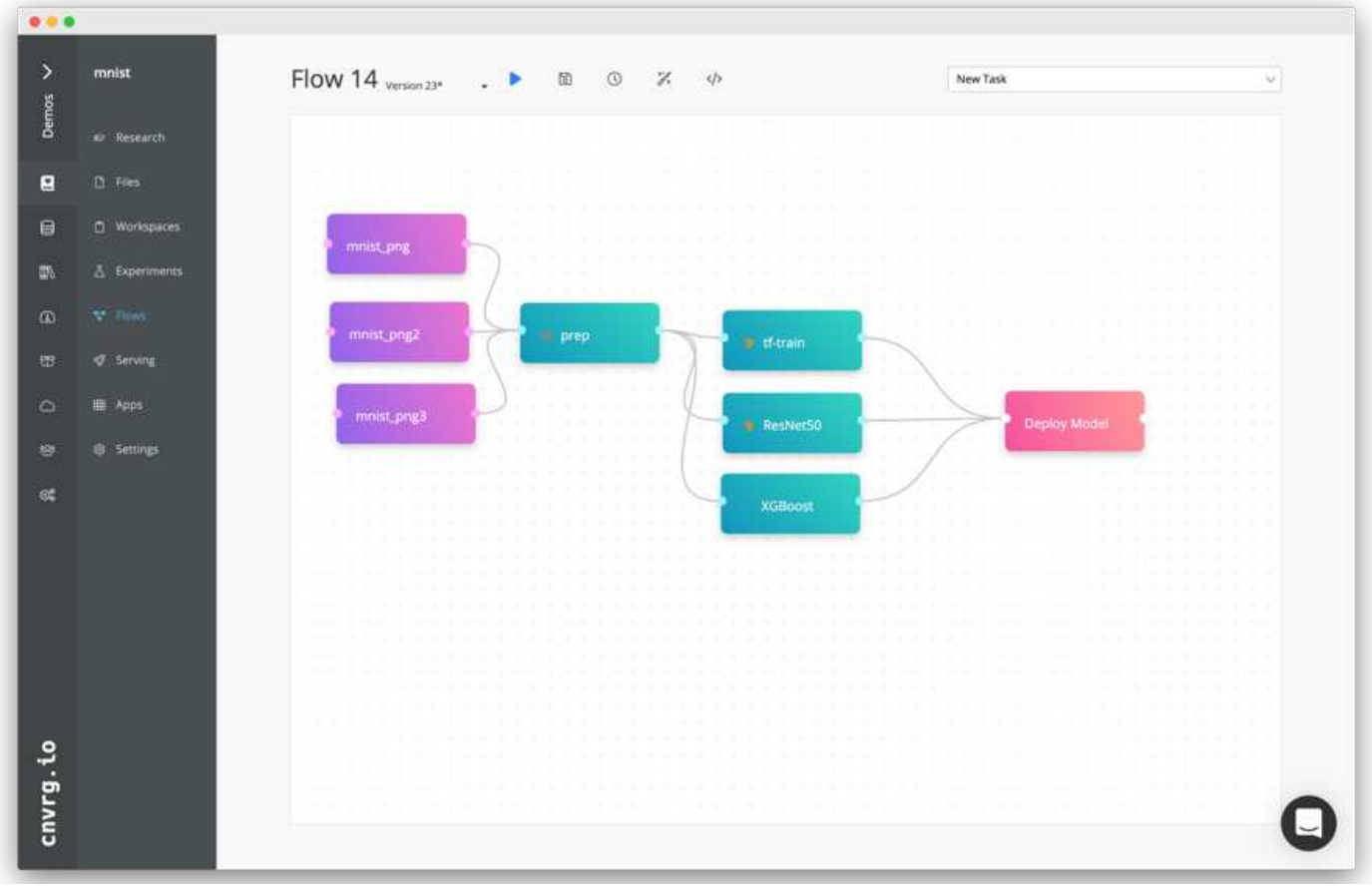
為了加快訓練速度、避免為每個模型訓練和實驗下載超過60萬個檔案、我們在資料一開始上傳至中央資料湖物件存放區之後、就使用了資料快取功能。



使用者按一下「Cache（快取）」後、cnvrg會從遠端物件存放區下載其特定提交的資料、並將其快取至ONTAP「SflexNFS Volume（更新資料）」。資料完成後、即可立即接受訓練。此外、如果資料未使用數天（例如模型訓練或探索）、cnvrg會自動清除快取。

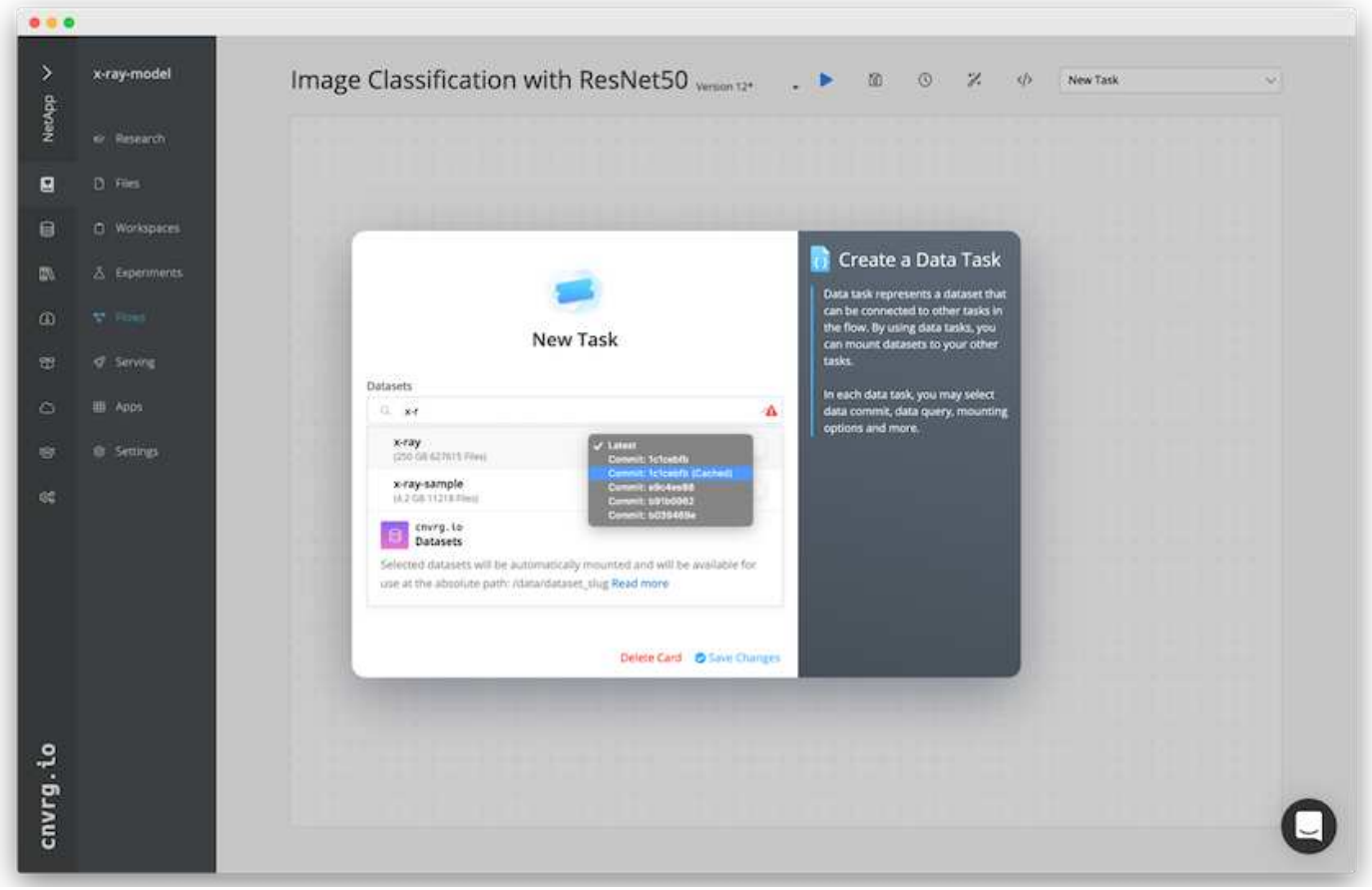
利用快取資料建立**ML**管道

Cnvrg流程可讓您輕鬆建立正式作業ML管線。流程很靈活、可用於任何類型的ML使用案例、並可透過GUI或程式碼建立。流程中的每個元件都能以不同的Docker映像檔在不同的運算資源上執行、因此能夠建置混合雲和最佳化的ML管線。



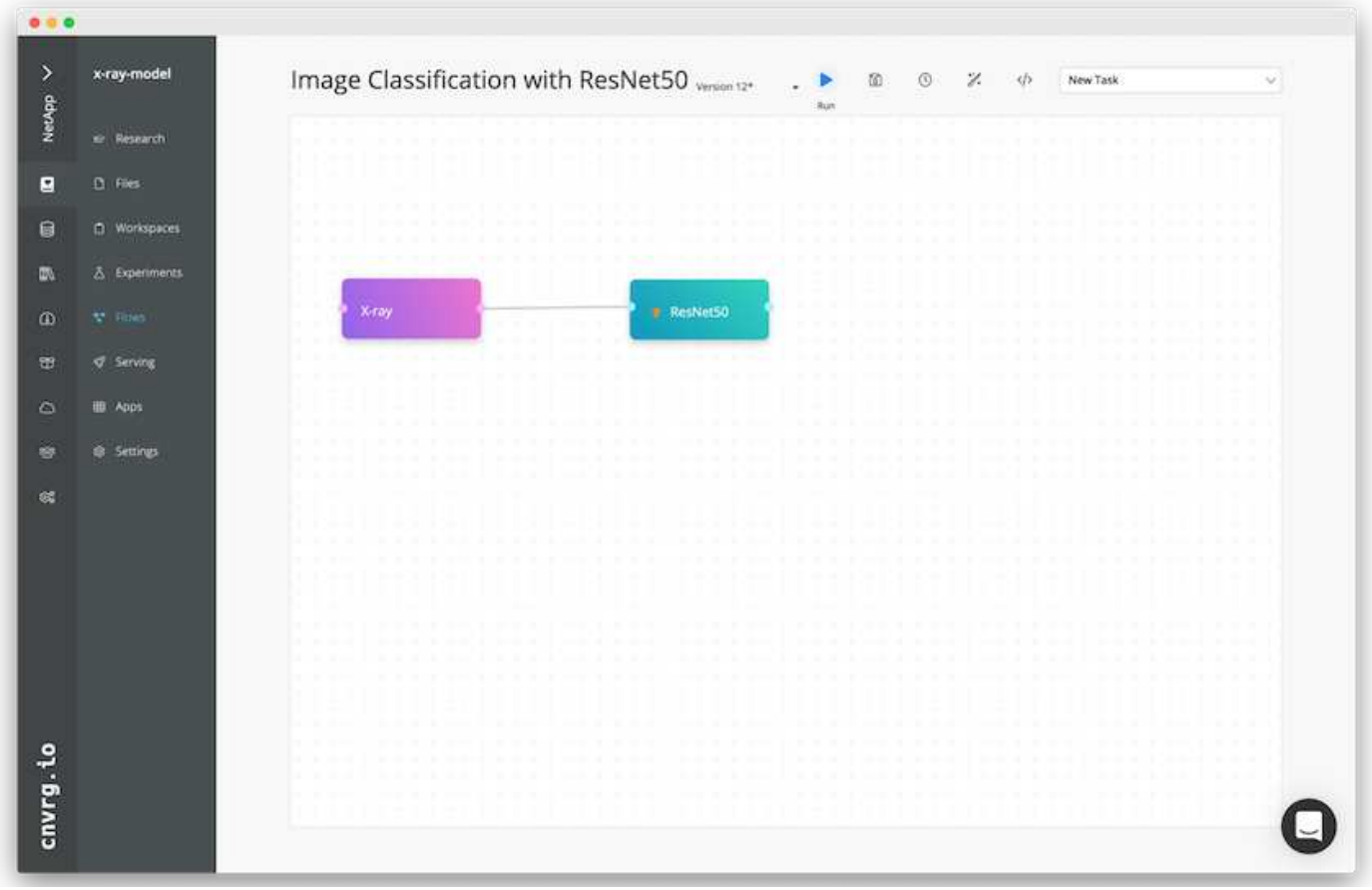
建立胸前**X**光流程：設定資料

我們將資料集新增至新建立的流程。新增資料集時、您可以選取特定版本（提交）、並指出是否要使用快取版本。在此範例中、我們選取了快取的commit。



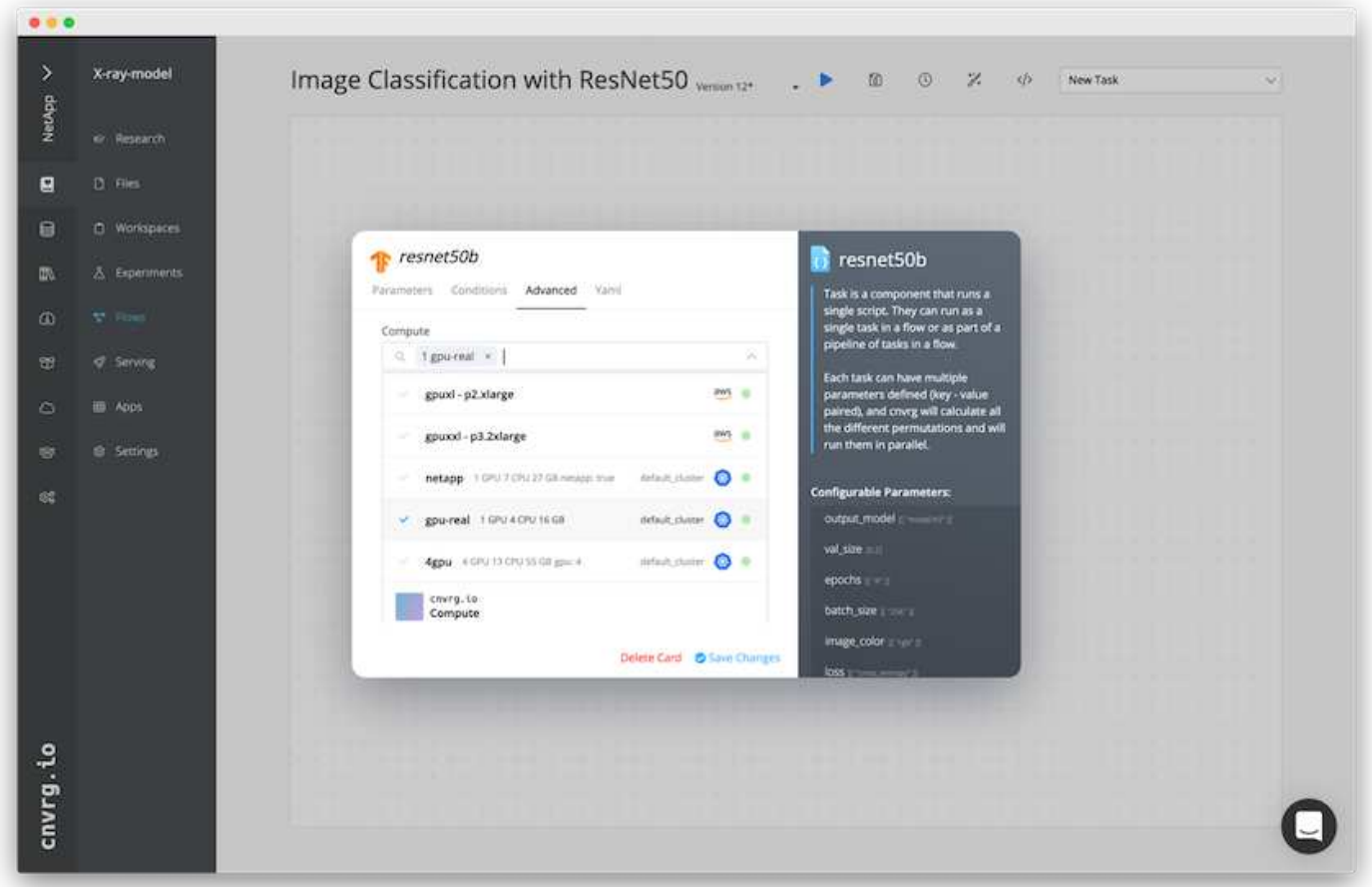
建立胸前**X**光流程：設定訓練模式：**ResNet50**

在管道中、您可以新增任何類型的自訂程式碼。在cnvrg中、還有AI程式庫、可重複使用的ML元件集合。AI程式庫中有演算法、指令碼、資料來源及其他解決方案、可用於任何ML或深度學習流程。在此範例中、我們選擇了預先建置的ResNet50模組。我們使用預設參數、例如batch_Size:128、epochs：10等。這些參數可在AI程式庫文件中檢視。下列螢幕快照顯示新流程、其中X光資料集已連線至ResNet50。



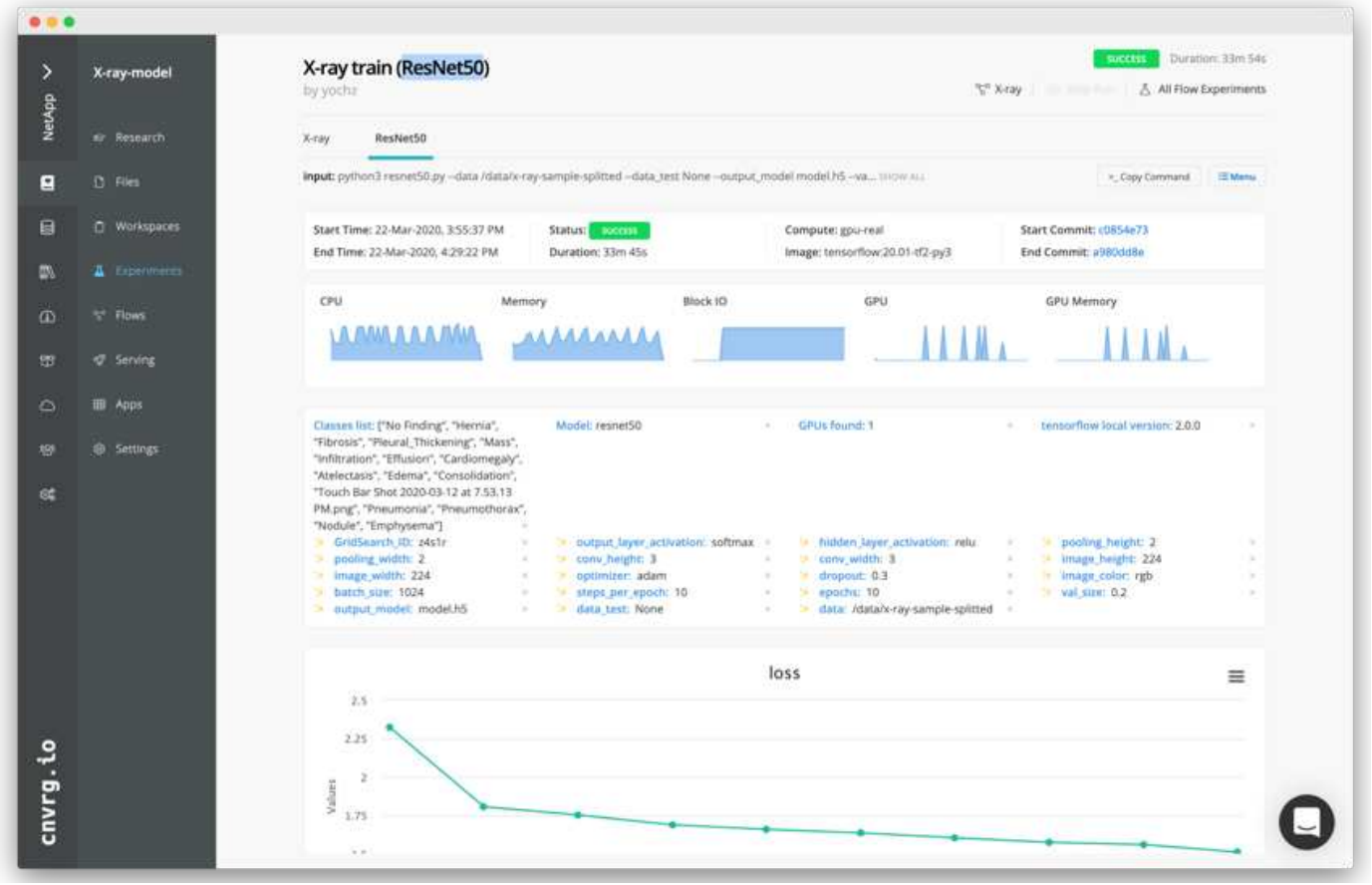
定義**ResNet50**的運算資源

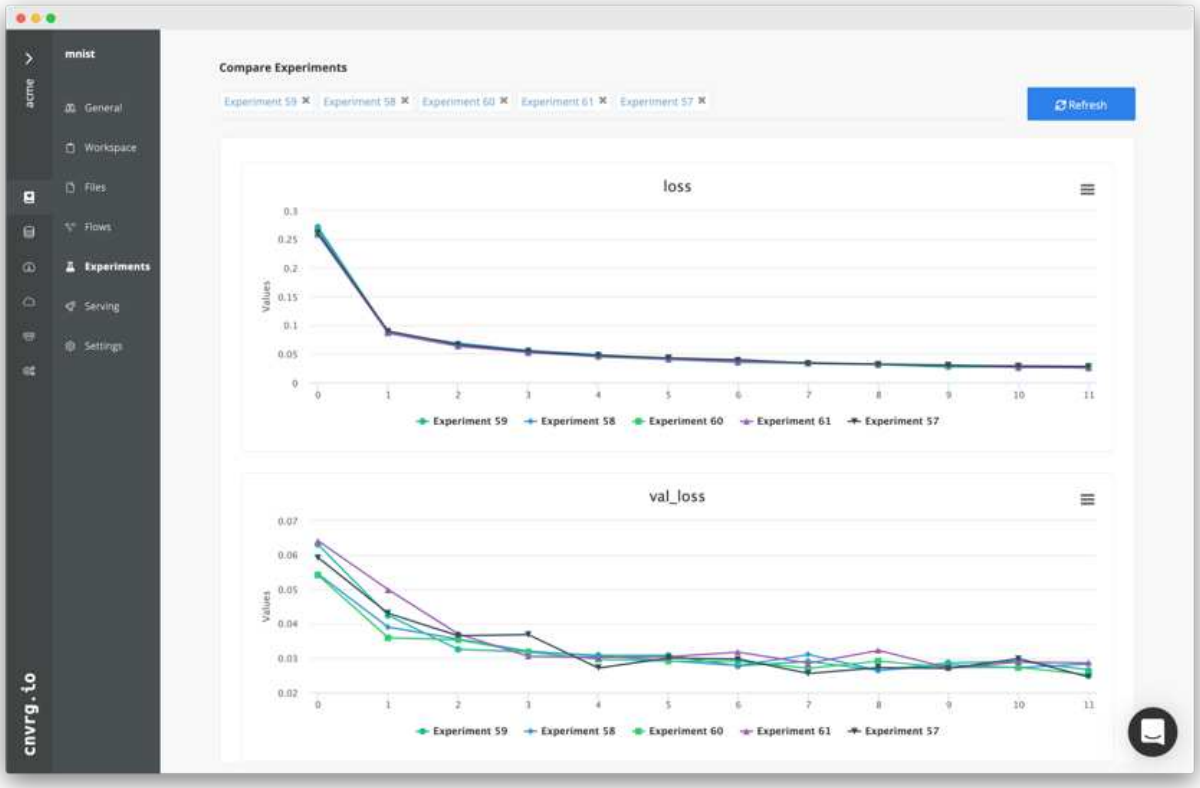
cnvrg流程中的每個演算法或元件都可以在不同的運算執行個體上執行、並使用不同的Docker映像檔。在我們的設定中、我們想要在採用NetApp ONTAP AI架構的NVIDIA DGX系統上執行訓練演算法。在下圖中、我們選擇了「GPU實際」、這是內部部署叢集的運算範本和規格。我們也建立了範本佇列、並選取多個範本。如此一來、如果無法分配「GPU實際」資源（例如其他資料科學家正在使用該資源）、您就可以新增雲端供應商範本來啟用自動雲端資源爆增功能。下列螢幕快照顯示如何使用GPU Real做為ResNet50的運算節點。



追蹤及監控結果

執行流程之後、cnvrg會觸發追蹤與監控引擎。每次流程執行都會自動記錄並即時更新。超參數、度量、資源使用率（GPU使用率等）、程式碼版本、成品、記錄、 「實驗」區段會自動提供這些功能、如下圖所示。


X-ray-model
X-ray train (ResNet50)
by yochz
SUCCESS
Duration: 33m 54s
X-ray
All Flow Experiments
Research
Files
Workspaces
Experiments
Flows
Serving
Apps
Settings
X-ray
ResNet50
input: python3 resnet50.py --data /data/x-ray-sample-splitted --data_test None --output_model model.h5 --va...
Copy Command
Menu
Start Time: 22-Mar-2020, 3:55:37 PM
End Time: 22-Mar-2020, 4:29:22 PM
Status: SUCCESS
Duration: 33m 45s
Compute: gpu-real
Image: tensorflow:20.01-tf2-py3
Start Commit: c0854e73
End Commit: a980dd8e
CPU
Memory
Block IO
GPU
GPU Memory
Classes list: ["No Finding", "Hernia", "Fibrosis", "Pleural_Thickening", "Mass", "Infiltration", "Effusion", "Cardiomegaly", "Atelectasis", "Edema", "Consolidation", "Touch Bar Shot 2020-03-12 at 7.53.13 PM.png", "Pneumonia", "Pneumothorax", "Nodule", "Emphysema"]
Model: resnet50
GPUs found: 1
tensorflow local version: 2.0.0
GridSearch_ID: z4s1r
pooling_width: 2
image_width: 224
batch_size: 1024
output_model: model.h5
output_layer_activation: softmax
conv_height: 3
optimizer: adam
steps_per_epoch: 10
data_test: None
hidden_layer_activation: relu
conv_width: 3
dropout: 0.3
epochs: 10
data: /data/x-ray-sample-splitted
pooling_height: 2
image_height: 224
image_color: rgb
val_size: 0.2
loss
Values
cnvrg.io


acme
mnist
General
Workspace
Files
Flows
Experiments
Serving
Settings
Compare Experiments
Experiment 59
Experiment 58
Experiment 60
Experiment 61
Experiment 57
Refresh
loss
Values
Experiment 59
Experiment 58
Experiment 60
Experiment 61
Experiment 57
val_loss
Values
Experiment 59
Experiment 58
Experiment 60
Experiment 61
Experiment 57
cnvrg.io

結論

NetApp與cnvrg-IO合作、為客戶提供完整的資料管理解決方案、以利ML與DL軟體開發。支援各種規模的作業、均可提供高效能的運算與儲存、而cnvrg-IO軟體則可簡化資料科學工作流程、並改善資源使用率。ONTAP

感謝

- Mike Oglesby、NetApp技術行銷工程師
- NetApp資深技術總監Santosh Rao

何處可找到其他資訊

若要深入瞭解本文所述資訊、請參閱下列資源：

- Cnvrg-IO（"https://cnvrg.io"）：
  - Cnvrg核心（免費ML平台）

    https://cnvrg.io/platform/core
  - Cnvrg文件

    "https://app.cnvrg.io/docs"
- NVIDIA DGX-1伺服器：
  - NVIDIA DGX-1伺服器

    https://www.nvidia.com/en-us/data-center/dgx-1/
  - NVIDIA Tesla V100 Tensor Core GPU

    https://www.nvidia.com/en-us/data-center/tesla-v100/
  - NVIDIA GPU雲端（NGC）

    https://www.nvidia.com/en-us/gpu-cloud/
- NetApp AFF 系統：
  - 資料表AFF

    https://www.netapp.com/us/media/d-3582.pdf
  - NetApp FlashAdvantage for AFF 功能

    https://www.netapp.com/us/media/ds-3733.pdf
  - 2.x文件ONTAP

    http://mysupport.netapp.com/documentation/productlibrary/index.html?productID=62286

- NetApp FlexGroup 技術報告

  https://www.netapp.com/us/media/tr-4557.pdf

- 適用於容器的NetApp持續儲存設備：
  - NetApp Trident

    https://netapp.io/persistent-storage-provisioner-for-kubernetes/

- NetApp互通性對照表：
  - NetApp 互通性對照表工具

    http://support.netapp.com/matrix

- AI網路：ONTAP
  - Cisco Nexus 3232C交換器

    https://www.cisco.com/c/en/us/products/switches/nexus-3232c-switch/index.html

  - Mellanox Spectrum 2000系列交換器

    http://www.mellanox.com/page/products_dyn?product_family=251&mtag=sn2000

- ML架構與工具：
  - 達利

    https://github.com/NVIDIA/DALI

  - TensorFlow：適用於所有人的開放原始碼機器學習架構

    https://www.tensorflow.org/

  - Horovod：Uber的開放原始碼分散式深度學習架構、適用於TensorFlow

    https://eng.uber.com/horovod/

  - 在Container執行時間生態系統中啟用GPU

    https://devblogs.nvidia.com/gpu-containers-runtime/

  - Docker

    https://docs.docker.com

  - Kubernetes

    https://kubernetes.io/docs/home/

  - NVIDIA DeepOps

    https://github.com/NVIDIA/deepops

- Kubeflow

  http://www.kubeflow.org/

- Jupyter筆記型電腦伺服器

  http://www.jupyter.org/

- 資料集與基準測試：
  - NIH胸前X光資料集

    https://nihcc.app.box.com/v/ChestXray-NIHCC

  - 王小鬆、彭葉文、盧、陸志勇、MohammadhADI Bagheri、 Ronald Summers、ChestX-RAY 8：醫院規模的ChestX-Ray X光資料庫、以及一般胸病弱監督分類與本地化的基準測試、IEEE CVPR、第頁3462-3471、2017TR-4841-0620