# データキャッシング機能を備えたハイブリッドクラウド AI オペレーティングシステム

## NetApp Solutions

NetApp
September 10, 2024

# 目次

# TR-4841 ：『 Hybrid Cloud AI Operating System with Data Caching 』

ネットアップ Yochay Ettun 、 cnvrg.io 、 David Arnette 、 Rick Huang 氏

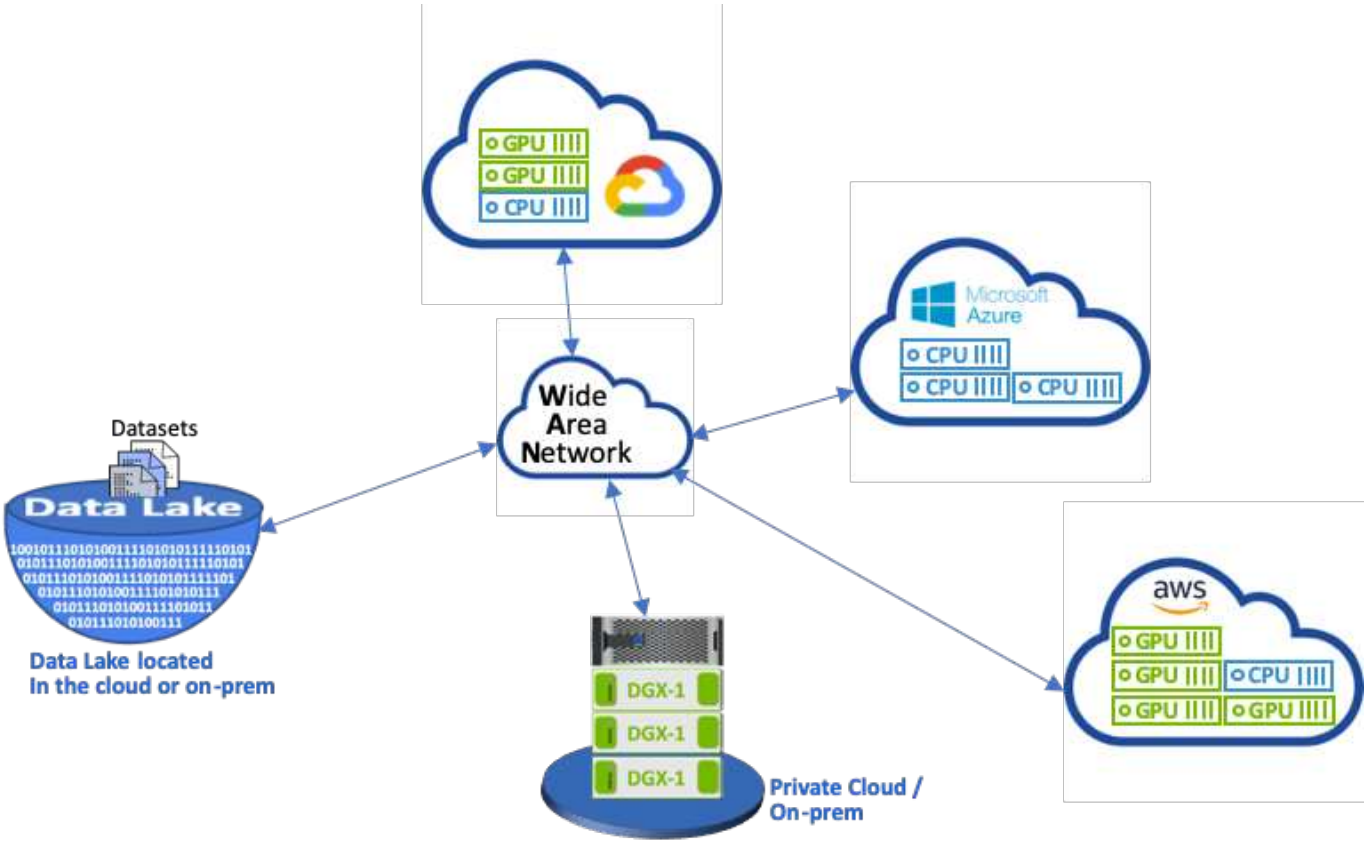
データの急増と ML と AI の急激な成長により、独自の開発と実装の課題を抱えるゼタバイト経済が生まれました。

ML モデルは大量のデータを必要とし、コンピューティングリソースにはハイパフォーマンスのデータストレージが必要であることは広く知られていますが、実際には、このモデルを実装するのはそれほど簡単ではありません。特にハイブリッドクラウドインスタンスや柔軟なコンピューティングインスタンスを使用する場合はそうです。一般に、大量のデータが低コストのデータレイクに保存されます。このデータレイクでは、 GPU などのハイパフォーマンスな AI コンピューティングリソースは効率的にアクセスできません。この問題は、一部のワークロードがクラウドで動作し、一部のワークロードがオンプレミス環境または別の HPC 環境に完全に配置されているハイブリッドクラウドインフラにさらに悪化しています。

このドキュメントでは、 IT プロフェッショナルやデータエンジニアがトポロジに対応したデータハブで真のハイブリッドクラウド AI プラットフォームを構築できる、新しい解決策を紹介します。これにより、データサイエンティストは、コンピューティングリソースに近接してデータセットのキャッシュを瞬時に自動作成できます。 どこにいても、その結果、高性能なモデルトレーニングを実施できるだけでなく、データセットバージョンハブ内のデータセットキャッシュ、バージョン、リネージにすぐにアクセスできる複数の AI 専門家のコラボレーションなど、さらなるメリットが得られます。

## ユースケースの概要と問題点

データセットとデータセットのバージョンは通常、 NetApp StorageGRID オブジェクトベースストレージなどのデータレイクに配置されるため、コストの削減やその他の運用上のメリットが得られます。データサイエンティストは、これらのデータセットを取得して複数の手順でエンジニアを配置し、特定のモデルを使用したトレーニングに備えます。多くの場合、途中で複数のバージョンが作成されます。次のステップとして、データサイエンティストは、モデルを実行するために最適化されたコンピューティングリソース（ GPU 、ハイエンド CPU インスタンス、オンプレミスクラスタなど）を選択する必要があります。次の図は、 ML コンピューティング環境にデータセットの距離がないことを示しています。




ただし、複数のトレーニング実験を異なるコンピューティング環境で並行して実行する必要があります。それぞれの環境では、データレイクからデータセットをダウンロードする必要があります。これはコストと時間のかかるプロセスです。データセットがコンピューティング環境（特にハイブリッドクラウド）に近接していることは保証されません。また、同じデータセットで独自の実験を行う他のチームメンバーも、同じ複雑なプロセスを実行する必要があります。データアクセスが遅いことが明らかなだけでなく、データセットのバージョン、データセットの共有、コラボレーション、再現性の追跡にも困難が伴います。

## お客様の要件

リソースを効率的に使用しながら、高パフォーマンスの ML を実行するためには、お客様の要件が異なる場合があります。たとえば、次のような場合があります。

- を実行する各コンピューティングインスタンスからデータセットに高速アクセス 高額なダウンロードやデータアクセスの複雑さを伴わないトレーニングモデル
- は任意のコンピューティングインスタンス（ GPU または CPU ）を使用する クラウドでもオンプレミスでも、場所を気にする必要はありません 」と入力します
- で複数のトレーニング実験を実行することで、効率と生産性が向上します を使用せずに、同一データセット上の異なるコンピューティングリソースと並行して実行できます 不要な遅延とデータ遅延
- コンピューティングインスタンスのコストを最小限に抑えます
- データセット、そのリネージ、バージョン、およびその他のメタデータの詳細の記録を保持するツールにより、再現性が向上しました
- 共有とコラボレーションを強化して、の権限を持つすべてのメンバーをサポートします チームはデータセットにアクセスして実験を実行できます

NetApp ONTAP データ管理ソフトウェアにデータセットのキャッシングを実装するには、次のタスクを実行

する必要があります。

- コンピューティングリソースに最も近い NFS ストレージを構成して設定します。
- キャッシュするデータセットとバージョンを決定します。
- キャッシュされたデータセットにコミットされた合計メモリと、追加のキャッシュコミットに使用できる NFS ストレージの量（キャッシュ管理など）を監視します。
- 特定の時間内に使用されなかったデータセットは、キャッシュ内でエージングアウトします。デフォルトは 1 日で、その他の設定オプションも使用できます。

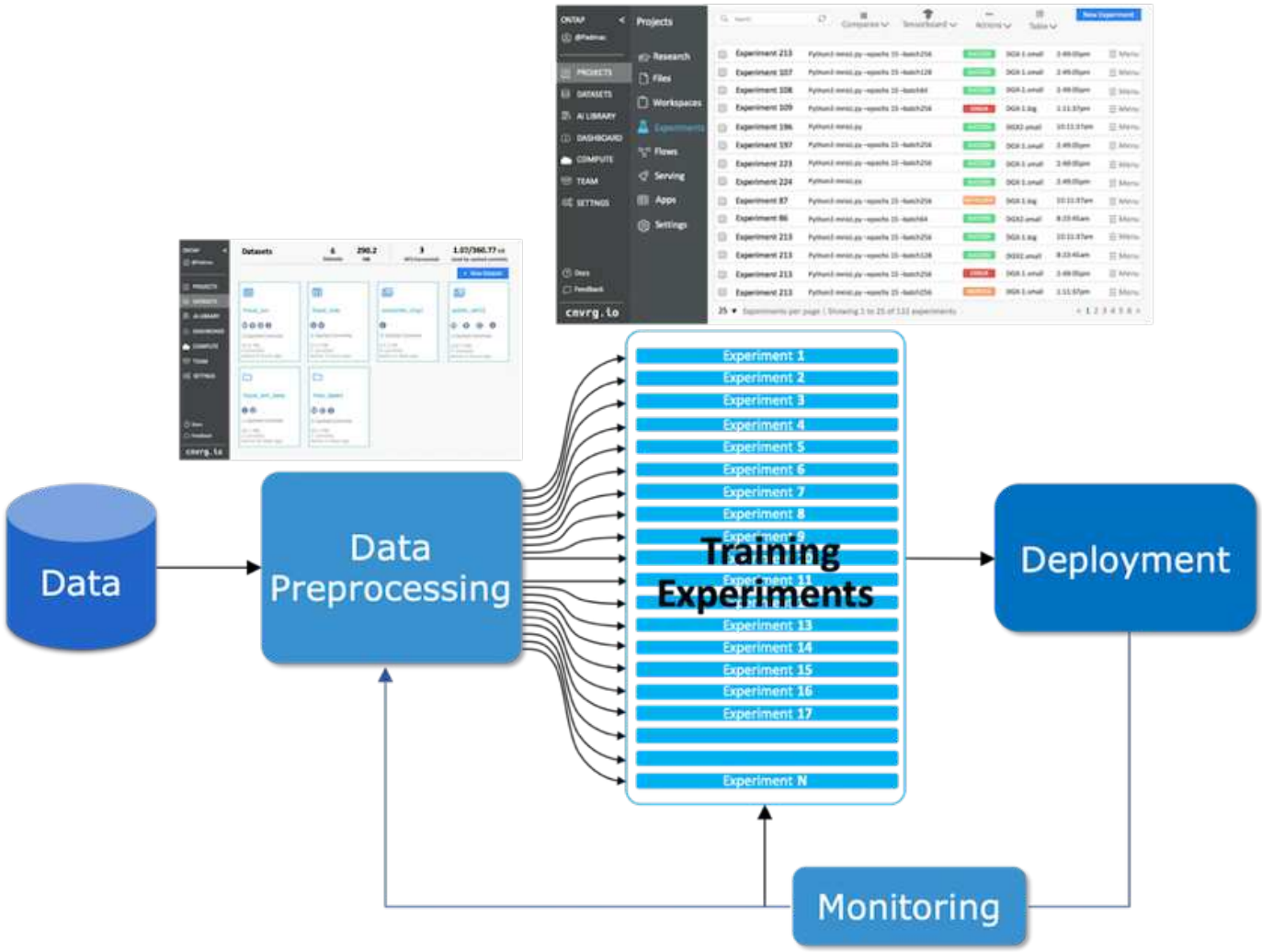
# 解決策の概要

このセクションでは、従来のデータサイエンスパイプラインとその欠点について説明します。また、提案するデータセットキャッシング解決策のアーキテクチャについても説明します。

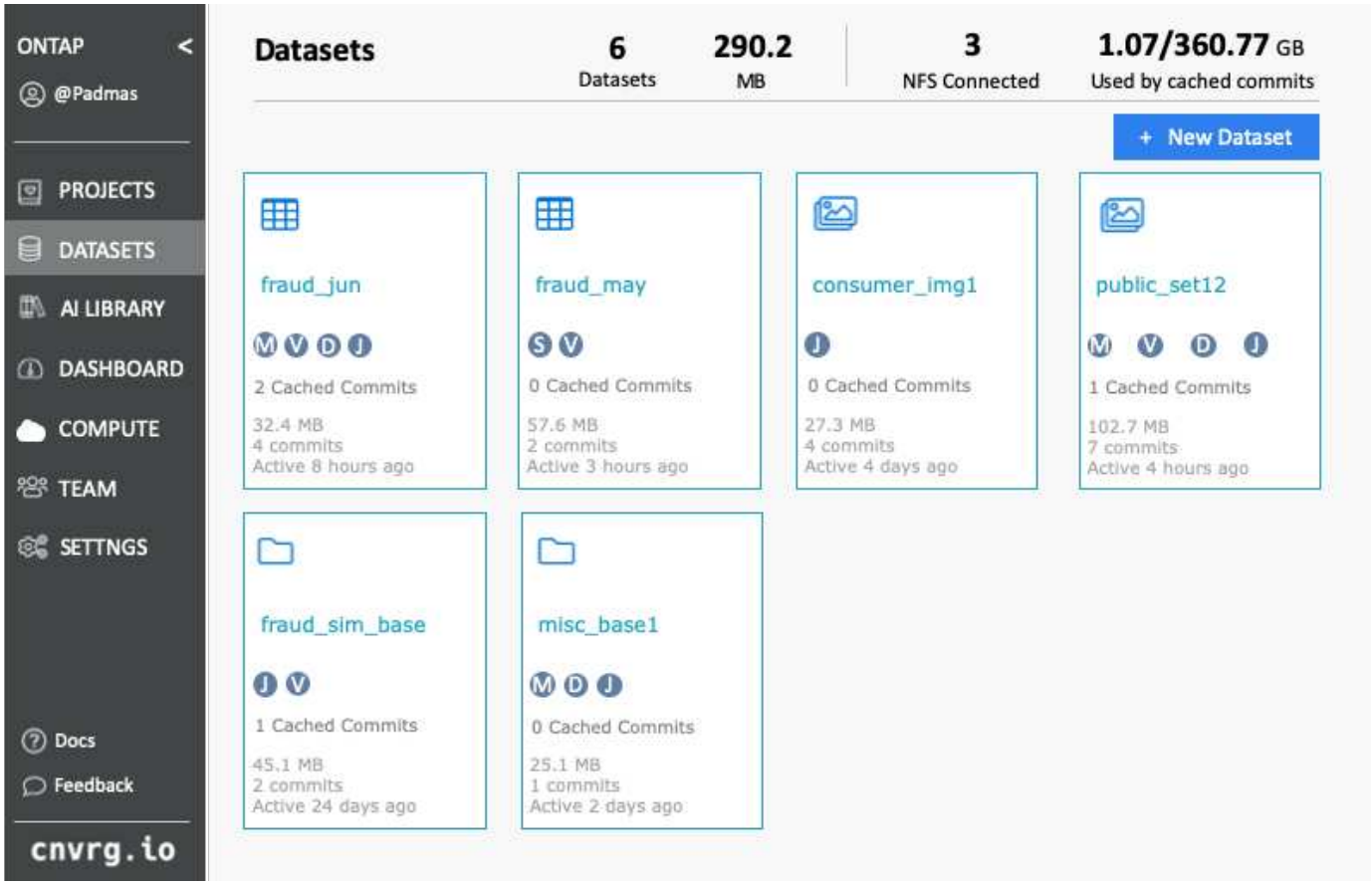
## 従来のデータサイエンスパイプラインと欠点

ML モデルの開発と導入の一般的な手順には、次のような反復的な手順が含まれます。

- データの取り込み
- データの前処理（データセットの複数のバージョンを作成）
- HyperParameter の最適化、さまざまなモデルなどを含む複数の実験を実行する
- 導入
- Monitoringcnvrg.io は、研究から導入までのすべてのタスクを自動化する包括的なプラットフォームを開発しました。次の図に、パイプラインに関するダッシュボードのスクリーンショットのごく一部を示します。




パブリックリポジトリやプライベートデータから複数のデータセットを使用するのは非常に一般的です。また、データセットのクリーンアップやフィーチャーエンジニアリングによって、各データセットに複数のバージョンが生成されることもよくあります。次の図に示すように、データセットハブとバージョンハブを提供するダッシュボードは、コラボレーションツールと整合性ツールをチームで確実に使用できるようにするために必要です。




パイプラインの次のステップではトレーニングを行います。トレーニングモデルには複数の並行インスタンスが必要で、それぞれがデータセットと特定のコンピューティングインスタンスに関連付けられている必要があります。データセットを特定のコンピューティングインスタンスと特定の実験にバインドすることは、一部の実験は Amazon Web Services （ AWS ）の GPU インスタンスによって実行され、それ以外の実験は DGX-1 インスタンスまたは DGX-2 インスタンスによってオンプレミスで実行される可能性があるため、難しい課題です。GCP の CPU サーバーでは他の実験が実行され、データセットの場所がトレーニングを実行するコンピューティングリソースにあまり近接していない場合があります。合理的なプロキシミティには、データセットストレージからコンピューティングインスタンスへの完全な 10GbE 以上の低レイテンシ接続が必要です。

データサイエンティストが、トレーニングを実行し、実験を実行するコンピューティングインスタンスにデータセットをダウンロードするのは、一般的に行われます。ただし、この方法にはいくつかの潜在的な問題があります。

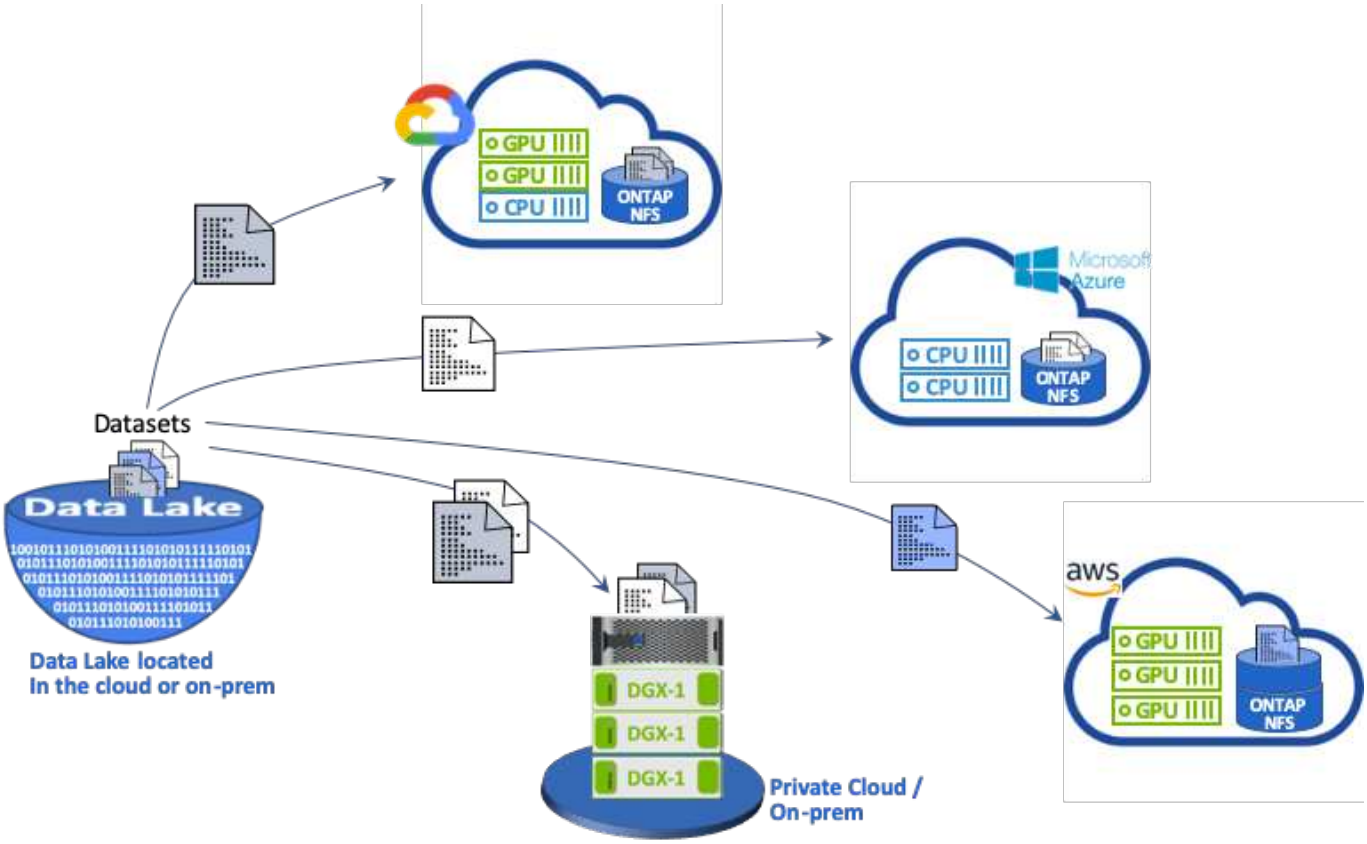
- データサイエンティストがデータセットをコンピューティングインスタンスにダウンロードした場合、統合コンピューティングストレージのパフォーマンスが高くても保証はありません（ハイパフォーマンスシステムの例としては ONTAP AFF A800 NVMe 解決策が挙げられます）。
- ダウンロードしたデータセットが 1 つのコンピューティングノードに存在する場合、複数のノードで分散モデルを実行すると（ NetApp ONTAP のハイパフォーマンス分散ストレージとは異なり）ストレージがボトルネックになることがあります。
- トレーニング実験の次の反復は、キューの競合や優先順位のために別のコンピューティングインスタンスで実行される場合もあります。これも、データセットから計算場所へのネットワーク距離が大幅に大きくなります。
- 同じコンピューティングクラスタ上でトレーニング実験を実行する他のチームメンバーは、このデータセットを共有できません。各チームメンバーは、任意の場所からデータセットの（高価な）ダウンロードを実行します。

- 後続のトレーニングジョブで同じデータセットの他のデータセットまたはバージョンが必要な場合、データサイエンティストは、 training.NetApp および cnvrg.io を実行しているコンピューティングインスタンスにデータセットの（高価な）ダウンロードを再度実行する必要があります。これにより、これらの障害を解消する新しいデータセットキャッシング解決策が作成されます。解決策は、ホットデータセットを ONTAP ハイパフォーマンスストレージシステムにキャッシュすることで、 ML パイプラインの実行を高速化します。ONTAP NFS では、ネットアップが提供するデータファブリック（ AFF A800 など）にデータセットが 1 回だけ（一度だけ）キャッシュされ、コンピューティングと一緒に配置されます。NetApp ONTAP NFS 高速ストレージが複数の ML コンピューティングノードに対応できるようになるため、トレーニングモデルのパフォーマンスが最適化され、コスト削減、生産性、運用効率が向上します。

## 解決策アーキテクチャ

この解決策は、次の図に示すように、ネットアップおよび cnvrg.io から提供されます。データセットのキャッシングにより、データサイエンティストは必要なデータセットまたはデータセットのバージョンを選択し、ML コンピューティングクラスタのすぐ近くにある ONTAP NFS キャッシュに移動できます。データサイエンティストは、遅延やダウンロードを発生させることなく、複数の実験を実行できるようになりました。さらに、コラボレーションするすべてのエンジニアは、データレイクから追加のダウンロードを行うことなく、接続されたコンピューティングクラスタ（任意のノードを自由に選択できる）で同じデータセットを使用できます。データサイエンティストは、すべてのデータセットとバージョンを追跡および監視するダッシュボードを提供し、キャッシュされたデータセットを確認します。

cnvrg.io プラットフォームは、一定の期間使用されていない古いデータセットを自動検出し、キャッシュから削除します。これにより、使用頻度の高いデータセット用に NFS キャッシュの空きスペースが維持されます。ONTAP を使用したデータセットのキャッシングは、クラウドとオンプレミスで機能するため、最大限の柔軟性が得られることに注意してください。



# コンセプトとコンポーネント

このセクションでは、 ML ワークフローのデータキャッシングに関連する概念とコンポーネントについて説明します。

## 機械学習

ML は、世界中の多くの企業や組織にとって急速に不可欠になっています。そのため、 IT チームと DevOps チームは、 ML ワークロードの標準化や、 ML のジョブやパイプラインで求められる動的で負荷の高いワークフローをサポートするクラウド、オンプレミス、ハイブリッドコンピューティングリソースのプロビジョニングという課題に直面しています。

## コンテナベースの機械学習と **Kubernetes**

コンテナは、共有ホストオペレーティングシステムカーネル上で実行される独立したユーザスペースインスタンスです。コンテナの採用が急速に増加しています。コンテナは、仮想マシン（ VM ）が提供するものと同じアプリケーションのサンドボックス化のメリットの多くを提供します。ただし、 VM が依存するハイパーバイザーレイヤとゲストオペレーティングシステムレイヤが排除されているため、コンテナの軽量化が大幅に向上しています。

コンテナを使用すると、アプリケーションの依存関係や実行時間などをアプリケーションで直接効率的にパッケージングできます。最も一般的に使用されるコンテナパッケージ形式は Docker コンテナです。Docker コンテナ形式でコンテナ化されたアプリケーションは、 Docker コンテナを実行できる任意のマシンで実行できます。これは、アプリケーションの依存関係がマシンに存在しない場合でも当てはまります。これは、すべての依存関係がコンテナ自体にパッケージ化されているためです。詳細については、を参照してください "Docker Web サイト"。

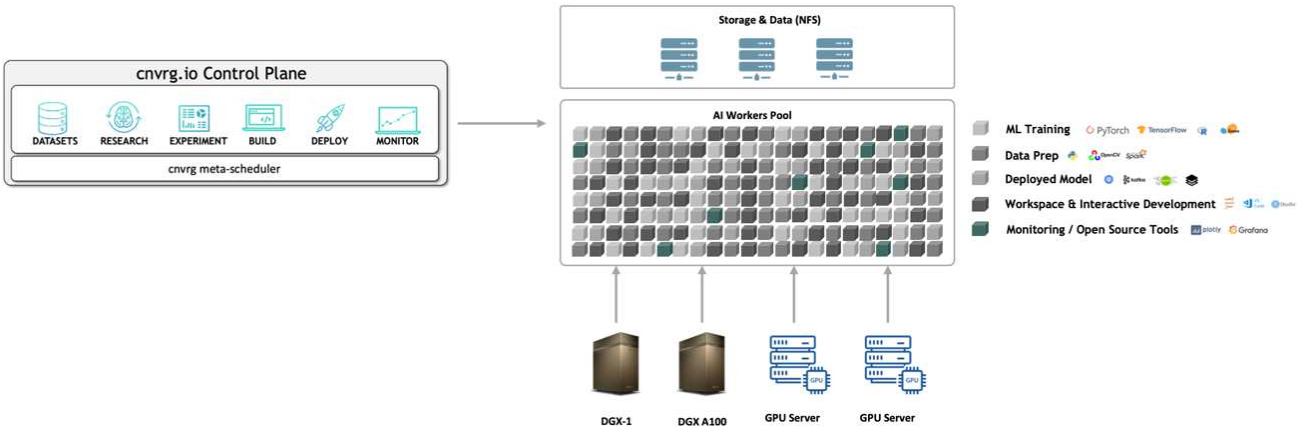
人気のあるコンテナオーケストレーションツールである Kubernetes を使用すると、データサイエンティストは柔軟なコンテナベースのジョブとパイプラインを開始できます。また、インフラチームは、管理された単一のクラウドネイティブ環境で ML ワークロードを管理および監視できます。詳細については、を参照してください "Kubernetes Web サイト"。

## cnvrg.io

cnvrg.io は、企業が AI やデータサイエンスの開発を研究から生産に至るまで管理、拡張、高速化する方法を変革する AI オペレーティングシステムです。コードファーストのプラットフォームは、データサイエンティストがデータサイエンティストのために構築し、オンプレミスとクラウドのどちらでも実行できる柔軟性を提供します。モデル管理、 MLOps 、継続的な ML ソリューションを備えた cnvrg.io は、データサイエンスチームに最先端のテクノロジを提供します。そのため、 DevOps に費やす時間を短縮し、真の魔法のアルゴリズムに集中できます。cnvrg.io を使用して以来、さまざまな業界のチームが生産モデルを増やし、ビジネス価値を高めてきました。

### **cnvrg.io** メタスケジューラ

cnvrg.IO には独自のアーキテクチャがあり、 IT エンジニアは異なるコンピューティングリソースを同じコントロールプレーンに接続し、すべてのリソースにわたって cnvrg.io で ML ジョブを管理できます。つまり、次の図に示すように、複数のオンプレミス Kubernetes クラスタ、 VM サーバ、クラウドアカウントを接続し、すべてのリソースで ML ワークロードを実行できます。




**cnvrg.io** データキャッシング

cnvrg.io を使用すると、データサイエンティストは、データキャッシングテクノロジを使用して、ホットデータセットとコールドデータセットのバージョンを定義できます。デフォルトでは、データセットは一元化されたオブジェクトストレージデータベースに格納されます。データサイエンティストは、選択したコンピューティングリソースに特定のデータバージョンをキャッシュして、ダウンロード時間を節約し、 ML の開発と生産性を向上させることができます。数日間キャッシュされていないデータセットは、選択した NFS から自動的に消去されます。キャッシュのキャッシュとクリアはワンクリックで実行でき、コーディング、 IT 、 DevOps の作業は必要ありません。

**cnvrg.io** フローと **ML** パイプライン

cnvrg.io フローは、本番 ML パイプラインを構築するためのツールです。フロー内の各コンポーネントは、ベースとなる Docker イメージを使用して選択したコンピューティング上で実行されるスクリプト / コードです。この設計により、データサイエンティストとエンジニアは、オンプレミスとクラウドの両方で実行できる単一のパイプラインを構築できます。cnvrg.io は、データ、パラメータ、およびアーティファクトが異なるコンポーネント間で移動していることを確認します。さらに、各フローを監視して追跡することで、再現性の高い 100% のデータサイエンスを実現します。

**cnvrg.io** コア

cnvrg.io コアは、データサイエンスコミュニティが DevOps よりもデータサイエンスに集中できるようにするための無償プラットフォームです。コアの柔軟なインフラストラクチャにより、データサイエンティストは、オンプレミスでもクラウドでも、あらゆる言語、 AI フレームワーク、コンピューティング環境を使用することができます。これにより、最適な処理を実行し、アルゴリズムを構築できます。cnvrg.io コアは、任意の Kubernetes クラスタ上で 1 つのコマンドを使用して簡単にインストールできます。

## NetApp ONTAP AI

ONTAP AI は、 ML ワークロードとディープラーニング（ DL ）ワークロード向けのデータセンターリファレンスアーキテクチャであり、 Tesla V100 GPU を搭載した NetApp AFF ストレージシステムと NVIDIA DGX システムを使用します。ONTAP AI は、業界標準の NFS ファイルプロトコルである 100Gb イーサネットを基盤としており、標準的なデータセンターテクノロジを使用して実装や管理のオーバーヘッドを軽減する、ハイパフォーマンスな ML / DL インフラを提供します。標準化されたネットワークとプロトコルを使用することで、 ONTAP AI をハイブリッドクラウド環境に統合しながら、運用の一貫性と簡易性を維持できます。解決策 AI は、事前検証済みのインフラ ONTAP として、導入にかかる時間とリスクを削減し、管理オーバーヘッドを大幅に削減することで、お客様はより短期間で価値を実現できます。

### NVIDIA DeepOps のことです

DeepOps は NVIDIA が開発したオープンソースプロジェクトです。 Ansible を使用することで、ベストプラクティスに従って GPU サーバクラスタの導入を自動化できます。DeepOps はモジュール方式であり、さまざまな導入タスクに使用できます。このドキュメントとこの検証の演習では、 DeepOps を使用して、 GPU サーバワーカーノードで構成される Kubernetes クラスタを導入します。詳細については、を参照してください "DeepOps の Web サイト"。

### NetApp Trident

Trident は、ネットアップが開発および管理しているオープンソースのストレージオーケストレーションツールで、 Kubernetes ワークロード向けの永続的ストレージの作成、管理、使用を大幅に簡易化します。Trident 自体は Kubernetes ネイティブのアプリケーションであり、 Kubernetes クラスタ内で直接実行されます。Trident を使用すると、 Kubernetes のユーザ（開発者、データサイエンティスト、 Kubernetes 管理者など）は、使い慣れた標準的な Kubernetes 形式で永続ストレージボリュームを作成、管理、操作できます。同時に、ネットアップの高度なデータ管理機能と、ネットアップテクノロジを基盤とするデータファブリックを活用できます。Trident は、複雑な永続的ストレージを抽象化して、消費を簡易化します。詳細については、を参照してください "Trident の Web サイト"。

### NetApp StorageGRID

NetApp StorageGRID は、ユーザが S3 プロトコルを使用してアクセスできるシンプルなクラウド型ストレージを提供することで、これらのニーズを満たすように設計された Software-Defined オブジェクトストレージプラットフォームです。StorageGRID は、距離に関係なく、インターネットに接続されたサイト全体で複数のノードをサポートするように設計されたスケールアウトシステムです。StorageGRID のインテリジェントポリシーエンジンを使用すると、サイト間でイレイジャーコーディングオブジェクトを選択して地理的な耐障害性を確保したり、リモートサイト間でオブジェクトレプリケーションを行ったりすることで、 WAN アクセスのレイテンシを最小限に抑えることができます。StorageGRID は、この解決策にある優れたプライベートクラウドプライマリオブジェクトストレージデータレイクを提供します。

### NetApp Cloud Volumes ONTAP の略

NetApp Cloud Volumes ONTAP データ管理ソフトウェアは、 AWS 、 Google Cloud Platform 、 Microsoft Azure などのパブリッククラウドプロバイダの柔軟性を活かして、ユーザデータの制御、保護、効率化を実現します。Cloud Volumes ONTAP は、 NetApp ONTAP ストレージソフトウェアを基盤としたクラウドネイティブなデータ管理ソフトウェアで、クラウドデータのニーズに対応する、汎用性に優れた優れたストレージプラットフォームをユーザに提供します。クラウドとオンプレミスで同じストレージソフトウェアを使用することで、ユーザはデータファブリックの価値を活用できます。まったく新しいデータ管理方法について IT 担当者をトレーニングする必要はありません。

ハイブリッドクラウドの導入モデルに関心があるお客様は、 Cloud Volumes ONTAP を使用することで、ほとんどのパブリッククラウドで同じ機能とクラス最高のパフォーマンスを実現し、一貫したシームレスなユーザエクスペリエンスをあらゆる環境で実現できます。

# ハードウェアとソフトウェアの要件

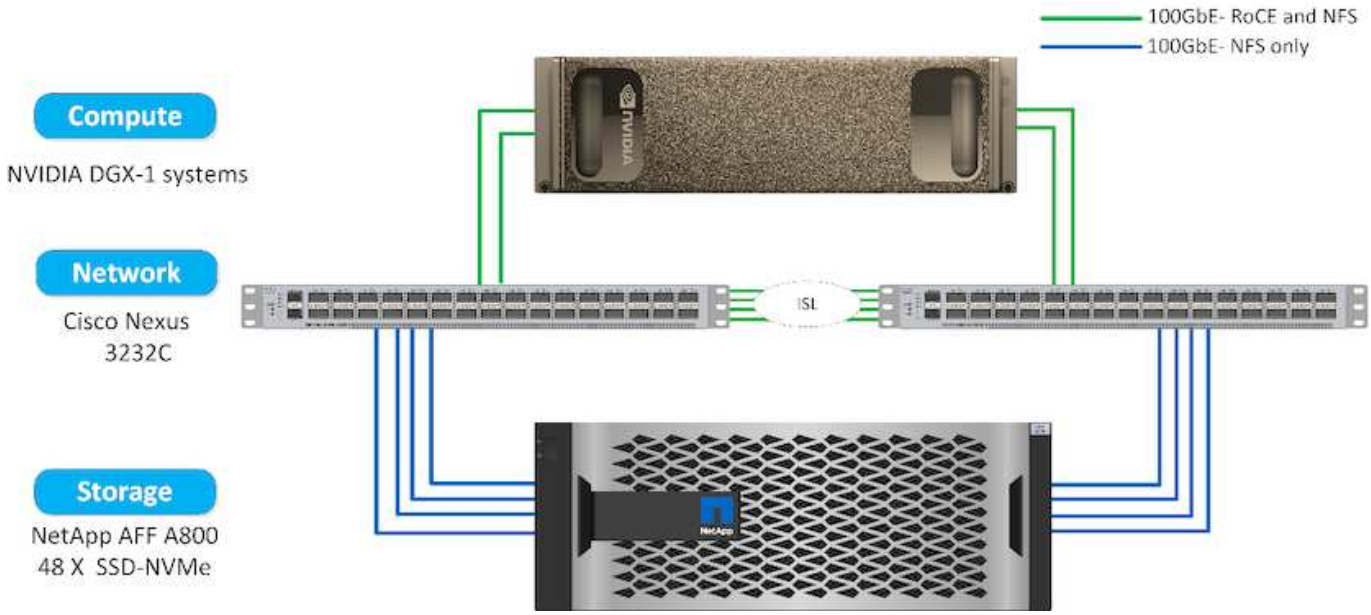
このセクションでは、 ONTAP AI 解決策のテクノロジ要件について説明します。

## ハードウェア要件

ハードウェア要件はお客様のワークロードによって異なりますが、 ONTAP AI は、大規模な ML/DL 運用向け

に、単一の GPU からラックスケール構成まで、あらゆる規模のデータエンジニアリング、モデルトレーニング、本番環境推論に導入できます。ONTAP AI の詳細については、を参照してください "ONTAP AI Web サイト"。

この解決策は、コンピューティングには DGX-1 システム、ネットワーク接続には NetApp AFF A800 ストレージシステム、 Cisco Nexus 3232C を使用して検証しました。この検証で使用される AFF A800 は、ほとんどの ML/DL ワークロードで最大 10 台の DGX-1 システムをサポートできます。次の図に、この検証でモデルのトレーニングに使用する ONTAP AI トポロジを示します。




この解決策をパブリッククラウドに拡張するために、 Cloud Volumes ONTAP をクラウド GPU コンピューティングリソースと一緒に導入し、ハイブリッドクラウドデータファブリックに統合することで、お客様は特定のワークロードに適したリソースを自由に使用できます。

## ソフトウェア要件

次の表に、この解決策検証で使用されるソフトウェアのバージョンを示します。

| コンポーネント | バージョン |
|---|---|
| Ubuntu | 18.04.4 LTS |
| NVIDIA DGX OS | 4.4.0 |
| NVIDIA DeepOps のことです | 20.02.1 |
| Kubernetes | 1.15 |
| Helm | 3.1.0 |
| cnvrg.io | 3.0.0 |
| NetApp ONTAP | 9.6P4 |

この解決策検証では、 Kubernetes を DGX-1 システム上にシングルノードクラスタとして導入しました。大規模な導入の場合は、管理サービスの高可用性を実現し、 ML ワークロードと DL ワークロードに貴重な DGX リソースを確保するために、独立した Kubernetes マスターノードを導入する必要があります。

# 解決策の導入と検証の詳細

以降のセクションでは、解決策の導入と検証の詳細について説明します。

## ONTAP AI の導入

ONTAP AI を導入するには、ネットワーク、コンピューティング、ストレージハードウェアのインストールと設定が必要です。ONTAP AI インフラの導入手順については、本ドキュメントでは説明していません。導入の詳細については、を参照してください "NVA-1121-deploy ： NetApp ONTAP AI 、 Powered by NVIDIA"。

この解決策検証では、 1 つのボリュームを作成して DGX-1 システムにマウントしました。その後、そのマウントポイントをコンテナにマウントして、トレーニング用のデータにアクセスできるようにしました。大規模な環境では、 NetApp Trident によってボリュームの作成とマウントが自動化されるため、管理上のオーバーヘッドが発生しないとともに、エンドユーザによるリソースの管理が可能になります。

## Kubernetes の導入

NVIDIA DeepOps で Kubernetes クラスタを導入および設定するには、導入ジャンプホストから次のタスクを実行します。

1. の手順に従って、 NVIDIA DeepOps をダウンロードします "「はじめに」ページ" NVIDIA DeepOps GitHub サイトで入手できます。
2. の手順に従って、クラスタに Kubernetes を導入します 。 "Kubernetes 導入ガイド" NVIDIA DeepOps GitHub サイトで入手できます。

> DeepOps Kubernetes 環境を使用するには、 Kubernetes マスターノードとワーカーノードがすべて同じユーザである必要があります。

配備に失敗した場合は 'kubectl_localhost' の値を 'd eepops/config/group_vars/k8s-cluster.yml' で false に変更し ' 手順 2 を繰り返します'Copy kubectl binary to Ansible host' タスクは 'kubectl_localhost' の値が true の場合にのみ実行され ' メモリ使用に関する既知の問題がある FETCH Ansible モジュールに依存しますこのようなメモリ使用の問題により、原因がタスクを失敗させることがあります。メモリ問題が原因でタスクが失敗した場合は、以降の導入処理は正常に完了しません。

「 kubectl_localhost 」の値を「 false 」に変更した後で展開が正常に完了した場合、「 kubectl binary 」を Kubernetes マスターノードから配備ジャンプホストに手動でコピーする必要があります。特定のマスター・ノード上で 'kubectl binary' の場所を確認するには 'which kubectl' コマンドをそのノード上で直接実行します

## cnvrg.io の展開

このセクションでは、Helmチャートを使用してcnvrgコアを導入する方法について詳しく説明します。

### Helm を使用して cnvrg コアを導入します

Helm は、任意のクラスタ、オンプレミス、 Minikube 、または任意のクラウドクラスタ（ AKS 、 EKS 、 GKE など）を使用して、 cnvrg を迅速に導入する最も簡単な方法です。このセクションでは、 Kubernetes が

インストールされたオンプレミス（ DGX-1 ）インスタンスに cnvrg がインストールされた方法について説明します。

前提条件

インストールを完了する前に、ローカルマシンに次の依存関係をインストールして準備する必要があります。

- Kubectl のように入力する
- Helm 3.x
- Kubernetes クラスタ 1.15 以降

**Helm** を使用して展開します

1. 最新の cnvrg Helm チャートをダウンロードするには、次のコマンドを実行します。

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

2. cnvrg を導入する前に、クラスタの外部 IP アドレス、および cnvrg を導入するノードの名前が必要です。オンプレミスの Kubernetes クラスタに cnvrg を導入するには、次のコマンドを実行します。

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

3. 「 helm install 」コマンドを実行します。すべてのサービスとシステムがクラスタに自動的にインストールされます。この処理には最大 15 分かかることがあります。
4. 「 helm install 」コマンドの所要時間は最大 10 分です。展開が完了したら、新しく展開した cnvrg の URL に移動するか、新しいクラスタを組織内のリソースとして追加します。「 helm' 」コマンドは正しい URL を通知します。

```
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. すべてのコンテナのステータスが「 Running 」または「 Complete 」の場合、 cnvrg は正常に展開されています。次のような出力が表示されます。

```
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
```

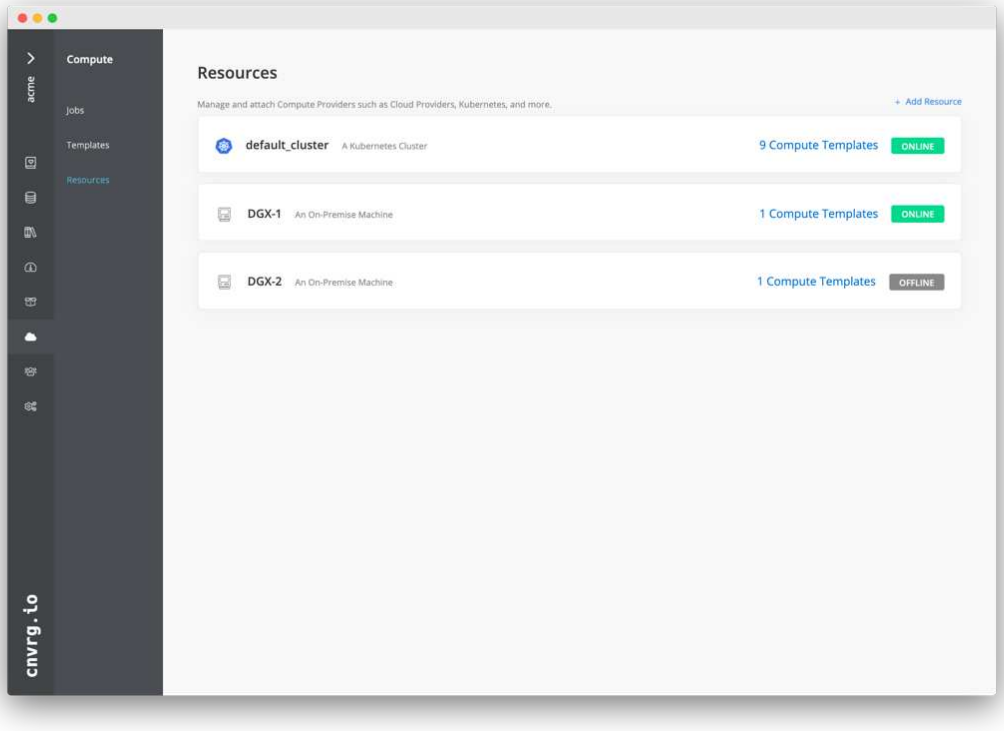
**ResNet50** および胸部 **X** 線を使用したコンピュータビジョンモデルトレーニング データセット

NVIDIA DGX システムを基盤とする NetApp ONTAP AI アーキテクチャ上の Kubernetes セットアップに、cnvrg.io AI OS が導入されました。検証には、胸部 X 線の匿名画像からなる NIH 胸部 X 線データセットを使用しました。画像は PNG 形式でした。このデータは NIH クリニカルセンタおよびによって提供された は、から使用できます "NIH ダウンロードサイト"。250 GB のサンプルデータを 15 クラスの 627 、 615 イメージで使用しました。

データセットは cnvrg プラットフォームにアップロードされ、 NetApp AFF A800 ストレージシステムからの NFS エクスポートにキャッシュされました。

コンピューティングリソースをセットアップする

cnvrg アーキテクチャおよびメタスケジューリング機能により、エンジニアおよび IT プロフェッショナルは、異なるコンピューティングリソースを 1 つのプラットフォームに接続できます。今回のセットアップでは、ディープラーニングワークロードの実行用に導入されたクラスタ cnvrg を使用しました。追加のクラスタを接続する必要がある場合は、次のスクリーンショットに示すように、 GUI を使用してください。




データをロードします

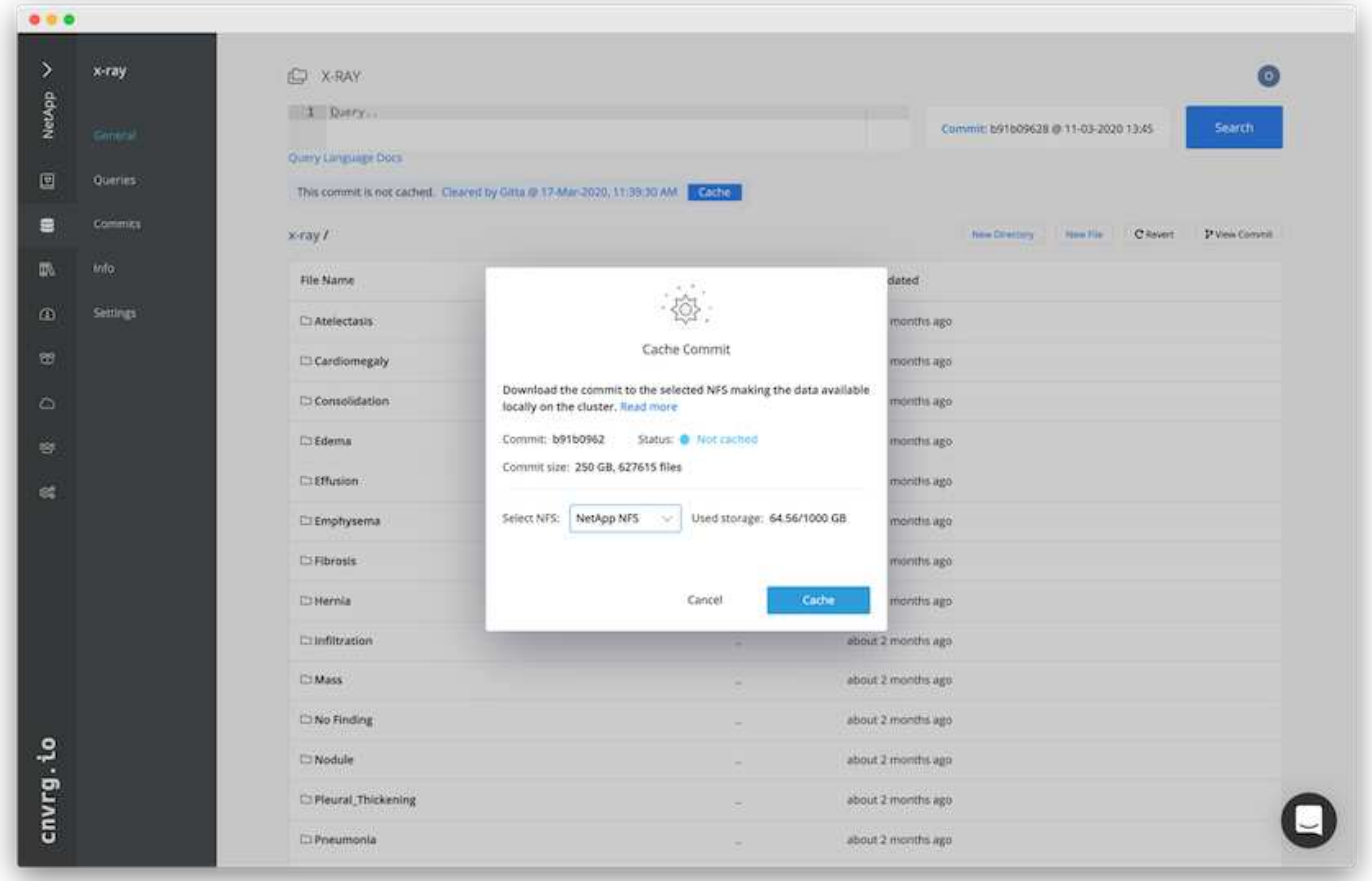
cnvrg プラットフォームにデータをアップロードするには、 GUI または cnvrg CLI を使用します。大規模なデータセットの場合は、 CLI の使用を推奨します。 CLI は、多数のファイルを処理できる、拡張性と信頼性に優れた強力なツールです。

データをアップロードするには、次の手順を実行します。

1. をダウンロードします "cnvrg CLI"。
2. X 線ディレクトリに移動します。
3. 「 cnvrg data init 」コマンドを使用して、プラットフォーム内のデータセットを初期化します。
4. 「 cnvrg data sync 」コマンドを使用して、ディレクトリのすべての内容を中央のデータレイクにアップロードします。データが中央のオブジェクトストア（ StorageGRID 、 S3 、またはその他）にアップロードされたら、 GUI で参照できます。次の図は、ロードされた胸部 X 線線維症画像 PNG ファイルを示しています。さらに、 cnvrg は、ビルドしたすべてのモデルをデータバージョンに複製できるように、データをバージョン化します。


マッハデータ

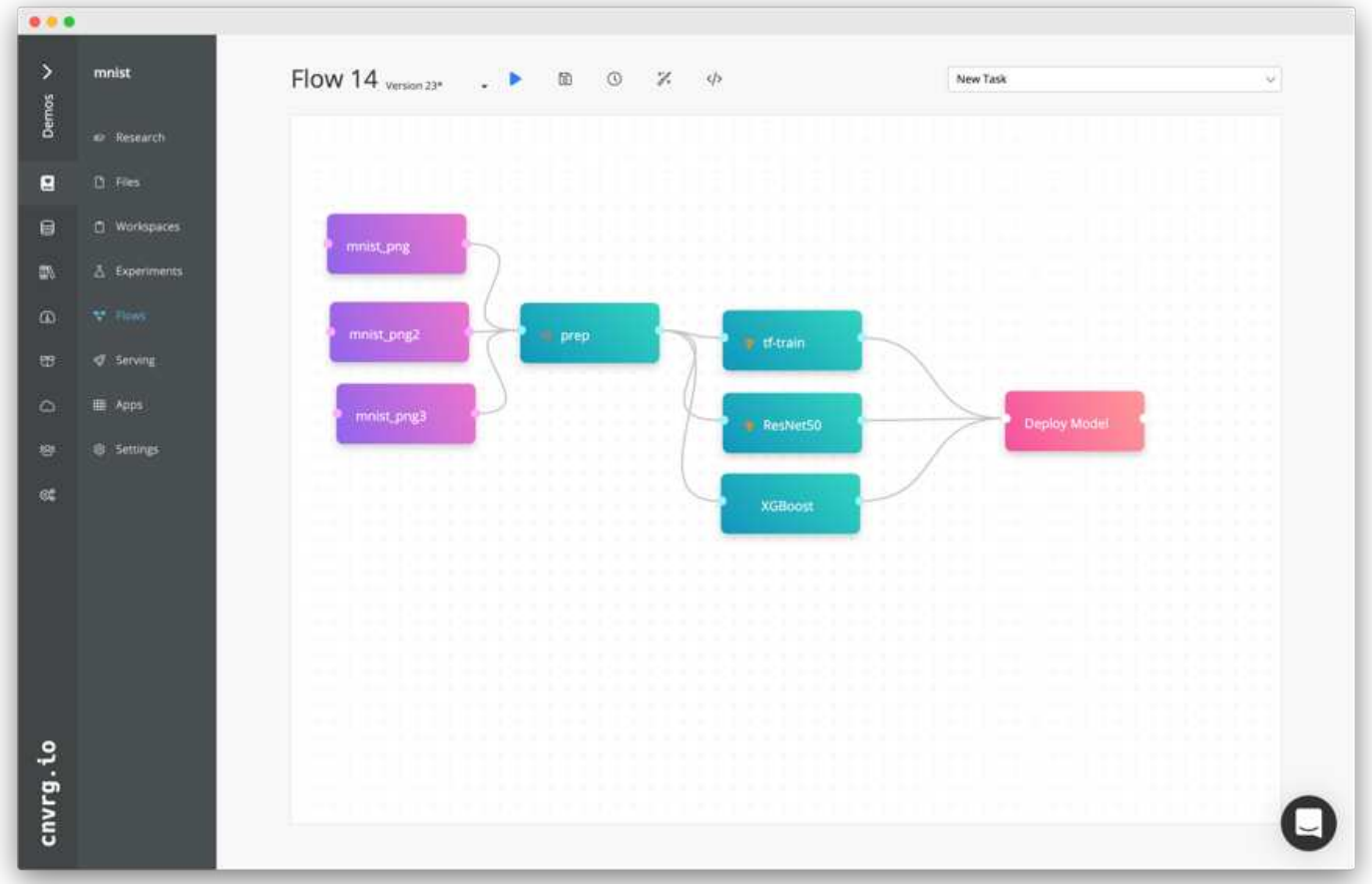
トレーニングを高速化し、モデルのトレーニングや実験ごとに 600k 以上のファイルをダウンロードしないようにするために、データを最初に中央のデータレイクオブジェクトストアにアップロードしたあとにデータキャッシュ機能を使用しました。




ユーザーが Cache をクリックすると、 cnvrg はリモートオブジェクトストアから特定のコミットでデータをダウンロードし、 ONTAP NFS ボリュームにキャッシュします。完了すると、データをすぐにトレーニングに利用できるようになります。さらに、データが数日間使用されていない場合（たとえば、モデルのトレーニングや探索など）、 cnvrg は自動的にキャッシュをクリアします。

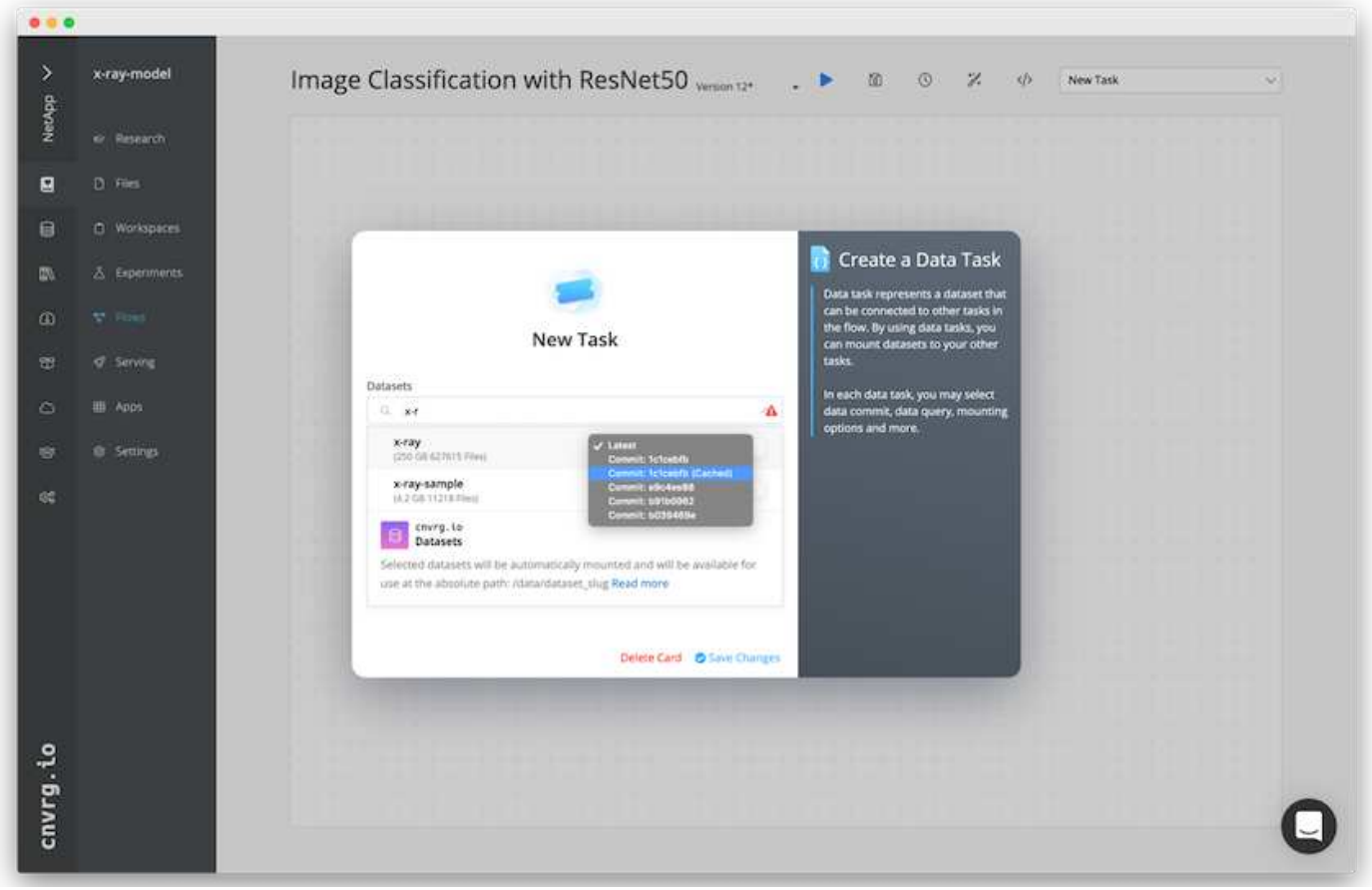
キャッシュデータで **ML** パイプラインを構築

cnvrg フローを使用すると、本番 ML パイプラインを簡単に構築できます。フローは柔軟性が高く、あらゆる種類の ML ユースケースに対応し、 GUI またはコードを使用して作成できます。フロー内の各コンポーネントは、異なる Docker イメージを使用して異なるコンピューティングリソース上で実行できるため、ハイブリッドクラウドを構築し、 ML パイプラインを最適化できます。



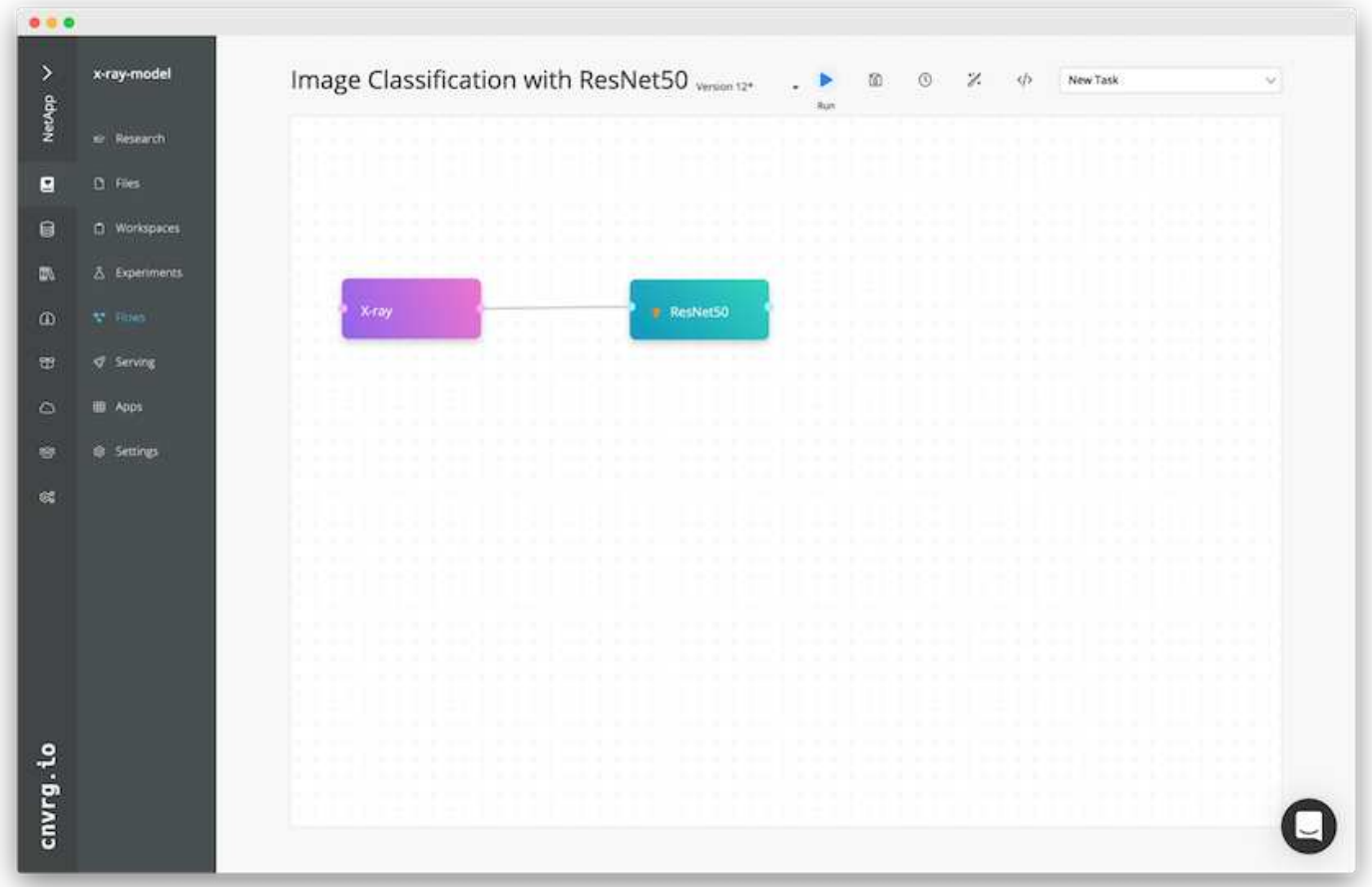

胸部 **X** 線フローの構築：データの設定

新しく作成したフローにデータセットを追加しました。データセットを追加する際には、特定のバージョン（commit ）を選択し、キャッシュされたバージョンが必要かどうかを指定できます。この例では、キャッシュされたコミットを選択しました。



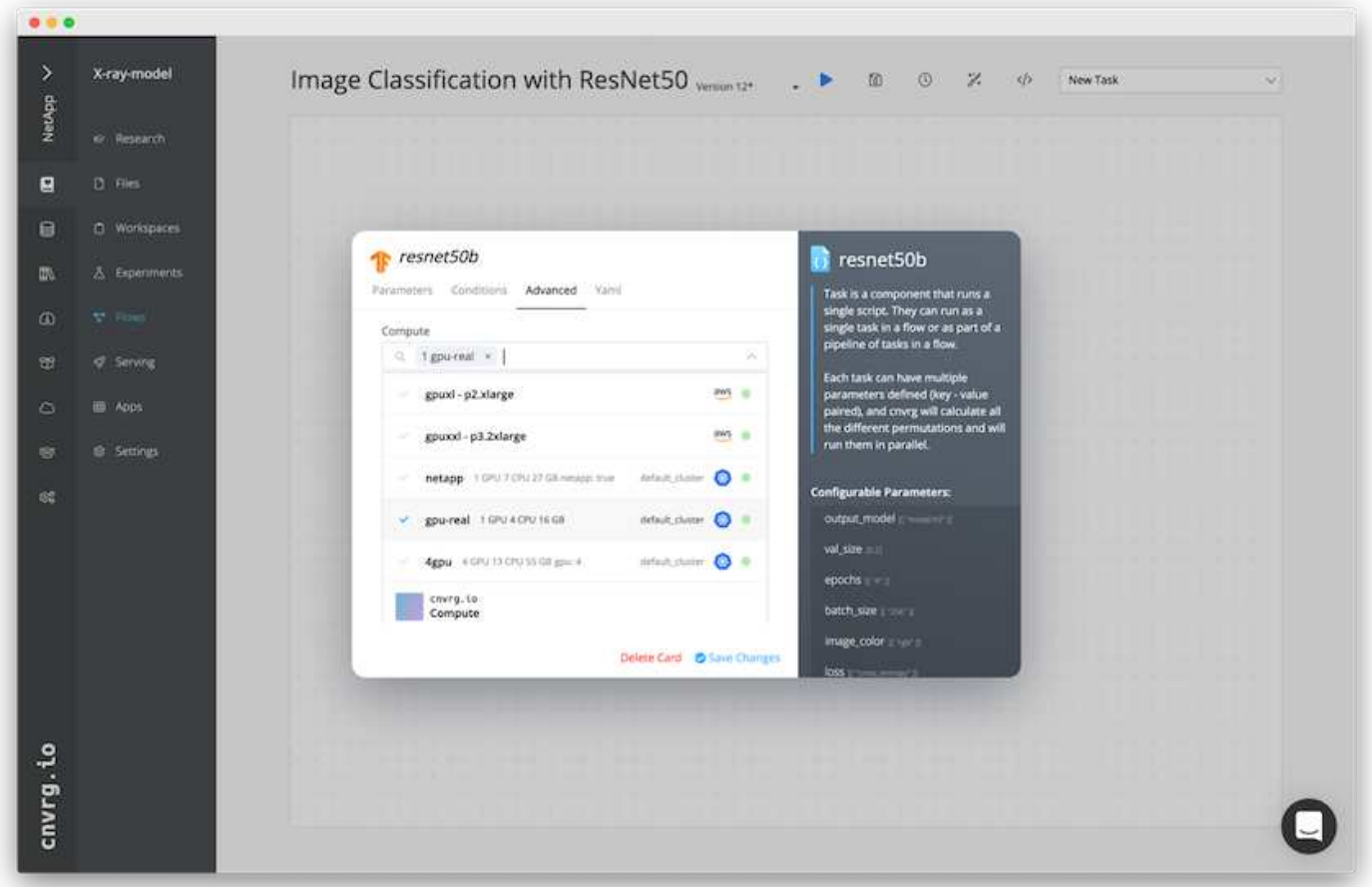

胸部 **X** 線フローの構築：トレーニングモデルの設定： **ResNet50**

パイプラインでは、任意の種類のカスタムコードを追加できます。cnvrg には、再利用可能な ML コンポーネントコレクションである AI ライブラリもあります。AI ライブラリには、アルゴリズム、スクリプト、データソースなど、あらゆる ML やディープラーニングフローで使用できるソリューションがあります。この例では、 ResNet50 の事前ビルドモジュールを選択しました。batch_size ： 128 、 epochs ： 10 などのデフォルトパラメータを使用しました。これらのパラメータは AI ライブラリのドキュメントで確認できます。次のスクリーンショットは、 X 線データセットが ResNet50 に接続された新しいフローを示しています。



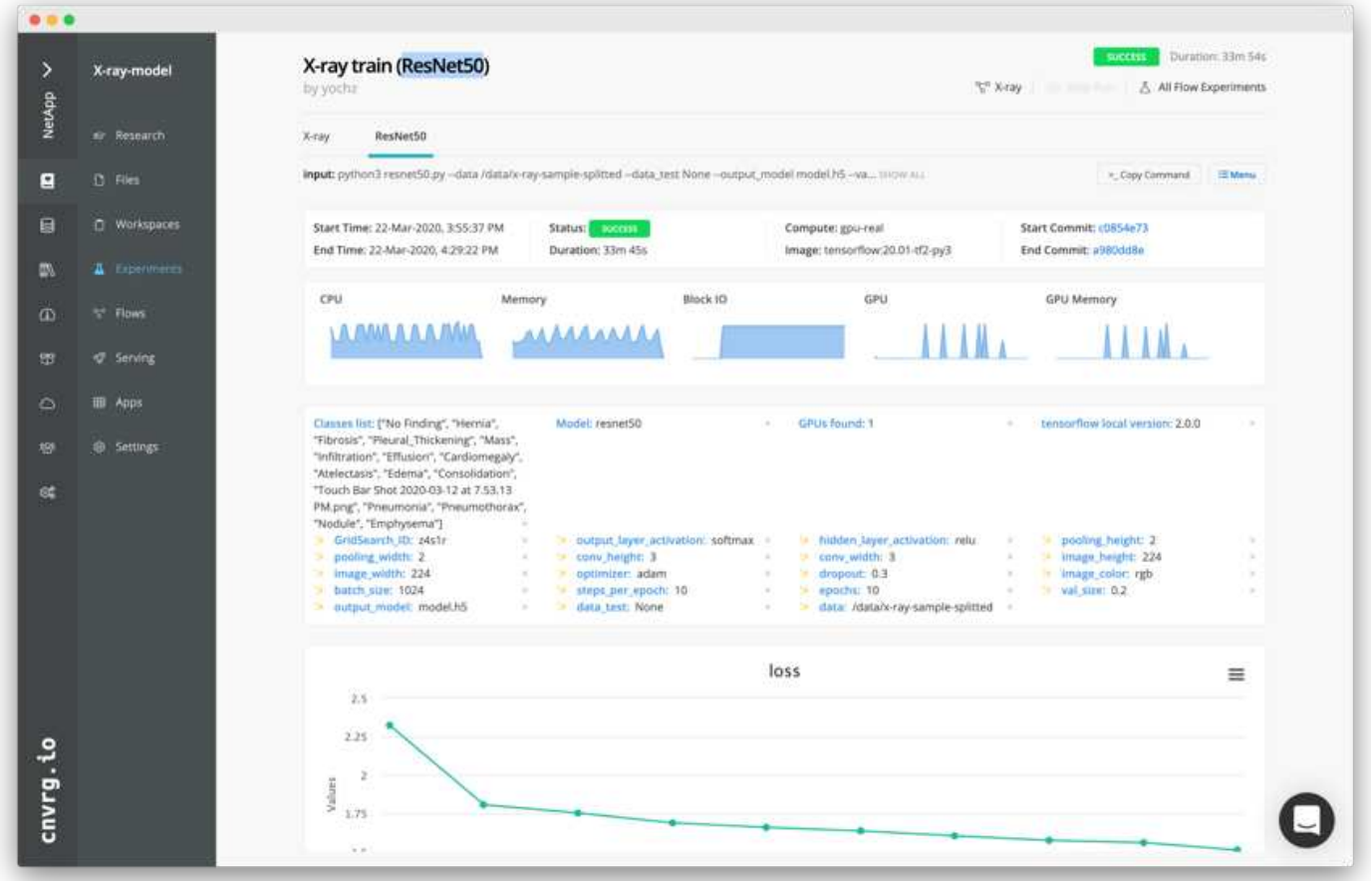

**ResNet50** の計算リソースを定義します

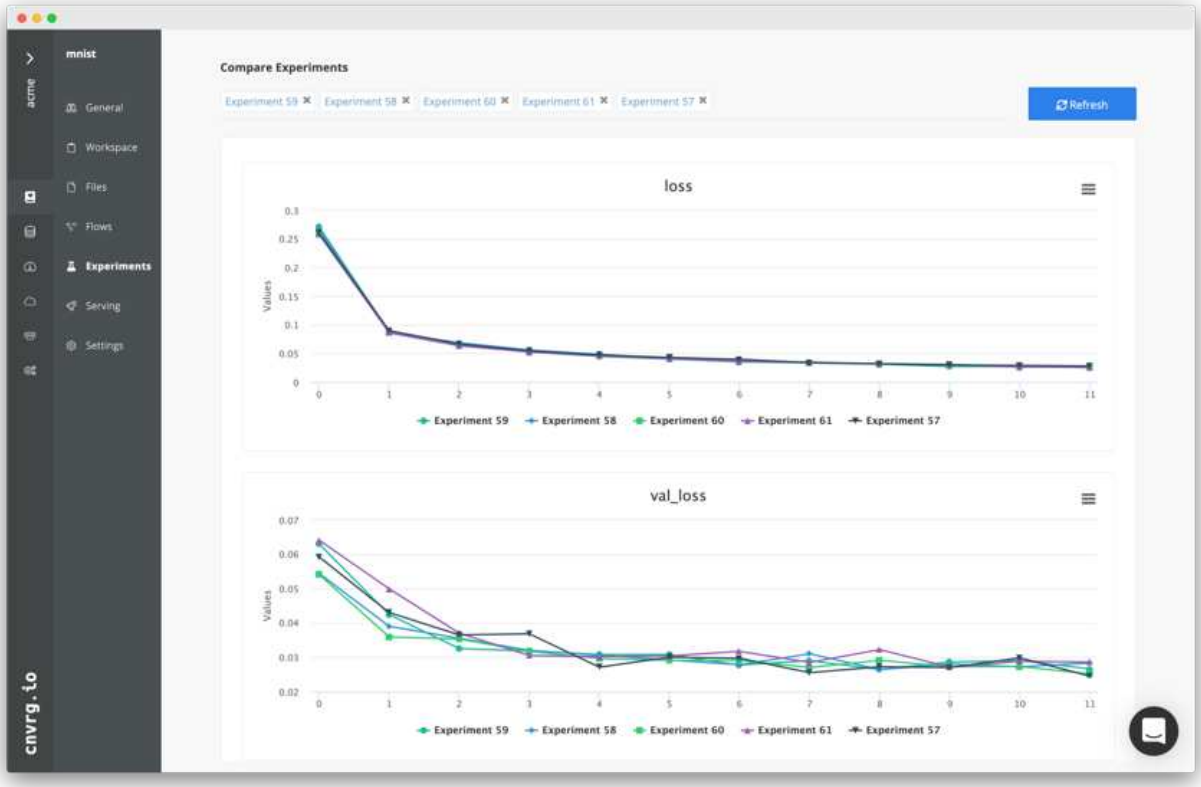
cnvrg フロー内の各アルゴリズムまたはコンポーネントは、異なる Docker イメージを使用して、異なるコンピューティングインスタンス上で実行できます。セットアップでは、 NetApp ONTAP AI アーキテクチャを採用した NVIDIA DGX システムでトレーニングアルゴリズムを実行したいと考えていました。次の図では、「 GPU - REAL 」を選択しました。これは、オンプレミスクラスタのコンピューティングテンプレートであり、仕様です。また、テンプレートのキューを作成し、複数のテンプレートを選択しました。このようにして 'GPU 実数のリソースを割り当てることができない場合 ( たとえば ' 他のデータ・サイエンティストがリソースを使用している場合 ) は ' クラウド・プロバイダ・テンプレートを追加して ' 自動クラウド・バーストを有効にできます次のスクリーンショットは、 ResNet50 のコンピューティングノードとしての GPU 実数の使用を示しています。




結果の追跡と監視

フローが実行されると、 cnvrg はトラッキングおよびモニタリングエンジンをトリガーします。フローの各実行は自動的に文書化され、リアルタイムで更新されます。ハイパーパラメータ、指標、リソース使用率（ GPU 利用率など）、コードバージョン、アーティファクト、ログ また、次の 2 つのスクリーンショットに示すように、 ［テスト］ セクションで自動的に使用できるようになります。


X-ray-model
X-ray train (ResNet50)
by yochz
SUCCESS
Duration: 33m 54s
X-ray
All Flow Experiments
X-ray
ResNet50
Research
Files
Workspaces
Experiments
Flows
Serving
Apps
Settings
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

# まとめ

NetApp と cnvrg.io はパートナーとして提携し、 ML および DL ソフトウェア開発向けの包括的なデータ管理解決策をお客様に提供しています。ONTAP AI は、あらゆる規模の運用に対応できる高性能なコンピューティングとストレージを提供します。 cnvrg.io ソフトウェアは、データサイエンスのワークフローを合理化し、リソース利用率を向上させます。


## 謝辞

- ネットアップテクニカルマーケティングエンジニア、 Mike Oglesby 氏
- ネットアップシニアテクニカルディレクター Santosh Rao 氏


# 追加情報の検索場所

このドキュメントに記載されている情報の詳細については、次のリソースを参照してください。

- Cnvrg.io （ "https://cnvrg.io"）：
  - Cnvrg コア（無償の ML プラットフォーム）

    https://cnvrg.io/platform/core
  - Cnvrg のドキュメント

    "https://app.cnvrg.io/docs"
- NVIDIA DGX-1 サーバ：
  - NVIDIA DGX-1 サーバ

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

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

    https://www.nvidia.com/en-us/gpu-cloud/
- NetApp AFF システム：
  - AFF データシート

    https://www.netapp.com/us/media/d-3582.pdf
  - AFF 向け NetApp FlashAdvantage プログラム

    https://www.netapp.com/us/media/ds-3733.pdf

- ONTAP 9.x のドキュメント

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

- NetApp FlexGroup テクニカルレポート

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

- コンテナ向けのネットアップの永続的ストレージ：
  - NetApp Trident

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

- NetApp Interoperability Matrix を参照してください
  - NetApp Interoperability Matrix Tool で確認できます

    https://mysupport.netapp.com/matrix/#welcome

- ONTAP AI ネットワーク：
  - Cisco Nexus 3232C スイッチ

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

  - Mellanox Spectrum 2000 シリーズスイッチ

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

- ML フレームワークとツール：
  - 大理

    https://github.com/NVIDIA/DALI

  - TensorFlow ：あらゆる環境に対応するオープンソースの機械学習フレームワーク

    https://www.tensorflow.org/

  - Horovod ： Uber が開発したオープンソースの TensorFlow 用分散学習フレームワーク

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

  - コンテナランタイムエコシステムでの GPU の有効化

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

  - Docker です

    https://docs.docker.com

  - Kubernetes

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

- NVIDIA DeepOps のことです

  https://github.com/NVIDIA/deepops

- クビフロー

  http://www.kubeflow.org/

- Jupyter Notebook Server の 2 つのツールを使用

  http://www.jupyter.org/

- データセットとベンチマーク：
  - NIH 胸部 X 線データセット

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

  - Xiaosong Wang 、 Yifan Peng 、 Le Lu 、 Zhiyong Lu 、 Mohammadhadi Bagheri 、 ロナルド・サマーズ、 ChestX-Ray8 ：『 Hospital scale Chest X-ray Database and Benchmarks on weakly Supervised Classification and Localization of Common Thorax Diseases 、 IEEE CVR 、 pp3462-3471 、 2017TR-4841-0620