



テクニカル レポート

自動運転ワークロード向けの NetApp StorageGRID データレイク ソリューションの設計

NetApp
David Arnette
2020年9月 | TR-4851

概要

本ドキュメントでは、NetApp® StorageGRID® オブジェクトストレージを、機械学習 (ML) およびディープラーニング (DL) ソフトウェア開発のデータリポジトリおよび管理システムとして使用する方法について説明します。このホワイトペーパーでは、自律走行車のソフトウェア開発におけるデータフローと要件、およびデータライフサイクルを合理化するStorageGRIDの機能について説明します。この解決策環境は、MLやDLの開発プロセスで一般的に見られる、マルチステージのデータパイプラインワークフローです。

<<本レポートは機械翻訳による参考訳です。公式な内容はオリジナルである英語版をご確認ください。>>

目次

1	エグゼクティブサマリー	4
2	解決策の概要	4
21	自動運転のユースケースのまとめ	4
22	データに関する課題	5
23	StorageGRIDデータレイクとパイプライン解決策の概要	6
3	テクノロジー	6
31	StorageGRID	6
32	ONTAP AI	7
33	NetApp AIコントロールプレーン	7
34	ハードウェア要件	7
35	ソフトウェア要件	9
36	テスト環境	9
4	データパイプラインのワークフロー検証	10
41	データの取り込み	10
42	データファクトリ処理	11
43	モデルトレーニング	15
44	シミュレーションと再生	17
5	まとめ	18
	謝辞	18
	追加情報の入手方法	18
	バージョン履歴	18

表一覧

表1)	ソフトウェア要件	9
-----	----------	---

図一覧

図1)	AV/ADAS開発プロセス	6
図2)	データレイク検証環境	8
図3)	StorageGRID ILMポリシー	9

図4) AVパイプライン-データの取り込み	10
図5) StorageGRIDオブジェクトのElasticsearchインデックス	11
図6) AVパイプライン- Data Factory	12
図7) StorageGRIDオブジェクトのタグ付け	13
図8) StorageGRIDデータレイクからのSparkジョブの例	14
図9) データセットオブジェクトタグ	15
図10) AVパイプライン-モデルのトレーニング	16
図11) モデルトレーニングの例	16

1 概要

自律走行車（AV）ソフトウェアを開発するには、地理的に分散した場所で大量のデータが生成される必要があります。これらのデータ資産をラベル付け、処理、モデルトレーニングに統合することは、高度なドライバーアシスタンスとAV機能を開発する際にエンジニアが直面する最大の課題の1つです。データ量が数十エクサバイトから数百エクサバイトに増え続けているため、従来型のストレージシステムでは、コスト効率に優れた方法で拡張して容量を確保することはできません。また、容量やアクセス可能なスペースに制限があるため、運用も複雑になります。

このような制限を克服するために、オブジェクトストレージは、このような膨大なデータを管理するストレージプラットフォームとして急速に選ばれるようになりつつあります。オブジェクトストレージは、単一のネームスペースでほぼ無制限のストレージ容量を提供し、地理的に分散したストレージオプションを提供して、ある場所から別の場所へのデータ移動を容易にします。また、オブジェクトストレージを使用すると、アクセスセマンティクスが簡易化され、データへのアクセスが容易になります。

NetApp® StorageGRID®は、これらすべての機能を提供するエンタープライズクラスのSoftware-Definedオブジェクトストレージシステムで、包括的なデータパイプライン解決策に必要な他の製品とシームレスに統合します。この解決策設計ガイドでは、AVソフトウェア開発データパイプラインのプライマリデータリポジトリおよびバックボーンとしてStorageGRIDオブジェクトストレージを使用する方法について説明します。ここでは、AVおよびADAS（Advanced Driver Assistance System）用のディープラーニング（DL）モデルの開発に使用されるワークフローの概要について説明します。また、AV開発ワークフローのさまざまな段階でStorageGRIDデータレイクを使用するためのガイダンスも記載されています。

この解決策ガイドの対象読者は次のとおりです。

- AVを含む自動車業界のユースケース向けAIモデルやソフトウェア開発のためのソリューションを設計するインフラアーキテクト。
- AVソフトウェア開発サイクル全体でデータ管理タスクを合理化したいと考えているデータエンジニア。
- AIへの取り組みから市場投入までの時間を短縮したいと考えている経営幹部の意思決定者。

2 ソリューション概要

2.1 自動運転のユースケースのまとめ

自動運転車には、道路の安全性と効率性を向上させる大きな可能性があります。ディープニューラルネットワーク（DNN）と畳み込みニューラルネットワーク（CNN）は、この自律性を実現します。コンポーネントには、画像やセンサーのデータを解釈するためのコンピュータビジョンモデル、物体の位置や動きを推定するための予測モデル、特定のアクションを提案または実行するための推奨モデルが含まれます。[National Highway Traffic Safety Administration](#)は、自律性のレベルを0～5のスケールで評価しています。

- **レベル0**。自動化なし。自律性がゼロ。運転者はすべての運転タスクを実行します。
- **レベル1**。ドライバーアシスタンス車両は運転者によって制御されますが、一部の運転支援機能が車両の設計に含まれている場合があります。
- **レベル2**部分的な自動化：車両には、加速やステアリングなどの自動機能が統合されていますが、運転者は常に運転タスクに従事し、環境を監視する必要があります。
- **レベル3**条件付き自動化：ドライバーは存在する必要がありますが、環境の監視には必要ありません。運転者は、常に注意を払って車両を制御する準備ができています。
- **レベル4**高度な自動化：車両は、特定の条件下ですべての運転機能を実行できます。運転者は車両を制御するオプションを持つことができます。
- **レベル5**完全自動化：車両は、あらゆる条件下ですべての運転機能を実行できます。運転者は車両を制御するオプションを持つことができます。

L2以上からの自律性のレベルごとに、複数のAIモデルを開発する必要があります。各モデルは複雑なニューラルネットワークアーキテクチャを持ち、非常に大規模なトレーニングデータセットが必要です。これらの数値を定量化し、方向性を見積もるには、次の点を考慮してください。

- 1日8時間、年間250日運転するサーベイカーは、年間最大2,000時間のビデオデータを作成できます。
- 30フレーム/秒の5台のカメラでは、このビデオデータは年間10億枚の画像に相当します。
- 2メガピクセルのカメラ解像度(1画像あたり約2MB)を使用して、1台の車が1時間あたり約1TBのデータを生成することができます。
- このデータの約3分の1は、モデルトレーニングに役立つようにクレンジングされ、ラベル付けされています。
- ベストプラクティスでは、単純なネットワークのトレーニングに数百万枚の画像を使用し、高度に複雑なネットワークのトレーニングに500万~800万枚の画像を使用する必要があります。

この初期データ量は、AVおよびADASシステムの開発における最大の課題の1つです。多くのAV開発企業は、エッジから処理データセンターに移動する生データの量を最小限に抑える手法に取り組んでいます。しかし、ニューラルネットワークのトレーニングには全体的なデータ量が必要なため、依然として膨大な量のデータが関与しています。

自動運転のためのAIソフトウェアの開発は、いくつかのステップで構成される反復的なワークフロープロセスです。

- **データ収集** は、搭載されたカメラとレーダーやライダーなどのセンサーを使用して現実のデータを収集するサーベイカーのフリートによって達成されます。
- **データファクトリ処理** には、データの取り込みとトランスコーディング、メタデータの管理と処理、アクティブラーニングが含まれます。
- **モデルトレーニング** では、ラベル付けされたデータセットを使用して、特定のタスク用にニューラルネットワークをトレーニングします。
- **シミュレーションと再生** には、実際のデータに対するコンポーネントレベルのテストと、運転シナリオと条件のシミュレーションの両方が含まれ、シミュレーションを使用して特定の条件下でモデルの動作を検証します。

各プロセスには、それぞれ異なるデータアクセス要件とパフォーマンス要件があります。詳細については、セクション4「データパイプラインワークフローの検証」で説明します。

2.2 データに関する課題

より高いレベルの自律性を実現するためのデータと高速コンピューティングの要件は、非常に重要です。データはプロジェクトのライフサイクルを通じて急激に増加し、このプロセスでは時間が重要な要素となります。企業は、開発者リソースを効果的に活用し、競争の激しい時期に新しい機能を市場に提供したいと考えています。この開発プロセスのインフラストラクチャは、個々の開発者の有効性を妨げることなく、データと組織の成長に応じて各ステップと規模に関わる時間を短縮できなければなりません。

データレイクは、AVソフトウェアの開発に加えて、多くの場合、ビジネスの他の部分にも使用されます。大規模な製造企業がデジタル変革と次世代の製造プロセスを採用すると、データは多くの場所から湖に流れ、湖から他の多くのプロセスに提供されます。既存のプロセスではPOSIXファイルシステムアクセス機能が必要になる場合があり、新しいクラウドネイティブアプリケーションでは通常、データアクセスにSimple Storage Service (S3)を使用します。セクション4.2で説明したラベル付けプロセスなど、一部のデータ利用者はストレージのパフォーマンス要件を最小限に抑えています。モデルのトレーニングでは最高レベルのパフォーマンスが求められます。

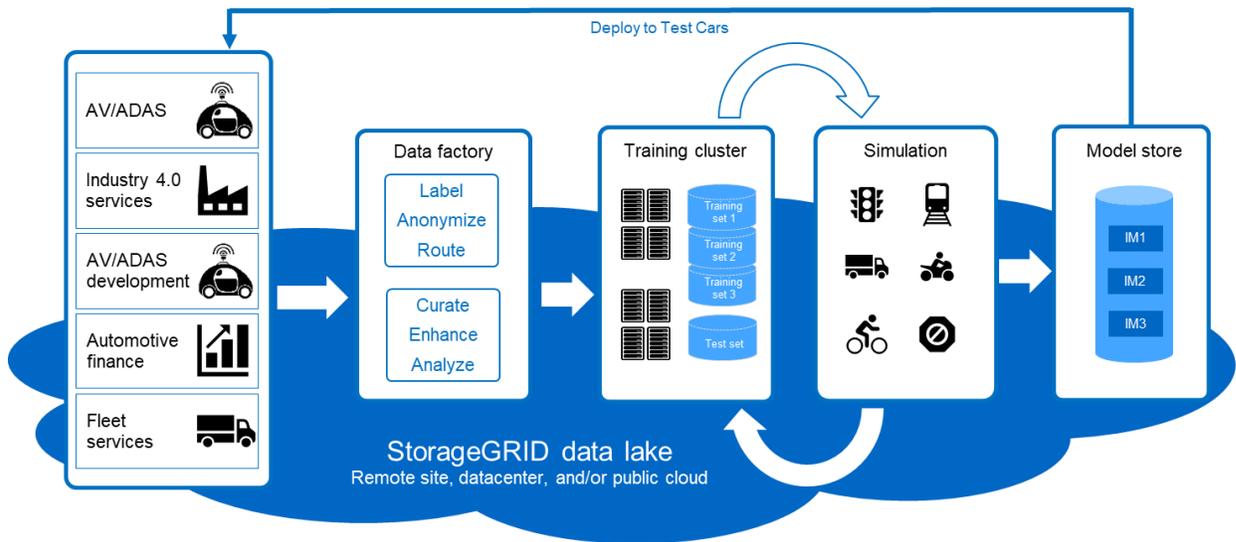
これらの要件から、大容量のストレージを提供し、データのライフサイクルステージに合わせてコストモデルを調整し、複数のプロトコルを介してアクセスを提供できる解決策の必要性が定義されています。また、解決策では、既存システム間、サイト間やコスト階層間、特殊な目的の環境へのデータ移動を最小限のユーザ操作で行うこともできなければなりません。最終的に、このような解決策では、特定のアクセス要件やパフォーマンス要件に基づいて複数のストレージタイプが必要になります。また、管理オーバーヘッドを最小限に抑え、リソース利用率を最大限に高めるために、ソフトウェア統合も必要になります。

2.3 StorageGRIDデータレイクとパイプライン解決策の概要

NetApp StorageGRIDは、膨大な量の初期データを保存するためのコスト効率と拡張性に優れた方法を提供し、ある場所でデータを入力して別の場所で処理するためにデータを取得する機能を提供します。オブジェクトストレージでは、豊富なメタデータを関連付けて、データとモデルのライフサイクル全体にわたって保持と保護に関する詳細なデータ管理ポリシーを作成できます。

これらの機能を使用すると、複数の場所にまたがるデータレイクや導入モダリティを構築して、データレイクをあらゆる方向に同時に流れるデータパイプラインに変換できます。リモートサイトに取り込まれたデータは、以降の処理やアーカイブのために自動的に移動できます。データセンターで作成されたデータは、リモートまたはパブリッククラウドのリソース全体にシームレスに伝播できるため、あらゆる処理やワークフローのニーズに対応できます。図1は、AV/ADASソフトウェア開発ワークフローで使用されるデータレイクとパイプラインの組み合わせの概要を示しています。

図1) AV/ADASの開発プロセス



3 テクノロジ

3.1 StorageGRID

NetApp StorageGRIDはソフトウェアで定義されるオブジェクトベースのストレージ解決策で、S3 APIやOpenStack Swift APIなどの業界標準のオブジェクトAPIをサポートします。非構造化データ向けにセキュリティとデータ保持性に優れたストレージを大規模に展開できます。メタデータベースの統合ライフサイクル管理ポリシーにより、データのライフサイクルを通じて最適な保存先を選択できます。コストを削減するために、コンテンツは適切な場所、適切なタイミング、適切なストレージ階層に配置されます。StorageGRIDは、グローバルに分散された冗長な異機種混在ノードで構成され、既存および次世代のクライアントアプリケーションと統合できます。

StorageGRIDシステムの主なメリットは次のとおりです。

- 拡張性にきわめて優れ、使いやすい非構造化データ用グローバルデータリポジトリ
- ハイブリッドクラウドに対応したシステム：ポリシーベースの情報ライフサイクル管理 (ILM) は、AWSやMicrosoft Azureなどのパブリッククラウドにオブジェクトを格納します。StorageGRIDプラットフォームサービスを使用すると、パブリッククラウドでコンテンツのレプリケーション、イベント通知、メタデータの検索が可能になります。

- **データの保持性と可用性を高める柔軟なデータ保護**：レプリケーションと階層型イレイジャーコーディングを使用してデータを保護できます。保存データと転送中データの検証により、長期保持の整合性を確認できます。
- **複数のストレージテナントアカウントをサポートし、システムに格納されているオブジェクトをエンティティ別に分離できます。**
- **ソフトウェアベースまたはハードウェアベースの導入のサポート**。StorageGRIDは、次のコンポーネントを任意に組み合わせて導入できます。
 - VMwareで実行されている仮想マシン。
 - Linuxホスト上のDockerコンテナ。
 - StorageGRIDアプライアンス：ストレージアプライアンスはオブジェクトストレージを提供します。サービスアプライアンスは、グリッド管理サービスとロードバランシングサービスを提供します。
- **フェデレーテッドアイデンティティ管理**：StorageGRIDは、ユーザ認証のためにActive Directory、OpenLDAP、またはOracleディレクトリサービスと統合されています。Security Assertion Markup Language 2.0 (SAML 2.0) 標準を使用してシングルサインオン (SSO) をサポートし、StorageGRIDとActive Directoryフェデレーションサービス (AD FS) 間で認証と承認のデータを交換します。

詳細については、netapp.comの『[NetApp StorageGRID](https://netapp.com)』を参照してください。

3.2 ONTAP AI

NVIDIA DGX™システムとNetAppクラウド対応ストレージシステムを基盤とするNetApp ONTAP®AIの実証済みアーキテクチャは、NetAppとNVIDIAによって開発、検証されました。NetApp ONTAP AIは、DGXシステムとNetApp AFFオールフラッシュストレージシステムと最先端のネットワークを緊密に統合します。NetApp ONTAP AIは、設計の複雑さと推測を排除することで、AIの導入を簡易化します。小規模構成から始めて、システムを中断することなく拡張しながら、エッジ、コア、クラウドの間でデータをインテリジェントに管理できます。

ONTAP AIは、この検証で使用されるニューラルネットワークトレーニングインフラを提供します。オブジェクトストアのデータは、DGX-1システムにマウントされたONTAP NFSエクスポートにローカルにキャッシュされます。このアプローチでは、複数のGPUサーバから同時に非常に高速でデータにアクセスでき、データを何度もステージングする必要はありません。

ONTAP AIの詳細については、NetApp Verified Architectures ([NVA-1121](#)、[NVA-1135](#)、および[NVA-1138](#))を参照してください。

3.3 NetApp AIコントロールプレーン

NetApp AIコントロールプレーンは、データサイエンティストとデータエンジニア向けのAIデータと実験を管理するためのフルスタック解決策です。ONTAPのパフォーマンスと機能をKubernetesとKubeflowと組み合わせることで、トレーサビリティとバージョン管理のためのデータとモデルのベースラインをほぼ瞬時に作成できるAI開発ワークフローを定義して実装できます。また、AIコントロールプレーンを使用すると、IT事業者はサイトやリージョン間でシームレスにデータを複製し、大規模なデータセットにアクセスできるJupyter Notebookワークスペースを迅速にプロビジョニングできます。AIコントロールプレーンのKubeflowパイプライン機能を使用すると、[セクション4](#)で説明するワークフロープロセス全体を自動化できます。NetApp AIコントロールプレーンの詳細については、[TR-4798](#)を参照してください。

3.4 ハードウェア要件

この解決策のハードウェア要件は、各サイトのストレージ容量とオブジェクトスループットのパフォーマンスに関する具体的な要件によって、サイトによって異なります。StorageGRIDには、容量、パフォーマンス、設置面積のあらゆる要件に対応する、物理アプライアンスと仮想アプライアンスのオプションが用意されています。

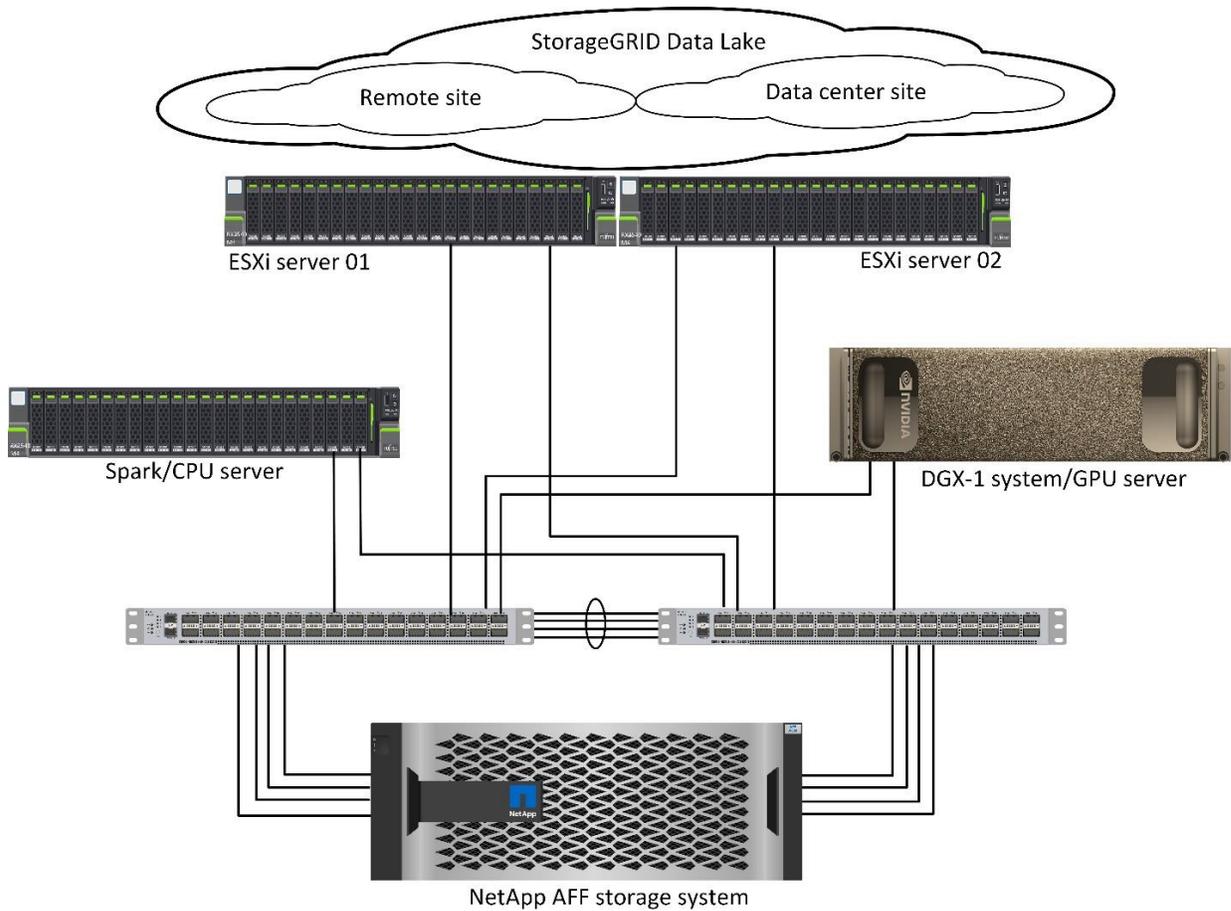
リモートサイトには、調査車両から取り込まれた生データを毎日保存し、そのデータをしばらく保持するのに十分な容量とパフォーマンスが必要です。通常、容量と取り込みパフォーマンスの要件を満たすためには、StorageGRID SG5760など、中容量のハードウェアノードを3つ以上設置する必要があります。

仮想ノードを使用すると、スループットを向上させたり、既存のハードウェアで容量を追加したりできます。設置面積の削減が必要なサイトの場合は、StorageGRID SG5712 ノードまたは仮想ノードを容量として使用し、Varnishソフトウェアのようなキャッシュメカニズムを使用してパフォーマンスを向上させることができます。

データセンターではさまざまなタスクが実行されるため、これらのサイトのパフォーマンスとスループットの要件ははるかに高くなります。SG5760などの低パフォーマンスノードは、高密度のコールドアーカイブストレージに最適です。SG6060ノードはハイパフォーマンスと大容量を提供します。また、非常に要件の厳しいスループット要件に対応するため、StorageGRID SGF6024ノードはフラッシュメモリのパフォーマンスを提供します。ILMポリシーでは、これらのハードウェア階層にわたってデータを管理できるため、パイプラインのパフォーマンス要件を満たしながら、最も対費用効果の高いストレージオプションにデータが自動的に配置されます。

この検証は、VMware仮想インフラに導入された完全な仮想グリッドを使用して実行されました。この解決策の他のプロセスは、物理Dockerホストを使用してCPU固有のタスクを実行することで検証されました。ONTAP AIインフラを使用して、セクション4.3「モデルトレーニング」で説明するモデルトレーニングタスクを実行しました。図2に、この検証に使用した環境を示します。この構成は、機能のデモに必要な最小限のStorageGRID構成を表しています。StorageGRIDでは同じグリッド内の物理ノードと仮想ノードがサポートされるため、この構成はテスト目的やコンセプトの実証目的で導入できます。ノード、サイト、ILMポリシーを追加したり、不要な仮想ノードを削除したりするだけで、エクサバイト規模の本番用オブジェクトストア環境にシームレスに拡張できます。

図2) データレイク検証環境



3.5 ソフトウェア要件

表1に、この解決策の検証で使用したソフトウェアコンポーネントを示します。

表1) ソフトウェア要件

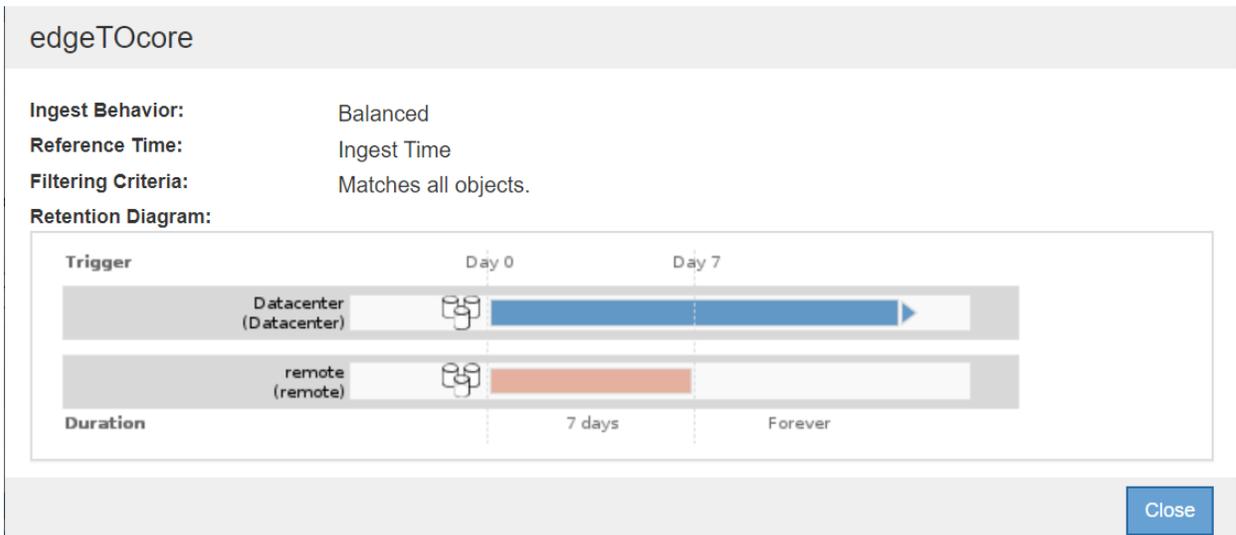
ソフトウェア	バージョン他
StorageGRIDオブジェクトストレージソフトウェア	11.3
Elasticsearch	7.6.2
AWS Boto3 S3ソフトウェア開発キット (SDK)	1.13.15
Sparkソフトウェア	PySpark 2.4.5

3.6 テスト環境

取り込み先には、3つの仮想ストレージノードと1つのロードバランサノードで構成されるリモートサイトを導入しました。データセンターサイトは、3つの仮想ストレージノード、ロードバランサ、StorageGRID管理ノードで構成されています。このセットアップでは、StorageGRIDデータパイプライン解決策のデータ管理機能をデモするための仮想マルチサイト環境を提供します。

各サイトは、イレイジャーコーディングデータ保護を使用して別々のストレージプールに構成されています。リモートサイトに取り込まれたすべてのデータに一致するようにILMルールが設定されています。このルールを使用してILMポリシーをアクティブ化し、取り込み直後にデータセンター内の各オブジェクトの2つ目のコピーを作成しました。このポリシーでは、新しいオブジェクトはリモートサイトに7日間だけ保持し、データセンターサイトのオブジェクトは無期限に保持するように指定されています。図3を参照してください。生産パイプラインでは、追加のポリシーが有効になります。ポリシーでは、処理に必要なに応じてデータを他の場所に移動したり、指定された時間が経過したあとにコールドストレージノードにデータをアーカイブしたり、保持期間が経過したあとにオブジェクトを削除したりできます。

図3) StorageGRIDのILMポリシー



StorageGRIDはElasticsearchインデックスと統合されており、リポジトリ内のStorageGRIDオブジェクトに対して高性能な検索機能を提供します。また、「きばな」は検索インデックスとの視覚的な対話に使用できます。今回の検証では、1つのElasticsearchインスタンスを使用しましたが、Elasticsearchでは、あらゆる規模の導入に対応できるように拡張可能なスケールアップアーキテクチャを使用しています。

この検証で使用された個々のプロセスはすべて、ONTAP AIトレーニングクラスタ内のCPUホストとDGX-1システムの両方で、Dockerコンテナで実行されました。Dockerコンテナは、ワークロードの完全なモビリティを提供し、NetApp AIコントロールプレーンを使用してこのプロセス全体の自動化を可能にします。

4 データパイプラインのワークフロー検証

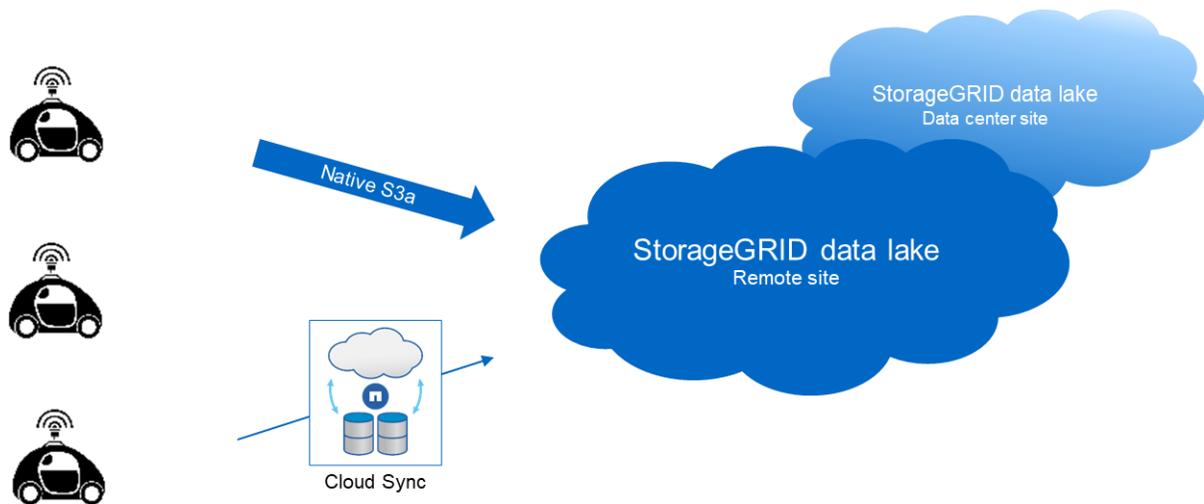
4.1 データの取り込み

データパイプラインは、データがどこで作成されていても、離れた場所にあっても企業のデータセンターにあっても開始されます。AVデータパイプラインでは、ほとんどのデータが調査車両で生成され、毎日何時間も運転して、可能な限り多くの現実世界のシナリオをキャプチャします。車両がガレージに戻ったら、そのデータを車両からAVソフトウェア開発サイクルの残りのステップを可能にする場所に移動する必要があります。調査車両が作成するデータの量は、多くの場合圧倒的です。リモートサイトからプライマリデータセンターへのデータの最初の取り込みと移動は、AVソフトウェア開発における最大の課題の1つです。

NetApp StorageGRIDは、さまざまな方法でリモートサイトに導入して、データパイプラインの取り込みポイントを提供できます。ハードウェアとソフトウェアのオプションを選択すると、各サイトに固有の運用をサポートするために必要なパフォーマンスと容量のレベルを達成できます。リモートサイトは、データの取り込みと格納を目的として独立して動作しますが、グリッド全体に対して確立されたILMポリシーにも参加します。データがグリッドに取り込まれると、メタデータタグを適用して場所、日付、またはその他の機能を指定できます。これらのタグを使用して、パイプライン内でのデータの移動を管理し、特定のタイプのデータを検索できます。

データは、アプリケーションやサーバイカーの構成に応じて、さまざまな方法で遠隔地のグリッドに取り込むことができます。新しいアプリケーションの多くは、S3プロトコルをネイティブに使用し、標準のHTTP API呼び出しを使用してデータをグリッドに直接アップロードできます。キャプチャされたデータを格納するために標準のファイルシステムに依存するアプリケーションもあります。その場合、何らかのツールを使用してデータをオブジェクトストアにコピーする必要があります。NetAppには、S3オブジェクトストアにデータをアップロードするために使用できる、NetApp Cloud SyncサービスやNetApp XCP移行ツールなどのデータムーバが用意されています。他にもいくつかのオープンソースツールやSDKが利用可能である。図4に、プロセスの第1フェーズで考えられる取り込みオプションを示します。

図4) AVパイプライン-データの取り込み



この検証では、サーバイカーからキャプチャされたRAW画像を表すImageNetデータセットから画像ファイルをアップロードしました。ImageNetテキスト注釈ファイルは、レーダーやライダーなど、キャプチャされる可能性のある他のデータタイプを表すために別々にアップロードされました。データは、AWS Boto3 S3 SDKを使用してS3プロトコル経由でファイルを転送するPythonスクリプトを使用してアップロードされました。画像とテキストファイルがアップロードされると、オブジェクトにも場所名とソースIDがタグ付けされます。データのコピーをローカルサイトに7日間保持し、データセンターサイトにもう1つコピーを作成して無期限に保持するように、アクティブなILMポリシーを設定しました。図3に、この検証用に設定されたILMポリシーを示します。

図5のスクリーンショットは、取り込まれたデータがグリッドに存在し、想定されるメタデータタグが設定されていることを示すElasticsearchクエリを示しています。同様のクエリを使用して、パイプラインの以降のフェーズで処理する特定のデータを特定します。

図5) StorageGRIDオブジェクトのElasticsearchインデックス

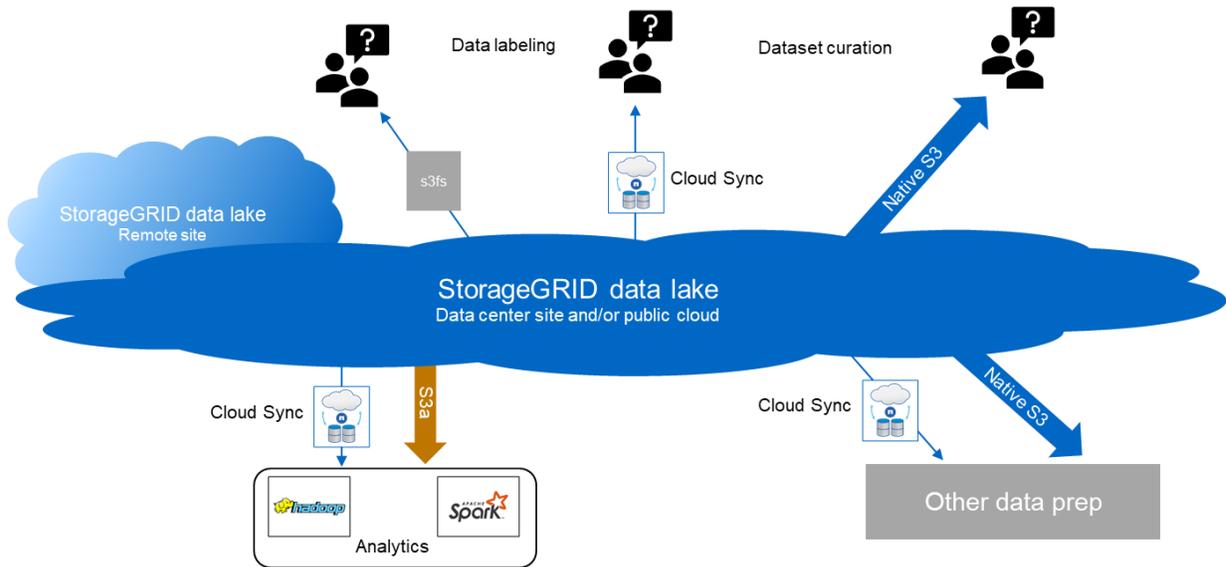
The screenshot shows the Elasticsearch Discover interface. The search query is 'n01440764'. The results show 1,804 hits. The interface includes a search bar, filters, and a list of search results. The results are displayed in a table format with the following fields: tags.Class_ID, bucket, key, accountId, size, md5, region, tags.site, tags.source, car1, and _id. The results are sorted by size in descending order.

tags.Class_ID	bucket	key	accountId	size	md5	region	tags.site	tags.source	car1	_id
n01440764	av-pipeline	n01440764_10029.JPEG	07997899040446180477	67,029	80bb097e038ca91b968731da5f106418	us-east-1	remote			av-pipeline_n01440764_10029.JPEG
n01440764	av-pipeline	n01440764_10026.JPEG	07997899040446180477	13,697	00c8d95c2695b34093e656be1412664	us-east-1	remote			av-pipeline_n01440764_10026.JPEG
n01440764	av-pipeline	n01440764_10042.JPEG	07997899040446180477	6,350	8e0978defd36099f14c00bc0d71b9ac1	us-east-1	remote			av-pipeline_n01440764_10042.JPEG
n01440764	av-pipeline	n01440764_10040.JPEG	07997899040446180477	146,489	9bf36a855e6a61cfb5bfa3552d0084d2	us-east-1	remote			av-pipeline_n01440764_10040.JPEG
n01440764	av-pipeline	n01440764_10027.JPEG	07997899040446180477	9,673	fbef96aa578fddcc2dd2f25f84be280	us-east-1	remote			av-pipeline_n01440764_10027.JPEG
n01440764	av-pipeline	n01440764_10043.JPEG	07997899040446180477	68,487	1e2d14641e2091e222e0f4a309a3f5	us-east-1	remote			av-pipeline_n01440764_10043.JPEG
n01440764	av-pipeline	n01440764_10048.JPEG	07997899040446180477	45,206	53e6917e834f53705d6d5b656337c8b9	us-east-1	remote			av-pipeline_n01440764_10048.JPEG

4.2 データファクトリ処理

取り込み後、AV開発サイクルのデータは、モデルのトレーニングに使用する前にいくつかの処理タスクを実行する可能性があります。データは、データファクトリーと呼ばれるものを通過します。ここでは、入力調査車からの生データであり、出力はAV開発サイクルの後続のステップで使用できるキューレーションデータです。データファクトリーは単一の場所にある場合もあれば、ユーザーやプロセスの一部が複数の場所に分散している場合もあります。図6は、仮想データファクトリーで発生する可能性のある多くのタスクのいくつかと、それらのステップがオブジェクトストレージパイプラインにどのように統合されるかを示しています。

図6) AVパイプライン-データファクトリ



データのラベル付けとアノテーション

データファクトリーの重要な作業の1つは、環境条件と物体識別のための画像データのラベル付けです。このタスクは通常人によって実行され、パフォーマンスは重要な要件ではありません。データには、通常、S3をネイティブにサポートするアプリケーションを使用して、オブジェクトストアから直接アクセスします。もう1つの方法は、クライアントシステムでs3fsなどのファイル変換レイヤを使用して、従来のPOSIXによるオブジェクトデータへのアクセスを提供する方法です。一般的に、このプロセスはデータを転送せずに実行できます。使用するアプリケーションに必要なのは、ソースデータにアクセスし、アノテーションやキューレーションデータをリポジトリにアップロードする方法だけです。

図7に、source 取り込みフェーズでタグが適用されたオブジェクトを検索するElasticsearchクエリを示します。

図7) StorageGRIDオブジェクトのタグ付け

The screenshot shows the NetApp Discover interface. At the top, there's a search bar with the query 'car1 and JPEG' and a 'Refresh' button. Below the search bar, there are options to 'Add filter' and 'Filter by type'. On the left, there's a sidebar with 'Selected fields' and 'Available fields'. The main area displays 1,350 hits for the search query. The results are shown as a list of objects with various fields highlighted in yellow and blue. The fields include 'key', 'md5', 'tags.source', 'bucket', 'accountId', 'region', 'tags.Class_ID', 'tags.site', '_id', '_type', '_index', and '_score'.

データのラベル付けプロセスをシミュレートするために、ユーザワークステーション上の別のPythonスクリプトを使用して前述のElasticsearchクエリを実行しました。クエリは、ラベル付けするオブジェクトを識別し、画像と注釈データのマッピングを確認しました。次に、検証済みデータにラベル付きデータを指定する新しいタグを付けました。このタグは、データセットキュレーションフェーズでラベル付けされたデータを識別し、使用可能なデータセットを作成するために使用されます。

データ分析

データファクトリーのもう1つの一般的なタスクは、特定の目的またはモデルのためのデータの選択を支援する非画像データの分析です。このタイプの分析は従来、Hadoopを使用して実行されてきましたが、アーキテクチャ上の大きな制限があるため、多くのお客様がこのユースケースにApache Sparkを利用しています。Sparkは、より少ないインフラでデータ分析を提供するためのより動的な機能を提供し、S3オブジェクトストアと直接インターフェイスしてデータをメモリにロードし、高速処理を実現します。

オブジェクトストレージパイプラインの一部として分析プロセスをシミュレートするために、取り込みフェーズでアップロードされたアノテーションファイルに対してスタンドアロンのSparkワードカウントジョブを実行しました。注釈ファイルは、画像内の特定のオブジェクトを囲むバウンディングボックスのX座標とY座標、およびその他のテキスト表記を含む単純なXMLファイルです。取り込みタグとファイルタイプを使用して分析対象のオブジェクトを特定する別のElasticsearchクエリを実行しました。このオブジェクトのリストは、コンテナ化されたスタンドアロンのSparkワードカウントジョブに渡され、オブジェクトがメモリにロードされ、要求された分析が実行されました。結果は出力として返され、別のファイルとしてオブジェクトストアにアップロードされました。図8は、テストスクリプトの出力を示しています。検索クエリで返されたファイルのリスト、word-countジョブの結果、オブジェクトストアにアップロードされたファイルが表示されます。

図8) StorageGRIDデータレイクからのSparkジョブの例

```
./run.sh

Starting pyspark container to execute wordcount.py
Searching annotation files for keywords 'pose', 'truncated', and 'difficult'

Started docker container spark-master

ELK query found 530 annotation files for class_id n01484850 in bucket av-pipeline

20/06/03 18:25:48 WARN NativeCodeLoader: Unable to load native-hadoop library for your
platform... using builtin-java classes where applicable
Using Spark's default log4j profile: org/apache/spark/log4j-defaults.properties
Setting default log level to "WARN".
To adjust logging level use sc.setLogLevel(newLevel). For SparkR, use setLogLevel(newLevel).

[Stage 0:=====>(529 + 1) / 530]

[Stage 1:=====>(529 + 1) / 530]

Saving results
Results saved to bucket av-pipeline as class_id_n01484850_posecount.txt
Confirming successful save of results file:
{
  "hits": {
    "hits": [
      {
        "_score": 11.963264,
        "_type": "all",
        "_id": "av-pipeline_class_id_n01484850_posecount.txt",
        "_source": {
          "region": "us-east-1",
          "bucket": "av-pipeline",
          "key": "class_id_n01484850_posecount.txt",
          "size": 19996,
          "accountId": "07997899040446180477",
          "md5": "fbd95fcd7adc4dedabd43a84bd6db0e"
        },
        "_index": "av-pipeline"
      }
    ],
    "total": {
      "relation": "eq",
      "value": 1
    },
    "max_score": 11.963264
  },
  "_shards": {
    "successful": 1,
    "failed": 0,
    "skipped": 0,
    "total": 1
  },
  "took": 64,
  "timed_out": false
}
```

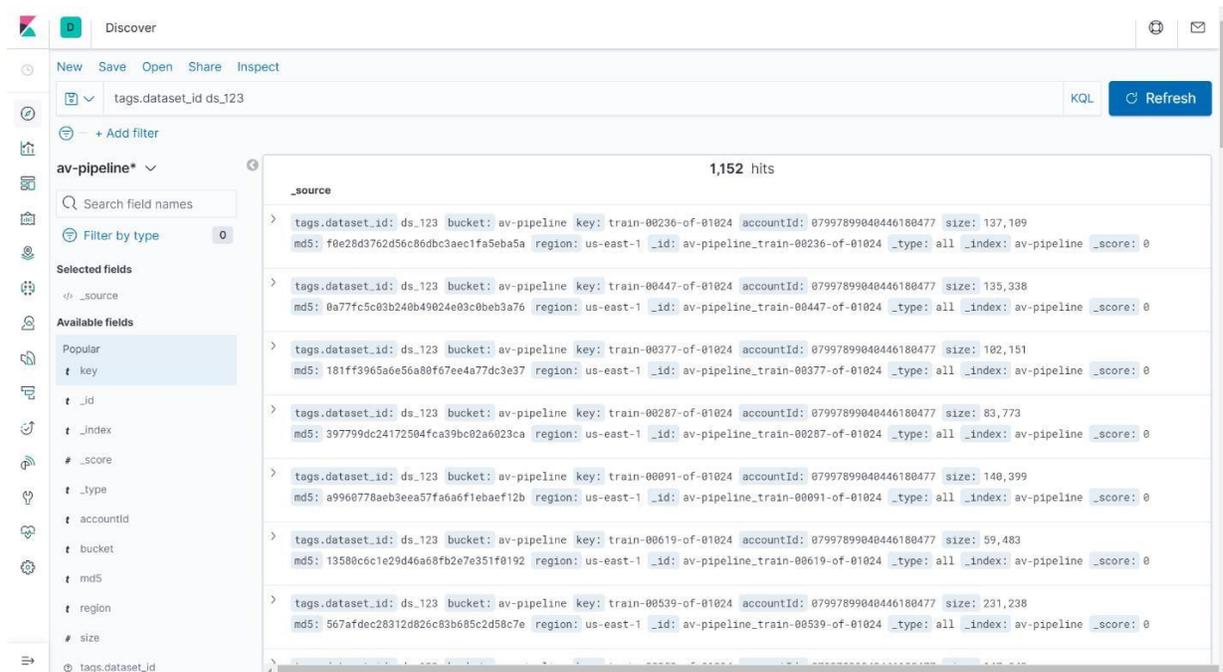
既存のHadoop環境を使用しているお客様は、NetApp Cloud Syncを使用して、StorageGRIDからHadoop分散ファイルシステム（HDFS）クラスタによってエクスポートされるNFS共有にデータを移動できます。

データセット期間

データファクトリ処理の最後のステップの1つは、モデルトレーニングのために特定のデータセットにデータを選択することです。このステップでは、特定のシーン、条件、オブジェクト、注釈など、モデルトレーニングの直接入力として必要なものをアセンブリし、使用可能な形式で処理します。

このデータファクトリプロセスのシミュレーションを完了するために、ImageNetデータセットの画像と注釈は、モデルトレーニングで使用するためにTFLord形式に処理されました。最後のPythonスクリプトを使用して、Elasticsearchインデックスに前の手順でラベル付けされたデータを照会し、S3 APIを使用してそのデータをローカルディレクトリにダウンロードしました。画像とアノテーションはTFCordファイルに処理され、データセットIDタグを使用してオブジェクトストアにアップロードされました。図9は、TFCord処理スクリプトによって適用されたデータセットIDに対する最終的なELKクエリの結果を示しています。

図9) データセットオブジェクトタグ



前述したように、これらのプロセスは単一の場所または複数の場所で発生する可能性があります。いずれの場合も、各手順で適用されたタグを使用してILMポリシーをトリガーし、必要に応じて別の場所にデータを移動したり、同じ場所にある別のグリッドストレージノードにデータを移動したりできます。この機能を使用すると、必要に応じてデータを自動的にアーカイブしたり、必要に応じてパフォーマンスの高いノードにデータを移動したりできます。また、このワークフローのどの時点でも、データをWrite Once Read Many (WORM) バケットに移動できます。この移動により、データは規制への準拠のために不変の形式で保存されるか、データが削除されます。これは、処理後に生の入力が必要なくなるためです。StorageGRIDでは、同じグリッド内で通常のバケットとWORMバケットがサポートされ、特定のオブジェクトが配置されているバケットに関係なく、同じILMポリシーとルールを適用できます。

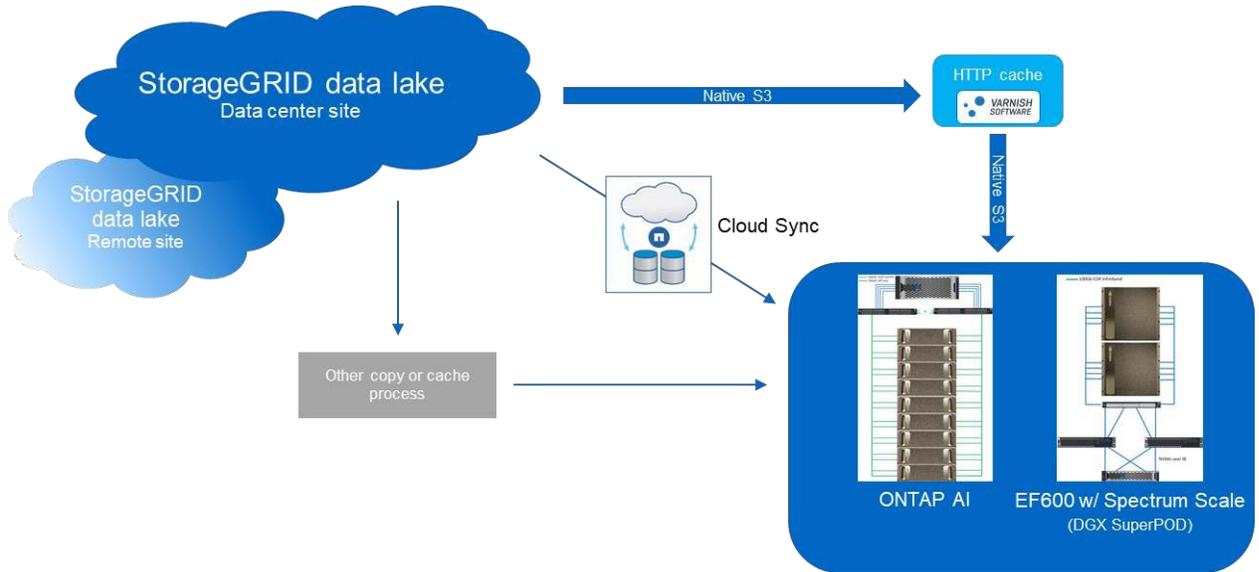
4.3 モデル トレーニング

DNNモデルトレーニングでは、AVソフトウェア開発プロセスのどの段階でも最高のパフォーマンスが求められます。データセットのサイズが膨大で、AVソフトウェア開発の大規模なGPUクラスターが必要とされる高いスループットには、オブジェクトストレージが通常提供できるパフォーマンスよりも高いパフォーマンスが必要です。これらの要件に対応するために、NetAppでは、ハイパフォーマンストレーニング向けに次の2つのインフラソリューションを提供しています。

- NetApp ONTAP AIリファレンスアーキテクチャ :
- NVIDIA DGX SuperPODアーキテクチャとNetApp EF600オールフラッシュアレイおよびIBM Spectrum Scaleファイルシステム

どのパフォーマンスストレージレイヤーが使用されているかに関係なく、データはオブジェクトストアのデータレイクからトレーニングクラスターに移動されます。図10は、NetApp Cloud SyncなどのData Mover、Varnish HTTPキャッシュなどのS3キャッシュプロセス、またはS3 APIを使用した別の自動コピープロセスを使用したデータの移動方法を示しています。

図10) AVパイプライン-モデルのトレーニング



このAVデータパイプライン例のトレーニングフェーズを完了するために、データキュレーションステップで指定されたデータセットIDのデータをElasticsearchインデックスに照会しました。そのデータはONTAP AIトレーニングクラスタにダウンロードされました。その後、データセットに対してTensorFlowトレーニングジョブが実行されました。図11は、サンプルのトレーニングプロセスの出力を示しています。共有NFSボリュームを使用してトレーニングデータを格納することで、データ移動プロセスを他のデータファクトリサーバで処理しながら、トレーニングクラスタへのハイパフォーマンスなボリュームアクセスを提供し、データセンターリソースを最適化できます。

図11) モデルトレーニングの例

```

root@gpu01:/# python3 download_train.py

Identified 1152 objects in ds_123, downloading to NFS mount point /mnt/av-pipeline/ds_123

Executing Tensorflow training job on 4 GPU

=====
== TensorFlow ==
=====

NVIDIA Release 20.01-tf1 (build 9338186)
TensorFlow Version 1.15.0

Container image Copyright (c) 2019, NVIDIA CORPORATION. All rights reserved.
Copyright 2017-2019 The TensorFlow Authors. All rights reserved.

Various files include modifications (c) NVIDIA CORPORATION. All rights reserved.
NVIDIA modifications are covered by the license terms that apply to the underlying project or
file.

NOTE: MOFED driver for multi-node communication was not detected.
Multi-node communication performance may be reduced.

===== Clean Cache !!! =====
mpirun -allow-run-as-root -np 1 -H localhost:1 bash -c 'sync; echo 1 > /proc/sys/vm/drop_caches'
=====
mpirun -allow-run-as-root -np 4 -H localhost:4 -bind-to none -map-by slot -x NCCL_DEBUG=INFO -x
LD_LIBRARY_PATH -x PATH python /netapp/tensorflow/benchmarks_190205/scripts/tf_cnn_benchmarks/tf_cnn_benchmarks.py --model=resnet50 --

```

```
batch_size=256 --device=gpu --force_gpu_compatible=True --num_intra_threads=1 --
num_inter_threads=48 --variable_update=horovod --batch_group_size=20 --num_batches=500 --
nodistortions - --num_gpus=1 --data_format=NCHW --use_fp16=True --use_tf_layers=False --
data_name=imagenet --use_datasets=True --datasets_parallel_interleave_cycle_length=10 --
datasets_sloppy_parallel_interleave=False --data_dir=/mnt/av-pipeline/ds_123/ --num_mounts=1 --
mount_prefix=/mnt/mount_%d --datasets_prefetch_buffer_size=2000 --
datasets_use_prefetch=True --datasets_num_private_threads=4 --horovod_device=gpu > /tmp/202006
05_105318_tensorflow_horovod_rdma_resnet50_gpu_4_256_b500_imagenet_nodistort_fp16_r10_m1_nockpt.t
xt 2>&1
Total images/sec = 4272.3825
```

AVソフトウェア開発サイクルのモデルトレーニングフェーズの詳細については、[TR-4799 『NetApp ONTAP AI Reference Architecture for Autonomous Driving Workloads』](#)を参照してください。

本番環境の開発環境では、結果として得られたモデルが他のデータセットと比較して評価され、検証と回帰テストが行われます。新しいモデルのパフォーマンスが許容範囲内であれば、最終モデルがオブジェクトストレージにコミットされます。この検証ケースでは、データセットIDとトレーニングプロセスログを含む小さなメタデータファイルとともに、モデルがWORMバケットにアップロードされました。これらのファイルは両方とも実験IDでタグ付けされており、それらを関連付けて簡単に検索できるようになっています。前の手順でこの処理をまだ実行していない場合は、データセット内の特定のデータオブジェクトをWORMバケットにコミットすることもできます。StorageGRIDを使用してWORMと標準のオブジェクトストレージを同じインフラ内で提供することで、トレーサビリティと再現性を1つのパイプラインで、必要に応じてプロセスの任意のステップで統合できます。

データライフサイクルポリシーは、プロセスのどの段階でも異なる方法で適用できます。たとえば、任意の手順でアーティファクトのコールドアーカイブを実行したり、スペース効率を高めるためにWORMリポジトリにコミットされたデータを削除したり、他の処理や用途のために別の場所に移動したりできます。

4.4 シミュレーションと再生

シミュレーションと再生には、実際のデータに対するコンポーネントレベルのテストと、モデルの動作を検証するための運転シナリオと条件のシミュレーションの両方が含まれます。AIを活用した自律走行車は、緊急車両、歩行者、動物など、さまざまな運転、交通、道路状況に正確に対応できなければなりません。また、その他の変動や障害、環境要因は事実上無限大です。これらの条件の多くは、現実世界でキャプチャしてテストするには危険すぎるか、または非常にまれであるため、効果的なモデルトレーニングに十分な速さでサンプルをキャプチャできません。AVプラットフォームをテスト車両に導入する前に、ハードウェアとソフトウェアのスタック全体を検証するには、シミュレーションとテストが不可欠です。

シミュレーションは、合成的に追加のシナリオを作成するための基盤として、事前にキャプチャされた画像またはビデオを使用するプロセスです。GPUレンダリング技術を使用すると、さまざまな自動車センサーに送信されるデータストリームに、より多くの要素または環境要因を追加できます。これらの追加により、特定の基本データオブジェクトを使用できる条件の数が拡張されます。必要とされる複雑なニューラルネットワークは、可能な限り多くのシナリオの例に公開する必要があります。したがって、このユースケースの拡張により、より高いレベルの自律性を達成するために必要な特定の条件付きデータサンプルの数が大幅に減少します。1つの基本シナリオがロードされると、プロセスのほぼすべてがコンピューティングに集中するため、ストレージに関しては、シミュレーションによって大きなストレージパフォーマンス要件は課されません。別の基本シナリオがロードされる前に、テストサイクル中に多数のバリエーションがリアルタイムで生成される可能性があります。

リプレイでは、センサー固有のデータの既存のデータセットを使用して、新しくトレーニングされたモデルとシステムのパフォーマンスを検証します。参照データセットは、新しいモデルを回帰テストのために既存のモデルのパフォーマンスと比較するために使用され、新しいデータセットは、新しいシナリオや条件に対する既存のモデルのパフォーマンスを評価するために使用される可能性があります。通常、Replayでは、データが既存の形式でハードウェアおよびソフトウェアスタックにストリーミングされるため、より重要なパフォーマンス要件が課されます。HIL(Hardware-in-the-Loop)環境では、通常、データはリアルタイムでストリーミングされます。センサーは、ハードウェアが移動しているかのように受信データを「認識」する必要があります。ソフトウェア・イン・ザ・ループ (SIL) テストにより、センサの要件が排除され、プロセスが大幅に高速化されます。SILの場合、ハイパフォーマンスストレージノードを使用してオブジェクトストアから直接データをストリーミングすることも、前述のセクション4で説明したいずれかの方法を使用して処理環境にデータをステージングすることもできます。

5 まとめ

AVおよびADASソフトウェア開発のデータパイプラインでは、トレーニング済みモデルが作成される前に多くの手順が実行されます。これらの手順では、データアクセス、パフォーマンス、保持に関するさまざまな要件が課されます。さらに、関連するデータの量が多いため、大規模環境でストレージシステムの対費用効果が非常に高いことが求められます。NetApp StorageGRIDは、大規模な非構造化データを管理するための自動データライフサイクル管理機能によって、堅牢で効率的なデータ保護機能とアーカイブ機能を提供します。ユーザレベルおよびデータレベルのセキュリティ機能を備えたNetApp StorageGRIDは、柔軟で統合された検索およびプロセスのワークフロー機能と組み合わせて、AVおよびADASソフトウェア開発用のデータレイクリポジトリに最適な解決策です。StorageGRIDは、その他のエンタープライズデータレイクのユースケースに適した優れた解決策でもあります。

謝辞

著者は、NetAppの主要なチームメンバーであるRobert Nagy、Aron Klein、Steven Pruchniewski、Erik Mulderの貢献を認めたいと思います。このホワイトペーパーの作成に大いに役立った洞察と専門知識を提供してくれたすべての個人に心からの感謝と感謝の意を表します。

詳細情報の入手方法

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。

- NetApp StorageGRID
<https://www.netapp.com/us/products/data-management-software/object-storage-grid-sds.aspx>
- NetApp ONTAP AI
<https://www.netapp.com/us/products/ontap-ai.aspx>
- Elasticsearch
<https://www.elastic.co/>
- PySpark
<https://spark.apache.org/docs/2.4.5/>
<https://spark.apache.org/docs/2.4.5/api/python/index.html>
- AWSボットS3 SDK
<https://aws.amazon.com/sdk-for-python/>

バージョン履歴

バージョン	日付	ドキュメントバージョン履歴
バージョン1.0	2020年9月	初版リリース

本ドキュメントに記載されている製品や機能のバージョンがお客様の環境でサポートされるかどうかについては、NetApp サポート サイトで [Interoperability Matrix Tool \(IMT\)](#) を参照してください。NetApp IMT には、NetApp がサポートする構成を構築するために使用できる製品コンポーネントやバージョンが定義されています。サポートの可否は、お客様の実際のインストール環境が公表されている仕様に従っているかどうかによって異なります。

機械翻訳に関する免責事項

原文は英語で作成されました。英語と日本語訳の間に不一致がある場合には、英語の内容が優先されます。公式な情報については、本資料の英語版を参照してください。翻訳によって生じた矛盾や不一致は、法令の順守や施行に対していかなる拘束力も法的な効力も持ちません。

著作権に関する情報

Copyright © 2024 NetApp, Inc. All Rights Reserved. Printed in the U.S. このドキュメントは著作権によって保護されています。著作権所有者の書面による事前承諾がある場合を除き、画像媒体、電子媒体、および写真複製、記録媒体、テープ媒体、電子検索システムへの組み込みを含む機械媒体など、いかなる形式および方法による複製も禁止します。

NetApp の著作物から派生したソフトウェアは、次に示す使用許諾条項および免責条項の対象となります。

このソフトウェアは、NetApp によって「現状のまま」提供されています。NetApp は明示的な保証、または商品性および特定目的に対する適合性の暗示的保証を含み、かつこれに限定されないいかなる暗示的な保証も行いません。NetApp は、代替品または代替サービスの調達、使用不能、データ損失、利益損失、業務中断を含み、かつこれに限定されない、このソフトウェアの使用により生じたすべての直接的損害、間接的損害、偶発的損害、特別損害、懲罰的損害、必然的損害の発生に対して、損失の発生の可能性が通知されていたとしても、その発生理由、根拠とする責任論、契約の有無、厳格責任、不法行為（過失またはそうでない場合を含む）にかかわらず、一切の責任を負いません。

NetApp は、ここに記載されているすべての製品に対する変更を随時、予告なく行う権利を保有します。NetApp による明示的な書面による合意がある場合を除き、ここに記載されている製品の使用により生じる責任および義務に対して、NetApp は責任を負いません。この製品の使用または購入は、NetApp の特許権、商標権、または他の知的所有権に基づくライセンスの供与とはみなされません。

このマニュアルに記載されている製品は、1 つ以上の米国特許、その他の国の特許、および出願中の特許により保護されている場合があります。

本書に含まれるデータは市販の製品および / またはサービス（FAR 2.101 の定義に基づく）に関係し、データの所有権は NetApp, Inc. にあります。米国政府は本データに対し、非独占的かつ移転およびサブライセンス不可で、全世界を対象とする取り消し不能の制限付き使用権を有し、本データの提供の根拠となった米国政府契約に関連し、当該契約の裏付けとする場合にのみ本データを使用できます。前述の場合を除き、NetApp, Inc. の書面による許可を事前に得ることなく、本データを使用、開示、転載、改変するほか、上演または展示することはできません。国防総省にかかる米国政府のデータ使用権については、DFARS 252.227-7015(b) 項で定められた権利のみが認められます。

商標に関する情報

NetApp、NetApp のロゴ、<https://www.netapp.com/company/legal/trademarks/> に記載されているマークは、NetApp, Inc. の商標です。その他の会社名と製品名は、それを所有する各社の商標である場合があります。