



Technisches Whitepaper

Entwicklung und Aufbau einer Daten-Pipeline für AI-Workflows

Einsatz von künstlicher Intelligenz (Artificial Intelligence, AI), Machine Learning (ML) und Deep Learning (DL) im gesamten Datacenter und in der Cloud des Unternehmens

Santosh Rao, NetApp
März 2018 | WP-7264

Executive Summary

Unternehmen setzen zunehmend auf AI-Technologien (künstliche Intelligenz) wie Deep Learning (DL), um neue Services einzuführen und bessere Erkenntnisse aus Unternehmensdaten zu gewinnen. Bei dem Übergang vom Proof of Concept zum Betrieb einer Deep-Learning-Technologie müssen Data-Science-Teams eine vollständige Architektur der Daten aufbauen, um Engpässe zu beseitigen und schnellere Modelliterationen zu ermöglichen.

Die Entwicklung dieser Architektur der Daten erfordert einen ganzheitlichen Denkansatz über die Daten-Pipeline von der Datenaufnahme und Edge-Analyse über die Datenvorbereitung und das Training im Core-Datacenter bis zur Archivierung in der Cloud. Dabei müssen die Performance-Anforderungen sowie die erforderlichen Datensätze und Datenservices unbedingt bekannt sein. Ebenso wichtige Kriterien sind Erweiterbarkeit und Unterstützung in der Zukunft, da sich Deep-Learning-Hardware und Cloud-Ansätze immer weiter entwickeln.

In diesem Whitepaper werden die Herausforderungen einer AI-Infrastruktur erläutert und es wird ausgeführt, wie NetApp Sie bei dem Aufbau einer Daten-Pipeline für Ihre aktuellen Deep-Learning-Workflows unterstützen kann, während Ihre Investitionen in die AI-Infrastruktur zukunftssicher bleiben. Eine sorgfältige Infrastrukturplanung erleichtert den Datenfluss durch die Deep-Learning-Pipeline, ermöglicht eine schnellere Implementierung und bietet potenziell maximale Wettbewerbsvorteile.

INHALTSVERZEICHNIS

1 Einführung: Ist Ihre Infrastruktur für AI-Workflows in der Produktion gerüstet?.....	3
2 Datenfluss in einer Deep-Learning-Pipeline	4
3 Beschleunigung der I/O in einer Deep-Learning-Pipeline.....	5
3.1 Beseitigung von Engpässen am Edge	5
3.2 Beseitigung von Engpässen vor Ort.....	6
3.3 Beseitigung von Engpässen in der Cloud	8
4 Filesystem und Architektur der Daten für eine Deep-Learning-Pipeline	9
4.1 Datenfluss in das Training-Cluster	11
4.2 Weitere Performance-Faktoren.....	12
5 NetApp Technologien und die Deep-Learning-Pipeline	15
6 Zukunftssichere Deep-Learning-Pipeline.....	15
6.1 Planung für die Weiterentwicklung der Core-Hardware	16
Fazit: Behalten Sie Ihre Daten-Pipeline und Ihre AI-Zukunft unter Kontrolle.....	17

LISTE DER TABELLEN

Tabelle 1: Wichtige Fragen und Überlegungen.....	9
--	---

ABBILDUNGSVERZEICHNIS

Abbildung 1: Eine für Deep Learning entwickelte Daten-Pipeline kann noch weitere AI- und Big-Data-Workflows umfassen.....	3
Abbildung 2: Eine Deep-Learning-Pipeline kann sich entweder vor Ort oder in der Cloud befinden.	4
Abbildung 3: Mithilfe von Edge-Analysen mit Daten-Tiering lassen sich Daten vom Edge mit hoher Priorität versehen und an den Core weiterleiten oder mit niedriger Priorität versehen und archivieren.	6
Abbildung 4: Deep-Learning-Pipeline mit Core vor Ort	7
Abbildung 5: Durch die Platzierung der Daten in Cloud-Nähe profitieren Sie von Cloud-Computing und Datenbeschleunigung, während Sie die vollständige Kontrolle behalten.	8
Abbildung 6: Unstrukturierte Daten können im Data Lake zusammengeführt und in das Training-Cluster übertragen werden.....	11
Abbildung 7: Strukturierte Daten werden mittels kleiner zufälliger I/O-Vorgänge gelesen und im Training-Cluster zusammengeführt.....	11
Abbildung 8: NetApp Data-Fabric-Technologien	15
Abbildung 9: Der Core der AI/ML/DL-Pipeline entwickelt sich kontinuierlich weiter.....	16

1 Einführung: Ist Ihre Infrastruktur für AI-Workflows in der Produktion gerüstet?

Unternehmen in allen Branchen setzen auf AI-Technologien (künstliche Intelligenz), um neue Services einzuführen und neue Erkenntnisse aus Unternehmensdaten zu gewinnen. Bei dem Übergang vom Proof of Concept zum Betrieb einer AI-Technologie stehen Data-Science-Teams jedoch häufig vor Problemen mit dem Datenmanagement. So kann beispielsweise das Verschieben oder Kopieren von Daten zwischen mehreren Daten-Repositorys sehr mühsam sein. Auch die Einhaltung von Service-Leveln in Produktionsqualität bei Performance und Sicherung großer und dynamischer Datensätze erweist sich häufig als schwierig.

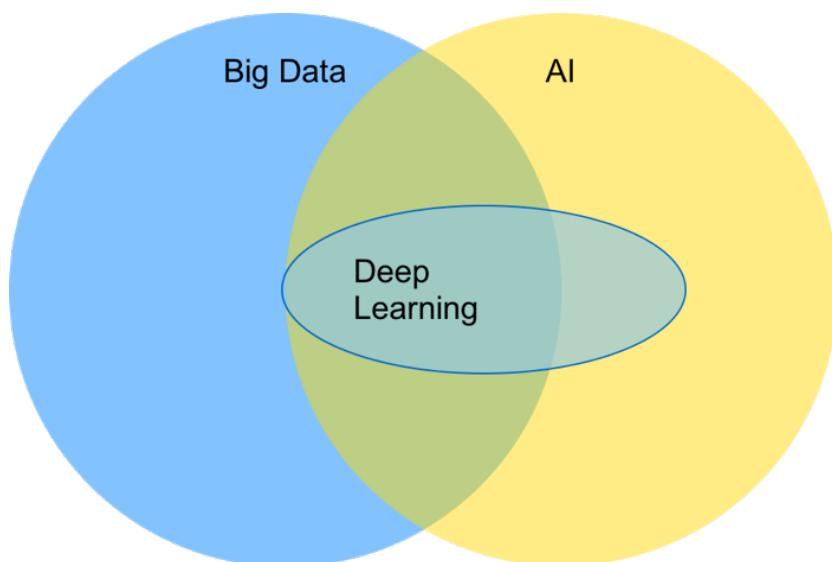
Einer der Gründe für diese Schwierigkeiten liegt darin, dass der für eine effektive AI-Technologie nötige Datenfluss nicht im Datacenter isoliert ist. Unternehmen verschiedenster Branchen und Größen, die das Internet of Things (IoT) und AI-Technologien nutzen, sind im gesamten Datacenter (Edge, Core) ebenso wie in der Cloud mit datenbezogenen Herausforderungen konfrontiert.

Viele Automobilhersteller erfassen beispielsweise Daten von immer mehr Fahrzeugen (Edge). Mit diesen Daten werden die für den autonomen Betrieb erforderlichen AI-Algorithmen trainiert (Core). Da die Datensätze exponentiell wachsen und zur Wiederverwendung gespeichert werden müssen, ist eine skalierbare, kostengünstige Plattform erforderlich (Cloud). Heute stoßen Automobilhersteller bereits häufig an die Grenzen der IT-Technologie. Internationale Einzelhändler stehen bei der Erstellung von Inferenzmodellen auf der Grundlage von Daten der Point-of-Sale-Geräte aus sämtlichen Filialen weltweit vor ähnlichen Herausforderungen.

Als größte Herausforderung bei AI-Daten wird häufig die richtige Performance genannt. Die Performance ist für den Core der AI-Pipeline ein wichtiges Kriterium. Noch wichtiger ist jedoch, dass die Daten-Pipeline den gesamten Datenfluss von der Aufnahme bis zur Archivierung umfasst, um eine erfolgreiche Umsetzung bei optimaler Performance, Effizienz und Wirtschaftlichkeit in jeder Phase zu ermöglichen.

In diesem Whitepaper werden die Herausforderungen einer AI-Infrastruktur erläutert und es wird ausgeführt, wie NetApp Sie bei dem Aufbau einer Daten-Pipeline für Deep Learning (DL) unterstützen kann. Da Deep Learning in Bezug auf Rechen- und I/O-Leistung der anspruchsvollste AI-Workflow ist, umfasst eine für Deep Learning entwickelte Daten-Pipeline darüber hinaus noch weitere AI- und Big-Data-Workflows. (Abb. 1)

Abbildung 1: Eine für Deep Learning entwickelte Daten-Pipeline kann noch weitere AI- und Big-Data-Workflows umfassen.

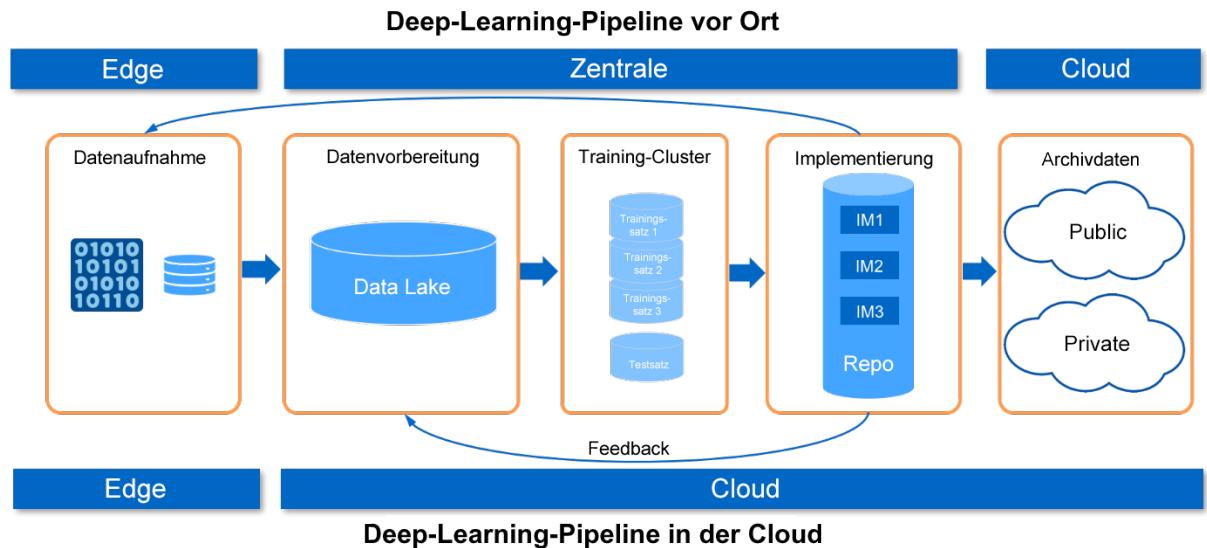


2 Datenfluss in einer Deep-Learning-Pipeline

Bei der Entwicklung einer Daten-Pipeline für AI oder Deep Learning sollten zunächst die folgenden, in Abbildung 2 gezeigten Schritte berücksichtigt werden:

1. **Datenaufnahme:** Die Aufnahme erfolgt in der Regel am Edge, etwa bei der Erfassung der Daten von Autos oder Point-of-Sale-Geräten. Je nach Anwendungsfall kann am Aufnahmepunkt oder in dessen Nähe eine IT-Infrastruktur erforderlich sein. So werden etwa bei Einzelhändlern die Daten mehrerer Geräte konsolidiert, da das Platzangebot in den Filialen begrenzt ist.
2. **Datenvorbereitung:** Die Vorbereitung ist zur Normalisierung der Daten vor dem Training erforderlich. Dieser Schritt erfolgt in einem Data Lake, und zwar häufig in der Cloud in Form einer S3 Tier oder vor Ort als File- oder Objektspeicher.
3. **Training:** Für die entscheidende Trainingphase von Deep Learning werden die Daten meist in regelmäßigen Abständen aus dem Data Lake in das Training-Cluster kopiert. In dieser Phase werden häufig Server mit Grafikprozessoren (Graphics Processing Units, GPUs) oder kundenspezifischen Chipsätzen eingesetzt, um die Vorgänge zu parallelisieren, wodurch eine enorme Datennachfrage entsteht. Daher ist die Brutto-I/O-Bandbreite ein entscheidendes Kriterium.
4. **Implementierung:** Das so entstandene Modell wird zunächst in eine Testumgebung und schließlich in die Produktion überführt. Je nach Anwendungsfall kann das Modell auch für Edge-Vorgänge eingesetzt werden. Die Ergebnisse in der Praxis werden überwacht und das Feedback fließt zusammen mit neuen Daten wieder in den Data Lake ein, um den Prozess zu wiederholen.
5. **Archivierung:** Kalte Daten aus vorherigen Zyklen können unendlich gespeichert werden. Viele AI-Teams nutzen dazu Objekt-Storage in einer Private oder Public Cloud.

Abbildung 2: Eine Deep-Learning-Pipeline kann sich entweder vor Ort oder in der Cloud befinden.



In vielen Unternehmen wurde der Versuch unternommen, diese Art von Daten-Pipeline entweder vor Ort oder in der Cloud aufzubauen. Dabei wird meist auf Standard-Hardware und Datenmanagement nach der Holzhammermethode gesetzt. Die Cloud kann dagegen hohe Kosten verursachen. Das Verschieben großer Datenmengen in die Cloud wird schnell teuer und wenn sich die Daten erst einmal dort befinden, folgt die gesamte Pipeline meist bald nach. In jedem Fall treten früher oder später Engpässe auf, wenn das Projekt in die Produktion überführt wird und die Datenmenge damit weiter zunimmt.

Die größten Engpässe entstehen dabei in der Trainingphase, in der sehr viel I/O-Bandbreite mit extremer I/O-Parallelität erforderlich ist, um das Training-Cluster für Deep Learning mit den zu verarbeitenden Daten zu speisen. Nach der Trainingphase werden die entstandenen Inferenzmodelle häufig in einem DevOps-artigen Repository gespeichert, sodass ein Zugriff mit äußerst niedriger Latenz möglich ist.

Wenn die Daten jedoch nicht ab der Aufnahme reibungslos durch die gesamte Pipeline fließen können, wird die AI-Daten-Pipeline niemals ihre volle Produktivität erreichen. Außerdem erfordert das Management der Pipeline immer mehr Arbeitszeit.

3 Beschleunigung der I/O in einer Deep-Learning-Pipeline

Unabhängig von der Ausführung des AI-Workflows vor Ort oder in der Cloud können Engpässe im Betrieb den Abschluss jedes Trainingzyklus verzögern. Dadurch sinkt nicht nur die Produktivität der Pipeline, sondern es geht auch wertvolle Arbeitszeit verloren.

In diesem Abschnitt werden Lösungen zur Beseitigung von I/O-Engpässen im gesamten Datacenter (Edge, Core) sowie in der Cloud beschrieben; im Einzelnen:

- Engpässe am Edge, die die Datenaufnahme verzögern
- Engpässe vor Ort
- Engpässe in der Cloud

Insbesondere für die drei Hauptphasen der Pipeline – Datenvorbereitung, Training und Implementierung – bestehen besondere I/O-Anforderungen, die zu erfüllen sind.

3.1 Beseitigung von Engpässen am Edge

Die von smarten Edge-Geräten und zahlreichen Aufnahmepunkten generierten Datenmengen können die Rechen-, Storage- und Netzwerkressourcen am Edge schnell überfordern. Bei dem Fluss dieser Daten in das Datacenter oder in die Cloud entstehen daher leicht Engpässe.

Mithilfe von Edge-Analysen lassen sich die Daten bei der Aufnahme verarbeiten und selektiv weiterleiten. Dieser Ansatz erfordert jedoch am Edge eine Infrastruktur mit High-Performance-Storage und äußerst niedriger Latenz. Viele Kunden von NetApp gehen an dieser Stelle hierarchisch vor: Sie setzen die Infrastruktur auf der letzten Meile ein und lassen Sensoren am Edge als Endpunkte agieren.

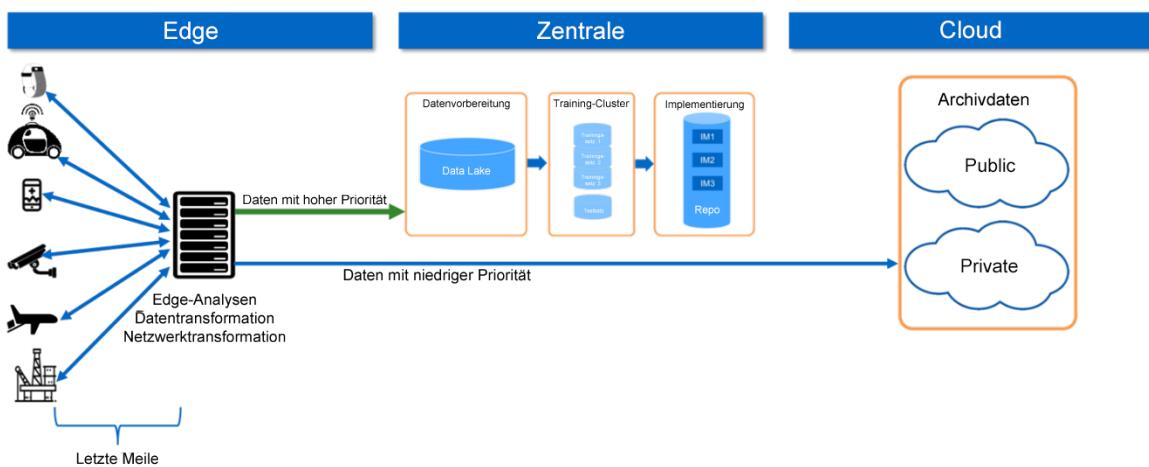
Beispielsweise geben die Sensoren von Fertigungsanlagen die Daten zur Aggregation und Analyse an die in jedem Werk eingesetzte Infrastruktur weiter, sodass sie selektiv in der Kette weitergeleitet werden können. Diese Vorgehensweise eignet sich auch für autonome Fahrzeuge (hier können täglich Datenmengen von bis zu 7 TB je Endpunkt generiert werden), den Einzelhandel und viele weitere Gebiete.

Datenmanagement-Tiering am Edge

Mit Edge-Analysen lassen sich verschiedene Datenservice-Tiers erstellen. Dabei werden bestimmte Daten entweder durch einfache Filterung oder durch erweiterte Analysen und AI priorisiert und effizient an die AI/ML/DL-Pipeline weitergeleitet. Die übrigen Daten werden mit niedrigerer Priorität behandelt und können entweder verworfen oder mit einer anderen Serviceklasse gemanagt werden.

Je nach Anforderungen kann jede Daten-Tier mit unterschiedlichen Transformationen verarbeitet werden, um die erforderliche Storage-Effizienz und Sicherheit zu erzielen. Daten mit niedriger Priorität lassen sich beispielsweise komprimieren, deduplizieren, verschlüsseln und zur späteren Verarbeitung oder aus Compliance-Gründen in einem Cloud-Repository speichern. (Abb. 3)

Abbildung 3: Mithilfe von Edge-Analysen mit Daten-Tiering lassen sich Daten vom Edge mit hoher Priorität versehen und an den Core weiterleiten oder mit niedriger Priorität versehen und archivieren.



Die Fähigkeit zur Verarbeitung der Edge-Analysen ist von der verfügbaren Rechenleistung abhängig. Anbieter von Computing- und Cloud-Ressourcen konkurrieren mit unterschiedlichen Strategien in Bezug auf den Platzbedarf am Edge. So bietet etwa NVIDIA Lösungen für die GPU-Leistung am Edge an, um AI für Applikationen wie selbstfahrende Autos zu ermöglichen. Aus Datensicht haben diese Lösungen eines gemeinsam, nämlich die Nutzung von Standard-DAS ohne intelligentes Datenmanagement. Der Bedarf an intelligentem Storage ist daher offensichtlich.

Smarte Data Mover

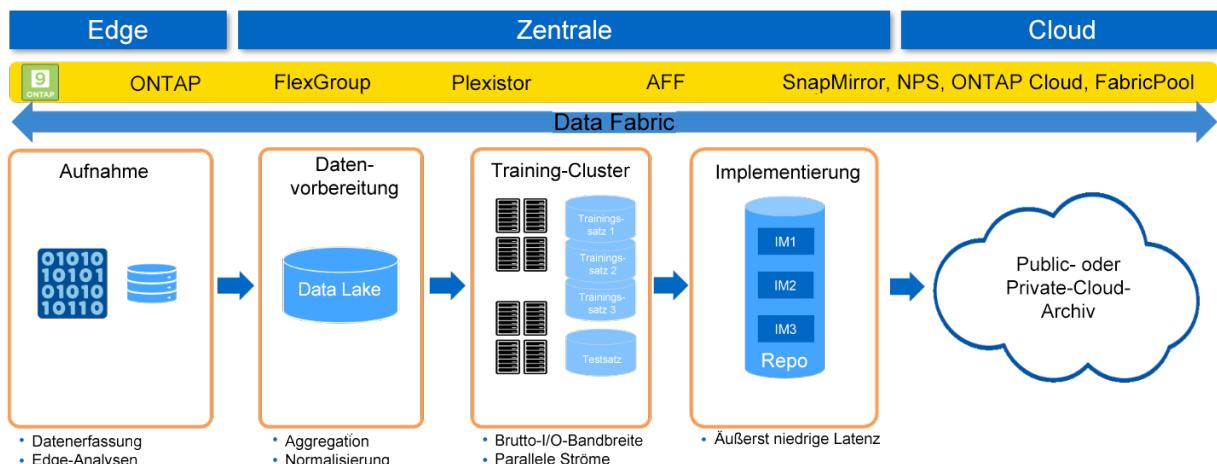
Ein smarter Data Mover ist eine zentrale Voraussetzung für hohe Bandbreite am Edge. In der heute am häufigsten eingesetzten Architektur werden die Daten in Form von S3 Put-Vorgängen verschoben. Dieser Ansatz hat jedoch den Nachteil, dass die Daten im großen Stil ohne Datentransformation verschoben werden.

Stattdessen können smarte Data Mover eingesetzt werden, um die Daten zusammenzuführen, durch Datentransformation den Speicherplatzbedarf zu reduzieren und durch Netzwerktransformation nur geänderte Blöcke zu verschieben. Auf diese Weise lässt sich die Datenverschiebung erheblich beschleunigen, während gleichzeitig die Bandbreitenanforderungen sinken.

3.2 Beseitigung von Engpässen vor Ort

Wenn sich der Core der Deep-Learning-Pipeline vor Ort befindet (wie in Abbildung 4 gezeigt), lassen sich Data Lake, Training-Cluster und Implementierung des Inferenzmodells direkt steuern.

Abbildung 4: Deep-Learning-Pipeline mit Core vor Ort



Data Lake

Die vom Edge eingehenden Daten werden in einem Data Lake erfasst. Ein unangemessen implementierter Data Lake wird bei zunehmenden Datenmengen unweigerlich zum Engpass. Bei einem Data Lake kann es sich um eine Hadoop Implementierung mit Hadoop Filesystem (HDFS) oder eine Implementierung mit Objekt- oder Filespeicher handeln. HDFS ist jedoch nicht für Performance optimiert und erstellt in der Regel drei Kopien jedes Datenobjekts, wodurch die Schreib-Performance sinkt und die Kosten steigen.

Objektspeicher wurden ursprünglich für die Archivierung in der Cloud statt für Performance optimiert, werden aber inzwischen meist als Standard-Datastore für Big-Data-Projekte genutzt. Wie bereits erwähnt, sind Objektspeicher für Deep Learning in Bezug auf die Performance allerdings nur bedingt geeignet.

Filespeicher mit Scale-out-Filesystemen wie Lustre und GPFS sind auf die Batch-Verarbeitung bei High-Performance-Computing (HPC) ausgelegt und können kleinere Datei-Workloads nicht angemessen verarbeiten. Die aus smarten Edge-Geräten in den Data Lake fließenden Daten stammen jedoch meist aus vielen kleinen Dateien und da diese Systeme dafür nicht optimiert sind, sinkt die Performance.

Mit NetApp All Flash FAS, insbesondere in Verbindung mit NFS und ONTAP FlexGroup Volumes, gehören die Einschränkungen anderer Data-Lake-Ansätze der Vergangenheit an. FlexGroup Gruppen liefern sowohl bei bandbreitenorientierten Batch-Workloads als auch bei kleinen Datei-Workloads eine hohe Performance. Die anderen genannten Data-Lake-Lösungen – HDFS, Objekt-Storage, Lustre, GPFS und anderer Scale-out-File-Storage – können für bestimmte Anwendungsfälle geeignet sein. Sie liefern jedoch nicht sowohl für sequenzielle als auch für zufällige I/O-Vorgänge eine gleichermaßen gute Performance.

Training-Cluster

Der aktuelle Stand bei Deep-Learning-Training-Clustern ist ein Scale-out-Cluster mit 32 bis 64 Servern und 4 bis 8 GPUs pro Server. Aus I/O-Sicht müssen alle GPUs immer zu 100 % ausgelastet sein. Jeder CPU-Core muss also mit einem parallelen I/O-Datenstrom versorgt werden. Jede CPU ist wiederum einer GPU zugeordnet. Die CPU verarbeitet den Datenstrom, führt die I/O-Vorgänge zusammen und leitet die Daten an die GPU weiter.

Bei diesem Prozess entstehen auf folgende Weise I/O-Engpässe:

- Die Daten müssen schnell und effizient vom Data Lake an das Training-Cluster gestreamt werden.
- Dazu müssen bis zu 256 parallele I/O-Datenströme (32 bis 64 Server mit jeweils 4 bis 8 GPUs) jederzeit vollständig ausgelastet und bereitgestellt sein, damit die GPUs zu keinem Zeitpunkt auf Daten warten müssen.

Die Softwarearchitektur von NetApp ONTAP erfüllt beide Anforderungen auf einzigartige Weise. Der Data Lake kann mit Hybrid-Flash-Nodes konzipiert werden, da diese die Daten mit äußerst hoher Bandbreite an das Training-Cluster streamen können. All-Flash-Storage-Nodes für das Training-Cluster liefern Bandbreiten von bis zu 18 GB/s je HA-Paar mit zwei Controllern und Latenzen von unter 500 Mikrosekunden und unterstützen damit zahlreiche parallele I/O-Datenströme. Darüber hinaus bietet NetApp eine Technologie-Roadmap, die eine weitere Erhöhung der I/O-Performance für die AI/ML/DL-Pipeline bei steigenden Anforderungen ermöglicht.

Implementierung

Nach Abschluss des Trainings werden die entstehenden Inferenzmodelle in einem DevOps-artigen Repository gespeichert und Inferenztests sowie einer Hypothesenvalidierung unterzogen. In dieser Phase ist die Nutzung von Storage-Systemen mit äußerst niedriger Latenz besonders wichtig.

Bei NetApp kann eine einzelne Storage-Architektur alle Performance-Anforderungen der zentralen Schritte einer Deep-Learning-Pipeline erfüllen. Obwohl dies unmittelbare Vorteile bietet, betreiben die meisten Kunden heute separate Cluster für jede Phase der Pipeline. So können für Big-Data-Pipelines bereits implementierte Data Lakes genutzt werden. Es empfiehlt sich möglicherweise, nur die neuen, für Deep Learning erforderlichen Elemente im Rahmen eines separaten Projekts zu implementieren und die Daten von einer Phase in die andere zu kopieren. Mit zunehmendem Datenwachstum muss die Pipeline jedoch weiter vereinheitlicht werden. Auch dies ist mit All Flash FAS möglich.

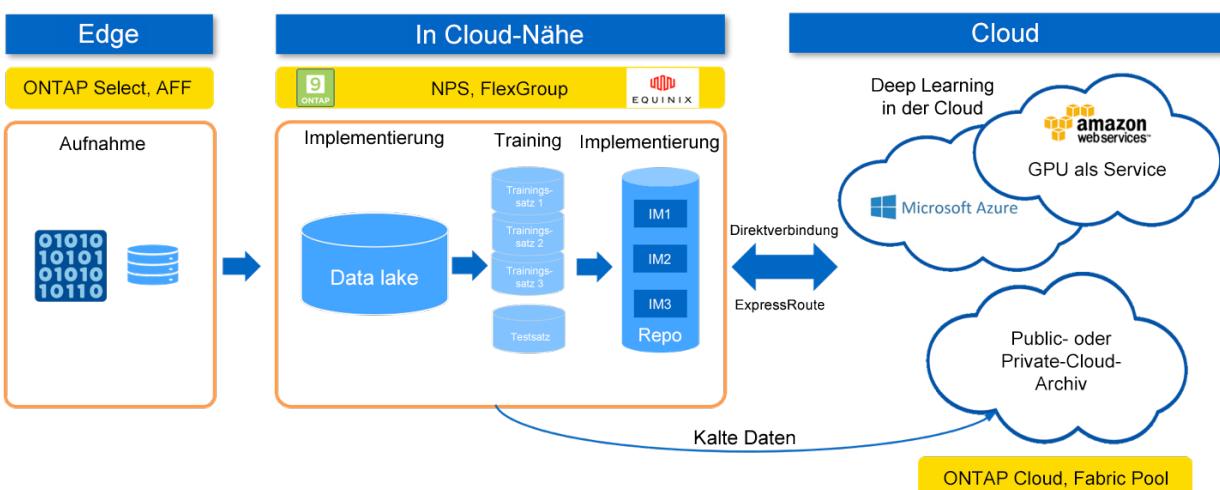
3.3 Beseitigung von Engpässen in der Cloud

Möglicherweise entscheiden Sie sich aufgrund der Agilität und der einfachen Nutzung für eine Implementierung von Deep Learning in der Cloud. Auch bei der Ausführung der Deep-Learning-Pipeline in der Cloud können jedoch dieselben potenziellen Engpässe auftreten:

- Liefert der Data Lake die geeignete Performance für die Datenaufnahme? Können die Daten in das Training-Cluster gestreamt werden?
- Kann Ihr Cloud-Provider die für das Training-Cluster erforderliche I/O-Parallelität liefern?
- Wie können Sie die für die Inferenzmodelle erforderliche extrem niedrige Latenz erzielen?
- Können Sie die Datenhoheit für sensible Daten gewährleisten?

Mit NetApp Private Storage (NPS) können Sie die Daten in Cloud-Nähe speichern und so die Funktionen von Public-Cloud-Computing und andere Services nutzen, während Sie die vollständige Kontrolle über Ihre Daten behalten. (Abb. 5) NPS stellt die im vorherigen Abschnitt beschriebene Architektur und Performance in einer Public Cloud bereit. Damit sind Probleme mit der Datenhoheit beseitigt und Ihre Daten sind nicht allein auf die Cloud angewiesen.

Abbildung 5: Durch die Platzierung der Daten in Cloud-Nähe profitieren Sie von Cloud-Computing und Datenbeschleunigung, während Sie die vollständige Kontrolle behalten.



Wenn Ihre Daten unbedingt in der Cloud gespeichert werden müssen, können Sie mit NetApp Data Fabric Ihre NFS Daten im Azure oder AWS Cloud-Service speichern und nahtlos managen.

4 Filesystem und Architektur der Daten für eine Deep-Learning-Pipeline

In einer AI-Pipeline haben die vom Edge eingehenden Daten und die vom Data Lake in das Training-Cluster fließenden Daten unterschiedliche I/O-Eigenschaften. In diesem Abschnitt werden die individuellen Optionen erläutert, die für einen reibungslosen Datenfluss durch die Daten-Pipeline in das Training-Cluster zu berücksichtigen sind.

Stellen Sie sich die GPUs im Training-Cluster als PS-starkes Auto vor. Der Unterschied einer guten Daten-Pipeline ist vergleichbar mit einer Fahrt auf einer Rennstrecke gegenüber einer Autobahnfahrt zur Stoßzeit. Um optimale Ergebnisse mit einer Implementierung für AI einschließlich ML und DL zu erzielen, ist die Daten-Pipeline wohl mit Abstand das wichtigste Kriterium, das dennoch häufig übersehen wird. Eine optimale Architektur der Daten berücksichtigt die I/O-Anforderungen am Edge, im Data Lake und im Training-Cluster.

Objekt-Storage ist nicht auf das von einer Daten-Pipeline benötigte Maß an Performance ausgelegt. Objektspeicher wurden ursprünglich für die Archivierung in der Cloud statt für Performance optimiert, werden aber inzwischen meist als Standard-Datastore für Big-Data-Projekte genutzt. Besonders für Deep Learning sind Objektspeicher in Bezug auf die Performance allerdings nur bedingt geeignet.

Wählen Sie daher ein Filesystem und eine Architektur der Daten unter Berücksichtigung aller für Ihre AI-Umgebung wichtigen Faktoren. Zwar ist File-basierter Storage meist eine gute Wahl; darüber hinaus sind jedoch noch viele weitere Faktoren zu berücksichtigen (siehe Tabelle 1).

Tabelle 1: Wichtige Fragen und Überlegungen

Wichtige Fragen	Wichtige Überlegungen
Welche Filesysteme kommen in Betracht?	<ul style="list-style-type: none"> • Scale-out-Filesysteme wie Lustre oder GPFS • HDFS, ein häufig genutztes Big-Data-Filesystem • NFS, das seit 30 Jahren am meisten eingesetzte Shared Filesystem für technische Applikationen
Kann das Filesystem sowohl strukturierte als auch unstrukturierte Daten aus verschiedenen Datenquellen unterstützen und bündeln, ohne dass die Performance beeinträchtigt wird?	<ul style="list-style-type: none"> • Protokoll- und Sensordaten • Datenbanken wie RDBMS und NoSQL • Zufällige I/O-Vorgänge für verschiedene Datenbanktypen: Tabellenscans, Lesevorgänge für Dokumente und erfasste Daten in NoSQL, Lesevorgänge für Spalten in spaltenbasierten Datenbanken und zufällige Lesevorgänge für Schlüssel/Wert-Paare in Schlüssel/Wert-Datenbanken • Sequenzielle I/O-Vorgänge für In-Memory-Datenbanken und In-Memory-Engines wie Spark • E-Mail-Protokolle • Home Directories • Andere Quellen
Liefert das System eine ausreichende Performance für kleine, zufällige I/O-Vorgänge im Vergleich zu sequenziellen I/O-Vorgängen?	<ul style="list-style-type: none"> • Einige Datenquellen generieren zufällige, andere dagegen sequenzielle I/O-Vorgänge. • Das Filesystem muss die Performance für beide I/O-Typen in Einklang bringen können.
Welche Performance und Funktionen bieten die Data Mover?	<ul style="list-style-type: none"> • Höchste Performance • Effizienteste Datenverschiebung
Kann das System den Daten-Lebenszyklus automatisieren?	<ul style="list-style-type: none"> • Intelligente Filterung zur Aufteilung der Daten nach Core/Archiv-Tiers • Echtzeit-Performance für Filterentscheidungen
Unterstützt das System die neuesten Storage- und Speichermedien für unterbrechungsfreie Fortschritte bei Performance und Latenz?	<ul style="list-style-type: none"> • Storage-Tiers mit dem erforderlichen Preis-Leistungs-Verhältnis für den Datastore, einschließlich Storage-Class Memory (SCM), Nonvolatile Memory Express (NVMe), Flash, Hybrid Flash, Festplatte und Cloud • Unterbrechungsfreie Datenverschiebung zwischen den Tiers • Scale-out-Designs zur inkrementellen Performance-Erhöhung

4.1 Datenfluss in das Training-Cluster

Neben den in Tabelle 1 aufgeführten Überlegungen gibt es noch einige wichtige Besonderheiten bei dem Datenfluss in das Training-Cluster. Diese Faktoren wirken sich auf die folgenden Punkte aus:

- Ort der I/O-Zusammenführung
- Anforderungen an einen Single Namespace
- Skalierung von Metadaten

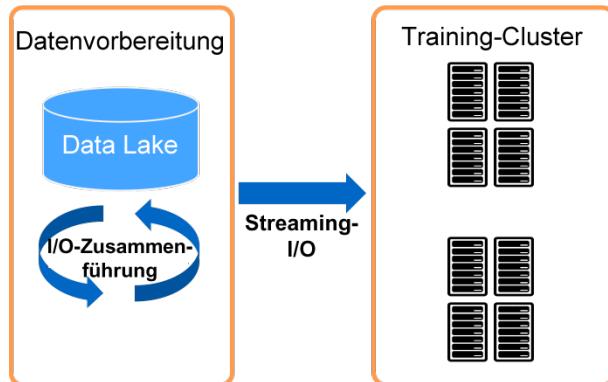
I/O-Zusammenführung

Die Datenkuratorierung ist Bestandteil der Datenquelle. Die I/O-Zusammenführung kann an zwei unterschiedlichen Orten erfolgen:

- im Data Lake als Teil der Datenkuratorierung und -transformation mit der Folge des I/O-Datenstroms in das Training-Cluster
- im Training-Cluster selbst mit der Folge zufälliger I/O-Vorgänge aus dem Data Lake

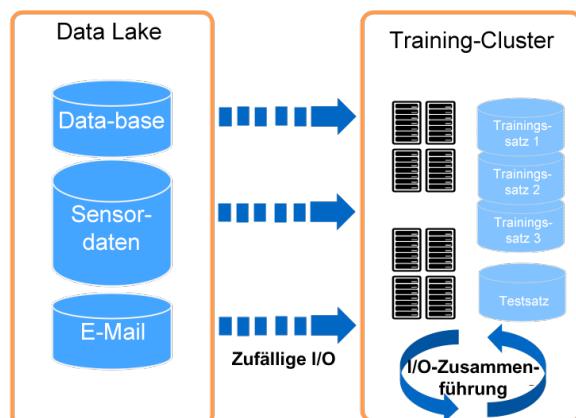
Bei einem unstrukturierten Data Lake handelt es sich beinahe schon per Definition um ein Filesystem. Er ist in der Lage, die Daten zu katalogisieren und als Gruppe zusammengehörter Dateiströme anzugeben. Diese Dateiströme lassen sich mit dem Training-Cluster harmonisieren, sodass die Daten direkt an die Cluster-CPUs fließen können, um die GPUs zu speisen. (Abb. 6)

Abbildung 6: Unstrukturierte Daten können im Data Lake zusammengeführt und in das Training-Cluster übertragen werden.



Dagegen können bei Datenquellen wie Datenbanken, Sensor- und Dateiprotokollen, E-Mails usw. katalogisierte Lesevorgänge zur Datenübertragung an das Cluster unmöglich sein. In diesen Fällen erfolgt der Datenzugriff mittels zufälliger Lesevorgänge und die I/O-Zusammenführung geschieht im Training-Cluster selbst. (Abb. 7)

Abbildung 7: Strukturierte Daten werden mittels kleiner zufälliger I/O-Vorgänge gelesen und im Training-Cluster zusammengeführt.



Abhängig von den Datenquellentypen muss die Architektur der Daten möglicherweise sowohl große sequenzielle Lesevorgänge als auch kleine zufällige Lesevorgänge im Training-Cluster unterstützen können.

Single Namespace

Alle Datensätze können potenziell sehr groß werden und die Folge ist eine enormes unkontrolliertes Datenwachstum. Um dieses Wachstum aufzufangen, ist ein Scale-out-Filesystem mit einem Single Namespace erforderlich, das die Performance linear für einzelne oder mehrere Client-Nodes skalieren kann, die parallel auf dieselben Daten zugreifen. Eine Architektur, die mit zunehmender Rechenleistung und Kapazität skaliert werden kann, ist von entscheidender Bedeutung.

Der Client-Zugriff auf diesen Single Namespace kann nach verschiedenen Methoden erfolgen, die jeweils bestimmte Auswirkungen auf die Performance haben. Bestimmte Trainingmodelle werden als „asynchron“ bezeichnet: Der Datensatz wird statisch in den Training-Cluster-Nodes partitioniert, wobei einzelne Nodes auf Regionen des Namespace zugreifen können und jeweils nur ein Client aktiv ist.

Andere Trainingmodelle werden synchron ausgeführt. Das Trainingmodell und der zugehörige Datensatz sind eng miteinander verbunden und der Datensatz wird von allen Cluster-Nodes bei gleichzeitigem Zugriff gemeinsam genutzt. Dabei sind mehrere Clients aktiv, sodass die Performance-Anforderungen sehr hoch sind.

Bei anderen Anwendungsfällen werden die Netzwerkschichten auf verschiedenen Nodes durch ein mehrschichtiges neuronales Netz trainiert. Die Nodes fungieren dabei als Modell-Pipeline und werden nacheinander durchlaufen. Der gesamte Datensatz wird Node für Node kontinuierlich wiederholt gelesen.

Bei der Bewertung der für diese Nutzungsmuster infrage kommenden Filesysteme werden Sie feststellen, dass NFS für sehr verschiedenartige Workloads eingesetzt wird. Diese Workloads reichen von den ursprünglichen Einsatzgebieten HPC und Home Directoys über Datenbanken wie Oracle und SQL auf NAS-Storage bis hin zu SAP sowie neuerdings auch Virtualisierung und Big Data. Da NFS bereits seit Langem für diese verschiedenen Workloads genutzt wird, kann es sowohl zufällige als auch sequenzielle I/O-Vorgänge verarbeiten. Diese I/O-Vorgänge können durch unterschiedliche Zugriffsmuster für den Namespace generiert werden, besonders in Verbindung mit den Vorteilen von All-Flash-Storage in einem linearen Scale-out-Cluster.

HDFS wurde als relativ neues Filesystem bisher wenig für verschiedenartige Daten-Workloads mit entsprechenden Performance-Merkmalen eingesetzt. Big-Data-Anbieter haben sehr umfangreiche (proprietäre) Anpassungen vorgenommen, um die Performance-Anforderungen bei der Transition von MapReduce zu Spark zu erfüllen. Mit AI kommt noch ein weiterer Aspekt zu HDFS dazu.

Durch die Nutzung eines Big-Data-spezifischen Filesystems wie HDFS können mehr Datenkopien und Silos entstehen, wenn immer wieder Daten von HDFS in ein High-Performance-Scale-out-Filesystem für AI kopiert werden müssen.

Metadaten-Performance

Die bereits beschriebenen Zugriffsmuster wirken sich auch auf die Metadaten-Performance aus. Da jeder Node im Training-Cluster die Metadaten unabhängig abfragen kann, muss sich die Performance bei der Metadatenabfrage linear mit dem Filesystem skalieren lassen. Der Metadatenzugriff mit Filesystemen wie Lustre und GPFS kann zu einem Engpass werden, weil die Metadaten von separaten Servern und Storage-Systemen abhängig sind.

4.2 Weitere Performance-Faktoren

Bei der Auswahl eines Filesystems für die AI-Daten-Pipeline sollten noch einige weitere Faktoren berücksichtigt werden, die Performance und Benutzerfreundlichkeit beeinflussen. Dazu zählen:

- Einfaches Management
- Quality of Service (QoS)
- Klonfunktionen
- Ökosystem der Client-seitigen Caching-Lösungen

- Möglichkeit für In-Place-AI/DL mit einem einheitlichen Filesystem für Data-Lake- und AI/DL-Tier
- Hervorragende Medienunterstützung
- Zukunftssicherheit

Einfaches Management

Bei der Bewertung von Filesystemen sollten Sie sich auch Fragen zum Management stellen. Kann das Filesystem autonom und automatisch ohne Management-Eingriffe skaliert werden? Wie viel Zeit und welche technischen Fachkenntnisse erfordert das Management des Filesystems? Wie leicht lässt sich Personal mit den erforderlichen Fachkenntnissen finden?

Scale-out-Filesysteme wie Lustre und GPFS können in Bezug auf Konfiguration, Wartung, Überwachung und Management eine Herausforderung darstellen. Dagegen kann NFS mit einfacherem Management und weit verbreiteten Fachkenntnissen punkten.

Quality of Service

QoS kann ein weiteres wichtiges Element für die Architektur der Daten sein. Wenn Sie mandantenfähige Training-Cluster aufbauen, können die Kosten schnell in die Millionen gehen. QoS ist eine wichtige Voraussetzung für Mandantenfähigkeit, da dieselben Ressourcen für diverse Aktivitäten genutzt werden können.

- Bietet das Filesystem QoS?
- Ist QoS durchgängig integriert?
- Lassen sich Grenz- und Höchstwerte für die Performance-Nutzung durch Storage, Netzwerke und Computing festlegen, um Service-Level für verschiedene Trainingmodelle zu partitionieren?

Klonfunktionen

Eine weitere Anforderung an die Mandantenfähigkeit ist die Erfüllung verschiedener Aufgaben im Unternehmen. Möglicherweise befinden sich mehrere Trainingmodelle in unterschiedlichen Entwicklungsphasen, wodurch sich verschiedene Anwendungsfälle ergeben:

- Frühes Training
- Modellvalidierung
- Implementierungsplanung
- Implementierung in der Produktion

Durch das Klonen von Datensätzen und das Zuweisen verschiedener QoS-Einstellungen für jeden Klon sind separate Performance-SLAs (Service Level Agreements) für unterschiedliche Anwendungsfälle möglich. Platzsparendes Klonen ist daher eine unverzichtbare Funktion für mandantenfähige Cluster.

Client-seitiges Caching

Durch die Nutzung eines Client-seitigen Cache lässt sich die Performance weiter steigern, indem dieser bei dem Zugriff auf den Trainingdatensatz durch die Training-Cluster-Nodes als Datenpuffer einen ununterbrochenen Datenfluss ermöglicht. Daher kann ein Filesystem, das verschiedene Client-Caching-Produkte unterstützt (Open Source oder kommerziell), deutliche Vorteile bieten.

Für NFS-basierten Storage sind zahlreiche Open-Source- wie auch kommerzielle Optionen erhältlich. Für Lustre, GPFS oder HDFS gibt es dagegen bisher kaum oder gar keine Produkte für das Client-seitige Caching. Darunter ist nahezu keine weithin verfügbare Open-Source-Lösung.

In-Place-AI/DL mit einheitlichem Filesystem

In bestimmten Fällen sollen dieselben Daten für Big-Data-Analytics-Workloads und für AI/ML/DL-Workloads genutzt werden. Bei AI mit Nachbearbeitung – wie bei Überwachung, Betrugsermittlung usw. – können mit dem richtigen Filesystem beide Workloads unterstützt werden, ohne dass Datenkopien erforderlich sind. Der Datensatz residiert an einem einzelnen Speicherort. Dabei werden sowohl In-Place-Analysen als auch In-Place-AI/ML/DL-Rechenverarbeitungen durchgeführt

(möglicherweise mit Client-seitigem Caching, wie weiter oben erläutert), ohne dass Daten in dedizierte Filesysteme für den Data Lake und das Training-Cluster kopiert werden müssen.

Wenn allerdings Echtzeit-Performance eine wichtige Anforderung ist oder einen Wettbewerbsvorteil bietet, ist mit hoher Wahrscheinlichkeit eine dedizierte Datenkopie für das Training-Cluster erforderlich.

Unterstützung aktueller Medien und Speichersysteme

Entscheiden Sie sich schließlich für ein Filesystem, das die neuesten Medien- und Speichersysteme unterstützt, sodass die Performance Ihrer Daten-Pipeline im Einklang mit der Technologie-Roadmap weiterentwickelt werden kann. Ist das Filesystem bereits für Flash optimiert? Kann es nahtlos erweitert werden, um neue Technologien zu unterstützen, und treiben die Anbieter aktiv Innovationen in Bereichen wie NVMe over Fabrics (NVMe-oF), NVDIMM und 3D XPoint voran?

Flash ermöglicht heute Latenzen im Bereich von 500 Mikrosekunden. Mit NVMe-oF reduziert sich dieser Wert noch einmal auf 200 Mikrosekunden. Mit NVDIMM, 3D XPoint und persistentem Speicher kann die Latenz potenziell weiter auf 100 Mikrosekunden, weniger als 10 Mikrosekunden und schließlich sogar Nanosekunden reduziert werden. Der Anbieter Ihrer Daten-Pipeline sollte also kontinuierlich auf diesem Gebiet investieren, um mit dieser Entwicklung bei serverbasierten und Shared-Storage-Lösungen Schritt zu halten.

Zukunftssicherheit für die Architektur der Daten und das Filesystem

Das gesamte Gebiet der AI entwickelt sich sehr schnell weiter; ein Neuaufbau alle sechs bis zwölf Monate dürfte jedoch wenig praktikabel bis unmöglich sein. Als letzte Überlegung sollten Sie daher eine möglichst zukunftssichere Technologie wählen. Wichtig ist dabei, dass verschiedene Schichten der Technologie wie Filesystem, Interconnect, Speicherort, Medien und Speichertyp in der gewählten Infrastruktur sich nahtlos und unterbrechungsfrei weiterentwickeln lassen. Diese Fähigkeit bietet neben einem langfristigen Return on Investment auch die Möglichkeit einer sofortigen Nutzung technologischer Neuerungen.

Ihre heutige Entscheidung für ein Filesystem dürfte nicht zuletzt von der Vertrautheit, den Kompetenzen und den Fachkenntnissen Ihres Teams abhängig sein. Dabei werden Sie auch Ihre Erfahrungen mit früheren und aktuellen Implementierungen sowie mit der vorhandenen Infrastruktur berücksichtigen.

Wenn Sie beispielsweise gute Erfahrungen mit FC oder InfiniBand gemacht haben, entscheiden Sie sich vielleicht für eine SAN-Architektur mit Lustre oder GPFS. Mit der Zeit stellen Sie jedoch möglicherweise fest, dass die 100-GbE- oder 400-GbE-Roadmap mit NFS Ihren Anforderungen eher entspricht. Eine sorgfältig geplante Architektur der Daten kann die Lösung zukunftssicher machen und einen nahtlosen Wechsel des Filesystems ohne Infrastrukturaustausch ermöglichen.

Ebenso entscheiden Sie sich vielleicht heute für NFS, stellen aber später fest, dass Sie eher ein SAN-, NVMe- oder NVMe-oF-basiertes Filesystem oder ein auf persistentem Speicher basierendes Daten-Layout benötigen. Bei einer zukunftssicheren Architektur können Sie die Datastore-Technologien weiterentwickeln, ohne die gesamte implementierte Infrastruktur ersetzen zu müssen.

Die in diesem Kapitel beschriebenen Kriterien sollten eine gute Grundlage bei Ihrer Entscheidung für ein Filesystem und eine Architektur der Daten liefern, die für Ihre AI/ML/DL-Anforderungen am besten geeignet sind. Wir glauben, dass die Kombination aus NFS und NetApp All Flash FAS Storage diese Anforderungen am besten erfüllt und zudem nahtlos mit den neuesten Technologien weiterentwickelt werden kann.

5 NetApp Technologien und die Deep-Learning-Pipeline

Die Datenmanagementtechnologien von NetApp Data Fabric können die Anforderungen der gesamten Deep-Learning-Pipeline erfüllen. (Abb. 8) Cloud-Provider decken den Edge selbst nicht ab und können nicht immer die erforderliche I/O-Performance liefern. Andere Storage-Anbieter versuchen zwar, die Bandbreitenprobleme im Training zu beheben, können aber weder eine äußerst niedrige Latenz noch die erforderliche Technologie für den gesamten Workflow bereitstellen. An dieser Stelle bietet NetApp Data Fabric klare Vorteile.

Am Edge kann NetApp ONTAP Select auf Standard-Hardware ausgeführt werden, um Datenaggregation und erweitertes Datenmanagement zu ermöglichen. Unsere neue Plexistor Technologie erleichtert die Aufnahme, und zwar besonders bei extrem hohen Aufnahmeraten.

NetApp All Flash FAS Storage bietet sowohl hohe Performance als auch hohe Kapazität, während weniger zeitintensive Datenkopien erforderlich sind, und erfüllt damit die Storage-Anforderungen von Data Lake und Training-Cluster. NetApp arbeitet derzeit an der Implementierung von NVMe-oF und Plexistor zur Erweiterung der Funktionen von All Flash FAS. NPS bietet viele dieser Vorteile für Deep-Learning-Pipelines in der Cloud.

Zur Archivierung kalter Daten ermöglicht FabricPool zudem die automatische Datenmigration in Objekt-Storage anhand definierter Richtlinien.

Abbildung 8: NetApp Data-Fabric-Technologien

	Edge	Zentrale	Cloud
Heute	ONTAP Select Datenaggregation und -management auf Standard-Hardware	All Flash FAS All-Flash-Storage für hohe Performance	NetApp Private Storage High-Performance Storage in Cloud-Nähe
Zukunft	Plexistor serverseitiger Storage mit äußerst niedriger Latenz	• NVMe-oF • Plexistor	FabricPool Automatisiertes Tiering kalter Daten in die Cloud

6 Zukunftssichere Deep-Learning-Pipeline

Die Deep-Learning-Datensätze und I/O-Anforderungen Ihrer Deep-Learning-Pipeline werden mit sehr hoher Wahrscheinlichkeit weiter ansteigen, während Sie zusätzliche Server, CPUs, GPUs und speziell entwickelte AI-Chipsätze mit immer höherer Leistung einsetzen. Die NetApp Roadmap umfasst zahlreiche Elemente, die eine entsprechende I/O-Skalierung ermöglichen. Dazu zählen:

- **NVMe-oF:** Durch die Implementierung von NVMe-oF in der All Flash FAS Architektur kann NetApp die Latenz um eine ganze Größenordnung reduzieren.
- **Plexistor:** NetApp hat mit Plexistor im Juni 2017 eine Technologie für serverseitigen Storage erworben, mit der NetApp Data Fabric auf den Server erweitert und die Latenz noch weiter reduziert werden kann. Plexistor kann im gesamten Datacenter (Edge, Core) und in der Cloud implementiert werden und beschleunigt Datenaufnahme, Edge-Analysen und Training.

Wenn sich außerdem der Core Ihrer AI-Pipeline bereits vor Ort befindet, sollten Sie eine Strategie zur Weiterentwicklung der Core-Hardware erarbeiten. Darüber hinaus sollten Sie berücksichtigen, dass Sie in Zukunft möglicherweise von einer „Edge-Core-Cloud“-Strategie zu einer „Edge-Cloud“-Strategie wechseln müssen.

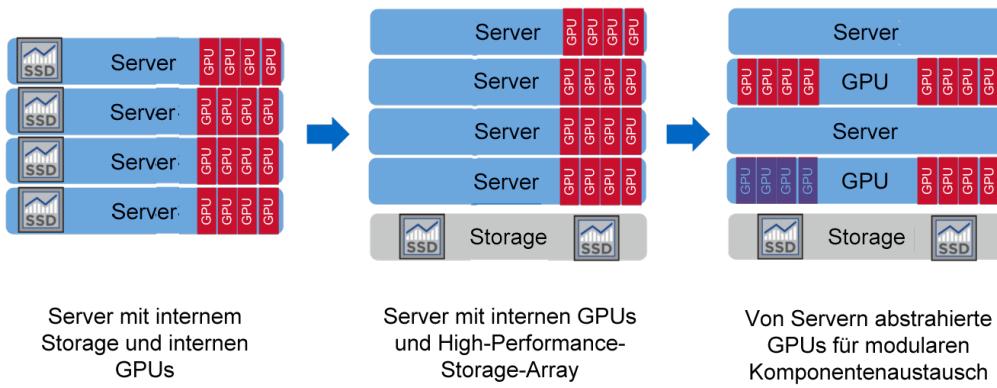
6.1 Planung für die Weiterentwicklung der Core-Hardware

Ein Vorteil der Cloud besteht in der Möglichkeit zur Nutzung eines Deep-Learning-Service, ohne die Besonderheiten des Hardware-Stacks kennen zu müssen. Dieser Komfort muss jedoch mit einem Kontrollverlust erkauft werden. Ein starker Trend bei dem Core der Deep-Learning-Pipeline vor Ort ist die kontinuierliche Weiterentwicklung der erforderlichen Hardware.

Die Position des bevorzugten Hardware-Anbieters für Deep Learning ist hart umkämpft. Derzeit hält NVIDIA einen klaren Vorsprung, es gibt aber auch viele neue Technologien. Jeder Cloud-Anbieter entwickelt kundenspezifische Hardware. Ein Beispiel dafür ist die Tensor Processing Unit (TPU) von Google. Auch viele Start-ups entwickeln kundenspezifische AI-Hardware.

Ein wichtiger Trend ist dabei die Möglichkeit einer Trennung der Serverinfrastruktur von der GPU-Infrastruktur, sodass beide unabhängig voneinander weiterentwickelt werden können. Eine Lösung, bei der die GPU-Hardware von der Server- und Storage-Hardware abstrahiert ist (siehe Abbildung 9), lässt sich leichter und kostengünstiger weiterentwickeln, um von neuen Entwicklungen zu profitieren.

Abbildung 9: Der Core der AI/ML/DL-Pipeline entwickelt sich kontinuierlich weiter.



NetApp bietet konvergente Infrastrukturen zusammen mit verschiedenen Serverpartnern an, darunter Cisco ([FlexPod](#)), Fujitsu ([NFLEX](#), eine konvergente Infrastruktur von NetApp und Fujitsu) und weitere Anbieter. Aus diesem Grund können Sie bei dem Aufbau Ihres Deep-Learning-Clusters mühelos alle Vorteile von NetApp Storage in Verbindung mit verschiedenen Serverplattformen nutzen.

Fazit: Behalten Sie Ihre Daten-Pipeline und Ihre AI-Zukunft unter Kontrolle

Die in diesem Whitepaper erläuterten Richtlinien sollen Sie bei der Planung Ihrer Daten-Pipeline unterstützen:

- Wählen Sie ein Filesystem und eine Architektur der Daten, die Ihre aktuellen Anforderungen am besten erfüllen. Berücksichtigen Sie dabei aber auch zukünftige Entwicklungen.
- Beschleunigen Sie den Datenfluss durch Ihre Pipeline unabhängig davon, ob diese sich vor Ort oder in der Cloud befindet.
- Implementieren Sie intelligentes Datenmanagement am Edge, um besser für Datenwachstum gerüstet zu sein.
- Verschieben Sie Daten intelligenter und effizienter vom Edge zum Core oder in die Cloud.
- Seien Sie auf eine eventuell erforderliche Umstellung auf ein „Edge-Cloud“-Modell vorbereitet.
- Bauen Sie eine agilere Core-Hardware-Architektur auf, die sich schnell weiterentwickeln lässt.

Wenn Sie diese Ratschläge befolgen, können Sie Engpässe beseitigen und einen höheren Durchsatz erzielen, während Sie gleichzeitig Ihre Investitionen in die AI-Infrastruktur zukunftssicher machen.

Überprüfen Sie mithilfe des [Interoperability Matrix Tools \(IMT\)](#) auf der NetApp Support-Website, ob die in diesem Dokument angegebenen Produktversionen und Funktionen in Ihrer IT-Umgebung unterstützt werden. NetApp IMT definiert die Produktkomponenten und -versionen, die für von NetApp unterstützte Konfigurationen verwendet werden können. Die jeweiligen Ergebnisse sind von der kundenspezifischen Installation bzw. den technischen Daten abhängig.

Copyright-Informationen

Copyright © 2018, NetApp. Alle Rechte vorbehalten. Gedruckt in den USA. Dieses urheberrechtlich geschützte Dokument darf ohne die vorherige schriftliche Genehmigung des Urheberrechtsinhabers in keiner Form und durch keine Mittel – weder grafische noch elektronische oder mechanische, einschließlich Fotokopieren, Aufnehmen oder Speichern in einem elektronischen Abrufsystem – auch nicht in Teilen vervielfältigt werden.

Software, die von urheberrechtlich geschütztem NetApp Material abgeleitet wird, unterliegt der folgenden Lizenz und dem folgenden Haftungsausschluss:

DIE VORLIEGENDE SOFTWARE WIRD IN DER VORLIEGENDEN FORM VON NETAPP ZUR VERFÜGUNG GESTELLT, D. H. OHNE JEGLICHE EXPLIZITE ODER IMPLIZITE GEWÄHRLEISTUNG, EINSCHLIESSLICH, JEDOCH NICHT BESCHRÄNKT AUF DER STILLSCHWEIGENDEN GEWÄHRLEISTUNG DER MARKTGÄNGIGKEIT UND EIGNUNG FÜR EINEN BESTIMMTEN ZWECK, DIE HIERMIT AUSGESCHLOSSEN WERDEN. NETAPP ÜBERNIMMT KEINERLEI HAFTUNG FÜR DIREKTE, INDIREKTE, ZUFÄLLIGE, BESONDERE ODER FOLGESCHÄDEN (EINSCHLIESSLICH, JEDOCH NICHT BESCHRÄNKT AUF DIE BESCHAFFUNG ODER DEN ERSATZ VON WAREN ODER DIENSTLEISTUNGEN, NUTZUNGS-, DATEN- ODER GEWINNVERLUST ODER UNTERBRECHUNG DES GESCHÄFTSBETRIEBS), DIE SICH UNABHÄNGIG VON DER URSACHE UND BELIEBIGER THEORETISCHER HAFTBARKEIT, OB VERTRÄGLICH FESTGELEGT, PER KAUSALHAFTUNG ODER DELIKTSHAFTUNG (EINSCHLIESSLICH FAHRLÄSSIGKEIT ODER AUF ANDEREM WEGE), ERGEBEN, DIE IN IRGENDERART UND WEISE AUS DER NUTZUNG DIESER SOFTWARE RESULTIEREN, SELBST WENN AUF DIE MÖGLICHKEIT DERARTIGER SCHÄDEN HINGEWIESEN WURDE.

NetApp behält sich das Recht vor, die hierin beschriebenen Produkte jederzeit und ohne Vorankündigung zu ändern. NetApp übernimmt keine Verantwortung oder Haftung für die Verwendung der hier beschriebenen Produkte, sofern nicht ausdrücklich in schriftlicher Form von NetApp angegeben. Die Verwendung oder der Erwerb dieses Produkts stellt keine Lizenzierung im Rahmen eines Patentrechts, Markenrechts oder eines anderen Rechts an geistigem Eigentum von NetApp dar.

Das in diesem Handbuch beschriebene Produkt kann durch ein oder mehrere US-amerikanische Patente, ausländische Patente oder Patentanmeldungen geschützt sein.

LEGENDE ZU „RESTRICTED RIGHTS“: Nutzung, Vervielfältigung oder Offenlegung durch die US-Regierung unterliegt den Einschränkungen gemäß Unterpunkt (c)(1)(ii) Klausel „Rights in Technical Data and Computer Software“ DFARS 252.277-7103 (Oktober 1988) und FAR 52-227-19 (Juni 1987).

Markeninformationen

NETAPP, das NETAPP Logo und die unter <http://www.netapp.com/TM> genannten Produktbezeichnungen sind Marken oder eingetragene Marken von NetApp Inc. in den USA und/oder in anderen Ländern. Alle anderen Marken- und Produktbezeichnungen sind möglicherweise Marken oder eingetragene Marken der jeweiligen Rechtsinhaber und werden hiermit anerkannt.