Big Data-Analyseoptionen in AWS

Januar 2016

Hinweise

Dieses Dokument wird nur zu Informationszwecken zur Verfügung gestellt. Es stellt das aktuelle Produktangebot und die Verfahren von AWS zum Ausstellungsdatum dieses Dokuments dar. Änderungen vorbehalten. Kunden sind für ihre eigene unabhängige Einschätzung der Informationen in diesem Dokument und jedwede Nutzung der AWS-Services verantwortlich. Jeder Service wird „wie besehen“ ohne Gewähr und ohne Garantie jeglicher Art, weder ausdrücklich noch impliziert, bereitgestellt. Dieses Dokument gibt keine Garantien, Gewährleistungen, vertraglichen Verpflichtungen, Bedingungen oder Zusicherungen von AWS, seinen Partnern, Zulieferern oder Lizenzgebern. Die Verantwortung und Haftung von AWS gegenüber seinen Kunden werden durch AWS-Vereinbarungen geregelt. Dieses Dokument ist weder ganz noch teilweise Teil der Vereinbarungen von AWS mit seinen Kunden und ändert diese Vereinbarungen auch nicht.

Inhalt

Kurzbeschreibung 4
Einleitung 4
Der AWS-Vorteil in Big Data-Analysen 5
Amazon Kinesis Streams 6
AWS Lambda 10
Amazon EMR 13
Amazon Machine Learning 19
Amazon DynamoDB 23
Amazon Redshift 27
Amazon Elasticsearch Service 31
Amazon QuickSight 35
Amazon EC2 36
Big Data-Probleme in AWS lösen 38
Beispiel 1: Enterprise-Data-Warehouse 40
Beispiel 2: Erfassen und Analysieren von Sensordaten 43
Beispiel 3: Meinungsanalyse von sozialen Medien 46
Fazit 49
Mitwirkende 50
Weitere Informationen 50
Am Dokument vorgenommene Änderungen 51
Anmerkungen 51

Kurzbeschreibung

Dieses Whitepaper hilft Architekten, Datenwissenschaftlern und Entwicklern, die in der AWS Cloud verfügbaren Big Data-Analyseoptionen zu verstehen, indem es einen Überblick über die Dienstleistungen mit den folgenden Informationen bietet:

- Ideale Nutzungsmuster
- Kostenmodell
- Leistung
- Haltbarkeit und Verfügbarkeit
- Skalierbarkeit und Elastizität
- Schnittstellen
- Anti-Patterns

Dieses Whitepaper bilden Szenarien, die die verwendeten Analyseoptionen sowie zusätzliche Ressourcen für den Einstieg in die Big Data-Analyse in AWS darstellen.

Einleitung

Während wir zu einer digitalen Gesellschaft werden, wächst und beschleunigt sich die Menge an Daten, die erstellt und gesammelt werden. Die Analyse dieser ständig wachsenden Daten wird mit herkömmlichen Analysetools zu einer Herausforderung. Wir brauchen Innovationen, um die Lücke zwischen den erzeugten Daten und den Daten, die effektiv analysiert werden können, zu überbrücken.

Big Data-Tools und -Technologien bieten Möglichkeiten, Daten effizient zu analysieren, um Kundenpräferenzen besser zu verstehen, Wettbewerbsvorteile zu erzielen und höhere Umsätze zu generieren. Damit sind aber auch Herausforderungen verbunden. Datenverwaltungsarchitekturen haben sich vom traditionellen Data-Warehousing-Modell zu komplexeren Architekturen entwickelt, die mehr Anforderungen wie Echtzeit- und Stapelverarbeitung erfüllen; strukturierte und unstrukturierte Daten; Hochgeschwindigkeits-Transaktionen; und so weiter.

Amazon Web Services (AWS) bietet eine breite Palette von Managed Services, mit denen Sie End-to-End-Big Data-Anwendungen schnell und einfach

erstellen, sichern und nahtlos skalieren können. Unabhängig davon, ob Ihre Anwendungen Echtzeit-Streaming oder Stapeldatenverarbeitung erfordern, bietet AWS die Infrastruktur und Tools, die Sie für Ihr nächstes Big Data-Projekt benötigen. Keine Hardware zu beschaffen, keine Infrastruktur zu warten und zu skalieren – nur das, was Sie zum Sammeln, Speichern, Verarbeiten und Analysieren von Big Data benötigen. AWS verfügt über ein Ökosystem von analytischen Lösungen, die speziell auf diese wachsende Datenmenge ausgelegt sind und Einblicke in Ihr Unternehmen bieten.

Der AWS-Vorteil in Big Data-Analysen

Das Analysieren großer Datensätze erfordert eine erhebliche Datenverarbeitungskapazität, deren Größe abhängig von der Menge der Eingabedaten und der Art der Analyse variieren kann. Diese Eigenschaft von Big Data-Workloads ist ideal für die nutzungsabhängige Abrechnung von Cloud-Computing-Services geeignet, bei der Anwendungen je nach Bedarf einfach skaliert und verkleinert werden können. Wenn sich die Anforderungen ändern, können Sie die Größe Ihrer Umgebung (horizontal oder vertikal) problemlos an AWS anpassen, um Ihre Anforderungen zu erfüllen, ohne auf zusätzliche Hardware warten zu müssen oder zu viel investieren zu müssen, um genügend Kapazität bereitzustellen.

Bei unternehmenskritischen Anwendungen auf einer traditionelleren Infrastruktur haben Systementwickler keine andere Wahl, als in Übergröße zu planen, da das System in der Lage sein muss, mit einem Anstieg zusätzlicher Daten aufgrund eines erhöhten Geschäftsbedarfs umzugehen. Im Gegensatz dazu können Sie mit AWS in wenigen Minuten mehr Kapazität und Datenverarbeitung bereitstellen, was bedeutet, dass Ihre Big Data-Anwendungen je nach Bedarf wachsen und schrumpfen und Ihr System so effizient wie möglich läuft.

Darüber hinaus erhalten Sie flexible Datenverarbeitung in einer globalen Infrastruktur mit Zugriff auf die vielen verschiedenen geographischen Regionen[1], die AWS bietet, sowie die Möglichkeit, andere skalierbare Dienstleistungen zu verwenden, die zur Erstellung anspruchsvoller Big Data-Anwendungen beitragen. Zu den weiteren Dienstleistungen gehören Amazon Simple Storage Service (Amazon S3)[2] zum Speichern von Daten und AWS Data Pipeline[3] zum Koordinieren von Jobs, um diese Daten einfach zu verschieben und zu transformieren. AWS IoT[4] lässt verbundene Geräte mit Cloud-Anwendungen und anderen verbundenen Geräten interagieren.

Darüber hinaus bietet AWS viele Möglichkeiten, um Daten in die Cloud zu übertragen, darunter sichere Geräte wie AWS Import/Export Snowball[5] zur Beschleunigung von Datenübertragungen im Petabyte-Bereich, Amazon Kinesis Firehose[6] zum Laden von Streaming-Daten und skalierbare private Verbindungen über AWS Direct Connect.[7] Da die Nutzung von Mobilgeräten weiterhin stark zunimmt, können Sie mithilfe der Services in AWS Mobile Hub[8] die App-Nutzung und Daten sammeln und messen oder diese Daten zur weiteren benutzerdefinierten Analyse an einen anderen Service exportieren.

Diese Funktionen der AWS-Plattform eignen sich ideal zur Lösung von Big Data-Problemen und viele Kunden haben in AWS erfolgreich Big Data-Analyse-Workloads implementiert. Weitere Informationen zu Fallstudien finden Sie unter Big Data & HPC. Unterstützt von der AWS Cloud.[9]

Die folgenden Services werden beschrieben, um Big Data zu erfassen, zu verarbeiten, zu speichern und zu analysieren:

- Amazon Kinesis Streams
- AWS Lambda
- Amazon Elastic MapReduce
- Amazon Machine Learning
- Amazon DynamoDB
- Amazon Redshift
- Amazon Elasticsearch Service
- Amazon QuickSight

Darüber hinaus stehen Amazon EC2-Instances für selbstverwaltete Big Data-Anwendungen zur Verfügung.

Amazon Kinesis Streams

Mit Amazon Kinesis Streams[10] können Sie benutzerdefinierte Anwendungen erstellen, die Streaming-Daten in Echtzeit verarbeiten oder analysieren. Amazon Kinesis Streams kann kontinuierlich Terabytes an Daten pro Stunde aus Hunderttausenden von Quellen erfassen und speichern, wie z. B. Clickstreams von Websites, Finanztransaktionen, Social Media-Feeds, IT-Protokollen und Ereignissen zur Standortverfolgung.

Mit der Amazon Kinesis Client Library (KCL) können Sie Amazon Kinesis-Anwendungen erstellen und Streaming-Daten nutzen, um Echtzeit-Dashboards zu steuern, Warnungen zu generieren und dynamische Preisgestaltung und Werbung zu implementieren. Sie können Daten von Amazon Kinesis Streams auch an andere AWS-Services wie Amazon Simple Storage Service (Amazon S3), Amazon Redshift, Amazon Elastic MapReduce (Amazon EMR) und AWS Lambda senden.

Stellen Sie den erforderlichen Ein- und Ausgabelevel für Ihren Datenstrom in Blöcken von 1 Megabyte pro Sekunde (MB/s) bereit, mithilfe der AWS Management Console, API,[11] oder SDKs.[12] Die Größe Ihres Streams kann nach oben oder unten angepasst werden jederzeit ohne Neustart des Streams und ohne Auswirkungen auf die Datenquellen die Daten in den Stream schieben. Innerhalb von Sekunden stehen Daten in einem Stream für die Analyse zur Verfügung.

Stream-Daten werden über mehrere Availability Zones in einer Region 24 Stunden lang gespeichert. In diesem Fenster stehen Daten zum Lesen, erneuten Lesen, Zurückfüllen und Analysieren zur Verfügung oder können in Langzeitspeicher verschoben werden (z. B. Amazon S3 oder Amazon Redshift). Die KCL ermöglicht es Entwicklern, sich auf die Erstellung ihrer Geschäftsanwendungen zu konzentrieren und gleichzeitig die undifferenzierte Schwerstarbeit in Verbindung mit Lastenausgleich von Streaming-Daten, Koordination verteilter Services und fehlertoleranter Datenverarbeitung zu beseitigen.

Ideale Nutzungsmuster

Amazon Kinesis Streams ist überall dort nützlich, wo Daten schnell von den Herstellern (Datenquellen) verschoben und kontinuierlich verarbeitet werden müssen. Diese Verarbeitung kann darin bestehen, die Daten vor dem Emittieren in einen anderen Datenspeicher zu transformieren, Echtzeitmetriken und -analysen zu steuern oder mehrere Daten-Streams zu komplexeren Streams oder Downstream-Verarbeitung abzuleiten und zu sammeln. Im Folgenden finden Sie typische Szenarien für die Verwendung von Amazon Kinesis Streams für Analysen.

- **Echtzeit-Datenanalyse** – Amazon Kinesis Streams ermöglicht Echtzeit-Datenanalysen zu Streaming-Daten, z. B. die Analyse von Clickstream-Daten von Websites und Analysen zur Kundenbindung.

- **Ein- und Verarbeitung von Log- und Daten-Feeds** – Mit Amazon Kinesis Streams können Produzenten Daten direkt in einen Amazon Kinesis-Stream pushen. Sie können beispielsweise System- und Anwendungsprotokolle an Amazon Kinesis Streams senden und innerhalb von Sekunden auf den Stream zur Verarbeitung zugreifen. Dies verhindert, dass die Protokolldaten verloren gehen, wenn der Front-End- oder Anwendungsserver ausfällt, und reduziert den lokalen Protokollspeicher auf der Quelle. Amazon Kinesis Streams bietet beschleunigte Datenaufnahme, da Sie die Daten auf den Servern nicht zusammenstellen, bevor Sie sie zur Aufnahme einsenden.
- **Echtzeitmetriken und -berichte** – Sie können die in Amazon Kinesis Streams aufgenommenen Daten verwenden, um Messwerte zu extrahieren und KPIs zu generieren, mit denen Berichte und Dashboards in Echtzeit beschleunigt werden können. Dadurch kann die Datenverarbeitungs-Anwendungslogik Daten verarbeiten, während diese kontinuierlich im Stream eingehen, anstatt auf das Eintreffen der Datenstapel zu warten.

Kostenmodell

Amazon Kinesis Streams bietet einfache bedarfsorientierte Preise ohne Vorlaufkosten oder Mindestgebühren, und Sie zahlen nur für die Ressourcen, die Sie verbrauchen. Ein Amazon Kinesis-Stream besteht aus einem oder mehreren Shards und jeder Shard bietet Ihnen eine Kapazität von 5 Lesetransaktionen pro Sekunde, bis zu einer maximalen Gesamtmenge von 2 MB an Daten, die pro Sekunde gelesen werden. Jeder Shard kann bis zu 1000 Schreibtransaktionen pro Sekunde und bis zu maximal 1 MB pro Sekunde unterstützen.

Die Datenkapazität Ihres Streams hängt von der Anzahl der Shards ab, die Sie für den Stream angeben. Die Gesamtkapazität des Streams ist die Summe der Kapazität jedes Shards. Es gibt nur zwei Preiskomponenten, eine Stundengebühr pro Shard und eine Gebühr für jede 1 Million PUT-Transaktionen. Weitere Informationen finden Sie unter Amazon Kinesis Streams-Preise.[13] Für Anwendungen, die auf Amazon EC2 ausgeführt werden und Amazon Kinesis Streams verarbeiten, fallen ebenfalls die Amazon EC2-Standardkosten an.

Leistung

Mit Amazon Kinesis Streams können Sie die Durchsatzkapazität auswählen, die Sie für Shards benötigen. Mit jedem Shard in einem Amazon Kinesis-Stream können

Sie bis zu 1 Megabyte Daten pro Sekunde bei 1.000 Schreibtransaktionen pro Sekunde erfassen. Ihre Amazon Kinesis-Anwendungen können Daten von jedem Shard mit bis zu 2 Megabyte pro Sekunde lesen. Sie können so viele Shards bereitstellen, wie Sie benötigen, um die gewünschte Durchsatzkapazität zu erzielen. Zum Beispiel würde ein 1 Gigabyte pro Sekunde Datenstrom 1024 Shards erfordern.

Haltbarkeit und Verfügbarkeit

Amazon Kinesis Streams repliziert Daten synchron über drei Availability Zones in einer AWS-Region und bietet so hohe Verfügbarkeit und Datenhaltbarkeit. Darüber hinaus können Sie einen Cursor in DynamoDB speichern, um dauerhaft zu verfolgen, was aus einem Amazon Kinesis-Stream gelesen wurde. Für den Fall, dass Ihre Anwendung während des Lesens von Daten aus dem Datenstrom fehlschlägt, können Sie die Anwendung neu starten und den Cursor verwenden, um an der Stelle aufzubrechen, an der die fehlgeschlagene Anwendung aufgehört hat.

Skalierbarkeit und Elastizität

Sie können die Kapazität des Streams jederzeit entsprechend Ihren geschäftlichen oder betrieblichen Anforderungen erhöhen oder verringern, ohne dass die laufende Stream-Verarbeitung unterbrochen wird. Mithilfe von API-Aufrufen oder Entwicklungstools können Sie die Skalierung Ihrer Amazon Kinesis Streams-Umgebung automatisieren, um die Nachfrage zu befriedigen und sicherzustellen, dass Sie nur für das bezahlen, was Sie benötigen.

Schnittstellen

Es gibt zwei Schnittstellen zu Amazon Kinesis Streams: Eingabe, die von Datenproduzenten verwendet wird, um Daten in Amazon Kinesis Streams zu speichern; und Ausgabe zur Verarbeitung und Analyse der eingehenden Daten. Hersteller können Daten mit der Amazon Kinesis PUT-API schreiben, ein AWS Software Development Kit (SDK) oder Toolkit[14] Abstraktion, der Amazon Kinesis Producer Library (KPL),[15] oder dem Amazon Kinesis Agent.[16]

Zur Verarbeitung von Daten, die bereits in einen Amazon Kinesis-Stream eingegeben wurden, sind Client-Bibliotheken zum Erstellen und Betreiben von Echtzeit-Streaming-Daten verfügbar. Die KCL[17] fungiert als Vermittler zwischen Amazon Kinesis Streams und Ihren Geschäftsanwendungen, die die spezifische Verarbeitungslogik enthalten. Es gibt auch eine Integration, um aus einem Amazon Kinesis-Stream in Apache Storm hinein zu lesen über den Amazon Kinesis Storm Spout.[18]

Anti-Patterns

Amazon Kinesis Streams hat folgende Anti-Patterns:

- **Geringer konsistenter Durchsatz** – Obwohl Amazon Kinesis Streams für das Streamen von Daten mit 200 KB/s oder weniger geeignet ist, wurde es für größere Datendurchsätze entworfen und optimiert.
- **Langzeitdatenspeicherung und -analyse** – Amazon Kinesis Streams ist nicht für die Langzeitspeicherung von Daten geeignet. Standardmäßig werden die Daten für 24 Stunden gespeichert und Sie können die Aufbewahrungsfrist um bis zu 7 Tage verlängern. Sie können alle Daten, die länger als 7 Tage gespeichert werden müssen, in einen anderen dauerhaften Speicherdienst wie Amazon S3, Amazon Glacier, Amazon Redshift oder DynamoDB verschieben.

AWS Lambda

Mit AWS Lambda[19] können Sie Code ohne Bereitstellung oder Verwaltung von Servern ausführen. Sie zahlen nur für die genutzte Rechenzeit. Wenn Ihr Code nicht ausgeführt wird, wird auch nichts berechnet. Mit Lambda können Sie Code für praktisch jeden Anwendungstyp oder Backend-Service ohne Verwaltungsaufwand ausführen. Laden Sie einfach Ihren Code hoch und Lambda kümmert sich um alles, was für die Ausführung und Skalierung Ihres Codes mit hoher Verfügbarkeit erforderlich ist. Sie können Ihren Code so einrichten, dass er automatisch von anderen AWS-Services ausgelöst wird, oder Sie können ihn direkt von einer beliebigen Web- oder mobilen Anwendung aus aufrufen.

Ideale Nutzungsmuster

Mit Lambda können Sie Code als Reaktion auf Auslöser wie Datenänderungen, Änderungen des Systemstatus oder Aktionen von Benutzern ausführen. Lambda kann direkt von AWS-Services wie Amazon S3, DynamoDB, Amazon Kinesis Streams, Amazon Simple Notification Service (Amazon SNS) und Amazon CloudWatch ausgelöst werden, sodass Sie eine Vielzahl von Echtzeit-Datenverarbeitungssystemen erstellen können.

- **Echtzeit-Dateiverarbeitung** – Sie können Lambda veranlassen, einen Prozess aufzurufen, bei dem eine Datei in Amazon S3 hochgeladen oder geändert wurde. Zum Beispiel, um ein Bild von Farbe zu Graustufen zu ändern, nachdem es auf Amazon S3 hochgeladen wurde.

- **Echtzeit-Stream-Verarbeitung** – Mit Amazon Kinesis Streams und Lambda können Sie Streaming-Daten für Clickstream-Analyse, Protokollfilterung und Social Media-Analyse verarbeiten.
- **Extrahieren, Transformieren und Laden** – Mit Lambda können Sie Jobs ausführen, die Daten transformieren und von einer Daten-Repository in eine andere laden.
- **Cron ersetzen** – Verwenden Sie Zeitplanausdrücke, um eine Lambda-Funktion in regelmäßigen Intervallen als eine billigere und besser verfügbare Lösung auszuführen, als wenn Cron auf einer EC2-Instance ausgeführt wird.
- **AWS-Ereignisse verarbeiten** – Viele andere Services, z. B. AWS CloudTrail, können als Ereignisquellen fungieren, indem sie sich einfach bei Amazon S3 anmelden und S3-Bucket-Benachrichtigungen zum Auslösen von Lambda-Funktionen verwenden.

Kostenmodell

Mit Lambda zahlen Sie nur für das, was Sie verwenden. Sie werden anhand der Anzahl der Anfragen für Ihre Funktionen und der Zeit, zu der Ihr Code ausgeführt wird, berechnet. Die kostenlose Lambda-Stufe umfasst 1 Mio. kostenlose Anfragen pro Monat und 400.000 GB-Sekunden Rechenzeit pro Monat. Sie zahlen danach 0,20 USD pro 1 Million Anfragen (0,0000002 USD pro Anfrage). Zusätzlich wird die Dauer der Ausführung Ihres Codes in Relation zum zugewiesenen Speicher gesetzt. Für jede verwendete GB-Sekunde werden 0,00001667 USD berechnet. Weitere Informationen finden Sie unter AWS Lambda Preise.

Leistung

Nachdem Sie Ihren Code zum ersten Mal in Lambda implementiert haben, sind Ihre Funktionen in der Regel innerhalb von Sekunden nach dem Hochladen bereit. Lambda wurde entwickelt, um Ereignisse innerhalb von Millisekunden zu verarbeiten. Die Latenz wird höher sein, unmittelbar nachdem eine Lambda-Funktion erstellt, aktualisiert oder wenn sie zuletzt nicht verwendet wurde.

Haltbarkeit und Verfügbarkeit

Lambda wurde für die Verwendung von Replikation und Redundanz entwickelt, um sowohl für den Service selbst als auch für die von ihm betriebenen Lambda-Funktionen eine hohe Verfügbarkeit bereitzustellen. Für beide gibt es weder

Wartungsfenster noch geplante Ausfallzeiten. Bei einem Fehler werden Lambda-Funktionen, die synchron aufgerufen werden, mit einer Ausnahme reagieren. Lambda-Funktionen, die asynchron aufgerufen werden, werden mindestens dreimal wiederholt, danach kann das Ereignis zurückgewiesen werden.

Skalierbarkeit und Elastizität

Es gibt keine Begrenzung für die Anzahl der Lambda-Funktionen, die Sie ausführen können. Lambda hat jedoch eine voreingestellte Sicherheitsdrossel von 100 gleichzeitigen Ausführungen pro Konto/Region. Ein Mitglied des AWS-Support-Teams kann dieses Limit erhöhen.

Lambda ist so konzipiert, dass es automatisch für Sie skaliert wird. Der Skalierung einer Funktion sind keine grundlegenden Grenzen gesetzt. Lambda teilt die Kapazität dynamisch zu, um die Rate eingehender Ereignisse zu berücksichtigen.

Schnittstellen

Lambda-Funktionen können auf verschiedene Arten verwaltet werden. Über das Dashboard in der Lambda-Konsole können Sie Ihre Lambda-Funktionen einfach auflisten, löschen, aktualisieren und überwachen. Sie können auch das AWS CLI und das AWS SDK verwenden, um Ihre Lambda-Funktionen zu verwalten.

Sie können eine Lambda-Funktion von einem AWS-Ereignis aus auslösen, z. B. Amazon S3-Bucket-Benachrichtigungen, DynamoDB-Streams, CloudWatch-Protokolle, Amazon SES, Amazon Kinesis Streams, Amazon SNS, Amazon Cognito und mehr. Jeder API-Aufruf in einem Service, der CloudTrail unterstützt, kann als Ereignis in Lambda verarbeitet werden, indem auf CloudTrail-Überwachungsprotokolle geantwortet wird. Weitere Informationen zu Ereignisquellen finden Sie unter Kernkomponenten: AWS Lambda-Funktion und Ereignisquellen.[20]

Lambda unterstützt Programmiersprachen wie Java, Node.js und Python. Ihr Code kann vorhandene Bibliotheken enthalten, sogar native. Lambda-Funktionen können problemlos Prozesse starten, die Sprachen verwenden, die von Amazon Linux AMI,[21] unterstützt werden, einschließlich Bash, Go und Ruby. Weitere Informationen finden Sie in der Dokumentation zu Node.js,[22] Python,[23] und Java[24].

Anti-Patterns

Lambda hat die folgenden Anti-Patterns:

- **Lang laufende Anwendungen** – Jede Lambda-Funktion muss innerhalb von 300 Sekunden abgeschlossen sein. Für lange laufende Anwendungen, die Jobs länger als fünf Minuten benötigen, wird Amazon EC2 empfohlen, oder Sie erstellen eine Kette von Lambda-Funktionen, wobei Funktion 1 Funktion 2 aufruft, die Funktion 3 aufruft usw., bis der Prozess abgeschlossen ist.
- **Dynamische Websites** – Obwohl es möglich ist, eine statische Website mit Lambda zu betreiben, kann die Ausführung einer hochdynamischen Website mit großem Volume eine hohe Leistung verbieten. Die Verwendung von Amazon EC2 und Amazon CloudFront wäre ein empfohlener Anwendungsfall.
- **Zustandsorientierte Anwendungen** – Lambda-Code muss in einem „statuslosen“ Stil geschrieben werden; das heißt, es sollte davon ausgegangen werden, dass keine Affinität zur zugrunde liegenden Datenverarbeitungs-Infrastruktur besteht. Der Zugriff auf lokale Dateisysteme, untergeordnete Prozesse und ähnliche Artefakte darf nicht länger als die Lebensdauer der Anforderung dauern. Jeder permanente Status sollte in Amazon S3, DynamoDB oder einem anderen im Internet verfügbaren Speicherdienst gespeichert werden.

Amazon EMR

Amazon EMR[25] ist ein stark verteiltes Datenverarbeitungs-Framework, mit dem Daten schnell und kostengünstig verarbeitet und gespeichert werden können. Amazon EMR verwendet Apache Hadoop, ein Open-Source-Framework, um Ihre Daten und Verarbeitung über einen größenveränderlichen Cluster von Amazon EC2-Instances zu verteilen und Ihnen den Einsatz der gängigsten Hadoop-Tools wie Hive, Pig, Spark usw. zu ermöglichen. Hadoop bietet einen Rahmen für die Verarbeitung großer Datenmengen und Analysen. Amazon EMR übernimmt die umfangreichen Aufgaben zur Bereitstellung, Verwaltung und Wartung der Infrastruktur und Software eines Hadoop-Clusters.

Ideale Nutzungsmuster

Das flexible Framework von Amazon EMR reduziert große Verarbeitungsprobleme und Datensätze in kleinere Jobs und verteilt sie auf viele Rechenknoten in einem Hadoop-Cluster. Diese Fähigkeit eignet sich für viele Verwendungsmuster mit Big Data-Analyse. Hier sind ein paar Beispiele:

- Protokollverarbeitung und -analyse
- Große ETL-Datenbewegung (Extrahieren, Transformieren und Laden)
- Risikomodellierung und Bedrohungsanalyse
- Anzeigenausrichtung und Klickstromanalyse
- Genomik
- Prognose-Analysen
- Ad-hoc Data-Mining und Analysen

Weitere Informationen finden Sie im Bewährte Praktiken für Amazon EMR[26] - Whitepaper.

Kostenmodell

Amazon EMR ermöglicht es, einen dauerhaft verfügbaren Cluster zu erstellen, oder einen temporären, der nach der Analyse automatisch beendet wird. In beiden Fällen zahlen Sie nur für die Zeit, die der Cluster ausgeführt wird.

Amazon EMR unterstützt eine Vielzahl von Amazon EC2-Instance-Typen (Standard, hohe CPU, hoher Speicher, hohe Ein-/Ausgaben usw.) und alle Amazon EC2-Preisoptionen (On-Demand, Reserviert und Spot). Wenn Sie einen Amazon EMR-Cluster (auch als „Auftragsablauf“ bezeichnet) starten, wählen Sie, wie viele und welche Art von Amazon EC2-Instances bereitgestellt werden sollen. Der Amazon EMR-Preis kommt zusätzlich zum Amazon EC2-Preis. Weitere Informationen finden Sie unter Amazon EMR-Preise.[27]

Leistung

Die Amazon EMR-Leistung wird von der Art der EC2-Instances bestimmt, auf denen Sie Ihren Cluster ausführen möchten und wie viele Sie für die Ausführung Ihrer Analysen ausgewählt haben. Sie sollten einen für Ihre Verarbeitungsanforderungen geeigneten Instance-Typ mit ausreichend Arbeitsspeicher-, Speicher- und Verarbeitungsleistung auswählen. Weitere Informationen zu EC2-Instance-Spezifikationen finden Sie unter Amazon EC2-Instance-Typen.[28]

Haltbarkeit und Verfügbarkeit

Standardmäßig ist Amazon EMR fehlertolerant für Core-Knotenfehler und setzt die Jobausführung fort, wenn ein Slave-Knoten ausfällt. Momentan stellt Amazon EMR nicht automatisch einen anderen Knoten zur Übernahme fehlgeschlagener Slaves bereit, aber Kunden können den Zustand von Knoten überwachen und ausgefallene Knoten durch CloudWatch ersetzen.

Um den unwahrscheinlichen Fall eines Ausfalls eines Master-Knotens zu behandeln, sollten Sie Ihre Daten in einem dauerhaften Speicher wie Amazon S3 sichern. Optional können Sie Amazon EMR mit der MapR-Verteilung,[29] ausführen, die eine No-NameNode-Architektur bietet, die mehrere gleichzeitige Fehler mit automatischem Failover und Rückgriff tolerieren kann. Die Metadaten werden wie die Daten verteilt und repliziert. Mit einer No-NameNode-Architektur gibt es keine praktische Begrenzung für die Anzahl der gespeicherten Dateien und auch keine Abhängigkeit von einem externen Netzwerkspeicher.

Skalierbarkeit und Elastizität

Mit Amazon EMR lässt sich die Größe eines laufenden Clusters leicht anpassen.[30] Sie können jederzeit Kernknoten hinzufügen, die das verteilte Hadoop-Dateisystem (HDFS) enthalten, um die Verarbeitungsleistung zu erhöhen und die HDFS-Speicherkapazität (und den Durchsatz) zu erhöhen. Darüber hinaus können Sie Amazon S3 nativ verwenden oder EMFS zusammen mit oder anstelle von lokalem HDFS verwenden, wodurch Sie Speicher und Rechenleistung von Ihrem Speicher entkoppeln können, was für mehr Flexibilität und Kosteneffizienz sorgt.

Sie können auch jederzeit Aufgabenknoten hinzufügen und entfernen, die Hadoop-Jobs verarbeiten, aber HDFS nicht verwalten können. Einige Kunden fügen ihren Clustern bei der Stapelverarbeitung Hunderte von Instances hinzu und entfernen die zusätzlichen Instances nach Abschluss der Verarbeitung. Beispielsweise wissen Sie möglicherweise nicht, wie viele Daten in Ihren Clustern in 6 Monaten verarbeitet werden, oder haben zeitweise stark erhöhten Verarbeitungsbedarf. Mit Amazon EMR müssen Sie Ihre zukünftigen Anforderungen oder Vorkehrungen für Spitzennachfrage nicht erraten, da Sie Kapazität jederzeit problemlos hinzufügen oder entfernen können.

Darüber hinaus können Sie alle neuen Cluster verschiedener Größen hinzufügen und jederzeit mit ein paar Klicks in der Konsole oder durch einen programmgesteuerten API[31] -Aufruf entfernen.

Schnittstellen

Amazon EMR unterstützt viele Tools zusätzlich zu Hadoop, die für Big Data-Analyse verwendet werden können und jeweils über eigene Schnittstellen verfügen. Hier eine kurze Zusammenfassung der beliebtesten Optionen:

Hive

Hive ist ein Open-Source-Data-Warehouse- und Analysepaket, das auf Hadoop ausgeführt wird. Hive wird von Hive QL betrieben, einer SQL-basierten Sprache, mit der Benutzer Daten strukturieren, zusammenfassen und abfragen können. Hive QL geht über Standard-SQL hinaus und bietet erstklassigen Support für Map-/Reduce-Funktionen und komplexe erweiterbare benutzerdefinierte Datentypen wie JSON und Thrift. Diese Fähigkeit ermöglicht die Verarbeitung komplexer und unstrukturierter Datenquellen wie Textdokumente und Protokolldateien.

Hive ermöglicht Benutzererweiterungen über benutzerdefinierte Funktionen, die in Java geschrieben sind. Amazon EMR hat zahlreiche Verbesserungen an Hive vorgenommen, einschließlich der direkten Integration in DynamoDB und Amazon S3. Mit Amazon EMR können Sie beispielsweise Tabellenpartitionen automatisch aus Amazon S3 laden, Sie können Daten in Tabellen in Amazon S3 schreiben, ohne temporäre Dateien zu verwenden, und Sie können auf Ressourcen in Amazon S3 zugreifen, z. B. Skripts für benutzerdefinierte Zuordnungen und/oder Reduzierungsoperationen und zusätzliche Bibliotheken. Weitere Informationen finden Sie unter Apache Hive[32] im *Amazon EMR-Versionshandbuch*.

Pig

Pig ist ein Open-Source-Analysepaket, das auf Hadoop läuft. Pig wird von Pig Latin betrieben, einer SQL-ähnlichen Sprache, mit der Benutzer Daten strukturieren, zusammenfassen und abfragen können. Neben den SQL-ähnlichen Operationen bietet Pig Latin auch erstklassige Unterstützung für Karten- und Reduzierungsfunktionen und komplexe erweiterbare benutzerdefinierte Datentypen. Diese Fähigkeit ermöglicht die Verarbeitung komplexer und unstrukturierter Datenquellen wie Textdokumente und Protokolldateien.

Pig ermöglicht Benutzererweiterungen über benutzerdefinierte Funktionen, die in Java geschrieben sind. Amazon EMR hat zahlreiche Verbesserungen an Pig vorgenommen, darunter die Möglichkeit, mehrere Dateisysteme zu verwenden (normalerweise kann Pig nur auf ein Fern-Dateisystem zugreifen), die Möglichkeit, Kunden-JARs und Skripts von Amazon S3 zu laden (z. B. „REGISTER s3://my-bucket/piggybank.jar“) und zusätzliche Funktionen für die String- und DateTime-Verarbeitung. Weitere Informationen finden Sie unter Apache Pig[33] im *Amazon EMR-Versionshandbuch*.

Spark

Spark ist eine Open-Source-Datenanalyse-Engine, die auf Hadoop mit den Grundlagen für In-Memory MapReduce basiert. Spark bietet zusätzliche Geschwindigkeit für bestimmte Analysen und ist die Grundlage für andere Tools wie Shark (SQL-gesteuertes Data Warehousing), Spark Streaming (Streaming-Anwendungen), GraphX (Graph-Systeme) und MLlib (Machine Learning). Weitere Informationen finden Sie im Blogeintrag der Installation von Apache Spark für ein Amazon EMR-Cluster.[34]

HBase

HBase ist eine Open-Source-, nicht-relationale, verteilte Datenbank nach dem Vorbild von Googles BigTable. Es wurde als Teil des Hadoop-Projekts der Apache Software Foundation entwickelt und wird auf dem Hadoop Distributed File System (HDFS) ausgeführt, um BigTable-ähnliche Funktionen für Hadoop bereitzustellen. HBase bietet Ihnen eine fehlertolerante, effiziente Möglichkeit zum Speichern großer Mengen von Daten mit geringer Dichte dank spaltenbasierter Komprimierung und Speicherung. Darüber hinaus bietet HBase eine schnelle Suche nach Daten, da die Daten im Speicher statt auf der Festplatte gespeichert werden.

HBase ist für sequenzielle Schreibvorgänge optimiert und äußerst effizient für Batch-Einfügungen, Aktualisierungen und Löschvorgänge. HBase arbeitet nahtlos mit Hadoop zusammen, teilt sein Dateisystem und dient als direkte Eingabe und Ausgabe für Hadoop-Jobs. HBase integriert sich auch in Apache Hive, ermöglicht SQL-ähnliche Abfragen über HBase-Tabellen, verbindet sich mit Hive-basierten Tabellen und unterstützt Java Database Connectivity (JDBC). Mit Amazon EMR können Sie HBase auf Amazon S3 sichern (vollständig oder inkrementell, manuell oder automatisiert), und Sie können von einer zuvor erstellten Sicherung wiederherstellen. Weitere Informationen finden Sie unter HBase und EMR[35] im *Amazon EMR-Entwicklerhandbuch*.

Impala

Impala ist ein Open-Source-Tool im Hadoop-Ökosystem für interaktive Ad-hoc-Abfragen mit SQL-Syntax. Anstatt MapReduce zu verwenden, verwendet es eine massiv parallele Verarbeitungs-Engine (MPP), die der in herkömmlichen relationalen Datenbank-Management-Systemen (RDBMS) ähnelt. Mit dieser Architektur können Sie Ihre Daten in HDFS- oder HBase-Tabellen sehr schnell abfragen und Hadoops Fähigkeit nutzen, verschiedene Datentypen zu verarbeiten und das Schema zur Laufzeit bereitzustellen. Dies macht Impala zu einem großartigen Tool für interaktive Analysen mit geringer Latenz.

Impala hat auch benutzerdefinierte Funktionen in Java und C++ und kann über ODBC- und JDBC-Treiber Verbindungen zu BI-Tools herstellen. Impala verwendet die Hive-Metasorte, um Informationen über die Eingabedaten einschließlich der Partitionsnamen und Datentypen zu speichern. Weitere Informationen finden Sie unter Impala und EMR[36] im *Amazon EMR-Entwicklerhandbuch*.

Hunk

Hunk wurde von Splunk entwickelt, um Maschinendaten zugänglich, nutzbar und für jedermann wertvoll zu machen. Mit Hunk können Sie in Amazon EMR und Amazon S3 gespeicherte Daten interaktiv erkunden, analysieren und visualisieren und Splunk-Analysen auf Hadoop nutzen. Weitere Informationen finden Sie unter Amazon EMR mit Hunk: Splunk-Analyse für Hadoop und NoSQL.[37]

Presto

Presto ist eine Open-Source-Engine für verteilte SQL-Abfragen, die für die Ad-hoc-Analyse von Daten mit geringer Latenz optimiert ist. Es unterstützt den SQL-Standard ANSI, einschließlich komplexer Abfragen, Sammlungen, Verbunde und Fensterfunktionen. Presto kann Daten aus mehreren Datenquellen verarbeiten, darunter das verteilte Dateisystem Hadoop (HDFS) und Amazon S3.

Weitere Tools von Drittanbietern

Amazon EMR unterstützt auch eine Vielzahl anderer gängiger Anwendungen und Tools im Hadoop-Ökosystem wie R (Statistik), Mahout (maschinelles Lernen), Ganglia (Überwachung), Accumulo (sichere NoSQL-Datenbank), Hue (Benutzeroberfläche zur Analyse von Hadoop-Daten), Sqoop (relationaler Datenbankkonnektor), HCatalog (Tabellen- und Speicherverwaltung) und mehr.

Darüber hinaus können Sie Ihre eigene Software auf Amazon EMR installieren, um Ihre geschäftlichen Anforderungen zu erfüllen. AWS ermöglicht das schnelle Verschieben großer Datenmengen von Amazon S3 nach HDFS, von HDFS nach Amazon S3 und zwischen Amazon S3-Buckets mithilfe von S3DistCp,[38] von Amazon EMR, einer Erweiterung des Open-Source-Tools DistCp, mit der MapReduce große Mengen effizient verschieben kann von Dateien.

Sie können optional das EMR File System (EMRFS) verwenden, eine Implementierung von HDFS, mit der Amazon EMR-Cluster Daten auf Amazon S3 speichern können. Sie können die serverseitige und die clientseitige Amazon S3-Verschlüsselung sowie die konsistente Ansicht für EMRFS aktivieren. Wenn Sie EMRFS verwenden, wird ein Metadatenspeicher transparent in DynamoDB erstellt, um die Interaktion mit Amazon S3 zu verwalten und Ihnen zu ermöglichen, dass mehrere EMR-Cluster dieselben EMRFS-Metadaten und denselben Speicher auf Amazon S3 verwenden können.

Anti-Patterns

Amazon EMR hat folgende Anti-Patterns:

- **Kleine Datenmengen** – Amazon EMR ist für massive Parallelverarbeitung ausgelegt. Wenn Ihr Datensatz klein genug ist, um auf einer einzelnen Maschine in einem einzigen Thread ausgeführt zu werden, ist der zusätzliche Aufwand zum Zuordnen und Reduzieren von Aufträgen für kleine Datensätze, die problemlos im Arbeitsspeicher eines einzelnen Systems verarbeitet werden können, nicht sinnvoll.
- **ACID-Transaktionsanforderungen** – Zwar gibt es Möglichkeiten, ACID (Atomizität, Konsistenz, Isolation, Haltbarkeit) oder einen begrenzten Grad an ACID auf Hadoop zu erreichen, eine andere Datenbank wie Amazon RDS oder eine relationale Datenbank, die auf Amazon EC2 läuft, ist jedoch eine bessere Option für Workloads mit strikteren Anforderungen.

Amazon Machine Learning

Amazon ML[39] ist ein Service, der es jedem ermöglicht, Prognoseanalysen und Machine-Learning-Technologie zu verwenden. Amazon ML bietet Visualisierungswerkzeuge und Assistenten, die Sie durch den Prozess des Erstellens von Machine Learning (ML) -Modellen leiten, ohne dass Sie

komplexe ML-Algorithmen und -Technologien erlernen müssen. Nachdem Ihre Modelle fertig sind, können Sie mithilfe von Amazon ML einfach Prognosen für Ihre Anwendung mithilfe von API-Vorgängen ermitteln, ohne benutzerdefinierten Prognosegenerierungscode implementieren oder eine Infrastruktur verwalten zu müssen.

Amazon ML kann ML-Modelle auf Grundlage der in Amazon S3, Amazon Redshift oder Amazon RDS gespeicherten Daten erstellen. Eingebaute Assistenten leiten Sie durch die Schritte des interaktiven Durchsuchens Ihrer Daten, zum Trainieren des ML-Modells, zum Bewerten der Modellqualität und zum Anpassen der Ergebnisse an Geschäftsziele. Nachdem ein Modell bereit ist, können Sie Vorhersagen in beiden Stapel oder mithilfe der Echtzeit-API mit niedriger Latenz anfordern.

Ideale Nutzungsmuster

Amazon ML eignet sich ideal zum Auffinden von Mustern in Ihren Daten und zum Erstellen von ML-Modellen, mit denen Sie Vorhersagen für neue, ungesehene Datenpunkte erstellen können. Als Beispiel:

- **Aktivieren Sie Anwendungen, um verdächtige Transaktionen zu melden** – Erstellen Sie ein ML-Modell, das vorhersagt, ob eine neue Transaktion legitim oder betrügerisch ist.
- **Prognostizierte Produktnachfrage** – Geben Sie historische Bestellinformationen ein, um zukünftige Bestellmengen vorherzusagen.
- **Personalisieren des Anwendungsinhalts** – Sagen Sie voraus, an welchen Elementen ein Benutzer am meisten interessiert ist, und rufen Sie diese Vorhersagen in Echtzeit aus Ihrer Anwendung ab.
- **Benutzeraktivitäten vorhersagen** – Analysieren Sie das Benutzerverhalten, um Ihre Website anzupassen und eine bessere Benutzererfahrung zu bieten.
- **Soziale Medien anhören** – Social Media-Feeds erfassen und analysieren, die sich auf geschäftliche Entscheidungen auswirken können.

Kostenmodell

Mit Amazon ML zahlen Sie nur für das, was Sie verwenden. Es gibt keine Mindestgebühren und keine Vorabverpflichtungen. Amazon ML berechnet einen Stundensatz für die Rechenzeit, die für die Erstellung von Prognosemodellen verwendet wird, und Sie bezahlen dann die Anzahl der Prognosen, die für Ihre

Anwendung generiert wurden. Für Echtzeit-Prognosen können Sie auch einen Stundensatz für die reservierte Kapazität auf der Grundlage des für Ihr Modell erforderlichen Arbeitsspeichers bezahlen.

Die Kosten für die Datenanalyse, das Modelltraining und die Auswertung basieren auf der Anzahl der Rechenstunden, die für die Ausführung benötigt werden, und hängen von der Größe der Eingabedaten, der Anzahl der darin enthaltenen Attribute und der Anzahl und Art der angewendeten Transformationen ab. Die Gebühren für Datenanalyse und Modellbildung liegen bei 0,42 USD pro Stunde. Prognosegebühren werden als Stapel und Echtzeit kategorisiert. Stapel-Prognosen sind 0,10 USD pro 1000 Prognosen, aufgerundet auf die nächsten 1000, während Echtzeit-Prognosen 0,0001 USD pro Prognose sind, aufgerundet auf den nächsten Cent. Für Echtzeit-Prognosen gibt es auch eine reservierte Kapazitätsgebühr von 0,001 USD pro Stunde für jede 10 MB Speicher, die für Ihr Modell bereitgestellt wird.

Während der Modellerstellung geben Sie die maximale Speichergröße jedes Modells an, um die Kosten zu steuern und die prognostische Leistung zu steuern.
Sie bezahlen die Gebühren für die reservierte Kapazität nur, solange Ihr Modell für Echtzeit-Prognosen aktiviert ist. Kosten für Daten, die in Amazon S3, Amazon RDS oder Amazon Redshift gespeichert sind, werden separat in Rechnung gestellt. Weitere Informationen finden Sie unter Amazon Machine Learning-Preise.[40]

Leistung

Die Zeit, die zum Erstellen von Modellen oder zum Anfordern von Stapelprognosen von diesen Modellen benötigt wird, hängt von der Anzahl der Eingabedatensätze, den Typen und der Verteilung der Attribute in diesen Datensätzen und der Komplexität des von Ihnen angegebenen Datenverarbeitungsrezepts ab.

Die meisten Echtzeit-Prognoseanforderungen geben eine Antwort innerhalb von 100 ms zurück und sind damit schnell genug für interaktive Web-, mobile oder Desktop-Anwendungen. Wie viel Zeit die Echtzeit-API genau benötigt, um eine Prognose zu generieren, hängt vom Umfang der Eingabedatensätze und der Komplexität des „Rezepts“[41] der Datenverarbeitung ab, das dem ML-Modell zugeordnet ist, das die Prognosen erzeugt. Jedes ML-Modell, das für Echtzeitvorhersagen aktiviert ist, kann standardmäßig für die Anforderung von

bis zu 200 Transaktionen pro Sekunde verwendet werden wobei diese Anzahl kann erhöht werden, indem der Kundendienst kontaktiert wird. Sie können die Anzahl der von Ihren ML-Modellen angeforderten Vorhersagen mithilfe von CloudWatch-Messdaten überwachen.

Haltbarkeit und Verfügbarkeit

Amazon ML ist auf hohe Verfügbarkeit ausgelegt. Es gibt keine Wartungsfenster oder geplante Ausfallzeiten. Der Service wird in den bewährten, hochverfügbaren Amazon-Datenzentren ausgeführt, wobei die Replikation von Service-Stapel in drei Einrichtungen in jeder AWS-Region konfiguriert ist, um Fehlertoleranz bei Ausfall eines Servers oder Ausfall der Availability Zone bereitzustellen.

Skalierbarkeit und Elastizität

Sie können Datensätze mit einer Größe von bis zu 100 GB verarbeiten, um ML-Modelle zu erstellen oder Stapelprognosen anzufordern. Bei großen Mengen von Batch-Prognosen können Sie Ihre Eingabedatensätze in separate Blöcke aufteilen, um die Verarbeitung eines größeren Prognose-Datenvolumens zu ermöglichen.

Standardmäßig können Sie bis zu fünf gleichzeitige Jobs ausführen und wenn Sie sich an den Kundendienst wenden, können Sie dieses Limit erhöhen. Da es sich bei Amazon ML um einen verwalteten Service handelt, müssen keine Server bereitgestellt werden und daher können Sie skalieren, wenn Ihre Anwendung wächst, ohne dass Ressourcen, die nicht verwendet werden, überproportional bereitgestellt oder bezahlt werden.

Schnittstellen

Das Erstellen einer Datenquelle ist so einfach wie das Hinzufügen Ihrer Daten zu Amazon S3 oder das direkte Abrufen von Daten aus Amazon Redshift oder MySQL-Datenbanken, die von Amazon RDS verwaltet werden. Nachdem Ihre Datenquelle definiert wurde, können Sie über die Konsole mit Amazon ML interagieren. Der programmatische Zugriff auf Amazon ML wird freigegeben durch die AWS SDKs und Amazon ML API.[42] Sie können Amazon ML-Entitäten auch mit der AWS CLI erstellen und verwalten, die auf Windows-, Mac- und Linux/UNIX-Systemen verfügbar ist.

Anti-Patterns

Amazon ML hat folgende Anti-Patterns:

- **Sehr große Datensätze** – Obwohl Amazon ML bis zu 100 GB Daten unterstützen kann, wird die Aufnahme von Daten im Terabyte-Maßstab derzeit nicht unterstützt. Die Verwendung von Amazon EMR zur Ausführung von Sparks Machine Learning Library (MLlib) ist ein gängiges Tool für einen solchen Anwendungsfall.
- **Nicht unterstützte Lernaufgaben** – Amazon ML kann verwendet werden, um ML-Modelle zu erstellen, die eine binäre Klassifizierung (wählen Sie eine von zwei Möglichkeiten und bieten ein gewisses Maß an Vertrauen), Mehrfach-Klassifikation (Auswahlmöglichkeiten auf mehr als zwei Optionen erweitern) oder numerische Regression (eine Zahl vorhersagen) direkt. Nicht unterstützte ML-Aufgaben wie Sequenzvorhersage oder unüberwachtes Clustering können mithilfe von Amazon EMR ausgeführt werden, um Spark und MLlib auszuführen.

Amazon DynamoDB

Amazon DynamoDB[43] ist ein schneller, vollständig verwalteter NoSQL-Datenbank-Service, der das Speichern und Abrufen beliebiger Datenmengen und das Beaufschlagen von Anfragen mit beliebigem Umfang auf einfache und kostengünstige Weise ermöglicht. DynamoDB entlastet den administrativen Aufwand beim Betrieb und der Skalierung eines hochverfügbaren verteilten Datenbank-Clusters. Diese Speicheralternative erfüllt die Latenz- und Durchsatzanforderungen hoch anspruchvoller Anwendungen, indem sie eine Latenz und eine vorhersagbare Leistung im einstelligen Millisekundenbereich mit nahtlosem Durchsatz und Speicher-Skalierbarkeit bietet.

DynamoDB speichert strukturierte Daten in Tabellen, indiziert nach Primärschlüssel und ermöglicht Lese- und Schreibzugriff mit geringer Latenz auf Objekte von 1 Byte bis 400 KB. DynamoDB unterstützt drei Datentypen (Zahl, Zeichenfolge und Binär) sowohl in skalaren als auch in mehrwertigen Sätzen. Es unterstützt Dokumentspeicher wie JSON, XML oder HTML in diesen Datentypen. Tabellen haben kein festes Schema, daher kann jedes Datenelement eine unterschiedliche Anzahl von Attributen haben. Der Primärschlüssel kann entweder ein Einzelattribut-Hash-Schlüssel oder ein zusammengesetzter Hash-Bereichsschlüssel sein.

DynamoDB bietet sowohl globale als auch lokale sekundäre Indizes, die zusätzliche Flexibilität für die Abfrage von Attributen bieten, die nicht dem Primärschlüssel entsprechen. DynamoDB bietet sowohl schlüssige Lesevorgänge (standardmäßig) als auch stark konsistente Lesevorgänge (optional) sowie implizite Transaktionen auf Elementebene für das Setzen, Aktualisieren, Löschen, bedingte Operationen und Inkrementieren/Dekrementieren von Elementen.

DynamoDB ist mit anderen Services wie Amazon EMR, Amazon Redshift, AWS Data Pipeline und Amazon S3 für Analysen, Data-Warehouse, Datenimport/-export, Backup und Archivierung integriert.

Ideale Nutzungsmuster

DynamoDB eignet sich ideal für vorhandene oder neue Anwendungen, die eine flexible NoSQL-Datenbank mit geringen Lese- und Schreiblatenzen benötigen, sowie die Möglichkeit, Speicher und Durchsatz je nach Bedarf ohne Codeänderungen oder Ausfallzeiten zu erhöhen oder zu reduzieren.

Häufige Anwendungsfälle sind u. a.:

- Mobile Anwendungen
- Computerspiele
- Digitales Adserving
- Live-Abstimmung
- Zielgruppeninteraktion für Live-Veranstaltungen
- Sensornetzwerke
- Logaufnahme
- Zugriff auf Web-basierte Inhalte
- Metadaten-Speicher für Amazon S3-Objekte
- E-Commerce-Einkaufswagen
- Web-Sitzungsverwaltung

Viele dieser Anwendungsfälle erfordern eine hoch verfügbare und skalierbare Datenbank, da Ausfallzeiten oder Leistungseinbußen sich unmittelbar negativ auf das Geschäft einer Organisation auswirken.

Kostenmodell

Mit DynamoDB zahlen Sie nur für das, was Sie verwenden, und es gibt keine Mindestgebühr. DynamoDB hat drei Preiskomponenten: Provisionierte Durchsatzkapazität (pro Stunde), indizierte Datenspeicherung (pro GB im Monat), Datentransfer nach innen oder außen (pro GB im Monat). Neue Kunden können DynamoDB kostenlos als Teil des kostenlosen Nutzungskontingents für AWS Free Usage Tier.[44] Weitere Informationen finden Sie unter Amazon DynamoDB-Preise.[45]

Leistung

SSDs und die Beschränkung der Indexierung auf Attribute bieten einen hohen Durchsatz und niedrige Latenz[46] und reduzieren drastisch die Kosten von Lese- und Schreibvorgängen. Wenn die Datensätze wachsen, ist eine vorhersehbare Leistung erforderlich, so dass für die Workloads eine niedrige Latenz aufrechterhalten werden kann. Diese vorhersehbare Leistung kann durch Definieren der bereitgestellten Durchsatzkapazität erreicht werden, die für eine gegebene Tabelle erforderlich ist.

Hinter den Kulissen wickelt der Service die Bereitstellung von Ressourcen ab, um die angeforderte Durchsatzrate zu erreichen. Sie müssen nicht über Instances, Hardware, Arbeitsspeicher und andere Faktoren nachdenken, die sich auf die Durchsatzrate einer Anwendung auswirken können. Die reservierten Durchsatzkapazitätsreservierungen sind elastisch und können bei Bedarf erhöht oder verringert werden.

Haltbarkeit und Verfügbarkeit

DynamoDB verfügt über eine integrierte Fehlertoleranz, die automatisch und synchron Daten für eine hohe Verfügbarkeit in drei Datenzentren in einer Region repliziert und dabei hilft, Daten vor Fehlern einzelner Maschinen oder sogar Anlagen zu schützen. DynamoDB-Streams[47] erfasst alle Datenaktivitäten, die in Ihrer Tabelle ausgeführt werden, und ermöglicht die Einrichtung regionaler Replikation von einer geografischen Region in eine andere, um eine noch höhere Verfügbarkeit bereitzustellen.

Skalierbarkeit und Elastizität

DynamoDB ist sowohl hochskalierbar als auch elastisch. Die Datenmenge, die Sie in einer DynamoDB-Tabelle speichern können, ist unbegrenzt. Der Service weist automatisch mehr Speicher zu, wenn Sie mit den DynamoDB-API-Schreiboperationen mehr Daten speichern. Daten werden bei Bedarf automatisch partitioniert und neu partitioniert, während die Verwendung von SSDs vorhersagbare Antwortzeiten mit geringer Latenzzeit in jeder Größenordnung bietet. Der Service ist auch elastisch, indem Sie die Lese- und Schreibkapazität einer Tabelle einfach „Einwählen“[48] oder „Auswählen“[49], wenn sich Ihre Bedürfnisse ändern.

Schnittstellen

DynamoDB bietet eine untergeordnete REST-API sowie übergeordnete SDKs für Java, .NET und PHP, die die untergeordnete REST-API umschließen und einige objektrelationale Zuordnungsfunktionen (ORM) bereitstellen. Diese APIs stellen sowohl eine Verwaltungs- als auch eine Datenschnittstelle für DynamoDB bereit. Die API bietet derzeit Operationen an, die die Tabellenverwaltung ermöglichen (Erstellen, Auflisten, Löschen und Abrufen von Metadaten) und Arbeiten mit Attributen (Attribute abrufen, schreiben und löschen; Abfragen über einen Index und vollständigen Scan).

Während Standard-SQL nicht verfügbar ist, können Sie die DynamoDB-Auswahloperation verwenden, um SQL-ähnliche Abfragen zu erstellen, die basierend auf den von Ihnen angegebenen Kriterien eine Reihe von Attributen abrufen. Sie können mit DynamoDB auch mit der Konsole arbeiten.

Anti-Patterns

DynamoDB hat folgende Anti-Patterns:

- **Vorgeschriebene Anwendung, die an eine traditionelle relationale Datenbank gebunden ist** – Wenn Sie eine vorhandene Anwendung in die AWS-Cloud portieren und weiterhin eine relationale Datenbank verwenden möchten, können Sie entweder Amazon RDS (Amazon Aurora, MySQL, PostgreSQL, Oracle oder SQL Server) oder eines der vielen vorkonfigurierten Amazon EC2-Datenbank-AMIs. Sie können auch Ihre Datenbanksoftware auf einer von Ihnen verwalteten EC2-Instance installieren.

- **Verbunde oder komplexe Transaktionen** – Während viele Lösungen DynamoDB zur Unterstützung ihrer Benutzer nutzen können, ist es möglich, dass Ihre Anwendung Verbunde, komplexe Transaktionen und andere relationale Infrastrukturen erfordert, die von herkömmlichen Datenbankplattformen bereitgestellt werden. Wenn dies der Fall ist, können Sie Amazon Redshift, Amazon RDS oder Amazon EC2 mit einer selbst verwalteten Datenbank erkunden.
- **BLOB-Daten (Binary Large Objects)** – Wenn Sie große BLOB-Daten (größer als 400 KB) wie digitales Video, Bilder oder Musik speichern möchten, sollten Sie Amazon S3 in Betracht ziehen. In diesem Szenario spielt DynamoDB jedoch immer noch eine Rolle, da es die Metadaten (z. B. Name, Größe, Erstellungsdatum, Besitzer, Standort usw.) der Objekte in Bezug auf Ihre binären Objekte protokolliert.
- **Große Daten mit niedriger Ein-/Ausgabe-Rate** – DynamoDB verwendet SSD-Laufwerke und ist für Workloads mit hoher Ein-/Ausgabe-Rate pro gespeichertem GB optimiert. Wenn Sie beabsichtigen, sehr große Datenmengen zu speichern, auf die nur selten zugegriffen wird, sind möglicherweise andere Speicheroptionen, z. B. Amazon S3, besser geeignet.

Amazon Redshift

Amazon Redshift[50] ist ein schneller, vollständig verwalteter Data-Warehouse-Service im Petabyte-Bereich, mit dem Sie alle Ihre Daten mithilfe Ihrer vorhandenen Business-Intelligence-Tools einfach und kostengünstig analysieren können. Es ist für Datenmengen von wenigen hundert Gigabytes bis zu einem Petabyte oder mehr optimiert und kostet weniger als ein Zehntel der Kosten herkömmlicher Data Warehousing-Lösungen.

Amazon Redshift bietet eine schnelle Abfrage- und Ein-/Ausgabe-Leistung für Datensätze mit praktisch jeder Größe, indem die Columnar Storage-Technologie verwendet wird, während Abfragen über mehrere Knoten hinweg parallelisiert und verteilt werden. Es automatisiert die meisten allgemeinen Verwaltungsaufgaben, die mit der Bereitstellung, Konfiguration, Überwachung, Sicherung und Sicherstellung eines Data-Warehouse verbunden sind, sodass es einfach und kostengünstig zu verwalten und zu warten ist. Dank dieser Automatisierung können Sie Datenspeicher in Petabyte-Größe in Minuten statt in Wochen oder Monaten erstellen, die von traditionellen lokalen Implementierungen übernommen wurden.

Ideale Nutzungsmuster

Amazon Redshift ist ideal für die Online-Analyseverarbeitung (OLAP) mit Ihren vorhandenen Business-Intelligence-Tools. Organisationen verwenden Amazon Redshift, um Folgendes zu tun:

- Analysieren Sie globale Verkaufsdaten für mehrere Produkte
- Speichern Sie historische Börsenhandelsdaten
- Analysieren Sie Werbeeindrücke und Klicks
- Ansammlung von Computerspiel-Daten
- Analyse sozialer Trends
- Messen Sie klinische Qualität, operative Effizienz und finanzielle Leistung im Gesundheitswesen

Kostenmodell

Ein Amazon Redshift Data-Warehouse-Cluster erfordert keine langfristigen Verpflichtungen oder Vorlaufkosten. Dies befreit Sie von den Kapitalkosten und der Komplexität der Planung und dem Kauf von Data-Warehouse-Kapazitäten für Ihre Anforderungen. Die Gebühren basieren auf der Größe und Anzahl der Knoten Ihres Clusters.

Für Sicherungsspeicher werden bis zu 100 % Ihres bereitgestellten Speichers keine zusätzlichen Kosten verursacht. Wenn Sie beispielsweise über einen aktiven Cluster mit 2 XL-Knoten für insgesamt 4 TB Speicher verfügen, stellt AWS bis zu 4 TB Sicherungsspeicher auf Amazon S3 ohne zusätzliche Kosten bereit. Sicherungsspeicher jenseits der bereitgestellten Speichergröße und Sicherung, die nach dem Beenden des Clusters gespeichert werden, werden mit Amazon S3 Raten abgerechnet.[51] Für die Kommunikation zwischen Amazon S3 und Amazon Redshift fallen keine Kosten für die Datenübertragung an. Weitere Informationen finden Sie unter Amazon Redshift-Preise.[52]

Leistung

Amazon Redshift verwendet eine Vielzahl von Innovationen, um eine sehr hohe Leistung bei Datensätzen von Hunderten von Gigabyte bis zu einem Petabyte oder mehr zu erreichen. Durch die Anwendung von Techniken wie Columnar Storage, Datenkomprimierung und Zone Maps wird die Anzahl der für Abfragen erforderlichen Ein- und Ausgaben reduziert.

Amazon Redshift verfügt über eine MPP-Architektur (Massively Parallel Processing), die SQL-Operationen parallelisiert und verteilt, um alle verfügbaren Ressourcen zu nutzen. Die zugrunde liegende Hardware ist für Hochleistungsdatenverarbeitung ausgelegt und verwendet lokal angeschlossene Speicher, um den Durchsatz zwischen den CPUs und den Festplatten zu maximieren. Der Durchsatz zwischen den Knoten wird mithilfe eines 10GbE-Mesh-Netzwerks maximiert. Die Leistung kann basierend auf Ihren Data Warehousing-Anforderungen angepasst werden: AWS bietet Dense Compute (DC) mit SSD-Laufwerken sowie Dense Storage (DS) -Optionen.

Haltbarkeit und Verfügbarkeit

Amazon Redshift erkennt und ersetzt automatisch einen ausgefallenen Knoten in Ihrem Data-Warehouse-Cluster. Der Data-Warehouse-Cluster ist schreibgeschützt, bis ein Ersatzknoten bereitgestellt und der Datenbank hinzugefügt wird. Dies dauert in der Regel nur wenige Minuten. Amazon Redshift stellt Ihren Ersatzknoten sofort zur Verfügung und lässt dieam häufigsten verwendeten Daten zuerst von Amazon S3 Streamen, damit Sie Ihre Daten so schnell wie möglich abfragen können.

Darüber hinaus bleibt Ihr Data-Warehouse-Cluster bei einem Laufwerksausfall verfügbar. Da Amazon Redshift Ihre Daten über den Cluster hinweg spiegelt, verwendet er die Daten von einem anderen Knoten, um fehlerhafte Laufwerke neu zu erstellen. Amazon Redshift-Cluster befinden sich in einer Availability Zone,[53] wenn Sie jedoch eine Multi-AZ für Amazon Redshift einrichten möchten, können Sie eine Spiegelung einrichten und dann Replikation und Failover selbst verwalten.

Skalierbarkeit und Elastizität

Mit ein paar Klicks in der Konsole oder einem API-Aufruf,[54] können Sie die Anzahl oder den Typ von Knoten in Ihrem Data-Warehouse ändern, wenn sich Ihre Leistungs- oder Kapazitätsanforderungen ändern. Mit Amazon Redshift können Sie mit nur einem 160-GB-Knoten beginnen und bis zu einem Petabyte oder mehr komprimierter Benutzerdaten mit vielen Knoten skalieren. Weitere Informationen finden Sie im Abschnitt Über Cluster und Knoten,[55] im Thema Amazon Redshift Clusters, im *Amazon Redshift-Verwaltungshandbuch*.

Bei der Größenänderung platziert Amazon Redshift Ihren vorhandenen Cluster in den schreibgeschützten Modus, stellt einen neuen Cluster mit der von Ihnen

gewählten Größe bereit und kopiert anschließend Daten aus Ihrem alten Cluster parallel in Ihren neuen Cluster. Während dieses Vorgangs zahlen Sie nur für den aktiven Amazon Redshift-Cluster. Sie können weiterhin Abfragen für Ihren alten Cluster ausführen, während der neue bereitgestellt wird. Nachdem Ihre Daten in Ihren neuen Cluster kopiert wurden, leitet Amazon Redshift automatisch Abfragen an Ihren neuen Cluster um und entfernt den alten Cluster.

Schnittstellen

Amazon Redshift verfügt über benutzerdefinierte JDBC- und ODBC-Treiber, die Sie von der Registerkarte „Connect Client“ der Konsole herunterladen können. Dadurch können Sie eine Vielzahl vertrauter SQL-Clients verwenden. Sie können auch Standard PostgreSQL JDBC- und ODBC-Treiber verwenden. Weitere Informationen zu Amazon Redshift-Treibern finden Sie unter Amazon Redshift und PostgreSQL.[56]

Es gibt zahlreiche Beispiele für validierte Integrationen mit vielen beliebten Anbietern von BI- und ETL-Lösungen.[57] Lade- und Entladevorgänge werden parallel in jedem Rechenknoten ausprobiert, um die Rate zu maximieren, mit der Daten in Ihrem Data-Warehouse-Cluster aufgenommen werden können sowie zu und von Amazon S3 und DynamoDB. Sie können problemlos Streaming-Daten mit Amazon Kinesis Firehose in Amazon Redshift laden, sodass Analysen mit vorhandenen Business-Intelligence-Tools und Dashboards, die Sie bereits heute verwenden, fast in Echtzeit möglich werden. Metriken für Datenverarbeitungs-, Arbeitsspeicher- und Speicherauslastung sowie den Datenverkehr durch Lese-/Schreibvorgänge Ihres Amazon Redshift Data-Warehouse-Cluster stehen kostenlos über die Konsole oder CloudWatch-API-Operationen.

Anti-Patterns

Amazon Redshift hat folgende Anti-Patterns:

- **Kleine Datensätze** – Amazon Redshift ist für die parallele Verarbeitung über einen Cluster ausgelegt. Wenn Ihr Datensatz weniger als hundert Gigabyte groß ist, werden Sie nicht alle Vorteile von Amazon Redshift nutzen und Amazon RDS ist möglicherweise eine bessere Lösung.
- **Online-Transaktionsverarbeitung (OLTP)** – Amazon Redshift ist für Data-Warehouse-Workloads ausgelegt, die extrem schnelle und kostengünstige Analyse-Funktionen produziert. Wenn Sie ein schnelles Transaktionssystem benötigen, sollten Sie ein traditionelles relationales

Datenbanksystem wählen, das auf Amazon RDS oder einem NoSQL-Datenbankangebot wie DynamoDB basiert.

- **Unstrukturierte Daten** – Daten in Amazon Redshift müssen nach einem definierten Schema strukturiert sein, anstatt eine beliebige Schemastruktur für jede Zeile zu unterstützen. Wenn Ihre Daten nicht strukturiert sind, können Sie ETL-Dateien auf Amazon EMR extrahieren, transformieren und laden, um die Daten für das Laden in Amazon Redshift bereit zu haben.
- **BLOB-Daten** – Wenn Sie große Binärdateien (z. B. digitale Videos, Bilder oder Musik) speichern möchten, sollten Sie die Daten in Amazon S3 speichern und auf ihren Speicherort in Amazon Redshift verweisen. In diesem Szenario behält Amazon Redshift die Metadaten (z. B. Name, Größe, Erstellungsdatum, Eigentümer, Standort usw.) der Objekte in Bezug auf Ihre binären Objekte, die großen Objekte selbst werden jedoch in Amazon S3 gespeichert.

Amazon Elasticsearch Service

Amazon ES[58] ist ein Managed Service, der die Bereitstellung, den Betrieb und die Skalierung von Elasticsearch in der AWS-Cloud vereinfacht. Elasticsearch ist eine in Echtzeit verteilte Such- und Analyse-Engine. Es ermöglicht Ihnen, Ihre Daten mit einer Geschwindigkeit und in einer Größenordnung zu erkunden, die noch nie zuvor möglich war. Es wird für die Volltextsuche, die strukturierte Suche, die Analyse und alle drei in Kombination verwendet.

Sie können Ihren Amazon ES-Cluster über die Konsole in wenigen Minuten einrichten und konfigurieren. Amazon ES verwaltet die Arbeit, die beim Einrichten einer Domäne erforderlich ist, von der Bereitstellung von Infrastrukturkapazität, die Sie anfordern, bis zur Installation der Elasticsearch-Software.

Nachdem Ihre Domäne ausgeführt wurde, automatisiert Amazon ES allgemeine Verwaltungsaufgaben, z. B. das Durchführen von Sicherungen, das Überwachen von Instanzen und das Patchen von Software, die Ihre Amazon ES-Instance antreibt. Es erkennt automatisch fehlerhafte Elasticsearch-Knoten und ersetzt sie, wodurch der mit der selbstverwalteten Infrastruktur und der Elasticsearch-Software verbundene Aufwand reduziert wird. Mit diesem Service können Sie Ihren Cluster einfach über einen einzigen API-Aufruf oder einige Klicks in der Konsole skalieren.

Mit Amazon ES erhalten Sie direkten Zugriff auf die OpenSource-API von Elasticsearch, sodass Code und Anwendungen, die Sie bereits mit Ihren vorhandenen Elasticsearch-Umgebungen verwenden, nahtlos zusammenarbeiten. Es unterstützt die Integration mit Logstash, einer Open-Source-Datenpipeline, mit der Sie Protokolle und andere Ereignisdaten verarbeiten können. Es enthält auch einen integrierten Support für Kibana, einer Open-Source-Analyse- und Visualisierungsplattform, mit der Sie Ihre Daten besser verstehen können.

Ideale Nutzungsmuster

Amazon ES ist ideal für die Abfrage und Suche großer Datenmengen.

Organisationen können Amazon ES für Folgendes verwenden:

- Analysieren Sie Aktivitätsprotokolle, z. B. Protokolle für Kundenanwendungen oder Websites
- Analysieren Sie CloudWatch-Protokolle mit Elasticsearch
- Analysieren Sie Produktnutzungsdaten, die von verschiedenen Services und Systemen stammen
- Analysieren Sie Meinungen sozialer Medien und CRM-Daten und finden Sie Trends für Marken und Produkte
- Analysieren Sie Datenstromaktualisierungen von anderen AWS-Services, z. B. Amazon Kinesis Streams und DynamoDB
- Bieten Sie Ihren Kunden eine umfassende Such- und Navigationserfahrung
- Überwachen Sie die Nutzung für mobile Anwendungen

Kostenmodell

Bei Amazon ES zahlen Sie nur für die von Ihnen verwendeten Datenverarbeitungs- und Speicherressourcen. Es gibt keine Mindestgebühren oder Vorabverpflichtungen. Ihnen werden Gebühren für Amazon ES-Instance-Stunden, Amazon EBS-Speicher (wenn Sie diese Option wählen) und Standardgebühren für Datenübertragung berechnet.[59]

Wenn Sie EBS-Volumes für den Speicher verwenden, können Sie in Amazon ES den Volume-Typ auswählen. Wenn Sie bereitgestellte IOPS (SSD) -Speicher,[60] auswählen werden Ihnen sowohl der Speicherplatz als auch der von Ihnen bereitgestellte Durchsatz in Rechnung gestellt. Sie zahlen jedoch keine Kosten

für die Ein-/Ausgabe, die Sie verbrauchen. Sie haben auch die Option, für zusätzlichen Speicher zu zahlen, basierend auf der kumulativen Größe der EBS-Volumes, die an die Datenknoten in Ihrer Domäne angehängt sind.

Amazon ES stellt Speicherplatz für automatisierte Snapshots kostenlos für jede Amazon ES-Domäne bereit. Manuelle Snapshots werden gemäß den Amazon S3-Speicherraten berechnet. Weitere Informationen finden Sie unter Amazon Elasticsearch Service-Preise.[61]

Leistung

Die Leistung hängt von mehreren Faktoren ab, einschließlich Amazon ES-Instance-Typ, Workload, Index, Anzahl der verwendeten Bruchstücke, schreibgeschützte Replikate und Speicher-Konfigurationen (Instance-Speicher oder EBS-Speicher, z. B. allgemeine SSD). Indizes bestehen aus Bruchstücken von Daten, die auf verschiedenen Instances in mehreren Availability Zones verteilt werden können.

Schreibgeschützte Replikate der Bruchstücke werden von Amazon ES in einer anderen Availability Zone verwaltet, wenn die Zonensensitivität aktiviert ist. Amazon ES kann entweder den schnellen SSD-Instance-Speicher zum Speichern von Indizes oder mehrere EBS-Volumes verwenden. Eine Suchmaschine verwendet häufig Speichergeräte und macht Festplatten schneller zu schnellerer Abfrage- und Suchleistung.

Haltbarkeit und Verfügbarkeit

Sie können Ihre Amazon ES-Domänen für eine hohe Verfügbarkeit konfigurieren, indem Sie die Zone Awareness-Option entweder zum Zeitpunkt der Domänenerstellung oder durch Ändern einer Live-Domäne aktivieren. Wenn die Zonenüberwachung aktiviert ist, verteilt Amazon ES die Instances, die die Domäne unterstützen, über zwei verschiedene Availability Zones. Wenn Sie anschließend Replikate in Elasticsearch aktivieren, werden die Instances automatisch so verteilt, dass sie die zonenübergreifende Replikation ermöglichen.

Sie können die Datenhaltbarkeit für Ihre Amazon ES-Domäne durch automatisierte und manuelle Snapshots erhöhen. Sie können Snapshots verwenden, um Ihre Domäne mit vorinstallierten Daten wiederherzustellen oder um eine neue Domäne mit vorinstallierten Daten zu erstellen. Snapshots werden in Amazon S3 gespeichert, einem sicheren, dauerhaften und hoch

skalierbaren Objektspeicher. Standardmäßig erstellt Amazon ES automatisch tägliche Snapshots jeder Domäne. Darüber hinaus können Sie die Amazon ES-Snapshot-APIs verwenden, um zusätzliche manuelle Snapshots zu erstellen. Die manuellen Snapshots werden in Amazon S3 gespeichert. Manuelle Snapshots können für die regionsübergreifende Notfallwiederherstellung verwendet werden und bieten zusätzliche Haltbarkeit.

Skalierbarkeit und Elastizität

Sie können Instances hinzufügen oder entfernen und Amazon EBS-Volumes problemlos ändern, um Datenwachstum zu ermöglichen. Sie können einige Codezeilen schreiben, die den Status Ihrer Domain mithilfe von CloudWatch-Messdaten überwachen, und die Amazon ES-API aufrufen, um Ihre Domain basierend auf den von Ihnen festgelegten Schwellenwerten nach oben oder unten zu skalieren. Der Service führt die Skalierung ohne Ausfallzeit durch.

Amazon ES unterstützt ein EBS-Volume (maximale Größe von 512 GB) pro Instance die einem Cluster zugeordnet ist. Mit maximal 10 zugelassenen Instances pro Amazon ES-Cluster können Kunden einer einzelnen Amazon ES-Domäne ca. 5 TB Speicherplatz zuweisen.

Schnittstellen

Amazon ES unterstützt die Elasticsearch API,[62] sodass Code, Anwendungen und gängige Tools, die Sie bereits mit vorhandenen Elasticsearch-Umgebungen verwenden, nahtlos zusammenarbeiten. Die AWS-SDKs unterstützen alle Amazon ES-API-Vorgänge, sodass Sie Ihre Domains mithilfe Ihrer bevorzugten Technologie einfach verwalten und mit ihnen interagieren können. Die AWS-CLI oder die Konsole kann auch zum Erstellen und Verwalten Ihrer Domänen verwendet werden.

Amazon ES unterstützt die Integration mit mehreren AWS-Services, einschließlich Streaming von Daten aus Amazon S3, Amazon Kinesis Streams und DynamoDB Streams. Die Integrationen verwenden eine Lambda-Funktion als Ereignisbehandlungsroutine in der Cloud, die auf neue Daten reagiert, indem sie sie verarbeitet und die Daten an Ihre Amazon ES-Domäne streamt. Amazon ES ist außerdem mit CloudWatch zur Überwachung von Amazon ES-Domänenmetriken und CloudTrail zur Überwachung von Konfigurations-API-Aufrufen in Amazon ES-Domänen integriert.

Amazon ES umfasst eine eingebaute Integration mit Kibana, einer Open-Source-Analyse- und Visualisierungsplattform, und unterstützt die Integration mit Logstash, einer Open-Source-Datenpipeline, mit der Sie Protokolle und andere Ereignisdaten verarbeiten können. Sie können Ihre Amazon ES-Domain als Backend-Speicher für alle Protokolle einrichten, die über Ihre Logstash-Implementierung eingehen, sodass strukturierte und unstrukturierte Daten aus einer Vielzahl von Quellen auf einfache Art einbezogen werden können.

Anti-Patterns

Amazon ES hat die folgenden Anti-Patterns:

- **Online-Transaktionsverarbeitung (OLTP)** – Amazon ES ist eine in Echtzeit verteilte Such- und Analyse-Engine. Transaktionen oder Datenmanipulation werden nicht unterstützt. Wenn Sie ein schnelles Transaktionssystem benötigen, ist ein traditionelles relationales Datenbanksystem, das auf Amazon RDS basiert, oder eine NoSQL-Datenbank mit Funktionen wie DynamoDB die bessere Wahl.
- **Petabyte-Speicher** – Mit maximal 10 zulässigen Instances pro Amazon ES-Cluster können Sie einer einzelnen Amazon ES-Domäne etwa 5 TB Speicher zuweisen. Bei größeren Workloads empfiehlt sich die Verwendung von selbstverwalteten Elasticsearch auf Amazon EC2.

Amazon QuickSight

Im Oktober 2015 führte AWS die Vorschau von Amazon QuickSight ein, einem schnellen Cloud-basierten Business Intelligence (BI) -Service, mit dem Sie problemlos Visualisierungen erstellen, Ad-hoc-Analysen durchführen und schnell Geschäftsdaten aus Ihren Daten gewinnen können.

QuickSight verwendet eine neue, superschnelle, parallele Berechnungs-Engine (SPICE), um erweiterte Berechnungen durchzuführen und Visualisierungen schnell zu rendern. QuickSight lässt sich automatisch in AWS-Datenservices integrieren, ermöglicht Organisationen die Skalierung auf Hunderttausende von Benutzern und bietet eine schnelle und reaktionsschnelle Abfrageleistung über die Abfrage-Engine von SPICE. Mit einem Zehntel der Kosten herkömmlicher Lösungen ermöglicht Ihnen QuickSight die Bereitstellung von erschwinglichen BI-Funktionen für alle Mitarbeiter in Ihrem Unternehmen. Um mehr zu erfahren und sich für die Vorschau anzumelden, siehe QuickSight.[63]

Amazon EC2

Amazon EC2,[64] mit Instances, die als virtuelle AWS-Maschinen fungieren, ist dies eine ideale Plattform für den Betrieb eigener selbstverwalteter Big Data-Analyseanwendungen in der AWS-Infrastruktur. Fast jede Software, die Sie unter Linux oder Windows virtualisierten Umgebungen installieren können, kann auf Amazon EC2 ausgeführt werden, und Sie können das bedarfsorientierte Preismodell verwenden. Was Sie nicht erhalten, sind die verwalteten Services auf Anwendungslevel, die mit den anderen erwähnten Services in diesem Whitepaper geliefert werden. Es gibt viele Optionen für selbstverwaltete Big Data-Analysen. Hier sind einige Beispiele:

- Ein NoSQL-Angebot wie MongoDB
- Ein Data-Warehouse oder ein Columnar Store wie Vertica
- Ein Hadoop-Cluster
- Ein Apache Storm-Cluster
- Eine Apache Kafka-Umgebung

Ideale Nutzungsmuster

- **Spezialisierte Umgebung** – Wenn Sie eine benutzerdefinierte Anwendung, eine Variante eines Standard-Hadoop-Satz oder eine Anwendung ausführen, die nicht von einem unserer anderen Angebote abgedeckt wird, bietet Amazon EC2 die Flexibilität und Skalierbarkeit, um Ihre Computing-Anforderungen zu erfüllen.
- **Compliance-Anforderungen** – Bestimmte Compliance-Anforderungen erfordern möglicherweise, dass Sie Anwendungen selbst auf Amazon EC2 statt auf einem Managed Service-Angebot ausführen.

Kostenmodell

Amazon EC2 verfügt über verschiedene Instance-Typen in einer Reihe von Instance-Familien (Standard, hohe CPU, hoher Speicher, hohe Ein-/Ausgabe usw.) und verschiedene Preisoptionen (On-Demand, Reserviert und Spot). Abhängig von Ihren Anwendungsanforderungen möchten Sie möglicherweise zusätzliche Services zusammen mit Amazon EC2 verwenden, z. B. Amazon Elastic Block Store (Amazon EBS) für direkt angeschlossenen persistenten Speicher oder Amazon S3 als dauerhaften Objektspeicher. Jeder kommt mit einem eigenen Preismodell. Wenn Sie Ihre Big Data-Anwendung auf Amazon EC2 ausführen, sind Sie für alle Lizenzgebühren genauso verantwortlich wie in

Ihrem eigenen Datenzentrum. Die AWS Marketplace[65] bietet viele verschiedene Big Data-Softwarepakete von Drittanbietern, die so vorkonfiguriert sind, dass sie mit einem einfachen Mausklick gestartet werden können.

Leistung

Die Leistung in Amazon EC2 wird von dem für Ihre Big Data-Plattform ausgewählten Instance-Typ bestimmt. Jeder Instance-Typ verfügt über unterschiedliche CPU-, RAM-, Speicher-, IOP- und Netzwerkfunktionen, sodass Sie die für Ihre Anwendungsanforderungen richtige Leistungsstufe auswählen können.

Haltbarkeit und Verfügbarkeit

Kritische Anwendungen sollten in einem Cluster über mehrere Availability Zones hinweg in einer AWS-Region ausgeführt werden, damit ein Fehler in der Instance oder im Datenzentrum keine Auswirkungen auf die Anwendungsbenutzer hat. Bei kritischen Anwendungen, die keine Verfügbarkeit erfordern, können Sie Ihre Anwendung bei Amazon S3 sichern und bei einem Ausfall einer Instance oder Zone in jeder Availability Zone in derRegion wiederherstellen. Abhängig von der Anwendung, die Sie ausführen, und den Anforderungen, z. B. der Spiegelung Ihrer Anwendung, gibt es weitere Optionen.

Skalierbarkeit und Elastizität

Auto Scaling[66] ist ein Service, mit dem Sie Ihre Amazon EC2-Kapazität entsprechend den von Ihnen definierten Bedingungen automatisch nach oben oder unten skalieren können. Mit Auto Scaling können Sie sicherstellen, dass die Anzahl der von Ihnen verwendeten EC2-Instances während der Bedarfsspitzen nahtlos skaliert wird, um die Leistung zu erhalten, und während der Bedarfswartezeiten automatisch herunterskaliert wird, um die Kosten zu minimieren. Auto Scaling eignet sich besonders gut für Anwendungen, die stündliche, tägliche oder wöchentliche Schwankungen aufweisen. Auto Scaling wird von CloudWatch aktiviert und ist ohne zusätzliche Kosten, die über die CloudWatch-Gebühren hinausgehen, verfügbar.

Schnittstellen

Amazon EC2 kann programmgesteuert über API, SDK oder die Konsole verbunden werden. Metriken für die Rechen-, Arbeitsspeicher- und Speichernutzung, Netzwerkverbrauch und Lese-/Schreibverkehr für Ihre Instances sind kostenlos über die Konsolen- oder CloudWatch-API-Vorgänge.

Die Schnittstellen für Ihre Big Data-Analyse-Software, die Sie auf Amazon EC2 ausführen, hängen von den Eigenschaften der von Ihnen ausgewählten Software ab.

Anti-Patterns

Amazon EC2 bietet die folgenden Anti-Patterns:

- **Managed Service** – Wenn es sich bei Ihrem Bedarf um ein Managed Service-Angebot handelt, bei dem Sie die Infrastrukturschicht und Administration aus den Big Data-Analysen abstrahieren, ist dieses Modell der Verwaltung Ihrer eigenen Analysesoftware auf Amazon EC2 möglicherweise nicht die richtige Wahl.
- **Mangel an Fachwissen oder Ressourcen** – Wenn Ihre Organisation keine Ressourcen oder Fachkenntnisse zur Installation und Verwaltung einer Hochverfügbarkeitsinstallation für das fragliche System zur Verfügung hat oder nicht ausgeben möchte, sollten Sie die AWS-Entsprechung wie Amazon EMR in Erwägung ziehen, DynamoDB, Amazon Kinesis Streams oder Amazon Redshift.

Big Data-Probleme in AWS lösen

In diesem Whitepaper haben wir einige Tools von AWS zur Analyse von Big Data untersucht. Dies ist ein guter Anhaltspunkt, wenn Sie mit dem Entwurf Ihrer Big Data-Anwendungen beginnen. Es gibt jedoch zusätzliche Aspekte, dieSie bei der Auswahl der richtigen Tools für Ihren speziellen Anwendungsfall berücksichtigen sollten. Im Allgemeinen hat jeder analytische Workload bestimmte Merkmale und Anforderungen, die vorgeben, welcher Tool zu verwenden ist, wie z. B.

- Wie schnell brauchen Sie Analyse-Ergebnisse? in Echtzeit, in Sekunden oder ist eine Stunde ein passender Zeitrahmen?
- Welchen Wert bieten diese Analysen Ihrer Organisation und welche Budgetbeschränkungen gibt es?
- Wie groß sind die Daten und wie hoch ist ihre Wachstumsrate?
- Wie sind die Daten strukturiert?
- Welche Integrationsfähigkeit haben Produzenten und Konsumenten?
- Wie viel Latenz ist zwischen Produzenten und Konsumenten akzeptabel?

- Wie hoch sind die Kosten für Ausfallzeiten oder wie verfügbar und langlebig muss die Lösung sein?
- Ist der Analyse-Workload konsistent oder elastisch?

Jede dieser Eigenschaften oder Anforderungen hilft Ihnen, die richtige Richtung für den Tool zu finden. In einigen Fällen können Sie Ihren Big Data-Analyse-Workload einfach auf der Grundlage einer Reihe von Anforderungen in einen der Services zuordnen. In den meisten Big Data-Analyse-Workloads der realen Welt gibt es jedoch viele verschiedene und manchmal widersprüchliche Merkmale und Anforderungen für denselben Datensatz.

Zum Beispiel können einige Ergebnismengen Echtzeitanforderungen haben, wenn ein Benutzer mit einem System interagiert, während andere Analysen täglich gestapelt und ausgeführt werden können. Diese unterschiedlichen Anforderungen über denselben Datensatz sollten entkoppelt und mit mehr als einem Tool gelöst werden. Wenn Sie versuchen, beide oben genannten Beispiele im selben Toolset zu lösen, führt dies entweder zu einer Überprovisionierung und damit zu unnötigen Antwortzeiten oder zu einer Lösung, die nicht schnell genug ist, um in Echtzeit auf Ihre Benutzer zu reagieren. Durch die Anpassung des am besten geeigneten Tools an jedem einzelnen Analyse-Problem wird die kosteneffizienteste Nutzung Ihrer Rechen- und Speicherressourcen erreicht.

Big Data muss nicht „große Kosten“ bedeuten. Wenn Sie Ihre Anwendungen entwerfen, ist es daher wichtig, dass Ihr Design kosteneffizient ist. Wenn es nicht entsprechend zu den Alternativen ist, dann ist es wahrscheinlich nicht das richtige Design. Ein weiteres häufiges Missverständnis ist, dass mehrere Tools zur Lösung eines großen Datenproblems teurer oder schwieriger zu verwalten sind als ein großes Tool. Wenn Sie das gleiche Beispiel für zwei unterschiedliche Anforderungen für denselben Datensatz verwenden, ist die Echtzeitanforderung möglicherweise zu niedrig für die CPU, aber hoch zu Ein-/Ausgaben, während die langsamere Verarbeitungsanforderung sehr rechenintensiv sein kann. Die Entkopplung kann sehr viel kostengünstiger und einfacher zu verwalten sein, da Sie jeden Tool nach genauen Spezifikationen und nicht zu hoher Verfügbarkeit erstellen können. Mit dem AWS-Lohnabzugsverfahren und nur für das, was Sie als Servicemodell für die Infrastruktur verwenden, ist dies ein viel besserer Wert, da Sie die Stapelanalyse in nur einer Stunde ausführen und somit nur die Rechenressourcen für diese Stunde bezahlen müssen. Außerdem ist es möglicherweise einfacher, diesen Ansatz zu verwalten, anstatt ein einzelnes

System zu verwenden, das alle Anforderungen erfüllt. Die Lösung für unterschiedliche Anforderungen mit einem Tool führt dazu, dass versucht wird, einen quadratischen Rahmen (Echtzeitanforderungen) in ein rundes Loch (ein großes Data-Warehouse) einzupassen.

Die AWS-Plattform erleichtert das Entkoppeln Ihrer Architektur, indem verschiedene Tools denselben Datensatz analysieren. AWS-Services verfügen über eine eingebaute Integration, sodass das Verschieben einer Teilmenge von Daten von einem Tool zu einem anderen mithilfe der Parallelisierung sehr einfach und schnell erfolgen kann. Lassen Sie uns dies in die Praxis umsetzen, indem wir ein paar echte Big Data-Analyse-Problemszenarien untersuchen und eine AWS-Architekturlösung durchgehen.

Beispiel 1: Enterprise-Data-Warehouse

Ein multinationaler Bekleidungshersteller hat mehr als tausend Einzelhandelsgeschäfte, verkauft bestimmte Linien über Warenhäuser und Discounter und ist online präsent. Derzeit sind diese drei Kanäle unabhängig von einem technischen Standpunkt. Sie haben unterschiedliche Verwaltungs-, Point-of-Sale-Systeme und Buchhaltungsabteilungen. Es gibt kein System, das all diese Datensätze zusammenführt, um dem CEO Einblicke in das gesamte Unternehmen zu ermöglichen. Der CEO möchte ein unternehmensweites Bild seiner Kanäle haben und bei Bedarf Ad-hoc-Analysen durchführen können. Beispielanalysen, die das Unternehmen möchte, sind:

- Welche Trends gibt es in allen Kanälen?
- Welche geografischen Regionen schneiden besser über die Kanäle ab?
- Wie effektiv sind ihre Anzeigen und Gutscheine?
- Welche Trends gibt es in jeder Bekleidungslinie?
- Welche externen Kräfte können sich auf den Umsatz auswirken, zum Beispiel die Arbeitslosenquote oder das Wetter?
- Wie wirkt sich das Speichern von Attributseffekten auf den Verkauf aus, zum Beispiel Besitzdauer von Angestellten/Management, Einkaufszentrum im Vergleich zu einem geschlossenen Einkaufszentrum, Standort von Waren im Geschäft, Werbung, Endkappen, Verkaufsrunden, Ladenanzeigen usw.?

Ein Enterprise-Data-Warehouse ist eine hervorragende Möglichkeit, dieses Problem zu lösen. Das Data-Warehouse muss Daten von jedem der verschiedenen Systeme der drei Kanäle und von öffentlichen Aufzeichnungen für Wetter- und Wirtschaftsdaten sammeln. Jede Datenquelle sendet täglich Daten für den Verbrauch durch das Data-Warehouse. Da jede Datenquelle unterschiedlich strukturiert sein kann, wird ein Extraktions-, Transformations- und Ladeprozess (ETL) durchgeführt, um die Daten in eine gemeinsame Struktur umzuformatieren. Dann können Analysen über Daten aus allen Quellen gleichzeitig durchgeführt werden. Dazu verwenden wir die folgende Datenflussarchitektur:

Enterprise-Data-Warehouse

1. Der erste Schritt in diesem Prozess besteht darin, die Daten aus den vielen verschiedenen Quellen auf Amazon S3 zu übertragen. Amazon S3 wurde ausgewählt, weil es sich um eine sehr langlebige, kostengünstige und skalierbare Speicherplattform handelt, die parallel aus vielen verschiedenen Quellen zu sehr niedrigen Kosten geschrieben werden kann.
2. Amazon EMR wird verwendet, um die Daten vom Quellformat in das Ziel und ein Format umzuwandeln und zu bereinigen. Amazon EMR verfügt über eine eingebaute Integration mit Amazon S3, um parallele Durchsatz-Threads von jedem Knoten in Ihrem Cluster zu und von Amazon S3 zu ermöglichen. Typischerweise erhalten Data-Warehouses jede Nacht neue Daten aus ihren vielen verschiedenen Quellen. Da diese Analysen nicht mitten in der Nacht benötigt werden, besteht die einzige Anforderung in diesem Transformationsprozess darin, dass sie bis zum Morgen fertig ist, wenn der CEO und andere Geschäftsbenutzer Ergebnisse benötigen. Diese Anforderung bedeutet, dass Sie den Amazon EC2 Spot-Markt[67] nutzen können, um die Transformationskosten weiter zu senken. Eine gute Spot-Strategie könnte darin bestehen, um Mitternacht zu einem sehr

niedrigen Preis zu bieten und Ihren Preis im Laufe der Zeit kontinuierlich zu erhöhen, bis die Kapazität vergeben ist. Je näher Sie der Frist kommen, wenn die Spot-Gebote nicht erfolgreich waren, können Sie auf On-Demand-Preise zurückgreifen, um sicherzustellen, dass Sie Ihre Anforderungen an die Abschlusszeit erfüllen. Jede Quelle kann einen anderen Transformationsprozess auf Amazon EMR haben, aber mit dem bedarfsorientierten Modell von AWS können Sie für jede Transformation einen separaten Amazon EMR-Cluster erstellen und so abstimmen, dass alle Datentransformationsjobs ausgeführt werden können für den niedrigsten möglichen Preis, ohne mit den Mitteln der anderen Jobs zu konkurrieren.

3. Jeder Transformationsauftrag legt dann die formatierten und bereinigten Daten auf Amazon S3 ab. Amazon S3 wird hier erneut verwendet, da Amazon Redshift diese Daten für mehrere Threads von jedem Knoten parallel verarbeiten kann. Dieser Speicherort auf Amazon S3 dient auch als eine historische Aufzeichnung und ist die formatierte Quelle der Wahrheit zwischen Systemen. Daten zu Amazon S3 können von anderen Tools für Analysen verwendet werden, wenn im Laufe der Zeit zusätzliche Anforderungen eingeführt werden.
4. Amazon Redshift lädt, sortiert, verteilt und komprimiert die Daten in ihre Tabellen, sodass Analyse-Abfragen effizient und parallel ausgeführt werden können. Amazon Redshift wurde für Data-Warehouse-Workloads entwickelt und kann problemlos durch Hinzufügen eines weiteren Knotens erweitert werden, wenn die Datengröße im Laufe der Zeit zunimmt und das Unternehmen expandiert.
5. Zur Visualisierung der Analysen kann Amazon QuickSight oder eine der vielen Partner-Visualisierungsplattformen über die OBDC/JDBC-Verbindung zu Amazon Redshift verwendet werden. Hier können die Berichte und Grafiken vom CEO und seinen Mitarbeitern eingesehen werden. Diese Daten können nun von den Führungskräften genutzt werden, um bessere Entscheidungen über die Unternehmensressourcen zu treffen, was letztlich die Erträge und den Wert für die Aktionäre steigern kann.

Diese Architektur ist sehr flexibel und kann leicht erweitert werden, wenn das Unternehmen expandiert, mehr Datenquellen importiert werden, neue Kanäle geöffnet werden oder eine mobile Anwendung mit kundenspezifischen Daten gestartet wird. Zu jeder Zeit können zusätzliche Tools integriert und

das Lager mit wenigen Klicks skaliert werden, indem die Anzahl der Knoten im Amazon Redshift-Cluster erhöht wird.

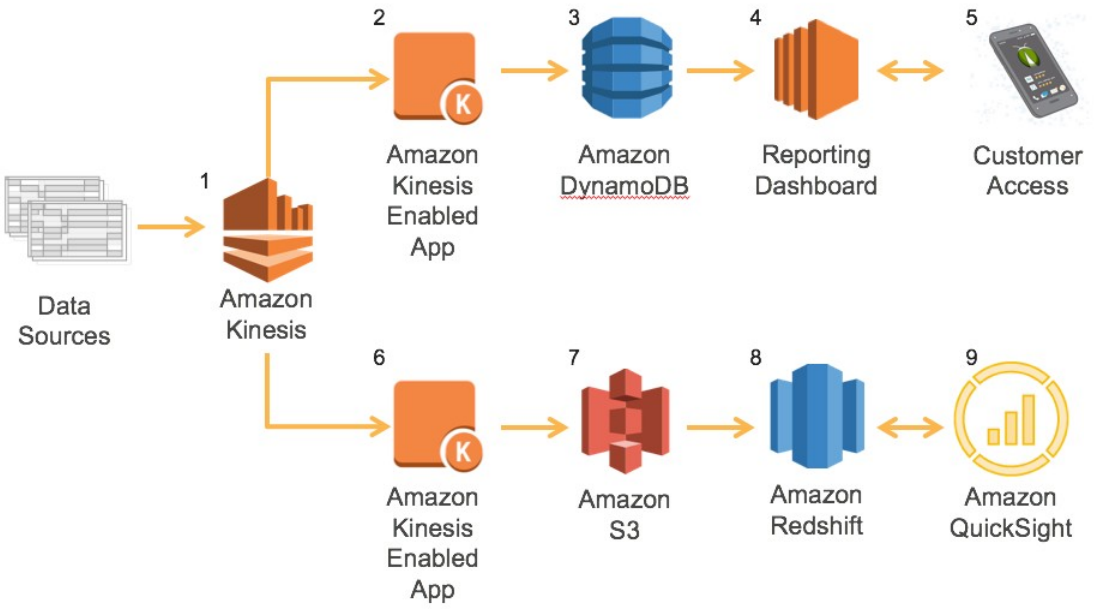
Beispiel 2: Erfassen und Analysieren von Sensordaten

Ein internationaler Klimaanlagenhersteller hat viele große Klimaanlagen, die er an verschiedene kommerzielle und industrielle Unternehmen verkauft. Sie verkaufen nicht nur die Klimageräte, sondern bieten, um sich besser als ihre Mitbewerber zu positionieren, auch zusätzliche Services an, bei denen Sie Echtzeit-Dashboards in einer mobilen Anwendung oder einem Webbrowser sehen können. Jede Einheit sendet ihre Sensorinformationen zur Verarbeitung und Analyse. Diese Daten werden vom Hersteller und seinen Kunden verwendet. Mit dieser Fähigkeit kann der Hersteller den Datensatz visualisieren und Trends erkennen.

Derzeit haben sie ein paar tausend im Voraus gekaufte Einheiten mit dieser Fähigkeit. Sie erwarten, dass sie diese in den nächsten Monaten an Kunden liefern werden, und hoffen, dass mit der Zeit Tausende von Einheiten auf der ganzen Welt diese Plattform nutzen werden. Wenn sie erfolgreich sind, möchten sie dieses Angebot auch auf ihre Verbraucherlinie mit einem viel größeren Volumen und einem größeren Marktanteil erweitern. Die Lösung muss in der Lage sein, große Datenmengen zu bewältigen und skalierbar zu sein, wenn sie ihr Geschäft ohne Unterbrechung ausbauen. Wie sollten Sie ein solches System entwerfen? Teilen Sie zuerst zwei Arbeits-Streams auf, die beide aus denselben Daten stammen:

- Aktuelle Informationen der Klimaanlage mit nahezu Echtzeitanforderungen und eine große Anzahl von Kunden, die diese Information konsumieren.
- Alle historischen Informationen zu den Klimaanlagen für Trends und Analysen für den internen Gebrauch.

Das folgende ist die Datenflussarchitektur, um dieses Big Data-Problem zu lösen:



Erfassen und Analysieren von Sensordaten

1. Der Prozess beginnt damit, dass jedes Klimagerät einen konstanten Datenstrom an Amazon Kinesis Streams liefert. Dies bietet eine elastische und dauerhafte Schnittstelle, mit der die Einheiten kommunizieren können, die nahtlos skaliert werden kann, da immer mehr Klimaanlagen verkauft und online gebracht werden.
2. Mit den von Amazon Kinesis Streams bereitgestellten Tools wie der Kinesis Client Library oder dem SDK wird eine einfache Anwendung auf Amazon EC2 erstellt, um Daten in Amazon Kinesis Streams auszulesen, zu analysieren und festzustellen, ob die Daten eine Aktualisierung auf das Echtzeit-Dashboard garantieren. Es sucht nach Änderungen im Systembetrieb, Temperaturschwankungen und allen Fehlern, auf die die Einheiten stoßen.
3. Dieser Datenfluss muss nahezu in Echtzeit erfolgen, damit Kunden und Wartungsteams bei einem Problem mit der Einheit so schnell wie möglich benachrichtigt werden können. Die Daten im Dashboard enthalten zwar einige zusammengefasste Trendinformationen, es handelt sich jedoch hauptsächlich um den aktuellen Status sowie um Systemfehler. Daher sind die Daten zum Füllen des Dashboards relativ klein. Darüber hinaus wird es aus den folgenden Quellen viel potenziellen Zugriff auf diese Daten geben:

- Kunden überprüfen ihr System über ein mobiles Gerät oder einen Browser
- Wartungsteams überprüfen den Status ihrer Flotte
- Daten und intelligente Algorithmen und Analysen in der Berichtsplattform erkennen Trends, die dann als Warnungen ausgegeben werden können, z. B. wenn der Klimaanlage-Lüfter ungewöhnlich lange läuft und die Gebäudetemperatur nicht sinkt.

DynamoDB wurde ausgewählt, um diesen nahezu Echtzeitdatensatz zu speichern, da er sowohl hoch verfügbar als auch skalierbar ist. Der Durchsatz zu diesen Daten kann leicht nach oben oder unten skaliert werden, um die Bedürfnisse seiner Verbraucher zu erfüllen, wenn die Plattform angenommen wird und die Nutzung zunimmt.

4. Das Berichts-Dashboard ist eine benutzerdefinierte Webanwendung, die auf diesem Datensatz erstellt wurde und auf Amazon EC2 ausgeführt wird. Es stellt Inhalte basierend auf dem Systemstatus und den Trends zur Verfügung und alarmiert Kunden und Wartungsteams über alle Probleme, die mit der Einheit auftreten können.
5. Der Kunde greift auf die Daten von einem mobilen Gerät oder einem Webbrowser zu, um den aktuellen Status des Systems zu erhalten und historische Trends zu visualisieren.

Der Datenfluss (Schritte 2-5), der gerade beschrieben wurde, ist für eine nahezu Echtzeit-Berichterstattung von Informationen an menschliche Konsumenten aufgebaut. Es ist für niedrige Latenzzeiten errichtet und entworfen und kann sehr schnell skaliert werden, um den Bedarf zu decken. Der Datenfluss (Schritte 6-9), der im unteren Teil des Diagramms dargestellt ist, hat keine so strengen Geschwindigkeits- und Latenzanforderungen. Dies ermöglicht es dem Architekten, einen anderen Lösungsspaket zu entwerfen, der größere Datenmengen zu wesentlich geringeren Kosten pro Informationsbyte aufnehmen und weniger teure Rechen- und Speicherressourcen auswählen kann.

6. Um aus dem Amazon Kinesis-Stream zu lesen, gibt es eine separate Amazon Kinesis-fähige Anwendung, die wahrscheinlich auf einer kleineren EC2-Instance läuft, die langsamer skaliert. Während diese Anwendung den gleichen Datensatz wie der obere Datenfluss analysiert, besteht der letztendliche Zweck dieser Daten darin, sie für eine langfristige Aufzeichnung zu speichern und den Datensatz in einem Data-Warehouse zu hosten. Dieser Datensatz besteht letztendlich aus allen Daten, die von den

Systemen gesendet werden, und ermöglicht es, einen viel breiteren Satz von Analysen ohne die Anforderungen in Echtzeit durchzuführen.

7. Die Daten werden von der Amazon Kinesis-fähigen Anwendung in ein Format umgewandelt, das für die Langzeitspeicherung, das Laden in dasData-Warehouse und das Speichern auf Amazon S3 geeignet ist. Die Daten auf Amazon S3 dienen nicht nur als parallele Aufnahmepunkte für Amazon Redshift, sondern sind ein dauerhafter Speicher, der alle Daten enthält, die jemals durch dieses System laufen. es kann die einzige Quelle der Wahrheit sein. Es kann verwendet werden, um andere Analyse-Tools zu laden, wenn zusätzliche Anforderungen entstehen. Amazon S3 bietet außerdem eine native Integration mit Amazon Glacier, wenn Daten in einen langfristigen, kostengünstigen Kühlspeicher übertragen werden müssen.
8. Amazon Redshift wird erneut als Data-Warehouse für den größeren Datensatz verwendet. Es kann leicht skaliert werden, wenn der Datensatz größer wird, indem ein weiterer Knoten zum Cluster hinzugefügt wird.
9. Zur Visualisierung der Analysen kann eine der vielen Partner-Visualisierungsplattformen über die OBDC/JDBC-Verbindung zu Amazon Redshift genutzt werden. Hier können die Berichte, Grafiken und Ad-hoc-Analysen für den Datensatz durchgeführt werden, um bestimmte Variablen und Trends zu finden, die dazu führen können, dass Klimaanlagen leistungsschwach Arbeiten oder sich ausschalten.

Diese Architektur kann klein anfangen und nach Bedarf wachsen. Durch die Entkopplung der beiden unterschiedlichen Arbeitsströme können sie je nach Bedarf ohne Vorabverpflichtung mit ihrer eigenen Rate wachsen, so dass der Hersteller den Erfolg oder Misserfolg dieses neuen Angebots ohne große Investitionen beurteilen kann. Sie können sich leicht weitere Ergänzungen wie Amazon ML vorstellen, um genau vorhersagen zu können, wie lange eine Klimaanlage dauert und präventiv Wartungsteams basierend auf ihren Prognose-Algorithmen aussenden, um ihren Kunden den bestmöglichen Service und die beste Erfahrung zu bieten. Dieses Serviceniveau würde sich von der Konkurrenz abheben und zu höheren zukünftigen Verkäufen führen.

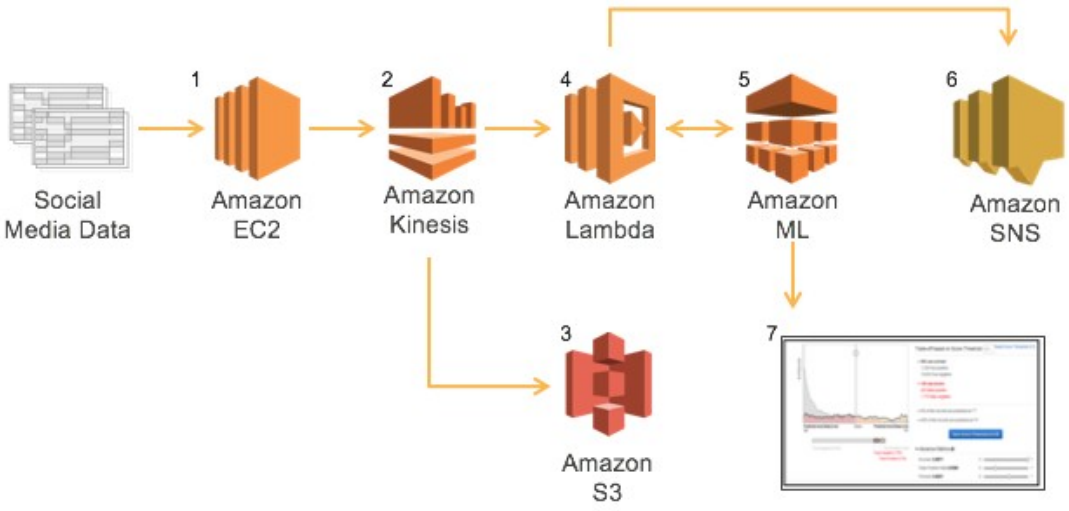
Beispiel 3: Meinungsanalyse von sozialen Medien

Ein großer Spielzeughersteller ist sehr schnell gewachsen und hat seine Produktpalette erweitert. Nach jeder neuen Spielzeugversion möchte

das Unternehmen verstehen, wie Konsumenten ihre Produkte genießen und verwenden. Darüber hinaus möchte das Unternehmen sicherstellen, dass seine Verbraucher eine gute Erfahrung mit ihren Produkten haben. Mit dem Wachstum des Spielzeug-Ökosystems möchte das Unternehmen sicherstellen, dass seine Produkte für seine Kunden immer noch relevant sind und dass sie zukünftige Leitpläne basierend auf Kunden-Feedback planen können. Das Unternehmen möchte Folgendes aus den sozialen Medien erfassen:

- Verstehen Sie, wie Verbraucher ihre Produkte verwenden
- Gewährleistung der Kundenzufriedenheit
- Planen Sie zukünftige Leitpläne

Das Erfassen der Daten aus verschiedenen sozialen Netzwerken ist relativ einfach, aber die Herausforderung besteht darin, die Informationen programmatisch aufzubauen. Nach der Aufnahme der Daten möchte das Unternehmen die Daten kostengünstig und programmatisch analysieren und klassifizieren können.
Zu diesem Zweck kann folgende Architektur verwendet werden:



Meinungsanalyse von sozialen Medien

1. Das erste, was zu tun ist, entscheiden Sie, auf welchen Social Media-Sites zu hören. Erstellen Sie dann eine Anwendung, die diese Websites über ihre entsprechenden APIs abfragt und auf Amazon EC2 ausführt.

2. Als Nächstes wird ein Amazon Kinesis-Stream erstellt, da wir möglicherweise mehrere Datenquellen haben: Twitter, Tumblr und so weiter. Auf diese Weise kann für jede hinzugefügte neue Datenquelle ein neuer Stream erstellt werden, und Sie können den vorhandenen Anwendungscode und die vorhandene Architektur verwenden. Darüber hinaus wird in diesem Beispiel ein neuer Amazon Kinesis-Stream erstellt, um die Rohdaten in Amazon S3 zu kopieren.
3. Zur Archivierung, Langzeitanalyse und historischen Referenz werden Rohdaten in Amazon S3 gespeichert. Weitere Amazon ML-Stapelmodelle können anhand von Daten in Amazon S3 ausgeführt werden, um eine vorausschauende Analyse durchzuführen und die Kauftrends der Verbraucher zu verfolgen.
4. Wie im Architekturdiagramm angegeben, wird Lambda für die Verarbeitung und Normalisierung der Daten und die Abfrage von Prognosen von Amazon ML verwendet. Nachdem die Amazon ML-Prognose zurückgegeben wurde, kann die Lambda-Funktion basierend auf der Vorhersage Maßnahmen ergreifen, z. B. um einen Social Media-Beitrag zur weiteren Überprüfung an das Kundendienstteam weiterzuleiten.
5. Amazon ML wird verwendet, um Prognosen für die Eingabedaten zu treffen. Zum Beispiel kann ein ML-Modell erstellt werden, um einen Social-Media-Kommentar zu analysieren, um festzustellen, ob der Kunde eine negative Meinung zu einem Produkt geäußert hat. Um genaue Vorhersagen mit Amazon ML zu erhalten, beginnen Sie mit Trainingsdaten und stellen Sie sicher, dass Ihre ML-Modelle ordnungsgemäß funktionieren. Wenn Sie ML-Modelle zum ersten Mal erstellen, lesen Sie im Tutorial: Verwendung von Amazon ML, zur Prognose von Antworten auf ein Marketing-Angebot.[68] Wie bereits erwähnt, wird bei Verwendung mehrerer Datenquellen für soziale Netzwerke ein anderes ML-Modell vorgeschlagen, um die Prognosegenauigkeit sicherzustellen.
6. Abschließend werden umsetzbare Daten mithilfe von Lambda an Amazon SNS gesendet und zur weiteren Untersuchung per Text oder E-Mail an die entsprechenden Ressourcen übermittelt.
7. Im Rahmen der Meinungsanalyse ist die Erstellung eines Amazon ML-Modells, das regelmäßig aktualisiert wird, unerlässlich für genaue Ergebnisse. Zusätzliche Metriken für ein bestimmtes Modell können über die Konsole grafisch angezeigt werden, z. B.: Genauigkeit, falsche positive Rate, Präzision

und Abruf. Weitere Informationen finden Sie unter Schritt 4: Überprüfung der Prognoseleistung des ML-Modells und Abschaltung festlegen.[69]

Mit einer Kombination aus Amazon Kinesis Streams, Lambda, Amazon ML und Amazon SES haben wir eine skalierbare und leicht anpassbare Social-Listening-Plattform geschaffen. Es ist wichtig zu beachten, dass dieses Bild nicht die Erstellung eines ML-Modells darstellt. Dieser Akt wird mindestens einmal durchgeführt, normalerweise aber regelmäßig, um das Modell auf dem neuesten Stand zu halten. Die Häufigkeit, mit der ein neues Modell erstellt wird, hängt vom Workload ab und wird nur durchgeführt, um das Modell genauer zu machen, wenn sich die Dinge ändern.

Fazit

Da immer mehr Daten generiert und gesammelt werden, erfordert die Datenanalyse skalierbare, flexible und leistungsstarke Tools, um rechtzeitig Erkenntnisse zu liefern. Unternehmen sehen sich jedoch einem wachsenden Big Data-Ökosystem gegenüber, in dem neue Tools entstehen und sehr schnell „sterben“. Daher kann es sehr schwierig sein, Schritt zu halten und die richtigen Toos zu wählen.

Dieses Whitepaper bietet einen ersten Schritt zur Lösung dieser Herausforderung. Mit einer breiten Palette von Managed Services zum Sammeln, Verarbeiten und Analysieren von Big Data vereinfacht die AWS-Plattform das Erstellen, Bereitstellen und Skalieren von Big Data-Anwendungen, sodass Sie sich auf Geschäftsprobleme konzentrieren können, anstatt diese Tools zu aktualisieren und zu verwalten.

AWS bietet viele Lösungen für die Analyse von Big Data-Anforderungen. Die meisten Big Data-Architekturlösungen verwenden mehrere AWS-Tools, um eine vollständige Lösung zu erstellen: Dies kann dabei helfen, die strengen Geschäftsanforderungen möglichst kostenoptimiert, performant und ausfallsicher zu erfüllen. Das Ergebnis ist eine flexible Big Data-Architektur, die in der globalen AWS-Infrastruktur mit Ihrem Unternehmen skalierbar ist.

Mitwirkende

Dieses Dokument ist unter der Mitarbeit folgender Personen und Organisationen entstanden:

- Erik Swensson, Manager, Lösungsarchitektur, Amazon Web Services
- Erick Dame, Lösungsarchitekt, Amazon Web Services
- Shree Kenghe, Lösungsarchitekt, Amazon Web Services

Weitere Informationen

Mithilfe der folgenden Ressourcen können Sie mit der Durchführung von Big Data-Analysen auf AWS beginnen:

- Besuchen Sie aws.amazon.com/de/big-data[70]

  Sehen Sie sich das umfangreiche Portfolio an Big Data-Services sowie Links zu anderen Ressourcen wie AWS Big Data-Partnern, Tutorials, Artikeln und AWS Marketplace -Angeboten zu Big Data-Lösungen an. Kontaktieren Sie uns wenn Sie Hilfe benötigen.

- Lesen Sie den AWS Big Data-Blog[71]

  Der Blog enthält Beispiele und Ideen aus dem echten Leben, die regelmäßig aktualisiert werden, um Sie beim Sammeln, Speichern, Bereinigen, Verarbeiten und Visualisieren von Big Data zu unterstützen.

- Probieren Sie einen der Big Data-Tests[72]

  Erkunden Sie das reichhaltige Ökosystem von Produkten, die mit AWS auf große Datenherausforderungen ausgerichtet sind. Tests werden von Partnern des AWS Partner Network (APN) für Beratung und Technologie entwickelt und werden kostenlos für Ausbildungs-, Demonstrations- und Evaluierungszwecke bereitgestellt.

- Nehmen Sie Teil an einem AWS-Schulungskurs zu Big Data[73]

Der Kurs von Big Data on AWS stellt Ihnen Cloud-basierte Big Data-Lösungen und Amazon EMR vor. Wir zeigen Ihnen, wie Sie Amazon EMR verwenden, um Daten mit dem breiten Ökosystem von Hadoop-Tools wie Pig und Hive zu verarbeiten. Wir zeigen Ihnen auch, wie Sie Big Data-Umgebungen erstellen, mit DynamoDB und Amazon Redshift arbeiten, die Vorteile von Amazon Kinesis Streams kennen und bewährte Praktiken für die Entwicklung von Big Data-Umgebungen für Sicherheit und Kosteneffizienz nutzen.

- Siehe unter Fallstudien zu Big Data-Kunden[74]

  Lernen Sie von den Erfahrungen anderer Kunden, die leistungsstarke und erfolgreiche Big Data-Plattformen in der AWS-Cloud entwickelt haben.

Am Dokument vorgenommene Änderungen

Januar 2016	Verändert, um Informationen zu Amazon Machine Learning, AWS Lambda und Amazon Elasticsearch Service hinzuzufügen; allgemeine Aktualisierung
Dezember 2014	Erstveröffentlichung

Anmerkungen

[1] http://aws.amazon.com/about-aws/globalinfrastructure/

[2] http://aws.amazon.com/s3/

[3] http://aws.amazon.com/datapipeline/

[4] https://aws.amazon.com/iot/

[5] https://aws.amazon.com/importexport/

[6] http://aws.amazon.com/kinesis/firehose

[7] https://aws.amazon.com/directconnect/

[8] https://aws.amazon.com/mobile/

[9] http://aws.amazon.com/solutions/case-studies/big-data/

[10] https://aws.amazon.com/kinesis/streams

[11] http://docs.aws.amazon.com/kinesis/latest/APIReference/Welcome.html

[12] http://docs.aws.amazon.com/aws-sdk-php/v2/guide/service-kinesis.html

[13] http://aws.amazon.com/kinesis/pricing/

[14] http://aws.amazon.com/tools/

[15] http://docs.aws.amazon.com/kinesis/latest/dev/developing-producers-with-kpl.html

[16] http://docs.aws.amazon.com/kinesis/latest/dev/writing-with-agents.html

[17] https://github.com/awslabs/amazon-kinesis-client

[18] https://github.com/awslabs/kinesis-storm-spout

[19] https://aws.amazon.com/lambda/

[20] http://docs.aws.amazon.com/lambda/latest/dg/intro-core-components.html

[21] https://aws.amazon.com/amazon-linux-ami/

[22] http://docs.aws.amazon.com/lambda/latest/dg/nodejs-create-deployment-pkg.html

[23] http://docs.aws.amazon.com/lambda/latest/dg/lambda-python-how-to-create-deployment-package.html

[24] http://docs.aws.amazon.com/lambda/latest/dg/lambda-java-how-to-create-deployment-package.html

[25] http://aws.amazon.com/elasticmapreduce/

[26] https://media.amazonwebservices.com/AWS_Amazon_EMR_Best_Practices. pdf

[27] http://aws.amazon.com/elasticmapreduce/pricing/

[28] http://aws.amazon.com/ec2/instance-types/

[29] http://aws.amazon.com/elasticmapreduce/mapr/

[30] http://docs.aws.amazon.com/ElasticMapReduce/latest/DeveloperGuide/emr-manage-resize.html

[31] http://docs.aws.amazon.com/ElasticMapReduce/latest/API/Welcome.html

[32] http://docs.aws.amazon.com/ElasticMapReduce/latest/ReleaseGuide/emr-hive.html

[33] http://docs.aws.amazon.com/ElasticMapReduce/latest/ReleaseGuide/emr-pig.html

[34] http://blogs.aws.amazon.com/bigdata/post/Tx15AY5C50K70RV/Installing-Apache-Spark-on-an-Amazon-EMR-Cluster

[35] http://docs.aws.amazon.com/ElasticMapReduce/latest/DeveloperGuide/emr-hbase.html

[36] http://docs.aws.amazon.com/ElasticMapReduce/latest/DeveloperGuide/emr-impala.html

[37] http://aws.amazon.com/elasticmapreduce/hunk/

[38] http://docs.aws.amazon.com/ElasticMapReduce/latest/DeveloperGuide/UsingEMR_s3distcp.html

[39] https://aws.amazon.com/machine-learning/

[40] https://aws.amazon.com/machine-learning/pricing/

[41] http://docs.aws.amazon.com/machine-learning/latest/dg/suggested-recipes.html

[42] http://docs.aws.amazon.com/machine-learning/latest/APIReference/Welcome.html

[43] https://aws.amazon.com/dynamodb

[44] http://aws.amazon.com/free/

[45] http://aws.amazon.com/dynamodb/pricing/

[46] Einstellige Millisekunden, typisch für durchschnittliche Antwortzeiten auf der Serverseite

[47] http://docs.aws.amazon.com/amazondynamodb/latest/developerguide/Streams.html

[48] Mit DynamoDB können Sie Ihren bereitgestellten Durchsatzlevel um bis zu 100 % durch einen einzigen UpdateTable-API-Betriebsanruf ändern. Wenn Sie Ihren Durchsatz um mehr als 100 % ändern möchten, rufen Sie den UpdateTable erneut auf.

[49] Sie können Ihren bereitgestellten Durchsatz so oft erhöhen, wie Sie möchten. Allerdings gibt es eine Begrenzung von zwei Abnahmen pro Tag.

[50] https://aws.amazon.com/redshift/

[51] http://aws.amazon.com/s3/pricing/

[52] http://aws.amazon.com/redshift/pricing/

[53] http://docs.aws.amazon.com/AWSEC2/latest/UserGuide/using-regions-availability-zones.html

[54] http://docs.aws.amazon.com/redshift/latest/APIReference/Welcome.html

[55] http://docs.aws.amazon.com/redshift/latest/mgmt/working-with-clusters.html#rs-about-clusters-and-nodes

[56] http://docs.aws.amazon.com/redshift/latest/dg/c_redshift-and-postgres- sql.html

[57] http://aws.amazon.com/redshift/partners/

[58] https://aws.amazon.com/elasticsearch-service/

[59] https://aws.amazon.com/ec2/pricing/

[60] https://aws.amazon.com/ebs/details/

[61] https://aws.amazon.com/elasticsearch-service/pricing/

[62] https://aws.amazon.com/elasticsearch-service/faqs/

[63] https://aws.amazon.com/quicksight

[64] https://aws.amazon.com/ec2/

[65] https://aws.amazon.com/marketplace

[66] http://aws.amazon.com/autoscaling/

[67] http://aws.amazon.com/ec2/spot/

[68] http://docs.aws.amazon.com/machine-learning/latest/dg/tutorial.html

[69] http://docs.aws.amazon.com/machine-learning/latest/dg/step-4-review-the-ml-model-predictive-performance-and-set-a-cut-off.html

[70] http://aws.amazon.com/big-data

[71] http://blogs.aws.amazon.com/bigdata/

[72] https://aws.amazon.com/testdrive/bigdata/

[73] http://aws.amazon.com/training/course-descriptions/bigdata/

[74] http://aws.amazon.com/solutions/case-studies/big-data/