Anwendung des AWS Well-Architected Frameworks für Amazon Neptune

AWS Präskriptive Leitlinien

Copyright © 2025 Amazon Web Services, Inc. and/or its affiliates. All rights reserved.

AWS Präskriptive Leitlinien: Anwendung des AWS Well-Architected Frameworks für Amazon Neptune

Copyright © 2025 Amazon Web Services, Inc. and/or its affiliates. All rights reserved.

Die Handelsmarken und Handelsaufmachung von Amazon dürfen nicht in einer Weise in Verbindung mit nicht von Amazon stammenden Produkten oder Services verwendet werden, durch die Kunden irregeführt werden könnten oder Amazon in schlechtem Licht dargestellt oder diskreditiert werden könnte. Alle anderen Handelsmarken, die nicht Eigentum von Amazon sind, gehören den jeweiligen Besitzern, die möglicherweise zu Amazon gehören oder nicht, mit Amazon verbunden sind oder von Amazon gesponsert werden.

Table of Contents

Einführung ... 1
Zielgruppe ... 1
Ziele ... 2
Operative Exzellenz ... 3
Automatisieren Sie die Bereitstellung mithilfe eines IaC-Ansatzes ... 3
Nehmen Sie häufig kleine, umkehrbare Änderungen vor ... 4
Rechnen Sie mit Ausfällen ... 4
Lernen Sie aus allen Betriebsausfällen ... 5
Verwenden Sie Protokollierungsfunktionen, um unbefugte oder ungewöhnliche Aktivitäten zu überwachen ... 6
Sicherheit ... 7
Implementieren Sie Datensicherheit ... 8
Schützen Sie Ihre Netzwerke ... 9
Implementieren Sie Authentifizierung und Autorisierung ... 9
Zuverlässigkeit ... 11
Neptune-Dienstkontingente verstehen ... 11
Verstehen Sie die Einsatzmuster von Neptune ... 12
Neptun-Cluster verwalten und skalieren ... 13
Backups und Failover-Ereignisse verwalten ... 14
Leistungseffizienz ... 15
Verstehen Sie die Graphmodellierung ... 15
Optimieren Sie Abfragen ... 16
Cluster in der richtigen Größe ... 18
Optimize schreibt ... 19
Kostenoptimierung ... 21
Verstehen Sie die Nutzungsmuster und die benötigten Dienste ... 21
Wählen Sie Ressourcen unter Berücksichtigung der Kosten aus ... 22
Wählen Sie die beste Neptune-Instanzkonfiguration für Ihren Workload ... 23
Datenspeicherung und -übertragung in der richtigen Größe ... 25
Nachhaltigkeit ... 26
Auswahl der AWS-Region ... 26
Der Konsum basiert auf Verhaltensmustern der Nutzer ... 27
Optimieren Sie die Softwareentwicklung und Architekturmuster ... 27
Ressourcen ... 29

Mitwirkende 30
Dokumentverlauf 31
Glossar 32
..... 32
A 33
B 36
C 38
D 41
E 46
F 48
G 50
H 51
I 53
L 55
M 56
O 61
P 64
Q 67
R 67
S 70
T 74
U 76
V 77
W 77
Z 78
..... lxxx

Anwendung des AWS Well-Architected Frameworks für Amazon Neptune

Amazon Web Services (Mitwirkende)

September 2023 (Verlauf der Dokumente)

Sie können graphbasierte Lösungen auf Amazon Web Services (AWS) mithilfe von Amazon Neptune erstellen. Dieser Leitfaden enthält Anleitungen zur Anwendung der Prinzipien des AWS Well-Architected Framework bei der Planung Ihrer Neptune-Bereitstellung. Die Anwendung des AWS Well-Architected Framework für Amazon Neptune Analytics behandelt dasselbe Thema für die Neptune Graph Analytics-Engine.

Das AWS Well-Architected Framework unterstützt Sie beim Aufbau sicherer, leistungsstarker, belastbarer und effizienter Infrastrukturen für eine Vielzahl von Anwendungen und Workloads. Es bietet Ihnen auch einen konsistenten Ansatz zur Bewertung von Architekturen und zur Implementierung skalierbarer Designs.

Das AWS Well-Architected Framework basiert auf den folgenden sechs Säulen:

- Operative Exzellenz
- Sicherheit
- Zuverlässigkeit
- Leistungseffizienz
- Kostenoptimierung
- Nachhaltigkeit

Dieses Handbuch enthält Informationen zu den Grundpfeilern und bewährten Methoden des Well-Architected Framework-Designs sowie Überlegungen, die Sie bei der Bereitstellung von Neptune berücksichtigen sollten. AWS

Zielgruppe

Dieser Leitfaden richtet sich an Dateningenieure, Lösungsarchitekten und Datenanalysten, die Lösungen entwerfen und implementieren, bei denen Grafiken verwendet werden. AWS

Ziele

Dieser Leitfaden kann Ihnen und Ihrer Organisation dabei helfen, Folgendes zu tun:

- Wählen Sie je nach Anwendungsfall und Abfragemustern aus den unterstützten Bereitstellungsoptionen und Abfragesprachen.
- Folgen Sie den Entwurfsmustern von AWS Well-Architected, die zur Verbesserung der Widerstandsfähigkeit und Sicherheit beitragen.
- Entwerfen Sie Ihre Abfragen im Hinblick auf optimale Leistung und Kosteneinsparungen.
- Erfahren Sie, wie Sie Ihren Neptune-Cluster in der Produktion betrieblich effizient verwalten können.

Säule „Operational Excellence“

Der Schwerpunkt Operational Excellence des AWS Well-Architected Framework liegt auf dem Betrieb und der Überwachung von Systemen sowie der kontinuierlichen Verbesserung von Prozessen und Verfahren. Dazu gehört die Fähigkeit, die Entwicklung zu unterstützen und Workloads effektiv auszuführen, Einblicke in deren Betrieb zu gewinnen und die unterstützenden Prozesse und Verfahren kontinuierlich zu verbessern, um einen Mehrwert für das Unternehmen zu erzielen. Sie können die betriebliche Komplexität reduzieren, indem Sie Workloads automatisch reparieren, wodurch die meisten Probleme ohne menschliches Eingreifen erkannt und behoben werden. Sie können auf dieses Ziel hinarbeiten, indem Sie die in diesem Abschnitt beschriebenen bewährten Methoden befolgen. Verwenden Sie die Kennzahlen und Mechanismen von Amazon Neptune APIs, um angemessen zu reagieren, wenn Ihre Arbeitslast vom erwarteten Verhalten abweicht.

Diese Diskussion über den Pfeiler Operational Excellence konzentriert sich auf die folgenden Schlüsselbereiche:

- Infrastructure as Code (IaC)
- Änderungsmanagement
- Strategien zur Resilienz
- Vorfallmanagement
- Auditberichte zur Einhaltung der Vorschriften
- Protokollierung und Überwachung

Automatisieren Sie die Bereitstellung mithilfe eines IaC-Ansatzes

Zu den bewährten Methoden für die Automatisierung der Bereitstellung auf Neptune mithilfe von IaC gehören:

- Wenden Sie nach Möglichkeit Infrastructure as Code (IaC) an, um Neptun-Cluster bereitzustellen. Verwenden Sie für eine konsistente Umgebungskonfiguration eine AWS CloudFormationVorlage oder HashiCorp Terraform AWS Cloud Development Kit (AWS CDK), um alle erforderlichen Ressourcen für Ihren Cluster zu erstellen.
- Automatisieren Sie Betriebsabläufe in Neptune, wie z. B. die Größenänderung von Instances, das Hinzufügen oder Entfernen von Read Replicas oder das Durchführen manueller Failovers für globale Tabellen, wann immer dies möglich ist.

- Speichern Sie Verbindungszeichenfolgen extern von Ihrem Client aus. Verwenden Sie ETL-Prozesse (Extrahieren, Transformieren und Laden), um Blue-Green-Implementierungsstrategien, Disaster Recovery (DR) und Migrationen zu neuen Clustern nahezu ohne Ausfallzeiten zu ermöglichen. Verbindungszeichenfolgen können in AWS Secrets ManagerAmazon DynamoDB oder an einem beliebigen Ort gespeichert werden, wo sie dynamisch geändert werden können.
- Verwenden Sie Tags, um Metadaten zu Ihren Neptune-Ressourcen hinzuzufügen, und verfolgen Sie die Nutzung anhand von Tags. Weitere Informationen finden Sie unter Tagging Amazon Neptune Resources.

Nehmen Sie häufig kleine, umkehrbare Änderungen vor

Die folgenden Empfehlungen konzentrieren sich auf kleine, umkehrbare Änderungen, um die Komplexität zu minimieren und die Wahrscheinlichkeit einer Unterbrechung der Arbeitslast zu verringern:

- Speichern Sie IaC-Vorlagen und -Skripts in einem Quellcodeverwaltungsdienst, z. B. GitHub oder GitLab.

  Wichtig: Speichern Sie keine AWS Anmeldeinformationen in der Quellcodeverwaltung.

- Erfordern Sie, dass IaC-Bereitstellungen einen CI/CD-Dienst (Continuous Integration and Continuous Delivery) verwenden, z. B. oder. AWS CodeDeployAWS CodeBuild Diese Services kompilieren, testen und implementieren Code in einer Nicht-Produktionsumgebung, die einen kurzlebigen Neptune-Cluster enthält, bevor sie sich auf Ihren Amazon Neptune Neptune-Produktionscluster auswirken.
- Testen Sie Infrastruktur- und Anwendungsabfragen in einer niedrigeren Umgebung, bevor Sie sie in der Produktion einsetzen. Auf diese Weise wird die Wahrscheinlichkeit einer Unterbrechung minimiert und es wird sichergestellt, dass sie Ihrer Arbeitslast und Ihrem Umfang entsprechend funktionieren.

Rechnen Sie mit Ausfällen

Eine selbstreparierende Infrastruktur ist ein Beispiel für betriebliche Exzellenz, da sie Ausfälle antizipiert und versucht, Probleme ohne Eingreifen zu lösen. Die folgenden Empfehlungen helfen Ihnen dabei, diese Reife mit Neptune zu erreichen:

- Erstellen Sie einen Überwachungsplan, der CloudWatch Amazon-Metriken verwendet, um die CPU- und Speicherauslastung Ihrer DB-Instance zu überwachen und die Nutzungsmuster zu verstehen. Erstellen Sie CloudWatch Dashboards und Alarme für wichtige Kennzahlen und die Antworten der Neptune-Kunden in Ihren Anwendungsprotokollen. Weitere Informationen zu Indikatoren für hohe oder niedrige CPU-Auslastung finden Sie unter Verwendung CloudWatch zur Überwachung der DB-Instance-Leistung in Neptune in der Neptune-Dokumentation.

  Wenn Sie häufig out-of-memory Ausnahmen bei Ihren Abfragen erhalten, während diese niedrig `FreeableMemory` ist, sollten Sie die Verwendung einer Instance aus der X2-Familie in Betracht ziehen.
- Richten Sie Benachrichtigungen ein, um den Zustand des Neptun-Clusters zu überwachen. Sie `BufferCacheHitRatio` sollte beispielsweise konstant hoch sein (über 99,9 Prozent), während sie konstant niedrig sein `MainRequestQueuePendingRequests` sollte (idealerweise 0, aber abhängig von Ihren Anforderungen und der Latenztoleranz).
- Erwägen Sie die Verwendung von Read Replicas, um eine hohe Verfügbarkeit innerhalb von Neptune zu erreichen. Sie sollten mindestens zwei Read Replicas in unterschiedlichen Availability Zones als die Writer-Instance haben, um sicherzustellen, dass während eines Failover-Ereignisses immer eine Instanz für Leseanfragen verfügbar ist.
- Automatische Skalierung von Read Replicas auf der Grundlage von Nutzungsmetriken. Weitere Informationen finden Sie unter Automatische Skalierung der Anzahl von Replikaten in einem Amazon Neptune Neptune-DB-Cluster.
- Testen Sie den Failover für Ihre DB-Instance, um zu verstehen, wie lange der Vorgang für Ihren Anwendungsfall dauert.
- Wenn Ihre Anwendung einen kompletten AWS-Region Ausfall überstehen muss, sollten Sie die Verwendung globaler Datenbanken als Teil Ihrer DR-Pläne in Betracht ziehen.

Lernen Sie aus allen Betriebsausfällen

Eine Infrastruktur zur Selbstheilung ist ein langfristiges Projekt, das sich in mehreren Schritten entwickelt, wenn seltene Probleme auftreten oder die Reaktionen nicht so effektiv sind wie gewünscht. Durch die Anwendung der folgenden Methoden wird die Konzentration auf dieses Ziel vorangetrieben:

- Treiben Sie Verbesserungen voran, indem Sie aus allen Fehlern lernen.

- Teilen Sie das Gelernte mit den Teams und der Organisation. Wenn mehrere Teams innerhalb einer Organisation Neptune verwenden, richten Sie einen gemeinsamen Chatroom oder eine Benutzergruppe ein, um Erfahrungen und bewährte Verfahren auszutauschen.

Verwenden Sie Protokollierungsfunktionen, um unbefugte oder ungewöhnliche Aktivitäten zu überwachen

Um ungewöhnliche Leistungs- und Aktivitätsmuster zu beobachten, speichern Sie Protokolle in Amazon CloudWatch Logs. Beachten Sie die folgenden bewährten Methoden:

- Aktivieren Sie die Protokollierung langsamer Abfragen. Überprüfen Sie regelmäßig das Protokoll und stellen Sie fest, warum bestimmte Abfragen langsam sind. Verwenden Sie Neptune Explain and Profile Endpoints für Gremlin, SPARQL oder OpenCypher, um zu verstehen, warum diese Abfragen langsam sind.
- Aktivieren Sie die Neptune-Auditprotokolle und überprüfen Sie die Protokolle regelmäßig auf unbefugten Zugriff oder Anomalien.
- Wenn Sie die Protokollierung mit langsamen Abfragen oder die Auditprotokollierung verwenden, aktivieren Sie die Veröffentlichung in Logs. CloudWatch Auf diese Weise können Sie vermeiden, dass Ihnen der Festplattenspeicher auf den Instanzen ausgeht. Neptune-Instances haben eine begrenzte Protokollspeicherkapazität und überschreiben ältere Protokolldateien, wenn der Protokollspeicher überschritten wird. CloudWatch Logs unterstützt die langfristige Aufbewahrung von Protokollen. Die erweiterten Überwachungsfunktionen in CloudWatch Logs verbessern Ihre Fähigkeit, Protokolle abzufragen und Probleme zu diagnostizieren.
- Um bessere Analysetools für Ihre Audit-Logs zu ermöglichen, können Sie einen Neptune-DB-Cluster so konfigurieren, dass er Audit-Log-Daten in einer Protokollgruppe in CloudWatch Logs veröffentlicht. Mit CloudWatch Logs können Sie die Protokolldaten in Echtzeit analysieren, Alarme erstellen und Metriken anzeigen und CloudWatch Logs verwenden, um Ihre Protokolldatensätze auf einem äußerst dauerhaften Speicher zu speichern. CloudWatch Weitere Informationen finden Sie unter Neptune-Protokolle in Amazon CloudWatch Logs veröffentlichen.
- Neptune unterstützt die Protokollierung von Aktionen auf der Kontrollebene mithilfe von. AWS CloudTrail Weitere Informationen finden Sie unter Protokollieren von Amazon Neptune Neptune-API-Aufrufen mit. AWS CloudTrail

Säule der Sicherheit

Cloud-Sicherheit hat höchste AWS Priorität. Als AWS Kunde profitieren Sie von einer Rechenzentrums- und Netzwerkarchitektur, die darauf ausgelegt sind, die Anforderungen der sicherheitssensibelsten Unternehmen zu erfüllen.

Sicherheit ist eine gemeinsame Verantwortung von Ihnen AWS und Ihnen. Im Modell der übergreifenden Verantwortlichkeit wird Folgendes mit „Sicherheit der Cloud“ bzw. „Sicherheit in der Cloud“ umschrieben:

- Sicherheit der Cloud — AWS ist verantwortlich für den Schutz der Infrastruktur, die AWS-Services in der läuft AWS Cloud. AWS bietet Ihnen auch Dienste, die Sie sicher nutzen können. Externe Prüfer testen und verifizieren regelmäßig die Wirksamkeit der AWS Sicherheit im Rahmen der AWS Compliance-Programme. Informationen zu den für Amazon Neptune geltenden Compliance-Programmen finden Sie unter Im Rahmen des Compliance-Programms zugelassene -Services.
- Sicherheit in der Cloud — Ihre Verantwortung richtet sich nach dem AWS-Service , was Sie verwenden. Sie sind auch für andere Faktoren verantwortlich, etwa für die Vertraulichkeit Ihrer Daten, die Anforderungen Ihres Unternehmens und die geltenden Gesetze und Vorschriften. Weitere Informationen zum Datenschutz finden Sie unter Häufig gestellte Fragen zum Datenschutz. Informationen zum Datenschutz in Europa finden Sie im Blogbeitrag AWS Shared Responsibility Model und GDPR.

Die Sicherheitssäule hilft Ihnen zu verstehen, wie Sie das Modell der gemeinsamen Verantwortung bei der Verwendung von Neptune anwenden können. Die folgenden Themen zeigen Ihnen, wie Sie Neptune zur Erfüllung Ihrer Sicherheits- und Compliance-Ziele konfigurieren können. Sie lernen auch, wie Sie andere verwenden können AWS-Services , die Ihnen helfen, Ihre Neptun-Ressourcen zu überwachen und zu sichern.

Die Sicherheitssäule umfasst die folgenden Schwerpunktbereiche:

- Datensicherheit
- Netzwerksicherheit
- Authentifizierung und Autorisierung

Implementieren Sie Datensicherheit

Datenlecks und Sicherheitsverletzungen gefährden Ihre Kunden und können erhebliche negative Auswirkungen auf Ihr Unternehmen haben. Die folgenden bewährten Methoden tragen dazu bei, Ihre Kundendaten vor unbeabsichtigter und böswilliger Offenlegung zu schützen:

- Clusternamen, Tags, Parametergruppen, AWS Identity and Access Management (IAM) -Rollen und andere Metadaten sollten keine vertraulichen oder sensiblen Informationen enthalten, da diese Daten in Abrechnungs- oder Diagnoseprotokollen erscheinen könnten.
- URIs oder Links zu externen Servern, die als Daten in Neptune gespeichert sind, sollten keine Anmeldeinformationen zur Validierung von Anfragen enthalten.
- Verschlüsselte Neptune-Instances bieten eine zusätzliche Datenschutzebene, da Sie Ihre Daten vor nicht autorisierten Zugriffen auf den zugrunde liegenden Speicher schützen. Sie können mithilfe der Neptune-Verschlüsselung den Datenschutz für Ihre in der Cloud bereitgestellten Anwendungen erhöhen. Sie können auch Neptune-Verschlüsselung verwenden, um die Compliance-Anforderungen für ruhende Daten zu erfüllen.

  Um die Verschlüsselung für eine neue Neptune-DB-Instance zu aktivieren, wählen Sie in der Neptune-Konsole im Abschnitt Verschlüsselung aktivieren die Option Ja aus (standardmäßig ausgewählt), oder legen Sie die Eigenschaft unter fest. AWS::Neptune::DBCluster::StorageEncrypted AWS CloudFormation Wenn die Verschlüsselung aktiviert ist, verwendet Neptune standardmäßig den von Amazon Relational Database Service (Amazon RDS) verwalteten AWS-Schlüssel, oder Sie können einen vom Kunden verwalteten Schlüssel erstellen. Informationen zum Erstellen einer Neptune-DB-Instance finden Sie unter Erstellen eines neuen Neptune-DB-Clusters. Weitere Informationen finden Sie unter Neptune-Ressourcen im Ruhezustand verschlüsseln. Ihre automatisierten und manuellen Snapshots verwenden dieselbe Verschlüsselung, die Sie für Ihren Neptune-Cluster ausgewählt haben.
- Wenn Sie die Sprachen SPARQL und OpenCypher verwenden, sollten Sie die richtigen Techniken zur Eingabevalidierung und Parametrisierung anwenden, um SQL-Injection und andere Formen von Angriffen zu verhindern. Vermeiden Sie es, Abfragen zu erstellen, die eine Verkettung von Zeichenketten mit vom Benutzer bereitgestellten Eingaben verwenden. Verwenden Sie parametrisierte Abfragen oder vorbereitete Anweisungen, um Eingabeparameter sicher an die Graphdatenbank zu übergeben. Weitere Informationen finden Sie unter Beispiele für parametrisierte OpenCypher-Abfragen und SPARQL Injection Defence.
- Verwenden Sie für die Gremlin-Sprache Gremlin-Sprachvarianten, anstatt direkt auf Zeichenketten basierende Gremlin-Skripte zu übergeben, um mögliche Injektionsprobleme zu vermeiden.

Schützen Sie Ihre Netzwerke

Ein Amazon Neptune Neptune-DB-Cluster kann nur in einer Virtual Private Cloud (VPC) erstellt werden. AWS Auf die Endpunkte des Neptune-DB-Clusters kann nur innerhalb dieser VPC zugegriffen werden, normalerweise von einer Amazon Elastic Compute Cloud (Amazon EC2) - Instance aus, die in dieser VPC ausgeführt wird. Sie können Ihre Neptune-Daten schützen, indem Sie den Zugriff auf die VPC einschränken, in der sich Ihr Neptune-DB-Cluster befindet. Weitere Informationen finden Sie unter Verbindung zu Ihrem Amazon Neptune Neptune-Diagramm herstellen.

Um Ihre Daten während der Übertragung zu schützen, erzwingt Neptune mithilfe sicherer Protokolle und Chiffren SSL-Verbindungen über HTTPS zu jeder Instanz oder jedem Cluster-Endpunkt. Neptune stellt SSL-Zertifikate für Ihre Neptune-DB-Instances bereit. Neptune SSL-Zertifikate unterstützen nur Hostnamen für Cluster-Endpunkte, Reader-Endpunkte und Instance-Endpunkte. Wenn Sie einen Load Balancer oder einen Proxyserver (z. B. HAProxy) verwenden, müssen Sie die SSL-Terminierung verwenden und Ihr eigenes SSL-Zertifikat auf dem Proxyserver haben. SSL-Passthrough funktioniert nicht, da die bereitgestellten SSL-Zertifikate nicht mit dem Hostnamen des Proxy-Servers übereinstimmen. Weitere Informationen zum Herstellen einer Verbindung zu Neptune-Endpunkten mit SSL finden Sie unter Verwenden des HTTP-REST-Endpunkts zur Verbindung mit einer Neptune-DB-Instance.

Implementieren Sie Authentifizierung und Autorisierung

Verwenden Sie IAM-Anmeldeinformationen, um zu steuern, wer Neptune-Verwaltungsaktionen auf Neptune-DB-Clustern und DB-Instances ausführen kann. Wenn Sie AWS mithilfe von IAM-Anmeldeinformationen eine Verbindung herstellen, muss Ihre IAM-Rolle über IAM-Richtlinien verfügen, die die für die Ausführung von Neptune-Verwaltungsoperationen erforderlichen Berechtigungen gewähren. Stellen Sie sicher, dass Sie dem Prinzip der geringsten Rechte folgen und nur die Berechtigungen gewähren, die für die Ausführung einer Aufgabe erforderlich sind. Weitere Informationen finden Sie unter Verschiedene Arten von IAM-Richtlinien zur Steuerung des Zugriffs auf Neptune verwenden und IAM-Authentifizierung mithilfe temporärer Anmeldeinformationen.

Um zu kontrollieren, wer eine Verbindung zu einem Neptune-Cluster herstellen und die Daten abfragen kann, können Sie IAM verwenden, um sich bei Ihrer Neptune-DB-Instance oder Ihrem Neptune-DB-Cluster zu authentifizieren. Wenn Sie die IAM-Authentifizierung in einem Neptune-DB-Cluster aktivieren, muss jeder, der auf den DB-Cluster zugreift, zuerst authentifiziert werden. Weitere Informationen finden Sie unter Aktivieren der IAM-Datenbankauthentifizierung in Neptune für Schritte zur Aktivierung der IAM-Authentifizierung.

Wenn die IAM-Datenbank-Authentifizierung aktiviert ist, muss jede Anforderung mit AWS Signature Version 4 signiert werden. Informationen zum Senden signierter Anfragen an alle Neptune-Endpunkte mit aktivierter IAM-Authentifizierung finden Sie unter Connecting and Signing with Signature Version 4. AWS Viele Bibliotheken und Tools, wie zum Beispiel awscurl, unterstützen bereits Signature Version 4. AWS

Für die Interaktion mit anderen AWS-Services verwendet Amazon Neptune mit dem Service verknüpfte IAM-Rollen. Eine serviceverknüpfte Rolle ist eine spezielle IAM-Rolle, die direkt mit Neptune verknüpft ist. Dienstbezogene Rollen sind von Neptune vordefiniert und beinhalten alle Berechtigungen, die der Dienst benötigt, um andere AWS-Services in Ihrem Namen anzurufen. Weitere Informationen finden Sie unter Verwenden von dienstverknüpften Rollen für Neptune.

Säule der Zuverlässigkeit

Die Säule Zuverlässigkeit umfasst die Fähigkeit eines Workloads, die vorgesehene Funktion korrekt und konsistent auszuführen, wenn dies erwartet wird. Dies umfasst die Möglichkeit, die Workload während des gesamten Lebenszyklus zu betreiben und zu testen.

Ausgangspunkt für eine zuverlässige Workload sind vorab getroffene Designentscheidungen für Software und Infrastruktur. Ihre Auswahl in puncto Architektur wirkt sich in allen Well-Architected-Säulen auf das Verhalten der Workload aus. Zur Gewährleistung von Zuverlässigkeit sind bestimmte Muster zu befolgen.

Die Säule Zuverlässigkeit konzentriert sich auf die folgenden Schlüsselbereiche:

- Workload-Architektur, einschließlich Servicequotas und Bereitstellungsmustern
- Änderungsmanagement
- Fehlerverwaltung

Neptune-Dienstkontingente verstehen

Ein Neptun-Cluster-Volume kann auf eine maximale Größe von 128 Tebibyte (TiB) anwachsen, AWS-Regionen sofern nicht China unterstützt wird und GovCloud wo das Kontingent 64 TiB beträgt.

Das 128 TiB-Kontingent reicht aus, um etwa 200-400 Milliarden Objekte im Diagramm zu speichern. In einem Labeled Property Graph (LPG) ist ein Objekt ein Knoten, eine Kante oder eine Eigenschaft an einem Knoten oder einer Kante. In einem RDF-Diagramm (Resource Description Framework) ist ein Objekt ein Quad.

Für jeden Neptune Serverless-Cluster legen Sie sowohl die minimale als auch die maximale Anzahl von Neptune Capacity Units () fest. NCUs Jede NCU besteht aus 2 Gibibytes (GiB) Arbeitsspeicher und der zugehörigen vCPU und dem Netzwerk. Die minimalen und maximalen NCU-Werte gelten für alle serverlosen Instanzen im Cluster. Der höchste maximale NCU-Wert, den Sie festlegen können, ist 128,0 NCUs, und der niedrigste Mindestwert ist 1,0. NCUs Optimieren Sie den NCU-Bereich, der für Ihre Anwendung am besten geeignet ist, indem Sie die CloudWatch Amazon-Metriken beobachten `ServerlessDatabaseCapacity` und `NCUUtilization` den Bereich erfassen, in dem Sie häufig arbeiten, und unerwünschtes Verhalten oder Kosten innerhalb dieses Bereichs korrelieren. Wenn Sie feststellen, dass Ihr Workload nicht schnell genug skaliert, erhöhen Sie den Mindestwert, NCUs um ausreichend Rechenleistung für den anfänglichen Anstieg während der Skalierung bereitzustellen.

In AWS-Konto jeder Region gibt es Kontingente für die Anzahl der Datenbankressourcen, die Sie erstellen können. Zu diesen Ressourcen gehören DB-Instances und DB-Cluster. Nachdem eine Größenbeschränkung für eine Ressource erreicht wurde, schlagen zusätzliche Aufrufe zum Erstellen dieser Ressource mit einer Ausnahme fehl. Bei einigen Kontingenten handelt es sich um unverbindliche Kontingente, die auf Anfrage erhöht werden können. Eine Liste der Kontingente, die von Amazon Neptune und Amazon RDS, Amazon Aurora und Amazon DocumentDB gemeinsam genutzt werden (mit MongoDB-Kompatibilität), zusammen mit Links zur Beantragung von Kontingenterhöhungen, sofern verfügbar, finden Sie unter Kontingente in Amazon RDS.

Verstehen Sie die Einsatzmuster von Neptune

In Neptune-DB-Clustern gibt es eine primäre DB-Instance und bis zu 15 Neptune-Repliken. Die primäre DB-Instance unterstützt Lese- und Schreibvorgänge und führt alle Datenänderungen am Cluster-Volume durch. Neptune-Replikate stellen eine Verbindung zu demselben Speichervolume her wie die primäre DB-Instance und unterstützen nur Lesevorgänge. Neptune-Replicas können Lese-Workloads von der primären DB-Instance auslagern.

Verwenden Sie Read Replicas, um eine hohe Verfügbarkeit zu erreichen. Wenn eine oder mehrere Read Replica-Instanzen in verschiedenen Availability Zones verfügbar sind, kann dies die Verfügbarkeit erhöhen, da Read Replicas als Failover-Ziele für die primäre Instanz dienen. Wenn die Writer-Instanz ausfällt, befördert Neptune eine Read Replica-Instanz zur primären Instanz. In diesem Fall kommt es zu einer kurzen Unterbrechung (in der Regel weniger als 30 Sekunden), während die hochgestufte Instanz neu gestartet wird. Während dieser Zeit schlagen Lese- und Schreibanforderungen an die primäre Instance mit einer Ausnahme fehl. Für höchste Zuverlässigkeit sollten Sie zwei Read Replicas in verschiedenen Availability Zones in Betracht ziehen. Wenn die primäre Instance in Availability Zone 1 offline geht, wird die Instance in Availability Zone 2 zur primären Instanz heraufgestuft, kann aber solange keine Abfragen bearbeiten. Daher ist eine Instance in Availability Zone 3 erforderlich, um Leseabfragen während des Übergangs zu verarbeiten.

Wenn Sie Neptune Serverless verwenden, werden Reader- und Writer-Instances in allen Availability Zones unabhängig voneinander je nach Datenbanklast hoch- und herunterskaliert. Sie können die Promotion-Stufe einer Reader-Instanz auf 0 oder 1 setzen, sodass sie zusammen mit der Kapazität der Writer-Instanz nach oben oder unten skaliert wird. Dadurch ist sie jederzeit bereit, die aktuelle Arbeitslast zu übernehmen.

Neptun-Cluster verwalten und skalieren

Sie können Neptune auto-scaling verwenden, um die Anzahl der Neptune-Replikate in einem DB-Cluster automatisch an Ihre Konnektivitäts- und Workload-Anforderungen auf der Grundlage von CPU-Auslastungsschwellenwerten anzupassen. Mit der auto-scaling kann Ihr Neptune-DB-Cluster plötzlichen Anstieg der Arbeitslast bewältigen. Wenn die Arbeitslast abnimmt, entfernt die auto-scaling unnötige Replikate, sodass Sie nicht für ungenutzte Kapazität zahlen müssen. Beachten Sie, dass der Start neuer Instances bis zu 15 Minuten dauern kann. auto-scaling allein ist also keine ausreichende Lösung für schnelle Bedarfsänderungen.

Sie können auto-scaling nur mit einem Neptune-DB-Cluster verwenden, der bereits über eine primäre Writer-Instance und mindestens eine Read-Replica-Instance verfügt (siehe Amazon Neptune DB Clusters and Instances). Außerdem müssen alle Read-Replica-Instances im Cluster verfügbar sein. Wenn sich eine Read-Replica in einem anderen Status als verfügbar befindet, tut Neptune auto-scaling nichts, bis jede Read-Replica im Cluster verfügbar ist.

Wenn sich die Nachfrage schnell ändert, sollten Sie die Verwendung serverloser Instances in Betracht ziehen. Die serverlosen Instances können über kurze Zeiträume vertikal skaliert werden, während die auto-scaling über längere Zeiträume horizontal skaliert wird. Diese Konfiguration bietet optimale Skalierbarkeit, da die serverlosen Instances vertikal skaliert werden, während die auto-scaling neue Read Replicas instanziiert, um die Arbeitslast zu bewältigen, die über die maximale Kapazität einer einzelnen serverlosen Instanz hinausgeht. Weitere Informationen zur Kapazitätsskalierung von Amazon Neptune Serverless finden Sie unter Kapazitätsskalierung in einem Neptune Serverless DB-Cluster.

Wenn sich Ihre Skalierungsanforderungen zu vorhersehbaren Zeiten ändern, können Sie Änderungen der Mindestanzahl an Instances, der maximalen Anzahl von Instances und Schwellenwerten planen, um diesen wechselnden Anforderungen besser gerecht zu werden. Denken Sie daran, Scale-Out-Ereignisse mindestens 15 Minuten im Voraus zu planen, damit diese Instances bei Bedarf online gehen können.

Sie können Ihre Datenbankkonfiguration in Amazon Neptune mittels Parametern in einer Parametergruppe verwalten. Parametergruppen dienen als Container für Engine-Konfigurationswerte, die auf eine oder mehrere DB-Instances angewendet werden. Wenn Sie Cluster-Parameter in Parametergruppen ändern, sollten Sie den Unterschied zwischen statischen und dynamischen Parametern verstehen und wissen, wie und wann sie angewendet werden. Verwenden Sie den Status-Endpunkt, um die aktuell angewendete Konfiguration zu sehen.

Backups und Failover-Ereignisse verwalten

Neptune sichert Ihr Cluster-Volume automatisch und bewahrt die gesicherten Daten für die Dauer des Backup-Aufbewahrungszeitraums auf. Neptune-Sicherungen sind kontinuierlich und inkrementell, so dass Sie schnell eine Sicherung zu einem beliebigen Punkt im Aufbewahrungszeitraum für Sicherungen durchführen können. Sie können einen Aufbewahrungszeitraum für Backups von 1—35 Tagen angeben, wenn Sie einen DB-Cluster erstellen oder ändern.

Um ein Backup über den Aufbewahrungszeitraum des Backups hinaus aufzubewahren, können Sie auch einen Snapshot der Daten in Ihrem Cluster-Volume erstellen. Für das Speichern von Snapshots fallen die Standardgebühren für die Speicherplatznutzung in Neptune an.

Wenn Sie einen Amazon Neptune Neptune-Snapshot eines DB-Clusters erstellen, erstellt Neptune einen Speicher-Volume-Snapshot des Clusters und sichert alle seine Daten, nicht nur einzelne Instances. Sie können einen DB-Cluster erstellen, indem Sie eine Wiederherstellung aus diesem DB-Cluster-Snapshot durchführen. Wenn Sie den DB-Cluster wiederherstellen, geben Sie den Namen des DB-Cluster-Snapshots an, aus dem wiederhergestellt werden soll, und dann geben Sie einen Namen für den neuen DB-Cluster an, der bei der Wiederherstellung erstellt wird.

Testen Sie, wie Ihr System auf Failover-Ereignisse reagiert. Verwenden Sie die Neptune-API, um ein Failover-Ereignis zu erzwingen. Ein Neustart mit Failover ist nützlich, wenn Sie einen Ausfall einer DB-Instance zu Testzwecken simulieren oder um nach einem Failover Operationen in der ursprünglichen Availability Zone wiederherzustellen. Weitere Informationen finden Sie unter Konfiguration und Verwaltung einer Multi-AZ-Bereitstellung. Wenn Sie eine DB-Writer-Instance neu starten, erfolgt ein Failover auf das Standby-Replikat. Das Neustarten eines Neptune-Replikats leitet kein Failover ein.

Gestalten Sie Ihre Clients im Hinblick auf Zuverlässigkeit. Testen Sie ihr Verhalten bei Failover-Ereignissen. Implementieren Sie in Ihrem Client eine Wiederholungslogik mit exponentieller Backoff-Logik. Codebeispiele, die diese Logik implementieren, finden Sie in AWS Lambda Funktionsbeispielen für Amazon Neptune.

Ziehen Sie die Verwendung in Betracht, AWS Backupwenn Sie gemeinsame Sicherungsanforderungen haben, die Sie für mehrere Datenbank-Engines anwenden.

Säule der Leistungseffizienz

Die Säule der Leistungseffizienz des AWS Well-Architected Framework konzentriert sich darauf, wie die Leistung beim Einlesen oder Abfragen von Daten optimiert werden kann. Die Leistungsoptimierung ist ein inkrementeller und kontinuierlicher Prozess, der Folgendes umfasst:

- Bestätigung der Geschäftsanforderungen
- Messung der Workload-Leistung
- Identifizierung leistungsschwacher Komponenten
- Abstimmung der Komponenten auf Ihre Geschäftsanforderungen

Der Schwerpunkt Leistungseffizienz enthält anwendungsfallspezifische Richtlinien, die Ihnen dabei helfen können, das richtige Grafikdatenmodell und die richtigen Abfragesprachen zu finden. Es enthält auch bewährte Methoden, die bei der Aufnahme von Daten in Amazon Neptune und der Nutzung von Daten aus Amazon Neptune zu beachten sind.

Der Schwerpunkt der Leistungseffizienz konzentriert sich auf die folgenden Schlüsselbereiche:

- Graphmodellierung
- Optimierung von Abfragen
- Richtige Clustergröße
- Optimierung schreiben

Verstehen Sie die Graphmodellierung

Verstehen Sie den Unterschied zwischen den Modellen Labeled Property Graph (LPG) und Resource Description Framework (RDF). In den meisten Fällen ist dies eine Frage der Präferenz. Es gibt jedoch mehrere Anwendungsfälle, in denen ein Modell besser geeignet ist als das andere. Wenn Sie den Pfad kennen müssen, der zwei Knoten in Ihrem Diagramm verbindet, wählen Sie LPG. Wenn Sie Daten aus Neptun-Clustern oder anderen Graph Triple Stores zusammenführen möchten, wählen Sie RDF.

Wenn Sie eine SaaS-Anwendung (Software as a Service) oder eine Anwendung entwickeln, die Mehrmandantenfähigkeit erfordert, sollten Sie die logische Trennung von Mandanten in Ihr Datenmodell integrieren, anstatt einen Mandanten für jeden Cluster zu haben. Um diese Art von

Design zu erreichen, können Sie benannte SPARQL-Diagramme und Kennzeichnungsstrategien verwenden, z. B. Kundenkennungen den Bezeichnungen voranstellen oder Eigenschaftsschlüssel-Wert-Paare hinzufügen, die Mandantenkennungen darstellen. Stellen Sie sicher, dass Ihre Client-Ebene diese Werte einfügt, um diese logische Trennung beizubehalten.

Die Leistung Ihrer Abfragen hängt von der Anzahl der Diagrammobjekte (Knoten, Kanten, Eigenschaften) ab, die bei der Verarbeitung Ihrer Abfrage ausgewertet werden müssen. Daher kann das Graphmodell erhebliche Auswirkungen auf die Leistung Ihrer Anwendung haben. Verwenden Sie nach Möglichkeit detaillierte Beschriftungen und speichern Sie nur die Eigenschaften, die Sie für die Pfadbestimmung oder Filterung benötigen. Um eine höhere Leistung zu erzielen, sollten Sie erwägen, Teile Ihres Diagramms vorab zu berechnen, z. B. Zusammenfassungsknoten oder direktere Kanten zu erstellen, die gemeinsame Pfade verbinden.

Vermeiden Sie es, zwischen Knoten zu navigieren, die eine ungewöhnlich hohe Anzahl von Kanten mit derselben Bezeichnung haben. Solche Knoten haben oft Tausende von Kanten (wobei die meisten Knoten eine Kantenanzahl im Zehnerbereich haben). Das Ergebnis ist eine viel höhere Rechen- und Datenkomplexität. Diese Knoten sind bei einigen Abfragemustern möglicherweise nicht problematisch, wir empfehlen jedoch, Ihre Daten anders zu modellieren, um dies zu vermeiden, insbesondere wenn Sie als Zwischenschritt über den Knoten navigieren. Sie können Protokolle für langsame Abfragen verwenden, um Abfragen zu identifizieren, die zwischen diesen Knoten navigieren. Sie werden wahrscheinlich deutlich höhere Latenz- und Datenzugriffsmetriken als Ihre durchschnittlichen Abfragemuster beobachten, insbesondere wenn Sie den Debug-Modus verwenden.

Verwenden Sie deterministischen Knoten IDs für Knoten und Kanten, wenn Ihr Anwendungsfall dies unterstützt, anstatt Neptune zu verwenden, um zufällige GUID-Werte zuzuweisen. IDs Der Zugriff auf Knoten anhand der ID ist die effizienteste Methode.

Optimieren Sie Abfragen

Die Sprachen OpenCypher und Gremlin können auf LPG-Modellen synonym verwendet werden. Wenn Leistung ein Hauptanliegen ist, sollten Sie erwägen, die beiden Sprachen synonym zu verwenden, da eine Sprache bei bestimmten Abfragemustern möglicherweise besser abschneidet als die andere.

Neptune ist dabei, zu seiner Alternative Query Engine (DFE) zu konvertieren. OpenCypher läuft nur auf dem DFE, aber sowohl Gremlin- als auch SPARQL-Abfragen können optional so eingestellt werden, dass sie auf dem DFE ausgeführt werden, indem Abfrageanmerkungen verwendet werden.

Erwägen Sie, Ihre Abfragen mit aktiviertem DFE zu testen und die Leistung Ihres Abfragemusters zu vergleichen, wenn Sie DFE nicht verwenden.

Neptune ist für transaktionale Abfragen optimiert, die an einem einzelnen Knoten oder einer Gruppe von Knoten beginnen und sich von dort aus ausbreiten, und nicht für analytische Abfragen, die den gesamten Graphen auswerten. Erwägen Sie für Ihre analytischen Abfrage-Workloads die Verwendung des AWS-SDK für Pandas oder die Verwendung von Neptune-Export in Kombination mit Amazon EMR. AWS Glue

Um Ineffizienzen und Engpässe in Ihren Modellen und Abfragen zu identifizieren, verwenden Sie das und `explain` APIs für jede Abfragesprache, um detaillierte Erläuterungen zum `profile` Abfrageplan und zu den Abfragemetriken zu erhalten. Weitere Informationen finden Sie unter Gremlin-Profil, OpenCypher Explain und SPARQL Explain.

Verstehen Sie Ihre Abfragemuster. Wenn die Anzahl der unterschiedlichen Kanten in einem Diagramm groß wird, kann die standardmäßige Neptun-Zugriffsstrategie ineffizient werden. Die folgenden Abfragen könnten ziemlich ineffizient werden:

- Abfragen, die rückwärts über Kanten navigieren, wenn keine Kantenbeschriftungen angegeben sind.
- Klauseln, die intern dasselbe Muster verwenden, z. B. `.both()` in Gremlin, oder Klauseln, die Knoten in einer beliebigen Sprache löschen (was das Löschen eingehender Kanten ohne Kenntnis der Labels erfordert).
- Abfragen, die auf Eigenschaftswerte zugreifen, ohne Eigenschaftsbezeichnungen anzugeben. Diese Abfragen könnten ziemlich ineffizient werden. Wenn dies Ihrem Nutzungsmuster entspricht, sollten Sie erwägen, den OSGP-Index (Objekt, Subjekt, Diagramm, Prädikat) zu aktivieren.

Verwenden Sie die Protokollierung langsamer Abfragen, um langsame Abfragen zu identifizieren. Langsame Abfragen können durch nicht optimierte Abfragepläne oder unnötig viele Indexsuchen verursacht werden, was die I/O-Kosten erhöhen kann. Die Neptune Explain and Profile Endpoints für Gremlin, SPARQL oder OpenCypher können Ihnen helfen zu verstehen, warum diese Abfragen langsam sind. Zu den möglichen Ursachen gehören:

- Knoten mit einer ungewöhnlich hohen Anzahl von Kanten im Vergleich zum durchschnittlichen Knoten im Diagramm (z. B. Tausende im Vergleich zu Zehnern) können die Rechenkomplexität erhöhen und somit die Latenz und den Ressourcenverbrauch erhöhen. Stellen Sie fest, ob diese Knoten korrekt modelliert sind oder ob die Zugriffsmuster verbessert werden können, um die Anzahl der Kanten, die überquert werden müssen, zu reduzieren.

- Nicht optimierte Abfragen enthalten eine Warnung, dass bestimmte Schritte nicht optimiert sind. Das Umschreiben dieser Abfragen zur Verwendung optimierter Schritte kann die Leistung verbessern.
- Redundante Filter können zu unnötigen Indexsuchen führen. Ebenso können redundante Muster zu doppelten Indexsuchen führen, die durch eine Verbesserung der Abfrage optimiert werden können (siehe `Index Operations - Duplication ratio` in der Profilausgabe).
- In einigen Sprachen wie Gremlin gibt es keine stark typisierten numerischen Werte und sie verwenden stattdessen die Typ-Heraufstufung. Wenn der Wert beispielsweise 55 ist, sucht Neptune nach Werten, die Ganzzahlen, Longs, Floats und andere numerische Typen sind, die 55 entsprechen. Dies führt zu zusätzlichen Operationen. Wenn Sie im Voraus wissen, dass Ihre Typen übereinstimmen, können Sie dies vermeiden, indem Sie einen Abfragehinweis verwenden.
- Ihr Grafikmodell kann sich erheblich auf die Leistung auswirken. Erwägen Sie, die Anzahl der Objekte, die ausgewertet werden müssen, zu reduzieren, indem Sie detailliertere Beschriftungen verwenden oder Abkürzungen für lineare Multiple-Hop-Pfade im Voraus berechnen.

Wenn die Abfrageoptimierung allein es Ihnen nicht ermöglicht, Ihre Leistungsanforderungen zu erfüllen, sollten Sie erwägen, verschiedene Caching-Techniken mit Neptune zu verwenden, um diese Anforderungen zu erfüllen.

Cluster in der richtigen Größe

Passen Sie die Größe Ihres Clusters an Ihre Anforderungen an Parallelität und Durchsatz an. Die Anzahl der gleichzeitigen Abfragen, die von jeder Instance im Cluster verarbeitet werden können, entspricht dem Zweifachen der Anzahl virtueller Abfragen CPUs (vCPUs) auf dieser Instance. Zusätzliche Abfragen, die eintreffen, während alle Worker-Threads belegt sind, werden in eine serverseitige Warteschlange gestellt. Diese Abfragen werden auf FIFO-Basis bearbeitet, wenn Worker-Threads verfügbar werden. first-in-first-out Die `MainRequestQueuePendingRequests` CloudWatch Amazon-Metrik zeigt die aktuelle Warteschlangentiefe für jede Instance.
Wenn dieser Wert häufig über Null liegt, sollten Sie eine Instance mit mehr v wählenCPUs.
Wenn die Warteschlangentiefe 8.192 überschreitet, gibt Neptune einen Fehler zurück. `ThrottlingException`

Ungefähr 65 Prozent des RAM für jede Instanz sind für den Puffercache reserviert. Der Puffercache enthält den Arbeitsdatensatz (nicht das gesamte Diagramm, sondern nur die Daten, die abgefragt werden). Überwachen Sie die Metrik, um festzustellen, welcher Prozentsatz der Daten aus dem Puffer-Cache und nicht aus dem Speicher abgerufen wird. CloudWatch `BufferCacheHitRatio`

Wenn diese Metrik häufig unter 99,9 Prozent fällt, sollten Sie es mit einer Instance mit mehr Arbeitsspeicher versuchen, um festzustellen, ob dadurch Ihre Latenz und I/O-Kosten gesenkt werden.

Read Replicas müssen nicht dieselbe Größe wie Ihre Writer-Instance haben. Starke Schreiblasten können jedoch dazu führen, dass kleinere Replikate ins Hintertreffen geraten und neu gestartet werden, da sie mit der Replikation nicht Schritt halten können. Aus diesem Grund empfehlen wir, Replikate gleich oder größer als die Writer-Instance zu erstellen.

Wenn Sie auto-scaling für Ihre Read Replicas verwenden, denken Sie daran, dass es bis zu 15 Minuten dauern kann, bis eine neue Read Replica online ist. Wenn der Client-Verkehr schnell, aber vorhersehbar zunimmt, sollten Sie eine geplante Skalierung in Betracht ziehen, um die Mindestanzahl von Read Replicas entsprechend dieser Initialisierungszeit zu erhöhen.

Serverlose Instanzen unterstützen verschiedene Anwendungsfälle und Workloads. Ziehen Sie in den folgenden Szenarien serverlose statt bereitgestellte Instanzen in Betracht:

- Ihre Arbeitslast schwankt im Laufe des Tages häufig.
- Sie haben eine neue Anwendung erstellt und sind sich nicht sicher, wie groß die Arbeitslast sein wird.
- Sie entwickeln und testen.

Es ist wichtig zu beachten, dass serverlose Instanzen teurer sind als vergleichbare bereitgestellte Instanzen, wenn man einen Dollar pro GB RAM berechnet. Jede serverlose Instanz besteht aus 2 GB RAM zusammen mit der zugehörigen vCPU und dem Netzwerk. Führen Sie eine Kostenanalyse zwischen Ihren Optionen durch, um überraschende Rechnungen zu vermeiden. Im Allgemeinen erzielen Sie mit Serverless nur dann Kosteneinsparungen, wenn Ihre Arbeitslast nur wenige Stunden am Tag sehr hoch ist und den Rest des Tages fast Null ist oder wenn Ihre Arbeitslast im Laufe des Tages stark schwankt.

Optimize schreibt

Beachten Sie Folgendes, um Schreibvorgänge zu optimieren:

- Der Neptune Bulk Loader ist die optimale Methode, um Ihre Datenbank zunächst zu laden oder an bestehende Daten anzuhängen. Der Neptune-Loader ist nicht transaktional und kann keine Daten löschen. Verwenden Sie ihn daher nicht, wenn dies Ihre Anforderungen sind.
- Transaktionsaktualisierungen können mithilfe der unterstützten Abfragesprachen vorgenommen werden. Um I/O-Schreibvorgänge zu optimieren, schreiben Sie Daten in Batches von 50-100

Objekten pro Commit. Ein Objekt ist ein Knoten, eine Kante oder eine Eigenschaft an einem Knoten oder einer Kante in LPG oder ein Triple Store oder ein Quad in RDF.

- Alle Neptune-Schreiboperationen werden für jede Verbindung in einem Thread ausgeführt. Wenn Sie eine große Datenmenge an Neptune senden, sollten Sie mehrere parallel Verbindungen in Betracht ziehen, die jeweils Daten schreiben. Wenn Sie sich für eine von Neptune bereitgestellte Instanz entscheiden, wird die Instanzgröße einer Zahl von v zugeordnet. CPUs Neptune erstellt zwei Datenbank-Threads für jede vCPU auf der Instance. Beginnen Sie also mit der doppelten Anzahl von v, CPUs wenn Sie die optimale Parallelisierung testen. Serverlose Instanzen skalieren die Anzahl von v mit einer CPUs Rate von ungefähr eins für jede 4. NCUs
- Planen Sie alle Schreibvorgänge ein ConcurrentModificationExceptionsund verarbeiten Sie sie effizient, auch wenn zu einem beliebigen Zeitpunkt nur eine einzige Verbindung Daten schreibt. Gestalten Sie Ihre Clients so, dass sie zuverlässig sind, wenn sie `ConcurrentModificationExceptions` auftreten.
- Wenn Sie alle Ihre Daten löschen möchten, sollten Sie die Fast-Reset-API verwenden, anstatt gleichzeitig Löschabfragen zu stellen. Letzteres dauert im Vergleich zu ersterem viel länger und verursacht erhebliche I/O-Kosten.
- Wenn Sie die meisten Ihrer Daten löschen möchten, sollten Sie erwägen, die Daten, die Sie behalten möchten, zu exportieren, indem Sie die Daten mithilfe von neptune-export in einen neuen Cluster laden. Löschen Sie dann den ursprünglichen Cluster.

Säule der Kostenoptimierung

Die Säule der Kostenoptimierung des AWS Well-Architected Framework konzentriert sich auf die Vermeidung unnötiger Kosten. Die folgenden Empfehlungen können Ihnen helfen, die Entwurfsprinzipien zur Kostenoptimierung und die bewährten Architekturpraktiken für Amazon Neptune zu erfüllen.

Die Säule Kostenoptimierung konzentriert sich auf die folgenden Schlüsselbereiche:

- Überblick über die Ausgaben im Zeitverlauf und Kontrolle der Mittelzuweisung
- Auswahl von Ressourcen der richtigen Art und Menge
- Skalierung zur Erfüllung der Geschäftsanforderungen ohne Mehrausgaben

Verstehen Sie die Nutzungsmuster und die benötigten Dienste

Neptune eignet sich gut für Ihren Workload, wenn Ihr Datenmodell eine erkennbare Graphstruktur hat und Ihre Abfragen Beziehungen untersuchen und mehrere Hops durchlaufen müssen. Eine Graphdatenbank eignet sich nicht für die folgenden Muster:

- Hauptsächlich Single-Hop-Abfragen (überlegen Sie, ob Ihre Daten besser als Attribute eines Objekts dargestellt werden könnten)
- JSON- oder BLOB-Daten, die als Eigenschaften gespeichert werden
- Abfragen, die sich über einen Datensatz hinweg aggregieren, z. B. die Berechnung der Summe einer numerischen Eigenschaft über eine große Anzahl von Knoten

Überlegen Sie, ob die gemeinsame Verwendung mehrerer speziell entwickelter Datenbanken für bestimmte Zugriffsmuster all Ihre Anforderungen erfüllen könnte. Zum Beispiel:

- Eine API, die weniger häufige komplexe Graphnavigationen erfordert und gleichzeitig Eigenschaften für einen einzelnen Knoten gleichzeitig abgerufen werden muss, lässt sich am besten mit einem oder mehreren von Neptune, DynamoDB oder Amazon DocumentDB präsentieren.
- Relationale Datenbanken können mit Neptune koexistieren, um Ihre bestehende Funktionalität beizubehalten. Verwenden Sie Neptune jedoch nur für Multiple-Hop-Traversals, die in relationalen Datenbanken nicht gut funktionieren und nicht gut skalieren.

Machen Sie sich mit den Kosten vertraut, die mit Diensten verbunden sind, die mit Neptune interagieren und diese ergänzen, einschließlich der folgenden:

- Speicherkosten für Amazon Simple Storage Service (Amazon S3) für Datendateien, die massenweise in Neptune geladen werden
- Lambda-Funktionen, die für Insert- oder Upsert-Abfragen, Leseabfragen und die Verarbeitung von Neptune-Streams verwendet werden
- Die API-Schicht, die auf Neptune basiert, um mit der Client-Anwendung zu interagieren (anstatt direkte Verbindungen zur Datenbank herzustellen) in oder AWS AppSync
- AWS Glue Jobs, die zum Übertragen von Daten zu und von Neptune verwendet werden
- Amazon Kinesis- oder Amazon Managed Streaming for Apache Kafka (Amazon MSK) -Instances empfangen Streaming-Daten für die Aufnahme nahezu in Echtzeit in Neptune.
- AWS Database Migration Service für die Migration relationaler Daten nach Neptune
- Amazon SageMaker Runtime-Kosten für Jupyter-Notebooks und Machine-Learning-Modelle mit Deep Graph Library

Wählen Sie Ressourcen unter Berücksichtigung der Kosten aus

Die Neptune-Preise basieren auf den stündlichen Instanzkosten (oder den Verbrauch von Neptune-Recheneinheiten bei serverlosem Betrieb), Daten-I/O und Speichernutzung. Instanzen machen im Durchschnitt 85 Prozent der Gesamtkosten aus, sodass die richtige Dimensionierung erhebliche Auswirkungen auf die Kosten haben kann. Die beste Methode zur richtigen Größe von Instances besteht darin, die Anwendungsleistung auf einer Vielzahl von Instances zu testen und die folgenden Faktoren zu vergleichen:

- Bleibt die `MainRequestQueuePendingRequests` CloudWatch Metrik auf einem konstant niedrigen Wert nahe Null?
- Bleibt die `BufferCacheHitRatio` CloudWatch Kennzahl die meiste Zeit bei oder über 99,9 Prozent?
- Wie sehen die Kosten- und Leistungskurven für Instance-Kosten und die damit verbundenen Daten-I/O-Kosten aus? Die Kosten für das Lesen von Daten können bei einer unterdimensionierten Instance, die ein häufiges Austauschen des Puffer-Caches mit dem Speicher erfordert, erheblich steigen. `BufferCacheHitRatio`werden in diesen Szenarien häufig sinken.

Die Instanzkosten innerhalb derselben Instance-Familie skalieren linear mit der Größe. Die Stundenkosten der `db.r6i.2xlarge` Instance sind doppelt so hoch wie die der `db.r6i.xlarge` Instance und haben zudem die doppelte Ressourcenzuweisung. Die `db.r6i.24xlarge` Instanz kostet das 24-fache der stündlichen Kosten der `db.r6i.xlarge` Instanz.

Schätzen Sie die Anzahl der gleichzeitigen Abfragen, die Sie unterstützen müssen. Sie können zwischen null und fünfzehn Read Replicas für die Verarbeitung schreibgeschützter Abfragen verwenden. Wenn Ihre Anforderungen je nach Tages-, Wochen- oder Monatszeit variieren, können Sie mehrere kleinere Instances verwenden, um nach einem Zeitplan zu skalieren. Jede vCPU auf einer Instanz stellt zwei Threads für die Verarbeitung gleichzeitiger Abfragen bereit. Drei `db.r6i.xlarge` Read Replicas mit jeweils 4 vCPUs können 24 gleichzeitige Abfragen verarbeiten.

Wenn Ihr Datenverkehrsvolumen stattdessen in Abfragen pro Sekunde (QPS) gemessen wird, müssen Sie experimentieren, um die durchschnittliche Latenz Ihrer Abfragen zu ermitteln. Die Anzahl der Abfragen pro Sekunde, die ein Neptun-Cluster unterstützen kann, entspricht. `vCPU × 2 × (1 second/average query latency)` Wenn Sie beispielsweise über 4 vCPUs und eine Abfragelatenz von 100 Millisekunden (0,1 Sekunden) verfügen,. `QPS = 4 × 2 × (1s/0.1s) = 80 queries per second`

Für kontinuierliche, stabile und vorhersehbare Workloads sind bereitgestellte Instanzen günstiger als serverlose Instanzen. Serverless bietet Möglichkeiten zur Kostenoptimierung, wenn Sie eine Arbeitslast haben, die nur wenige Stunden pro Tag sehr stark ausgelastet ist (z. B.`db.r6i.4xlarge`) und dann für den Rest des Tages fast keinen Verkehr erfordert (z. B. 1 Neptune Compute Unit). Eine serverlose Instanz, die für einige Stunden hochskaliert und dann wieder heruntergefahren wird, ist günstiger als die ganztägige Nutzung einer bereitgestellten `db.r6i.4xlarge` Instanz.

Wählen Sie die beste Neptune-Instanzkonfiguration für Ihren Workload

Für Experimente auf Einstiegsebene mit Neptune können Sie das kostenlose Kontingent für AWS nutzen. Die 750 kostenlosen Stunden `db.t3.medium` und die Nutzung der `db.t4g.medium` Instanzen reichen aus, um Neptune in geringem Umfang gut zu verstehen. Ihr Cluster bleibt auch nach Ablauf der kostenlosen Testphase bestehen, obwohl Ihnen ab diesem Zeitpunkt die Nutzung in Rechnung gestellt wird.

Die `db.t4g.medium` Instances `db.t3.medium` und eignen sich gut für kostengünstige Entwicklungsumgebungen. Beachten Sie jedoch, dass sie ein kleineres Verhältnis von RAM

zu vCPU (2:1) als die Instances der R-Familie (8:1) oder Instanzen der X-Familie (16:1) haben. Leistungsprofile können sich von diesen Klassen unterscheiden, insbesondere in Bezug auf `OutOfMemoryExceptions` und wann Abfragen über einen wesentlichen Teil des Diagramms navigieren. Überprüfen Sie die `BufferCacheHitRatio` CloudWatch Metrik, um festzustellen, ob die letztgenannte Bedingung betroffen sein könnte.

Wir raten dringend davon ab, Leistungs- oder Lasttests mit Instances der T-Familie durchzuführen, da es zu inkonsistenten Ergebnissen kommen kann, die nicht auf eine Produktionsumgebung hinweisen.

Bereitgestellte Instances bieten Ihnen die beste Kombination aus Kosten und Leistung, wenn Ihre Arbeitslast relativ stabil und vorhersehbar ist. Wählen Sie die Instanzgröße auf der Grundlage der erforderlichen Parallelität der Anfragen und der Komplexität der Abfrage. Für eine höhere Parallelität ist mehr v erforderlich. CPUs Eine höhere Abfragekomplexität erfordert mehr RAM. Ermitteln Sie anhand der `MainRequestQueuePendingRequests` CloudWatch Metrik, wie sich Ersteres auswirkt (ein Wert größer als Null steht für mehr gleichzeitige Anfragen, als bearbeitet werden können). Verwenden Sie die `BufferCacheHitRatio` CloudWatch Metrik, um die Auswirkung der letzteren zu ermitteln. Ein Verhältnis, das häufig unter 99,9 Prozent fällt, deutet darauf hin, dass nicht genug RAM vorhanden ist, um den aktiven Teil des auszuwertenden Diagramms aufzunehmen, was zu häufigerem Cache-Swapping führt. Wenn die R-Instance-Familie ausreichend Parallelität, aber nicht genug RAM bietet, sollten Sie die X-Instance-Familie ausprobieren.

Ideale Anwendungsfälle für serverlose Instanzen sind in der Neptune-Dokumentation beschrieben. Wenn Sie sich nicht sicher sind, ob bereitgestellte oder serverlose Workloads für Sie am besten geeignet sind und die Kosten Ihr Hauptanliegen sind, testen Sie Ihren Workload im serverlosen Modus, um die Anzahl der NCUs genutzten Workloads zu ermitteln und die Kosten für bereitgestellte () mit denen von serverlos () zu vergleichen. `N hours × hourly provisioned cost sum of NCUs × hourly cost per NCU` Wenn Sie sich nicht sicher sind, welche Provision-Instanz die entsprechende Größe hat, entspricht eine NCU etwa 2 GB RAM und der zugehörigen vCPU und dem Netzwerk. Wenn Ihre bereitgestellte Instance aus der R6i-Familie stammt, beträgt das Verhältnis 1 vCPU pro 8 GB RAM oder 4 NCUs, zusammen mit dem zugehörigen Netzwerk.

Wenn Sie Serverless für Primär- und Replikatinstanzen verwenden, denken Sie daran, dass Read Replicas der Upgrade-Stufen 0 und 1 entsprechend der Writer-Instanz skaliert werden, sodass sie bei einem Failover-Ereignis ordnungsgemäß skaliert werden. NCUs Legen Sie Ihre NCU-Grenzwerte für diese Instances danach fest, welche Ihrer Instances — Writer oder Reader — den meisten Traffic erhalten.

In Umgebungen, in denen der Cluster nicht 24 Stunden am Tag, 7 Tage die Woche benötigt wird, sollten Sie in Erwägung ziehen, Skripte zu schreiben, die die Neptune-Instanzen ausschalten,

wenn sie nicht verwendet werden, und sie erneut starten, bevor sie verwendet werden. Neptune-Instanzen werden automatisch alle 7 Tage neu gestartet, um sicherzustellen, dass die erforderlichen Wartungsupdates installiert werden. Wenn Sie beabsichtigen, die Instanzen für längere Zeit ausgeschaltet zu lassen, verwenden Sie ein wöchentliches Skript, um sie wieder herunterzufahren.

Datenspeicherung und -übertragung in der richtigen Größe

Effizientere Abfragen (z. B. Abfragen, die weniger Knoten, Kanten und Eigenschaften im Diagramm berühren müssen) erfordern weniger I/O-Übertragung und können möglicherweise kleinere Instanzen verwenden, da weniger Puffer-Cache erforderlich ist. Verwenden Sie das Profil oder die Explain-Endpunkte für Ihre Abfragesprache, um Ihre Abfrage zu optimieren, und ziehen Sie in Betracht, Ihr Grafikmodell im Hinblick auf Ihre Abfrageleistung zu optimieren.

Neptune verwendet die Wörterbuchkodierung für große Zeichenketten, und dieses Wörterbuch ist für Leistung und nicht für Effizienz optimiert. Wenn Sie große BLOBs, JSON-Zeichenketten oder häufig wechselnde Zeichenketten für Eigenschaften haben, sollten Sie erwägen, diese außerhalb von Neptune in Amazon S3, Amazon DynamoDB oder Amazon DocumentDB zu speichern und nur eine Referenz innerhalb des Neptune-Knotens zu speichern.

In einigen Fällen kann es günstiger sein, eine größere Instance-Größe zu wählen. Wenn Ihre I/O-Kosten aufgrund einer niedrigen Größe sehr hoch sind`BufferCacheHitRatio`, ist es möglich, dass der größere Puffer-Cache diese Kosten erheblich reduzieren würde. Das liegt daran, dass alle Daten in den Cache passen würden, anstatt häufig aus dem Speicher ausgelagert zu werden, was zu einer höheren I/O-Übertragungsrate führen würde.

Neptune verwendet copy-on-write Klonen. Beim Klonen, um ein Diagramm in mehrere Shards aufzuteilen, ist es möglicherweise effizienter, die unerwünschten Daten auf dem geklonten Cluster nicht zu löschen, da dafür neue Datenseiten erstellt werden müssen, was zu erhöhten Speicherkosten führt. Daten, die gegenüber der Zeit vor dem Klonen unverändert geblieben sind, befinden sich auf einer einzigen Datenseite, die von den beiden Clustern gemeinsam genutzt wird, und es fallen nur Gebühren für diese einzelne Kopie an.

Aktivieren Sie den OSGP-Index nicht und verwenden Sie keine R5d-Instances, es sei denn, Sie haben getestet, um zu bestätigen, dass sie einen wesentlichen Unterschied in Ihrer Arbeitslast bewirken. Beide sind für selten auftretende Szenarien konzipiert und können Ihre Kosten bei minimalem oder gar keinem Nutzen erhöhen.

Säule der Nachhaltigkeit

Der Schwerpunkt der Nachhaltigkeit liegt auf der Minimierung der Umweltauswirkungen der Ausführung von Cloud-Workloads. Zu den wichtigsten Themen gehören ein Modell der gemeinsamen Verantwortung für Nachhaltigkeit, das Verständnis der Auswirkungen und die Maximierung der Nutzung, um die benötigten Ressourcen zu minimieren und die nachgelagerten Auswirkungen zu reduzieren.

Die Säule Nachhaltigkeit umfasst die folgenden Schwerpunktbereiche:

- Ihr Einfluss
- Ziele im Bereich Nachhaltigkeit
- Maximierte Nutzung
- Antizipierung und Einführung neuer, effizienterer Hardware- und Softwareangebote
- Nutzung von Managed Services
- Reduzierung der nachgelagerten Auswirkungen

Dieser Leitfaden konzentriert sich auf Ihre Wirkung. Weitere Informationen zu den anderen Prinzipien des Nachhaltigkeitsdesigns finden Sie im AWS Well-Architected Framework.

Ihre Entscheidungen und Anforderungen wirken sich auf die Umwelt aus. Wenn Sie sich für eine geringere Kohlenstoffintensität entscheiden AWS-Regionen können und Ihre Anforderungen den tatsächlichen Arbeitsanforderungen entsprechen, anstatt nur die Verfügbarkeit und Haltbarkeit zu maximieren, erhöht sich die Nachhaltigkeit der Arbeitslast. In den nächsten Abschnitten werden bewährte Verfahren und durchdachte Überlegungen erörtert, die sich positiv auf die Umwelt auswirken können, wenn sie bei der Planung Ihrer Arbeitslast und Ihrem laufenden Betrieb berücksichtigt werden.

Auswahl der AWS-Region

Einige AWS-Regionen befinden sich in der Nähe von Amazonas-Projekten für erneuerbare Energien oder dort, wo das Netz eine veröffentlichte Kohlenstoffintensität aufweist, die niedriger ist als bei anderen. Berücksichtigen Sie die Auswirkungen auf die Nachhaltigkeit der Regionen, die für Ihre Arbeitsbelastung in Frage kommen könnten, und vergleichen Sie Ihre Liste mit den Regionen, in denen Neptune verfügbar ist.

Der Konsum basiert auf Verhaltensmustern der Nutzer

Wenn Sie Ihren Verbrauch an den Traffic und das Verhalten Ihrer Nutzer anpassen, können Sie die Auswirkungen von Diensten auf die Umwelt AWS minimieren. Beachten Sie bei der Entwicklung Ihrer Lösung die folgenden bewährten Methoden:

- Überwachen Sie CloudWatch Amazon-Metriken wie `CPUUtilizationMainRequestQueuePendingRequests`, und, `TotalRequestsPerSec` um festzustellen, wann Ihr Bedarf am höchsten und am niedrigsten ist, und stellen Sie sicher, dass Ihre Cluster-Ressourcen in diesen Zeiten die richtige Größe haben.
- Automatisieren Sie das Stoppen von Umgebungen außerhalb der Produktion während der Stunden, in denen sie nicht genutzt werden. Weitere Informationen finden Sie im Blogbeitrag Automatisieren Sie das Stoppen und Starten von Amazon Neptune Neptune-Umgebungsressourcen mithilfe von Ressourcen-Tags.
- Wenn Ihre Datenverkehrsmuster häufig und unvorhersehbar variieren, sollten Sie die Verwendung von Neptune Serverless-Instances in Betracht ziehen, die je nach Bedarf hoch- und herunterskalieren, anstatt eine Instanz zu verwenden, die für Spitzenverkehr bereitgestellt wurde.
- Erwägen Sie, Ihre Service Level Agreements neben den Zielen zur Geschäftskontinuität auch an Nachhaltigkeitszielen auszurichten. Durch die Vereinfachung von Anforderungen wie Disaster Recovery in mehreren Regionen, hohe Verfügbarkeit oder langfristige Aufbewahrung von Backups, insbesondere für Umgebungen außerhalb der Produktion oder für nicht geschäftskritische Workloads, kann die Menge der Ressourcen reduziert werden, die zur Erreichung dieser Ziele erforderlich sind.

Optimieren Sie die Softwareentwicklung und Architekturmuster

Um Verschwendung zu vermeiden, sollten Sie Ihre Modelle und Abfragen optimieren und Rechenressourcen gemeinsam nutzen, sodass Sie alle in Neptune-Instances und -Clustern verfügbaren Ressourcen nutzen können. Zu den spezifischen Best Practices gehören:

- Lassen Sie Entwickler Neptune-Instanzen und Jupyter Notebook-Anwendungsinstanzen gemeinsam nutzen, anstatt jeweils ihre eigenen zu erstellen. Geben Sie jedem Entwickler mithilfe von Multi-Tenancy-Partitionierungsstrategien seine eigene logische Partition in einem einzigen Neptune-Cluster und erstellen Sie separate Notebook-Ordner für jeden Entwickler auf einer einzigen Jupyter-Instanz.

- Implementieren Sie Muster, die den Ressourcenverbrauch maximieren und Leerlaufzeiten minimieren, z. B. parallel Threads zum Laden von Daten und zum Zusammenfügen von Datensätzen zu einer größeren Transaktion.
- Optimieren Sie Ihre Abfragen und Ihr Grafikmodell, um den Ressourcenaufwand für die Berechnung der Ergebnisse zu minimieren.
- Verwenden Sie für Gremlin-Abfrageergebnisse die Funktion zum Zwischenspeichern von Ergebnissen, um den Ressourcenaufwand für die Neuberechnung paginierter oder häufig wiederkehrender Abfragen zu minimieren.
- Halten Sie Ihre Neptun-Umgebungen auf dem neuesten Stand. Die neuesten Versionen von Neptune unterstützen die neuesten EC2 Instanzen wie Graviton, die effizienter sind. Sie bieten auch Verbesserungen bei der Abfrageoptimierung und Fehlerkorrekturen, die den Ressourcenaufwand für die Berechnung Ihrer Abfragen reduzieren.

Ressourcen

Referenzen

- AWS Well-Architected
- AWS Well-Architected-Framework-Dokumentation
- Anwendung des AWS Well-Architected Frameworks für Amazon Neptune Analytics
- Aktuelle Updates für Amazon Neptune
- Best Practices: Das Beste aus Neptune herausholen

Blog-Posts

- Automatisiertes Testen des Amazon Neptune Neptune-Datenzugriffs mit Apache Gremlin TinkerPop
- Automatisieren Sie das Stoppen und Starten von Amazon Neptune Neptune-Umgebungsressourcen mithilfe von Ressourcen-Tags
- Feinkörnige Zugriffskontrolle für Amazon Neptune Neptune-Aktionen auf der Datenebene
- Nutzen Sie semantisches Denken, um mithilfe der Integration mit Amazon Neptune neue Fakten aus Ihrem RDF-Diagramm abzuleiten RDFox
- Entwickeln Sie mit Amazon Neptune ML eine Lösung zur Betrugserkennung in Echtzeit

Kostenlose AWS Skill Builder-Kurse

- Erste Schritte mit Amazon Neptune
- Anwendungen auf Amazon Neptune erstellen
- Datenmodellierung für Amazon Neptune

Mitwirkende

Zu den Mitwirkenden an diesem Leitfaden gehören:

- Brian O'Keefe, leitender Architekt von Neptune Solutions, AWS
- Abhishek Mishra, leitender Architekt von Neptune Solutions, AWS
- Ganesh Sawhney, Teamleiter — Architekt für Erfolgslösungen für strategische Partner, AWS
- Michael Havey, leitender Architekt von Neptune Solutions, AWS
- Kevin Phillips, Architekt von Neptune Solutions, AWS
- Melissa Kwok, Architektin von Neptune Solutions, AWS
- Sakti Mishra, Hauptarchitektin für Lösungen AWS
- Javed Ali, leitender Lösungsarchitekt, AWS

Dokumentverlauf

In der folgenden Tabelle werden wichtige Änderungen in diesem Leitfaden beschrieben. Um Benachrichtigungen über zukünftige Aktualisierungen zu erhalten, können Sie einen RSS-Feed abonnieren.

Änderung	Beschreibung	Datum
Erste Veröffentlichung	—	27. September 2023

AWS Glossar zu präskriptiven Leitlinien

Die folgenden Begriffe werden häufig in Strategien, Leitfäden und Mustern von AWS Prescriptive Guidance verwendet. Um Einträge vorzuschlagen, verwenden Sie bitte den Link Feedback geben am Ende des Glossars.

Zahlen

7 Rs

Sieben gängige Migrationsstrategien für die Verlagerung von Anwendungen in die Cloud. Diese Strategien bauen auf den 5 Rs auf, die Gartner 2011 identifiziert hat, und bestehen aus folgenden Elementen:

- Faktorwechsel/Architekturwechsel – Verschieben Sie eine Anwendung und ändern Sie ihre Architektur, indem Sie alle Vorteile cloudnativer Feature nutzen, um Agilität, Leistung und Skalierbarkeit zu verbessern. Dies beinhaltet in der Regel die Portierung des Betriebssystems und der Datenbank. Beispiel: Migrieren Sie Ihre lokale Oracle-Datenbank auf die Amazon Aurora PostgreSQL-kompatible Edition.
- Plattformwechsel (Lift and Reshape) – Verschieben Sie eine Anwendung in die Cloud und führen Sie ein gewisses Maß an Optimierung ein, um die Cloud-Funktionen zu nutzen. Beispiel: Migrieren Sie Ihre lokale Oracle-Datenbank zu Amazon Relational Database Service (Amazon RDS) für Oracle in der. AWS Cloud
- Neukauf (Drop and Shop) – Wechseln Sie zu einem anderen Produkt, indem Sie typischerweise von einer herkömmlichen Lizenz zu einem SaaS-Modell wechseln. Beispiel: Migrieren Sie Ihr CRM-System (Customer Relationship Management) zu Salesforce.com.
- Hostwechsel (Lift and Shift) – Verschieben Sie eine Anwendung in die Cloud, ohne Änderungen vorzunehmen, um die Cloud-Funktionen zu nutzen. Beispiel: Migrieren Sie Ihre lokale Oracle-Datenbank zu Oracle auf einer EC2 Instanz in der. AWS Cloud
- Verschieben (Lift and Shift auf Hypervisor-Ebene) – Verlagern Sie die Infrastruktur in die Cloud, ohne neue Hardware kaufen, Anwendungen umschreiben oder Ihre bestehenden Abläufe ändern zu müssen. Sie migrieren Server von einer lokalen Plattform zu einem Cloud-Dienst für dieselbe Plattform. Beispiel: Migrieren Sie eine Microsoft Hyper-V Anwendung zu AWS.
- Beibehaltung (Wiederaufgreifen) – Bewahren Sie Anwendungen in Ihrer Quellumgebung auf. Dazu können Anwendungen gehören, die einen umfangreichen Faktorwechsel erfordern und

die Sie auf einen späteren Zeitpunkt verschieben möchten, sowie ältere Anwendungen, die Sie beibehalten möchten, da es keine geschäftliche Rechtfertigung für ihre Migration gibt.

- Außerbetriebnahme – Dekommissionierung oder Entfernung von Anwendungen, die in Ihrer Quellumgebung nicht mehr benötigt werden.

A

ABAC

Siehe attributbasierte Zugriffskontrolle.

abstrahierte Dienste

Weitere Informationen finden Sie unter Managed Services.

ACID

Siehe Atomarität, Konsistenz, Isolierung und Haltbarkeit.

Aktiv-Aktiv-Migration

Eine Datenbankmigrationsmethode, bei der die Quell- und Zieldatenbanken synchron gehalten werden (mithilfe eines bidirektionalen Replikationstools oder dualer Schreibvorgänge) und beide Datenbanken Transaktionen von miteinander verbundenen Anwendungen während der Migration verarbeiten. Diese Methode unterstützt die Migration in kleinen, kontrollierten Batches, anstatt einen einmaligen Cutover zu erfordern. Es ist flexibler, erfordert aber mehr Arbeit als eine aktiv-passive Migration.

Aktiv-Passiv-Migration

Eine Datenbankmigrationsmethode, bei der die Quell- und Zieldatenbanken synchron gehalten werden, aber nur die Quelldatenbank Transaktionen von verbindenden Anwendungen verarbeitet, während Daten in die Zieldatenbank repliziert werden. Die Zieldatenbank akzeptiert während der Migration keine Transaktionen.

Aggregatfunktion

Eine SQL-Funktion, die mit einer Gruppe von Zeilen arbeitet und einen einzelnen Rückgabewert für die Gruppe berechnet. Beispiele für Aggregatfunktionen sind `SUM` und`MAX`.

AI

Siehe künstliche Intelligenz.

AIOps

Siehe Operationen im Bereich künstliche Intelligenz.

Anonymisierung

Der Prozess des dauerhaften Löschens personenbezogener Daten in einem Datensatz. Anonymisierung kann zum Schutz der Privatsphäre beitragen. Anonymisierte Daten gelten nicht mehr als personenbezogene Daten.

Anti-Muster

Eine häufig verwendete Lösung für ein wiederkehrendes Problem, bei dem die Lösung kontraproduktiv, ineffektiv oder weniger wirksam als eine Alternative ist.

Anwendungssteuerung

Ein Sicherheitsansatz, bei dem nur zugelassene Anwendungen verwendet werden können, um ein System vor Schadsoftware zu schützen.

Anwendungsportfolio

Eine Sammlung detaillierter Informationen zu jeder Anwendung, die von einer Organisation verwendet wird, einschließlich der Kosten für die Erstellung und Wartung der Anwendung und ihres Geschäftswerts. Diese Informationen sind entscheidend für den Prozess der Portfoliofindung und -analyse und hilft bei der Identifizierung und Priorisierung der Anwendungen, die migriert, modernisiert und optimiert werden sollen.

künstliche Intelligenz (KI)

Das Gebiet der Datenverarbeitungswissenschaft, das sich der Nutzung von Computertechnologien zur Ausführung kognitiver Funktionen widmet, die typischerweise mit Menschen in Verbindung gebracht werden, wie Lernen, Problemlösen und Erkennen von Mustern. Weitere Informationen finden Sie unter Was ist künstliche Intelligenz?

Operationen mit künstlicher Intelligenz (AIOps)

Der Prozess des Einsatzes von Techniken des Machine Learning zur Lösung betrieblicher Probleme, zur Reduzierung betrieblicher Zwischenfälle und menschlicher Eingriffe sowie zur Steigerung der Servicequalität. Weitere Informationen zur Verwendung in der AWS Migrationsstrategie finden Sie im Operations Integration Guide. AIOps

Asymmetrische Verschlüsselung

Ein Verschlüsselungsalgorithmus, der ein Schlüsselpaar, einen öffentlichen Schlüssel für die Verschlüsselung und einen privaten Schlüssel für die Entschlüsselung verwendet. Sie können den

öffentlichen Schlüssel teilen, da er nicht für die Entschlüsselung verwendet wird. Der Zugriff auf den privaten Schlüssel sollte jedoch stark eingeschränkt sein.

Atomizität, Konsistenz, Isolierung, Haltbarkeit (ACID)

Eine Reihe von Softwareeigenschaften, die die Datenvalidität und betriebliche Zuverlässigkeit einer Datenbank auch bei Fehlern, Stromausfällen oder anderen Problemen gewährleisten.

Attributbasierte Zugriffskontrolle (ABAC)

Die Praxis, detaillierte Berechtigungen auf der Grundlage von Benutzerattributen wie Abteilung, Aufgabenrolle und Teamname zu erstellen. Weitere Informationen finden Sie unter ABAC AWS in der AWS Identity and Access Management (IAM-) Dokumentation.

autoritative Datenquelle

Ein Ort, an dem Sie die primäre Version der Daten speichern, die als die zuverlässigste Informationsquelle angesehen wird. Sie können Daten aus der maßgeblichen Datenquelle an andere Speicherorte kopieren, um die Daten zu verarbeiten oder zu ändern, z. B. zu anonymisieren, zu redigieren oder zu pseudonymisieren.

Availability Zone

Ein bestimmter Standort innerhalb einer AWS-Region , der vor Ausfällen in anderen Availability Zones geschützt ist und kostengünstige Netzwerkkonnektivität mit niedriger Latenz zu anderen Availability Zones in derselben Region bietet.

AWS Framework für die Cloud-Einführung (AWS CAF)

Ein Framework mit Richtlinien und bewährten Verfahren, das Unternehmen bei der Entwicklung eines effizienten und effektiven Plans für die erfolgreiche Umstellung auf die Cloud unterstützt. AWS AWS CAF unterteilt die Leitlinien in sechs Schwerpunktbereiche, die als Perspektiven bezeichnet werden: Unternehmen, Mitarbeiter, Unternehmensführung, Plattform, Sicherheit und Betrieb. Die Perspektiven Geschäft, Mitarbeiter und Unternehmensführung konzentrieren sich auf Geschäftskompetenzen und -prozesse, während sich die Perspektiven Plattform, Sicherheit und Betriebsabläufe auf technische Fähigkeiten und Prozesse konzentrieren. Die Personalperspektive zielt beispielsweise auf Stakeholder ab, die sich mit Personalwesen (HR), Personalfunktionen und Personalmanagement befassen. Aus dieser Perspektive bietet AWS CAF Leitlinien für Personalentwicklung, Schulung und Kommunikation, um das Unternehmen auf eine erfolgreiche Cloud-Einführung vorzubereiten. Weitere Informationen finden Sie auf der AWS -CAF-Webseite und dem AWS -CAF-Whitepaper.

AWS Workload-Qualifizierungsrahmen (AWS WQF)

Ein Tool, das Workloads bei der Datenbankmigration bewertet, Migrationsstrategien empfiehlt und Arbeitsschätzungen bereitstellt. AWS WQF ist in () enthalten. AWS Schema Conversion Tool AWS SCT Es analysiert Datenbankschemas und Codeobjekte, Anwendungscode, Abhängigkeiten und Leistungsmerkmale und stellt Bewertungsberichte bereit.

B

schlechter Bot

Ein Bot, der Einzelpersonen oder Organisationen stören oder ihnen Schaden zufügen soll.

BCP

Siehe Planung der Geschäftskontinuität.

Verhaltensdiagramm

Eine einheitliche, interaktive Ansicht des Ressourcenverhaltens und der Interaktionen im Laufe der Zeit. Sie können ein Verhaltensdiagramm mit Amazon Detective verwenden, um fehlgeschlagene Anmeldeversuche, verdächtige API-Aufrufe und ähnliche Vorgänge zu untersuchen. Weitere Informationen finden Sie unter Daten in einem Verhaltensdiagramm in der Detective-Dokumentation.

Big-Endian-System

Ein System, welches das höchstwertige Byte zuerst speichert. Siehe auch Endianness.

Binäre Klassifikation

Ein Prozess, der ein binäres Ergebnis vorhersagt (eine von zwei möglichen Klassen). Beispielsweise könnte Ihr ML-Modell möglicherweise Probleme wie „Handelt es sich bei dieser E-Mail um Spam oder nicht?“ vorhersagen müssen oder „Ist dieses Produkt ein Buch oder ein Auto?“

Bloom-Filter

Eine probabilistische, speichereffiziente Datenstruktur, mit der getestet wird, ob ein Element Teil einer Menge ist.

Blau/Grün-Bereitstellung

Eine Bereitstellungsstrategie, bei der Sie zwei separate, aber identische Umgebungen erstellen. Sie führen die aktuelle Anwendungsversion in einer Umgebung (blau) und die neue

Anwendungsversion in der anderen Umgebung (grün) aus. Mit dieser Strategie können Sie schnell und mit minimalen Auswirkungen ein Rollback durchführen.

Bot

Eine Softwareanwendung, die automatisierte Aufgaben über das Internet ausführt und menschliche Aktivitäten oder Interaktionen simuliert. Manche Bots sind nützlich oder nützlich, wie z. B. Webcrawler, die Informationen im Internet indexieren. Einige andere Bots, sogenannte bösartige Bots, sollen Einzelpersonen oder Organisationen stören oder ihnen Schaden zufügen.

Botnetz

Netzwerke von Bots, die mit Malware infiziert sind und unter der Kontrolle einer einzigen Partei stehen, die als Bot-Herder oder Bot-Operator bezeichnet wird. Botnetze sind der bekannteste Mechanismus zur Skalierung von Bots und ihrer Wirkung.

branch

Ein containerisierter Bereich eines Code-Repositorys. Der erste Zweig, der in einem Repository erstellt wurde, ist der Hauptzweig. Sie können einen neuen Zweig aus einem vorhandenen Zweig erstellen und dann Feature entwickeln oder Fehler in dem neuen Zweig beheben. Ein Zweig, den Sie erstellen, um ein Feature zu erstellen, wird allgemein als Feature-Zweig bezeichnet. Wenn das Feature zur Veröffentlichung bereit ist, führen Sie den Feature-Zweig wieder mit dem Hauptzweig zusammen. Weitere Informationen finden Sie unter Über Branches (GitHub Dokumentation).

Zugang durch Glasbruch

Unter außergewöhnlichen Umständen und im Rahmen eines genehmigten Verfahrens ist dies eine schnelle Methode für einen Benutzer, auf einen Bereich zuzugreifen AWS-Konto , für den er in der Regel keine Zugriffsrechte besitzt. Weitere Informationen finden Sie unter dem Indikator Implementation break-glass procedures in den AWS Well-Architected-Leitlinien.

Brownfield-Strategie

Die bestehende Infrastruktur in Ihrer Umgebung. Wenn Sie eine Brownfield-Strategie für eine Systemarchitektur anwenden, richten Sie sich bei der Gestaltung der Architektur nach den Einschränkungen der aktuellen Systeme und Infrastruktur. Wenn Sie die bestehende Infrastruktur erweitern, könnten Sie Brownfield- und Greenfield-Strategien mischen.

Puffer-Cache

Der Speicherbereich, in dem die am häufigsten abgerufenen Daten gespeichert werden.

Geschäftsfähigkeit

Was ein Unternehmen tut, um Wert zu generieren (z. B. Vertrieb, Kundenservice oder Marketing). Microservices-Architekturen und Entwicklungsentscheidungen können von den Geschäftskapazitäten beeinflusst werden. Weitere Informationen finden Sie im Abschnitt Organisiert nach Geschäftskapazitäten des Whitepapers Ausführen von containerisierten Microservices in AWS.

Planung der Geschäftskontinuität (BCP)

Ein Plan, der die potenziellen Auswirkungen eines störenden Ereignisses, wie z. B. einer groß angelegten Migration, auf den Betrieb berücksichtigt und es einem Unternehmen ermöglicht, den Betrieb schnell wieder aufzunehmen.

C

CAF

Weitere Informationen finden Sie unter Framework für die AWS Cloud-Einführung.

Bereitstellung auf Kanaren

Die langsame und schrittweise Veröffentlichung einer Version für Endbenutzer. Wenn Sie sich sicher sind, stellen Sie die neue Version bereit und ersetzen die aktuelle Version vollständig.

CCoE

Weitere Informationen finden Sie im Cloud Center of Excellence.

CDC

Siehe Erfassung von Änderungsdaten.

Erfassung von Datenänderungen (CDC)

Der Prozess der Nachverfolgung von Änderungen an einer Datenquelle, z. B. einer Datenbanktabelle, und der Aufzeichnung von Metadaten zu der Änderung. Sie können CDC für verschiedene Zwecke verwenden, z. B. für die Prüfung oder Replikation von Änderungen in einem Zielsystem, um die Synchronisation aufrechtzuerhalten.

Chaos-Technik

Absichtliches Einführen von Ausfällen oder Störungsereignissen, um die Widerstandsfähigkeit eines Systems zu testen. Sie können AWS Fault Injection Service (AWS FIS) verwenden, um Experimente durchzuführen, die Ihre AWS Workloads stress, und deren Reaktion zu bewerten.

CI/CD

Siehe Continuous Integration und Continuous Delivery.

Klassifizierung

Ein Kategorisierungsprozess, der bei der Erstellung von Vorhersagen hilft. ML-Modelle für Klassifikationsprobleme sagen einen diskreten Wert voraus. Diskrete Werte unterscheiden sich immer voneinander. Beispielsweise muss ein Modell möglicherweise auswerten, ob auf einem Bild ein Auto zu sehen ist oder nicht.

clientseitige Verschlüsselung

Lokale Verschlüsselung von Daten, bevor das Ziel sie AWS-Service empfängt.

Cloud-Exzellenzzentrum (CCoE)

Ein multidisziplinäres Team, das die Cloud-Einführung in der gesamten Organisation vorantreibt, einschließlich der Entwicklung bewährter Cloud-Methoden, der Mobilisierung von Ressourcen, der Festlegung von Migrationszeitplänen und der Begleitung der Organisation durch groß angelegte Transformationen. Weitere Informationen finden Sie in den CCoE-Beiträgen im AWS Cloud Enterprise Strategy Blog.

Cloud Computing

Die Cloud-Technologie, die typischerweise für die Ferndatenspeicherung und das IoT-Gerätemanagement verwendet wird. Cloud Computing ist häufig mit Edge-Computing-Technologie verbunden.

Cloud-Betriebsmodell

In einer IT-Organisation das Betriebsmodell, das zum Aufbau, zur Weiterentwicklung und Optimierung einer oder mehrerer Cloud-Umgebungen verwendet wird. Weitere Informationen finden Sie unter Aufbau Ihres Cloud-Betriebsmodells.

Phasen der Einführung der Cloud

Die vier Phasen, die Unternehmen bei der Migration in der Regel durchlaufen AWS Cloud:

- Projekt – Durchführung einiger Cloud-bezogener Projekte zu Machbarkeitsnachweisen und zu Lernzwecken
- Fundament — Tätigen Sie grundlegende Investitionen, um Ihre Cloud-Einführung zu skalieren (z. B. Einrichtung einer landing zone, Definition eines CCo E, Einrichtung eines Betriebsmodells)

- Migration – Migrieren einzelner Anwendungen
- Neuentwicklung – Optimierung von Produkten und Services und Innovation in der Cloud

Diese Phasen wurden von Stephen Orban im Blogbeitrag The Journey Toward Cloud-First & the Stages of Adoption im AWS Cloud Enterprise Strategy-Blog definiert. Informationen darüber, wie sie mit der AWS Migrationsstrategie zusammenhängen, finden Sie im Leitfaden zur Vorbereitung der Migration.

CMDB

Siehe Datenbank für das Konfigurationsmanagement.

Code-Repository

Ein Ort, an dem Quellcode und andere Komponenten wie Dokumentation, Beispiele und Skripts gespeichert und im Rahmen von Versionskontrollprozessen aktualisiert werden. Zu den gängigen Cloud-Repositorys gehören GitHub oderBitbucket Cloud. Jede Version des Codes wird Zweig genannt. In einer Microservice-Struktur ist jedes Repository einer einzelnen Funktionalität gewidmet. Eine einzelne CI/CD-Pipeline kann mehrere Repositorien verwenden.

Kalter Cache

Ein Puffer-Cache, der leer oder nicht gut gefüllt ist oder veraltete oder irrelevante Daten enthält. Dies beeinträchtigt die Leistung, da die Datenbank-Instance aus dem Hauptspeicher oder der Festplatte lesen muss, was langsamer ist als das Lesen aus dem Puffercache.

Kalte Daten

Daten, auf die selten zugegriffen wird und die in der Regel historisch sind. Bei der Abfrage dieser Art von Daten sind langsame Abfragen in der Regel akzeptabel. Durch die Verlagerung dieser Daten auf leistungsschwächere und kostengünstigere Speicherstufen oder -klassen können Kosten gesenkt werden.

Computer Vision (CV)

Ein Bereich der KI, der maschinelles Lernen nutzt, um Informationen aus visuellen Formaten wie digitalen Bildern und Videos zu analysieren und zu extrahieren. Amazon SageMaker AI bietet beispielsweise Bildverarbeitungsalgorithmen für CV.

Drift in der Konfiguration

Bei einer Arbeitslast eine Änderung der Konfiguration gegenüber dem erwarteten Zustand. Dies kann dazu führen, dass der Workload nicht mehr richtlinienkonform wird, und zwar in der Regel schrittweise und unbeabsichtigt.

Verwaltung der Datenbankkonfiguration (CMDB)

Ein Repository, das Informationen über eine Datenbank und ihre IT-Umgebung speichert und verwaltet, inklusive Hardware- und Softwarekomponenten und deren Konfigurationen. In der Regel verwenden Sie Daten aus einer CMDB in der Phase der Portfolioerkennung und -analyse der Migration.

Konformitätspaket

Eine Sammlung von AWS Config Regeln und Abhilfemaßnahmen, die Sie zusammenstellen können, um Ihre Konformitäts- und Sicherheitsprüfungen individuell anzupassen. Mithilfe einer YAML-Vorlage können Sie ein Conformance Pack als einzelne Entität in einer AWS-Konto AND-Region oder unternehmensweit bereitstellen. Weitere Informationen finden Sie in der Dokumentation unter Conformance Packs. AWS Config

Kontinuierliche Bereitstellung und kontinuierliche Integration (CI/CD)

Der Prozess der Automatisierung der Quell-, Build-, Test-, Staging- und Produktionsphasen des Softwareveröffentlichungsprozesses. CI/CD is commonly described as a pipeline. CI/CDkann Ihnen helfen, Prozesse zu automatisieren, die Produktivität zu steigern, die Codequalität zu verbessern und schneller zu liefern. Weitere Informationen finden Sie unter Vorteile der kontinuierlichen Auslieferung. CD kann auch für kontinuierliche Bereitstellung stehen. Weitere Informationen finden Sie unterKontinuierliche Auslieferung im Vergleich zu kontinuierlicher Bereitstellung.

CV

Siehe Computer Vision.

D

Daten im Ruhezustand

Daten, die in Ihrem Netzwerk stationär sind, z. B. Daten, die sich im Speicher befinden.

Datenklassifizierung

Ein Prozess zur Identifizierung und Kategorisierung der Daten in Ihrem Netzwerk auf der Grundlage ihrer Kritikalität und Sensitivität. Sie ist eine wichtige Komponente jeder Strategie für das Management von Cybersecurity-Risiken, da sie Ihnen hilft, die geeigneten Schutz- und Aufbewahrungskontrollen für die Daten zu bestimmen. Die Datenklassifizierung ist ein Bestandteil

der Sicherheitssäule im AWS Well-Architected Framework. Weitere Informationen finden Sie unter Datenklassifizierung.

Datendrift

Eine signifikante Variation zwischen den Produktionsdaten und den Daten, die zum Trainieren eines ML-Modells verwendet wurden, oder eine signifikante Änderung der Eingabedaten im Laufe der Zeit. Datendrift kann die Gesamtqualität, Genauigkeit und Fairness von ML-Modellvorhersagen beeinträchtigen.

Daten während der Übertragung

Daten, die sich aktiv durch Ihr Netzwerk bewegen, z. B. zwischen Netzwerkressourcen.

Datennetz

Ein architektonisches Framework, das verteilte, dezentrale Dateneigentum mit zentraler Verwaltung und Steuerung ermöglicht.

Datenminimierung

Das Prinzip, nur die Daten zu sammeln und zu verarbeiten, die unbedingt erforderlich sind. Durch Datenminimierung im AWS Cloud können Datenschutzrisiken, Kosten und der CO2-Fußabdruck Ihrer Analysen reduziert werden.

Datenperimeter

Eine Reihe präventiver Schutzmaßnahmen in Ihrer AWS Umgebung, die sicherstellen, dass nur vertrauenswürdige Identitäten auf vertrauenswürdige Ressourcen von erwarteten Netzwerken zugreifen. Weitere Informationen finden Sie unter Aufbau eines Datenperimeters auf. AWS

Vorverarbeitung der Daten

Rohdaten in ein Format umzuwandeln, das von Ihrem ML-Modell problemlos verarbeitet werden kann. Die Vorverarbeitung von Daten kann bedeuten, dass bestimmte Spalten oder Zeilen entfernt und fehlende, inkonsistente oder doppelte Werte behoben werden.

Herkunft der Daten

Der Prozess der Nachverfolgung des Ursprungs und der Geschichte von Daten während ihres gesamten Lebenszyklus, z. B. wie die Daten generiert, übertragen und gespeichert wurden.

betroffene Person

Eine Person, deren Daten gesammelt und verarbeitet werden.

Data Warehouse

Ein Datenverwaltungssystem, das Business Intelligence wie Analysen unterstützt. Data Warehouses enthalten in der Regel große Mengen historischer Daten und werden in der Regel für Abfragen und Analysen verwendet.

Datenbankdefinitionssprache (DDL)

Anweisungen oder Befehle zum Erstellen oder Ändern der Struktur von Tabellen und Objekten in einer Datenbank.

Datenbankmanipulationssprache (DML)

Anweisungen oder Befehle zum Ändern (Einfügen, Aktualisieren und Löschen) von Informationen in einer Datenbank.

DDL

Siehe Datenbankdefinitionssprache.

Deep-Ensemble

Mehrere Deep-Learning-Modelle zur Vorhersage kombinieren. Sie können Deep-Ensembles verwenden, um eine genauere Vorhersage zu erhalten oder um die Unsicherheit von Vorhersagen abzuschätzen.

Deep Learning

Ein ML-Teilbereich, der mehrere Schichten künstlicher neuronaler Netzwerke verwendet, um die Zuordnung zwischen Eingabedaten und Zielvariablen von Interesse zu ermitteln.

defense-in-depth

Ein Ansatz zur Informationssicherheit, bei dem eine Reihe von Sicherheitsmechanismen und -kontrollen sorgfältig in einem Computernetzwerk verteilt werden, um die Vertraulichkeit, Integrität und Verfügbarkeit des Netzwerks und der darin enthaltenen Daten zu schützen. Wenn Sie diese Strategie anwenden AWS, fügen Sie mehrere Steuerelemente auf verschiedenen Ebenen der AWS Organizations Struktur hinzu, um die Ressourcen zu schützen. Ein defense-in-depth Ansatz könnte beispielsweise Multi-Faktor-Authentifizierung, Netzwerksegmentierung und Verschlüsselung kombinieren.

delegierter Administrator

In AWS Organizations kann ein kompatibler Dienst ein AWS Mitgliedskonto registrieren, um die Konten der Organisation und die Berechtigungen für diesen Dienst zu verwalten. Dieses Konto

wird als delegierter Administrator für diesen Service bezeichnet. Weitere Informationen und eine Liste kompatibler Services finden Sie unter Services, die mit AWS Organizations funktionieren in der AWS Organizations -Dokumentation.

Bereitstellung

Der Prozess, bei dem eine Anwendung, neue Feature oder Codekorrekturen in der Zielumgebung verfügbar gemacht werden. Die Bereitstellung umfasst das Implementieren von Änderungen an einer Codebasis und das anschließende Erstellen und Ausführen dieser Codebasis in den Anwendungsumgebungen.

Entwicklungsumgebung

Siehe Umgebung.

Detektivische Kontrolle

Eine Sicherheitskontrolle, die darauf ausgelegt ist, ein Ereignis zu erkennen, zu protokollieren und zu warnen, nachdem ein Ereignis eingetreten ist. Diese Kontrollen stellen eine zweite Verteidigungslinie dar und warnen Sie vor Sicherheitsereignissen, bei denen die vorhandenen präventiven Kontrollen umgangen wurden. Weitere Informationen finden Sie unter Detektivische Kontrolle in Implementierung von Sicherheitskontrollen in AWS.

Abbildung des Wertstroms in der Entwicklung (DVSM)

Ein Prozess zur Identifizierung und Priorisierung von Einschränkungen, die sich negativ auf Geschwindigkeit und Qualität im Lebenszyklus der Softwareentwicklung auswirken. DVSM erweitert den Prozess der Wertstromanalyse, der ursprünglich für Lean-Manufacturing-Praktiken konzipiert wurde. Es konzentriert sich auf die Schritte und Teams, die erforderlich sind, um durch den Softwareentwicklungsprozess Mehrwert zu schaffen und zu steigern.

digitaler Zwilling

Eine virtuelle Darstellung eines realen Systems, z. B. eines Gebäudes, einer Fabrik, einer Industrieanlage oder einer Produktionslinie. Digitale Zwillinge unterstützen vorausschauende Wartung, Fernüberwachung und Produktionsoptimierung.

Maßtabelle

In einem Sternschema eine kleinere Tabelle, die Datenattribute zu quantitativen Daten in einer Faktentabelle enthält. Bei Attributen von Dimensionstabellen handelt es sich in der Regel um Textfelder oder diskrete Zahlen, die sich wie Text verhalten. Diese Attribute werden häufig zum Einschränken von Abfragen, zum Filtern und zur Kennzeichnung von Ergebnismengen verwendet.

Katastrophe

Ein Ereignis, das verhindert, dass ein Workload oder ein System seine Geschäftsziele an seinem primären Einsatzort erfüllt. Diese Ereignisse können Naturkatastrophen, technische Ausfälle oder das Ergebnis menschlichen Handelns sein, wie z. B. unbeabsichtigte Fehlkonfigurationen oder ein Malware-Angriff.

Disaster Recovery (DR)

Die Strategie und der Prozess, die Sie verwenden, um Ausfallzeiten und Datenverluste aufgrund einer Katastrophe zu minimieren. Weitere Informationen finden Sie unter Disaster Recovery von Workloads unter AWS: Wiederherstellung in der Cloud im AWS Well-Architected Framework.

DML

Siehe Sprache zur Datenbankmanipulation.

Domainorientiertes Design

Ein Ansatz zur Entwicklung eines komplexen Softwaresystems, bei dem seine Komponenten mit sich entwickelnden Domains oder Kerngeschäftszielen verknüpft werden, denen jede Komponente dient. Dieses Konzept wurde von Eric Evans in seinem Buch Domaingesteuertes Design: Bewältigen der Komplexität im Herzen der Software (Boston: Addison-Wesley Professional, 2003) vorgestellt. Informationen darüber, wie Sie domaingesteuertes Design mit dem Strangler-Fig-Muster verwenden können, finden Sie unter Schrittweises Modernisieren älterer Microsoft ASP.NET (ASMX)-Webservices mithilfe von Containern und Amazon API Gateway.

DR

Siehe Disaster Recovery.

Erkennung von Driften

Verfolgung von Abweichungen von einer Basiskonfiguration. Sie können es beispielsweise verwenden, AWS CloudFormation um Abweichungen bei den Systemressourcen zu erkennen, oder Sie können AWS Control Tower damit Änderungen in Ihrer landing zone erkennen, die sich auf die Einhaltung von Governance-Anforderungen auswirken könnten.

DVSM

Siehe Abbildung des Wertstroms in der Entwicklung.

E

EDA

Siehe explorative Datenanalyse.

EDI

Siehe elektronischer Datenaustausch.

Edge-Computing

Die Technologie, die die Rechenleistung für intelligente Geräte an den Rändern eines IoT-Netzwerks erhöht. Im Vergleich zu Cloud Computing kann Edge Computing die Kommunikationslatenz reduzieren und die Reaktionszeit verbessern.

elektronischer Datenaustausch (EDI)

Der automatisierte Austausch von Geschäftsdokumenten zwischen Organisationen. Weitere Informationen finden Sie unter Was ist elektronischer Datenaustausch.

Verschlüsselung

Ein Rechenprozess, der Klartextdaten, die für Menschen lesbar sind, in Chiffretext umwandelt.

Verschlüsselungsschlüssel

Eine kryptografische Zeichenfolge aus zufälligen Bits, die von einem Verschlüsselungsalgorithmus generiert wird. Schlüssel können unterschiedlich lang sein, und jeder Schlüssel ist so konzipiert, dass er unvorhersehbar und einzigartig ist.

Endianismus

Die Reihenfolge, in der Bytes im Computerspeicher gespeichert werden. Big-Endian-Systeme speichern das höchstwertige Byte zuerst. Little-Endian-Systeme speichern das niedrigwertigste Byte zuerst.

Endpunkt

Siehe Service-Endpunkt.

Endpunkt-Services

Ein Service, den Sie in einer Virtual Private Cloud (VPC) hosten können, um ihn mit anderen Benutzern zu teilen. Sie können einen Endpunktdienst mit anderen AWS-Konten oder AWS Identity and Access Management (IAM AWS PrivateLink -) Prinzipalen erstellen und diesen

Berechtigungen gewähren. Diese Konten oder Prinzipale können sich privat mit Ihrem Endpunktservice verbinden, indem sie Schnittstellen-VPC-Endpunkte erstellen. Weitere Informationen finden Sie unter Einen Endpunkt-Service erstellen in der Amazon Virtual Private Cloud (Amazon VPC)-Dokumentation.

Unternehmensressourcenplanung (ERP)

Ein System, das wichtige Geschäftsprozesse (wie Buchhaltung, MES und Projektmanagement) für ein Unternehmen automatisiert und verwaltet.

Envelope-Verschlüsselung

Der Prozess der Verschlüsselung eines Verschlüsselungsschlüssels mit einem anderen Verschlüsselungsschlüssel. Weitere Informationen finden Sie unter Envelope-Verschlüsselung in der AWS Key Management Service (AWS KMS) -Dokumentation.

Umgebung

Eine Instance einer laufenden Anwendung. Die folgenden Arten von Umgebungen sind beim Cloud-Computing üblich:

- Entwicklungsumgebung – Eine Instance einer laufenden Anwendung, die nur dem Kernteam zur Verfügung steht, das für die Wartung der Anwendung verantwortlich ist. Entwicklungsumgebungen werden verwendet, um Änderungen zu testen, bevor sie in höhere Umgebungen übertragen werden. Diese Art von Umgebung wird manchmal als Testumgebung bezeichnet.
- Niedrigere Umgebungen – Alle Entwicklungsumgebungen für eine Anwendung, z. B. solche, die für erste Builds und Tests verwendet wurden.
- Produktionsumgebung – Eine Instance einer laufenden Anwendung, auf die Endbenutzer zugreifen können. In einer CI/CD-Pipeline ist die Produktionsumgebung die letzte Bereitstellungsumgebung.
- Höhere Umgebungen – Alle Umgebungen, auf die auch andere Benutzer als das Kernentwicklungsteam zugreifen können. Dies kann eine Produktionsumgebung, Vorproduktionsumgebungen und Umgebungen für Benutzerakzeptanztests umfassen.

Epics

In der agilen Methodik sind dies funktionale Kategorien, die Ihnen helfen, Ihre Arbeit zu organisieren und zu priorisieren. Epics bieten eine allgemeine Beschreibung der Anforderungen und Implementierungsaufgaben. Zu den Sicherheitsthemen AWS von CAF gehören beispielsweise Identitäts- und Zugriffsmanagement, Detektivkontrollen, Infrastruktursicherheit,

Datenschutz und Reaktion auf Vorfälle. Weitere Informationen zu Epics in der AWS - Migrationsstrategie finden Sie im Leitfaden zur Programm-Implementierung.

ERP

Siehe Enterprise Resource Planning.

Explorative Datenanalyse (EDA)

Der Prozess der Analyse eines Datensatzes, um seine Hauptmerkmale zu verstehen. Sie sammeln oder aggregieren Daten und führen dann erste Untersuchungen durch, um Muster zu finden, Anomalien zu erkennen und Annahmen zu überprüfen. EDA wird durchgeführt, indem zusammenfassende Statistiken berechnet und Datenvisualisierungen erstellt werden.

F

Faktentabelle

Die zentrale Tabelle in einem Sternschema. Sie speichert quantitative Daten über den Geschäftsbetrieb. In der Regel enthält eine Faktentabelle zwei Arten von Spalten: Spalten, die Kennzahlen enthalten, und Spalten, die einen Fremdschlüssel für eine Dimensionstabelle enthalten.

schnell scheitern

Eine Philosophie, die häufige und inkrementelle Tests verwendet, um den Entwicklungslebenszyklus zu verkürzen. Dies ist ein wichtiger Bestandteil eines agilen Ansatzes.

Grenze zur Fehlerisolierung

Dabei handelt es sich um eine Grenze AWS Cloud, z. B. eine Availability Zone AWS-Region, eine Steuerungsebene oder eine Datenebene, die die Auswirkungen eines Fehlers begrenzt und die Widerstandsfähigkeit von Workloads verbessert. Weitere Informationen finden Sie unter Grenzen zur AWS Fehlerisolierung.

Feature-Zweig

Siehe Zweig.

Features

Die Eingabedaten, die Sie verwenden, um eine Vorhersage zu treffen. In einem Fertigungskontext könnten Feature beispielsweise Bilder sein, die regelmäßig von der Fertigungslinie aus aufgenommen werden.

Bedeutung der Feature

Wie wichtig ein Feature für die Vorhersagen eines Modells ist. Dies wird in der Regel als numerischer Wert ausgedrückt, der mit verschiedenen Techniken wie Shapley Additive Explanations (SHAP) und integrierten Gradienten berechnet werden kann. Weitere Informationen finden Sie unter Interpretierbarkeit von Modellen für maschinelles Lernen mit AWS.

Featuretransformation

Daten für den ML-Prozess optimieren, einschließlich der Anreicherung von Daten mit zusätzlichen Quellen, der Skalierung von Werten oder der Extraktion mehrerer Informationssätze aus einem einzigen Datenfeld. Das ermöglicht dem ML-Modell, von den Daten profitieren. Wenn Sie beispielsweise das Datum „27.05.2021 00:15:37“ in „2021“, „Mai“, „Donnerstag“ und „15“ aufschlüsseln, können Sie dem Lernalgorithmus helfen, nuancierte Muster zu erlernen, die mit verschiedenen Datenkomponenten verknüpft sind.

Eingabeaufforderung mit wenigen Klicks

Bereitstellung einer kleinen Anzahl von Beispielen, die die Aufgabe und das gewünschte Ergebnis veranschaulichen, bevor das LLM aufgefordert wird, eine ähnliche Aufgabe auszuführen. Bei dieser Technik handelt es sich um eine Anwendung des kontextbezogenen Lernens, bei der Modelle anhand von Beispielen (Aufnahmen) lernen, die in Eingabeaufforderungen eingebettet sind. Bei Aufgaben, die spezifische Formatierungs-, Argumentations- oder Fachkenntnisse erfordern, kann die Eingabeaufforderung mit wenigen Handgriffen effektiv sein. Siehe auch Zero-Shot Prompting.

FGAC

Siehe detaillierte Zugriffskontrolle.

Feinkörnige Zugriffskontrolle (FGAC)

Die Verwendung mehrerer Bedingungen, um eine Zugriffsanfrage zuzulassen oder abzulehnen.

Flash-Cut-Migration

Eine Datenbankmigrationsmethode, bei der eine kontinuierliche Datenreplikation durch Erfassung von Änderungsdaten verwendet wird, um Daten in kürzester Zeit zu migrieren, anstatt einen schrittweisen Ansatz zu verwenden. Ziel ist es, Ausfallzeiten auf ein Minimum zu beschränken.

FM

Siehe Fundamentmodell.

Fundamentmodell (FM)

Ein großes neuronales Deep-Learning-Netzwerk, das mit riesigen Datensätzen generalisierter und unbeschrifteter Daten trainiert wurde. FMs sind in der Lage, eine Vielzahl allgemeiner Aufgaben zu erfüllen, z. B. Sprache zu verstehen, Text und Bilder zu generieren und Konversationen in natürlicher Sprache zu führen. Weitere Informationen finden Sie unter Was sind Foundation-Modelle.

G

generative KI

Eine Untergruppe von KI-Modellen, die mit großen Datenmengen trainiert wurden und mit einer einfachen Textaufforderung neue Inhalte und Artefakte wie Bilder, Videos, Text und Audio erstellen können. Weitere Informationen finden Sie unter Was ist Generative KI.

Geoblocking

Siehe geografische Einschränkungen.

Geografische Einschränkungen (Geoblocking)

Bei Amazon eine Option CloudFront, um zu verhindern, dass Benutzer in bestimmten Ländern auf Inhaltsverteilungen zugreifen. Sie können eine Zulassungsliste oder eine Sperrliste verwenden, um zugelassene und gesperrte Länder anzugeben. Weitere Informationen finden Sie in der Dokumentation unter Beschränkung der geografischen Verteilung Ihrer Inhalte. CloudFront

Gitflow-Workflow

Ein Ansatz, bei dem niedrigere und höhere Umgebungen unterschiedliche Zweige in einem Quellcode-Repository verwenden. Der Gitflow-Workflow gilt als veraltet, und der Trunk-basierte Workflow ist der moderne, bevorzugte Ansatz.

goldenes Bild

Ein Snapshot eines Systems oder einer Software, der als Vorlage für die Bereitstellung neuer Instanzen dieses Systems oder dieser Software verwendet wird. In der Fertigung kann ein Golden Image beispielsweise zur Bereitstellung von Software auf mehreren Geräten verwendet werden und trägt zur Verbesserung der Geschwindigkeit, Skalierbarkeit und Produktivität bei der Geräteherstellung bei.

Greenfield-Strategie

Das Fehlen vorhandener Infrastruktur in einer neuen Umgebung. Bei der Einführung einer Neuausrichtung einer Systemarchitektur können Sie alle neuen Technologien ohne Einschränkung der Kompatibilität mit der vorhandenen Infrastruktur auswählen, auch bekannt als Brownfield. Wenn Sie die bestehende Infrastruktur erweitern, könnten Sie Brownfield- und Greenfield-Strategien mischen.

Integritätsschutz

Eine allgemeine Regel, die dazu beiträgt, Ressourcen, Richtlinien und die Einhaltung von Vorschriften in allen Unternehmenseinheiten zu regeln (OUs). Präventiver Integritätsschutz setzt Richtlinien durch, um die Einhaltung von Standards zu gewährleisten. Sie werden mithilfe von Service-Kontrollrichtlinien und IAM-Berechtigungsgrenzen implementiert. Detektivischer Integritätsschutz erkennt Richtlinienverstöße und Compliance-Probleme und generiert Warnmeldungen zur Abhilfe. Sie werden mithilfe von AWS Config, AWS Security Hub, Amazon GuardDuty AWS Trusted Advisor, Amazon Inspector und benutzerdefinierten AWS Lambda Prüfungen implementiert.

H

HEKTAR

Siehe Hochverfügbarkeit.

Heterogene Datenbankmigration

Migrieren Sie Ihre Quelldatenbank in eine Zieldatenbank, die eine andere Datenbank-Engine verwendet (z. B. Oracle zu Amazon Aurora). Eine heterogene Migration ist in der Regel Teil einer Neuarchitektur, und die Konvertierung des Schemas kann eine komplexe Aufgabe sein. AWS bietet AWS SCT, welches bei Schemakonvertierungen hilft.

hohe Verfügbarkeit (HA)

Die Fähigkeit eines Workloads, im Falle von Herausforderungen oder Katastrophen kontinuierlich und ohne Eingreifen zu arbeiten. HA-Systeme sind so konzipiert, dass sie automatisch ein Failover durchführen, gleichbleibend hohe Leistung bieten und unterschiedliche Lasten und Ausfälle mit minimalen Leistungseinbußen bewältigen.

historische Modernisierung

Ein Ansatz zur Modernisierung und Aufrüstung von Betriebstechnologiesystemen (OT), um den Bedürfnissen der Fertigungsindustrie besser gerecht zu werden. Ein Historian ist eine Art von Datenbank, die verwendet wird, um Daten aus verschiedenen Quellen in einer Fabrik zu sammeln und zu speichern.

Holdout-Daten

Ein Teil historischer, beschrifteter Daten, der aus einem Datensatz zurückgehalten wird, der zum Trainieren eines Modells für maschinelles Lernen verwendet wird. Sie können Holdout-Daten verwenden, um die Modellleistung zu bewerten, indem Sie die Modellvorhersagen mit den Holdout-Daten vergleichen.

Homogene Datenbankmigration

Migrieren Sie Ihre Quelldatenbank zu einer Zieldatenbank, die dieselbe Datenbank-Engine verwendet (z. B. Microsoft SQL Server zu Amazon RDS für SQL Server). Eine homogene Migration ist in der Regel Teil eines Hostwechsels oder eines Plattformwechsels. Sie können native Datenbankserviceprogramme verwenden, um das Schema zu migrieren.

heiße Daten

Daten, auf die häufig zugegriffen wird, z. B. Echtzeitdaten oder aktuelle Translationsdaten. Für diese Daten ist in der Regel eine leistungsstarke Speicherebene oder -klasse erforderlich, um schnelle Abfrageantworten zu ermöglichen.

Hotfix

Eine dringende Lösung für ein kritisches Problem in einer Produktionsumgebung. Aufgrund seiner Dringlichkeit wird ein Hotfix normalerweise außerhalb des typischen DevOps Release-Workflows erstellt.

Hypercare-Phase

Unmittelbar nach dem Cutover, der Zeitraum, in dem ein Migrationsteam die migrierten Anwendungen in der Cloud verwaltet und überwacht, um etwaige Probleme zu beheben. In der Regel dauert dieser Zeitraum 1–4 Tage. Am Ende der Hypercare-Phase überträgt das Migrationsteam in der Regel die Verantwortung für die Anwendungen an das Cloud-Betriebsteam.

I

IaC

Sehen Sie sich Infrastruktur als Code an.

Identitätsbasierte Richtlinie

Eine Richtlinie, die einem oder mehreren IAM-Prinzipalen zugeordnet ist und deren Berechtigungen innerhalb der AWS Cloud Umgebung definiert.

Leerlaufanwendung

Eine Anwendung mit einer durchschnittlichen CPU- und Arbeitsspeicherauslastung zwischen 5 und 20 Prozent über einen Zeitraum von 90 Tagen. In einem Migrationsprojekt ist es üblich, diese Anwendungen außer Betrieb zu nehmen oder sie On-Premises beizubehalten.

IIoT

Siehe Industrielles Internet der Dinge.

unveränderliche Infrastruktur

Ein Modell, das eine neue Infrastruktur für Produktionsworkloads bereitstellt, anstatt die bestehende Infrastruktur zu aktualisieren, zu patchen oder zu modifizieren. Unveränderliche Infrastrukturen sind von Natur aus konsistenter, zuverlässiger und vorhersehbarer als veränderliche Infrastrukturen. Weitere Informationen finden Sie in der Best Practice Deploy using immutable infrastructure im AWS Well-Architected Framework.

Eingehende (ingress) VPC

In einer Architektur AWS mit mehreren Konten ist dies eine VPC, die Netzwerkverbindungen von außerhalb einer Anwendung akzeptiert, überprüft und weiterleitet. Die AWS Security Reference Architecture empfiehlt, Ihr Netzwerkkonto mit eingehendem und ausgehendem Datenverkehr und Inspektion einzurichten, VPCs um die bidirektionale Schnittstelle zwischen Ihrer Anwendung und dem Internet im weiteren Sinne zu schützen.

Inkrementelle Migration

Eine Cutover-Strategie, bei der Sie Ihre Anwendung in kleinen Teilen migrieren, anstatt eine einziges vollständiges Cutover durchzuführen. Beispielsweise könnten Sie zunächst nur einige Microservices oder Benutzer auf das neue System umstellen. Nachdem Sie sich vergewissert haben, dass alles ordnungsgemäß funktioniert, können Sie weitere Microservices oder Benutzer

schrittweise verschieben, bis Sie Ihr Legacy-System außer Betrieb nehmen können. Diese Strategie reduziert die mit großen Migrationen verbundenen Risiken.

Industrie 4.0

Ein Begriff, der 2016 von Klaus Schwab eingeführt wurde und sich auf die Modernisierung von Fertigungsprozessen durch Fortschritte in den Bereichen Konnektivität, Echtzeitdaten, Automatisierung, Analytik und KI/ML bezieht.

Infrastruktur

Alle Ressourcen und Komponenten, die in der Umgebung einer Anwendung enthalten sind.

Infrastructure as Code (IaC)

Der Prozess der Bereitstellung und Verwaltung der Infrastruktur einer Anwendung mithilfe einer Reihe von Konfigurationsdateien. IaC soll Ihnen helfen, das Infrastrukturmanagement zu zentralisieren, Ressourcen zu standardisieren und schnell zu skalieren, sodass neue Umgebungen wiederholbar, zuverlässig und konsistent sind.

industrielles Internet der Dinge (T) IIo

Einsatz von mit dem Internet verbundenen Sensoren und Geräten in Industriesektoren wie Fertigung, Energie, Automobilindustrie, Gesundheitswesen, Biowissenschaften und Landwirtschaft. Weitere Informationen finden Sie unter Aufbau einer digitalen Transformationsstrategie für das industrielle Internet der Dinge (IIoT).

Inspektions-VPC

In einer Architektur AWS mit mehreren Konten eine zentralisierte VPC, die Inspektionen des Netzwerkverkehrs zwischen VPCs (in demselben oder unterschiedlichen AWS-Regionen), dem Internet und lokalen Netzwerken verwaltet. In der AWS Security Reference Architecture wird empfohlen, Ihr Netzwerkkonto mit eingehendem und ausgehendem Datenverkehr sowie Inspektionen einzurichten, VPCs um die bidirektionale Schnittstelle zwischen Ihrer Anwendung und dem Internet im weiteren Sinne zu schützen.

Internet of Things (IoT)

Das Netzwerk verbundener physischer Objekte mit eingebetteten Sensoren oder Prozessoren, das über das Internet oder über ein lokales Kommunikationsnetzwerk mit anderen Geräten und Systemen kommuniziert. Weitere Informationen finden Sie unter Was ist IoT?

Interpretierbarkeit

Ein Merkmal eines Modells für Machine Learning, das beschreibt, inwieweit ein Mensch verstehen kann, wie die Vorhersagen des Modells von seinen Eingaben abhängen. Weitere Informationen finden Sie unter Interpretierbarkeit des Modells für maschinelles Lernen mit. AWS

IoT

Siehe Internet der Dinge.

IT information library (ITIL, IT-Informationsbibliothek)

Eine Reihe von bewährten Methoden für die Bereitstellung von IT-Services und die Abstimmung dieser Services auf die Geschäftsanforderungen. ITIL bietet die Grundlage für ITSM.

T service management (ITSM, IT-Servicemanagement)

Aktivitäten im Zusammenhang mit der Gestaltung, Implementierung, Verwaltung und Unterstützung von IT-Services für eine Organisation. Informationen zur Integration von Cloud-Vorgängen mit ITSM-Tools finden Sie im Leitfaden zur Betriebsintegration.

BIS

Weitere Informationen finden Sie in der IT-Informationsbibliothek.

ITSM

Siehe IT-Servicemanagement.

L

Labelbasierte Zugangskontrolle (LBAC)

Eine Implementierung der Mandatory Access Control (MAC), bei der den Benutzern und den Daten selbst jeweils explizit ein Sicherheitslabelwert zugewiesen wird. Die Schnittmenge zwischen der Benutzersicherheitsbeschriftung und der Datensicherheitsbeschriftung bestimmt, welche Zeilen und Spalten für den Benutzer sichtbar sind.

Landing Zone

Eine landing zone ist eine gut strukturierte AWS Umgebung mit mehreren Konten, die skalierbar und sicher ist. Dies ist ein Ausgangspunkt, von dem aus Ihre Organisationen Workloads und Anwendungen schnell und mit Vertrauen in ihre Sicherheits- und Infrastrukturumgebung starten

und bereitstellen können. Weitere Informationen zu Landing Zones finden Sie unter Einrichtung einer sicheren und skalierbaren AWS -Umgebung mit mehreren Konten..

großes Sprachmodell (LLM)

Ein Deep-Learning-KI-Modell, das anhand einer riesigen Datenmenge vorab trainiert wurde. Ein LLM kann mehrere Aufgaben ausführen, z. B. Fragen beantworten, Dokumente zusammenfassen, Text in andere Sprachen übersetzen und Sätze vervollständigen. Weitere Informationen finden Sie unter Was sind. LLMs

Große Migration

Eine Migration von 300 oder mehr Servern.

SCHWARZ

Weitere Informationen finden Sie unter Label-basierte Zugriffskontrolle.

Geringste Berechtigung

Die bewährte Sicherheitsmethode, bei der nur die für die Durchführung einer Aufgabe erforderlichen Mindestberechtigungen erteilt werden. Weitere Informationen finden Sie unter Geringste Berechtigungen anwenden in der IAM-Dokumentation.

Lift and Shift

Siehe 7 Rs.

Little-Endian-System

Ein System, welches das niedrigwertigste Byte zuerst speichert. Siehe auch Endianness.

LLM

Siehe großes Sprachmodell.

Niedrigere Umgebungen

Siehe Umgebung.

M

Machine Learning (ML)

Eine Art künstlicher Intelligenz, die Algorithmen und Techniken zur Mustererkennung und zum Lernen verwendet. ML analysiert aufgezeichnete Daten, wie z. B. Daten aus dem Internet der

Dinge (IoT), und lernt daraus, um ein statistisches Modell auf der Grundlage von Mustern zu erstellen. Weitere Informationen finden Sie unter Machine Learning.

Hauptzweig

Siehe Filiale.

Malware

Software, die entwickelt wurde, um die Computersicherheit oder den Datenschutz zu gefährden. Malware kann Computersysteme stören, vertrauliche Informationen durchsickern lassen oder sich unbefugten Zugriff verschaffen. Beispiele für Malware sind Viren, Würmer, Ransomware, Trojaner, Spyware und Keylogger.

verwaltete Dienste

AWS-Services für die die Infrastrukturebene, das Betriebssystem und die Plattformen AWS betrieben werden, und Sie greifen auf die Endgeräte zu, um Daten zu speichern und abzurufen. Amazon Simple Storage Service (Amazon S3) und Amazon DynamoDB sind Beispiele für Managed Services. Diese werden auch als abstrakte Dienste bezeichnet.

Manufacturing Execution System (MES)

Ein Softwaresystem zur Nachverfolgung, Überwachung, Dokumentation und Steuerung von Produktionsprozessen, bei denen Rohstoffe in der Fertigung zu fertigen Produkten umgewandelt werden.

MAP

Siehe Migration Acceleration Program.

Mechanismus

Ein vollständiger Prozess, bei dem Sie ein Tool erstellen, die Akzeptanz des Tools vorantreiben und anschließend die Ergebnisse überprüfen, um Anpassungen vorzunehmen. Ein Mechanismus ist ein Zyklus, der sich im Laufe seiner Tätigkeit selbst verstärkt und verbessert. Weitere Informationen finden Sie unter Aufbau von Mechanismen im AWS Well-Architected Framework.

Mitgliedskonto

Alle AWS-Konten außer dem Verwaltungskonto, die Teil einer Organisation in sind. AWS Organizations Ein Konto kann jeweils nur einer Organisation angehören.

DURCHEINANDER

Siehe Manufacturing Execution System.

Message Queuing-Telemetrietransport (MQTT)

Ein leichtes machine-to-machine (M2M) -Kommunikationsprotokoll, das auf dem Publish/Subscribe-Muster für IoT-Geräte mit beschränkten Ressourcen basiert.

Microservice

Ein kleiner, unabhängiger Dienst, der über genau definierte Kanäle kommuniziert APIs und in der Regel kleinen, eigenständigen Teams gehört. Ein Versicherungssystem kann beispielsweise Microservices beinhalten, die Geschäftsfunktionen wie Vertrieb oder Marketing oder Subdomains wie Einkauf, Schadenersatz oder Analytik zugeordnet sind. Zu den Vorteilen von Microservices gehören Agilität, flexible Skalierung, einfache Bereitstellung, wiederverwendbarer Code und Ausfallsicherheit. Weitere Informationen finden Sie unter Integration von Microservices mithilfe serverloser Dienste. AWS

Microservices-Architekturen

Ein Ansatz zur Erstellung einer Anwendung mit unabhängigen Komponenten, die jeden Anwendungsprozess als Microservice ausführen. Diese Microservices kommunizieren mithilfe von Lightweight über eine klar definierte Schnittstelle. APIs Jeder Microservice in dieser Architektur kann aktualisiert, bereitgestellt und skaliert werden, um den Bedarf an bestimmten Funktionen einer Anwendung zu decken. Weitere Informationen finden Sie unter Implementierung von Microservices auf. AWS

Migration Acceleration Program (MAP)

Ein AWS Programm, das Beratung, Unterstützung, Schulungen und Services bietet, um Unternehmen dabei zu unterstützen, eine solide betriebliche Grundlage für die Umstellung auf die Cloud zu schaffen und die anfänglichen Kosten von Migrationen auszugleichen. MAP umfasst eine Migrationsmethode für die methodische Durchführung von Legacy-Migrationen sowie eine Reihe von Tools zur Automatisierung und Beschleunigung gängiger Migrationsszenarien.

Migration in großem Maßstab

Der Prozess, bei dem der Großteil des Anwendungsportfolios in Wellen in die Cloud verlagert wird, wobei in jeder Welle mehr Anwendungen schneller migriert werden. In dieser Phase werden die bewährten Verfahren und Erkenntnisse aus den früheren Phasen zur Implementierung einer Migrationsfabrik von Teams, Tools und Prozessen zur Optimierung der Migration von Workloads durch Automatisierung und agile Bereitstellung verwendet. Dies ist die dritte Phase der AWS - Migrationsstrategie.

Migrationsfabrik

Funktionsübergreifende Teams, die die Migration von Workloads durch automatisierte, agile Ansätze optimieren. Zu den Teams in der Migrationsabteilung gehören in der Regel Betriebsabläufe, Geschäftsanalysten und Eigentümer, Migrationsingenieure, Entwickler und DevOps Experten, die in Sprints arbeiten. Zwischen 20 und 50 Prozent eines Unternehmensanwendungsportfolios bestehen aus sich wiederholenden Mustern, die durch einen Fabrik-Ansatz optimiert werden können. Weitere Informationen finden Sie in Diskussion über Migrationsfabriken und den Leitfaden zur Cloud-Migration-Fabrik in diesem Inhaltssatz.

Migrationsmetadaten

Die Informationen über die Anwendung und den Server, die für den Abschluss der Migration benötigt werden. Für jedes Migrationsmuster ist ein anderer Satz von Migrationsmetadaten erforderlich. Beispiele für Migrationsmetadaten sind das Zielsubnetz, die Sicherheitsgruppe und AWS das Konto.

Migrationsmuster

Eine wiederholbare Migrationsaufgabe, in der die Migrationsstrategie, das Migrationsziel und die verwendete Migrationsanwendung oder der verwendete Migrationsservice detailliert beschrieben werden. Beispiel: Rehost-Migration zu Amazon EC2 mit AWS Application Migration Service.

Migration Portfolio Assessment (MPA)

Ein Online-Tool, das Informationen zur Validierung des Geschäftsszenarios für die Migration auf das bereitstellt. AWS Cloud MPA bietet eine detaillierte Portfoliobewertung (richtige Servergröße, Preisgestaltung, Gesamtbetriebskostenanalyse, Migrationskostenanalyse) sowie Migrationsplanung (Anwendungsdatenanalyse und Datenerfassung, Anwendungsgruppierung, Migrationspriorisierung und Wellenplanung). Das MPA-Tool (Anmeldung erforderlich) steht allen AWS Beratern und APN-Partnerberatern kostenlos zur Verfügung.

Migration Readiness Assessment (MRA)

Der Prozess, bei dem mithilfe des AWS CAF Erkenntnisse über den Cloud-Bereitschaftsstatus eines Unternehmens gewonnen, Stärken und Schwächen identifiziert und ein Aktionsplan zur Schließung festgestellter Lücken erstellt wird. Weitere Informationen finden Sie im Benutzerhandbuch für Migration Readiness. MRA ist die erste Phase der AWS - Migrationsstrategie.

Migrationsstrategie

Der Ansatz, der verwendet wurde, um einen Workload auf den AWS Cloud zu migrieren. Weitere Informationen finden Sie im Eintrag 7 Rs in diesem Glossar und unter Mobilisieren Sie Ihr Unternehmen, um umfangreiche Migrationen zu beschleunigen.

ML

Siehe maschinelles Lernen.

Modernisierung

Umwandlung einer veralteten (veralteten oder monolithischen) Anwendung und ihrer Infrastruktur in ein agiles, elastisches und hochverfügbares System in der Cloud, um Kosten zu senken, die Effizienz zu steigern und Innovationen zu nutzen. Weitere Informationen finden Sie unter Strategie zur Modernisierung von Anwendungen in der AWS Cloud.

Bewertung der Modernisierungsfähigkeit

Eine Bewertung, anhand derer festgestellt werden kann, ob die Anwendungen einer Organisation für die Modernisierung bereit sind, Vorteile, Risiken und Abhängigkeiten identifiziert und ermittelt wird, wie gut die Organisation den zukünftigen Status dieser Anwendungen unterstützen kann. Das Ergebnis der Bewertung ist eine Vorlage der Zielarchitektur, eine Roadmap, in der die Entwicklungsphasen und Meilensteine des Modernisierungsprozesses detailliert beschrieben werden, sowie ein Aktionsplan zur Behebung festgestellter Lücken. Weitere Informationen finden Sie unter Evaluierung der Modernisierungsbereitschaft von Anwendungen in der AWS Cloud.

Monolithische Anwendungen (Monolithen)

Anwendungen, die als ein einziger Service mit eng gekoppelten Prozessen ausgeführt werden. Monolithische Anwendungen haben verschiedene Nachteile. Wenn ein Anwendungs-Feature stark nachgefragt wird, muss die gesamte Architektur skaliert werden. Das Hinzufügen oder Verbessern der Feature einer monolithischen Anwendung wird ebenfalls komplexer, wenn die Codebasis wächst. Um diese Probleme zu beheben, können Sie eine Microservices-Architektur verwenden. Weitere Informationen finden Sie unter Zerlegen von Monolithen in Microservices.

MPA

Siehe Bewertung des Migrationsportfolios.

MQTT

Siehe Message Queuing-Telemetrietransport.

Mehrklassen-Klassifizierung

Ein Prozess, der dabei hilft, Vorhersagen für mehrere Klassen zu generieren (wobei eines von mehr als zwei Ergebnissen vorhergesagt wird). Ein ML-Modell könnte beispielsweise fragen: „Ist dieses Produkt ein Buch, ein Auto oder ein Telefon?“ oder „Welche Kategorie von Produkten ist für diesen Kunden am interessantesten?“

veränderbare Infrastruktur

Ein Modell, das die bestehende Infrastruktur für Produktionsworkloads aktualisiert und modifiziert. Für eine verbesserte Konsistenz, Zuverlässigkeit und Vorhersagbarkeit empfiehlt das AWS Well-Architected Framework die Verwendung einer unveränderlichen Infrastruktur als bewährte Methode.

O

OAC

Siehe Origin Access Control.

OAI

Siehe Zugriffsidentität von Origin.

COM

Siehe organisatorisches Change-Management.

Offline-Migration

Eine Migrationsmethode, bei der der Quell-Workload während des Migrationsprozesses heruntergefahren wird. Diese Methode ist mit längeren Ausfallzeiten verbunden und wird in der Regel für kleine, unkritische Workloads verwendet.

OI

Siehe Betriebsintegration.

OLA

Siehe Vereinbarung auf operativer Ebene.

Online-Migration

Eine Migrationsmethode, bei der der Quell-Workload auf das Zielsystem kopiert wird, ohne offline genommen zu werden. Anwendungen, die mit dem Workload verbunden sind, können während

der Migration weiterhin funktionieren. Diese Methode beinhaltet keine bis minimale Ausfallzeit und wird in der Regel für kritische Produktionsworkloads verwendet.

OPC-UA

Siehe Open Process Communications — Unified Architecture.

Offene Prozesskommunikation — Einheitliche Architektur (OPC-UA)

Ein machine-to-machine (M2M) -Kommunikationsprotokoll für die industrielle Automatisierung. OPC-UA bietet einen Interoperabilitätsstandard mit Datenverschlüsselungs-, Authentifizierungs- und Autorisierungsschemata.

Vereinbarung auf Betriebsebene (OLA)

Eine Vereinbarung, in der klargestellt wird, welche funktionalen IT-Gruppen sich gegenseitig versprechen zu liefern, um ein Service Level Agreement (SLA) zu unterstützen.

Überprüfung der Betriebsbereitschaft (ORR)

Eine Checkliste mit Fragen und zugehörigen bewährten Methoden, die Ihnen helfen, Vorfälle und mögliche Ausfälle zu verstehen, zu bewerten, zu verhindern oder deren Umfang zu reduzieren. Weitere Informationen finden Sie unter Operational Readiness Reviews (ORR) im AWS Well-Architected Framework.

Betriebstechnologie (OT)

Hardware- und Softwaresysteme, die mit der physischen Umgebung zusammenarbeiten, um industrielle Abläufe, Ausrüstung und Infrastruktur zu steuern. In der Fertigung ist die Integration von OT- und Informationstechnologie (IT) -Systemen ein zentraler Schwerpunkt der Industrie 4.0-Transformationen.

Betriebsintegration (OI)

Der Prozess der Modernisierung von Abläufen in der Cloud, der Bereitschaftsplanung, Automatisierung und Integration umfasst. Weitere Informationen finden Sie im Leitfaden zur Betriebsintegration.

Organisationspfad

Ein Pfad, der von erstellt wird und in AWS CloudTrail dem alle Ereignisse für alle AWS-Konten in einer Organisation protokolliert werden. AWS Organizations Diese Spur wird in jedem AWS-Konto , der Teil der Organisation ist, erstellt und verfolgt die Aktivität in jedem Konto. Weitere Informationen finden Sie in der CloudTrail Dokumentation unter Einen Trail für eine Organisation erstellen.

Organisatorisches Veränderungsmanagement (OCM)

Ein Framework für das Management wichtiger, disruptiver Geschäftstransformationen aus Sicht der Mitarbeiter, der Kultur und der Führung. OCM hilft Organisationen dabei, sich auf neue Systeme und Strategien vorzubereiten und auf diese umzustellen, indem es die Akzeptanz von Veränderungen beschleunigt, Übergangsprobleme angeht und kulturelle und organisatorische Veränderungen vorantreibt. In der AWS Migrationsstrategie wird dieses Framework aufgrund der Geschwindigkeit des Wandels, der bei Projekten zur Cloud-Einführung erforderlich ist, als Mitarbeiterbeschleunigung bezeichnet. Weitere Informationen finden Sie im OCM-Handbuch.

Ursprungszugriffskontrolle (OAC)

In CloudFront, eine erweiterte Option zur Zugriffsbeschränkung, um Ihre Amazon Simple Storage Service (Amazon S3) -Inhalte zu sichern. OAC unterstützt alle S3-Buckets insgesamt AWS-Regionen, serverseitige Verschlüsselung mit AWS KMS (SSE-KMS) sowie dynamische `PUT` und `DELETE` Anfragen an den S3-Bucket.

Ursprungszugriffsidentität (OAI)

In CloudFront, eine Option zur Zugriffsbeschränkung, um Ihre Amazon S3 S3-Inhalte zu sichern. Wenn Sie OAI verwenden, CloudFront erstellt es einen Principal, mit dem sich Amazon S3 authentifizieren kann. Authentifizierte Principals können nur über eine bestimmte Distribution auf Inhalte in einem S3-Bucket zugreifen. CloudFront Siehe auch OAC, das eine detailliertere und verbesserte Zugriffskontrolle bietet.

ORR

Weitere Informationen finden Sie unter Überprüfung der Betriebsbereitschaft.

NICHT

Siehe Betriebstechnologie.

Ausgehende (egress) VPC

In einer Architektur AWS mit mehreren Konten eine VPC, die Netzwerkverbindungen verarbeitet, die von einer Anwendung aus initiiert werden. Die AWS Security Reference Architecture empfiehlt die Einrichtung Ihres Netzwerkkontos mit eingehendem und ausgehendem Datenverkehr sowie Inspektion, VPCs um die bidirektionale Schnittstelle zwischen Ihrer Anwendung und dem Internet im weiteren Sinne zu schützen.

P

Berechtigungsgrenze

Eine IAM-Verwaltungsrichtlinie, die den IAM-Prinzipalen zugeordnet ist, um die maximalen Berechtigungen festzulegen, die der Benutzer oder die Rolle haben kann. Weitere Informationen finden Sie unter Berechtigungsgrenzen für IAM-Entitys in der IAM-Dokumentation.

persönlich identifizierbare Informationen (PII)

Informationen, die, wenn sie direkt betrachtet oder mit anderen verwandten Daten kombiniert werden, verwendet werden können, um vernünftige Rückschlüsse auf die Identität einer Person zu ziehen. Beispiele für personenbezogene Daten sind Namen, Adressen und Kontaktinformationen.

Personenbezogene Daten

Siehe persönlich identifizierbare Informationen.

Playbook

Eine Reihe vordefinierter Schritte, die die mit Migrationen verbundenen Aufgaben erfassen, z. B. die Bereitstellung zentraler Betriebsfunktionen in der Cloud. Ein Playbook kann die Form von Skripten, automatisierten Runbooks oder einer Zusammenfassung der Prozesse oder Schritte annehmen, die für den Betrieb Ihrer modernisierten Umgebung erforderlich sind.

PLC

Siehe programmierbare Logiksteuerung.

PLM

Siehe Produktlebenszyklusmanagement.

policy

Ein Objekt, das Berechtigungen definieren (siehe identitätsbasierte Richtlinie), Zugriffsbedingungen spezifizieren (siehe ressourcenbasierte Richtlinie) oder die maximalen Berechtigungen für alle Konten in einer Organisation definieren kann AWS Organizations (siehe Dienststeuerungsrichtlinie).

Polyglotte Beharrlichkeit

Unabhängige Auswahl der Datenspeichertechnologie eines Microservices auf der Grundlage von Datenzugriffsmustern und anderen Anforderungen. Wenn Ihre Microservices über dieselbe

Datenspeichertechnologie verfügen, kann dies zu Implementierungsproblemen oder zu Leistungseinbußen führen. Microservices lassen sich leichter implementieren und erzielen eine bessere Leistung und Skalierbarkeit, wenn sie den Datenspeicher verwenden, der ihren Anforderungen am besten entspricht. Weitere Informationen finden Sie unter Datenpersistenz in Microservices aktivieren.

Portfoliobewertung

Ein Prozess, bei dem das Anwendungsportfolio ermittelt, analysiert und priorisiert wird, um die Migration zu planen. Weitere Informationen finden Sie in Bewerten der Migrationsbereitschaft.

predicate

Eine Abfragebedingung, die `true` oder zurückgibt`false`, was üblicherweise in einer Klausel vorkommt. `WHERE`

Prädikat Pushdown

Eine Technik zur Optimierung von Datenbankabfragen, bei der die Daten in der Abfrage vor der Übertragung gefiltert werden. Dadurch wird die Datenmenge reduziert, die aus der relationalen Datenbank abgerufen und verarbeitet werden muss, und die Abfrageleistung wird verbessert.

Präventive Kontrolle

Eine Sicherheitskontrolle, die verhindern soll, dass ein Ereignis eintritt. Diese Kontrollen stellen eine erste Verteidigungslinie dar, um unbefugten Zugriff oder unerwünschte Änderungen an Ihrem Netzwerk zu verhindern. Weitere Informationen finden Sie unter Präventive Kontrolle in Implementierung von Sicherheitskontrollen in AWS.

Prinzipal

Eine Entität AWS , die Aktionen ausführen und auf Ressourcen zugreifen kann. Bei dieser Entität handelt es sich in der Regel um einen Root-Benutzer für eine AWS-Konto, eine IAM-Rolle oder einen Benutzer. Weitere Informationen finden Sie unter Prinzipal in Rollenbegriffe und -konzepte in der IAM-Dokumentation.

Datenschutz von Natur aus

Ein systemtechnischer Ansatz, der den Datenschutz während des gesamten Entwicklungsprozesses berücksichtigt.

Privat gehostete Zonen

Ein Container, der Informationen darüber enthält, wie Amazon Route 53 auf DNS-Abfragen für eine Domain und deren Subdomains innerhalb einer oder mehrerer VPCs Domains antworten

soll. Weitere Informationen finden Sie unter Arbeiten mit privat gehosteten Zonen in der Route-53-Dokumentation.

proaktive Steuerung

Eine Sicherheitskontrolle, die den Einsatz nicht richtlinienkonformer Ressourcen verhindern soll. Diese Steuerelemente scannen Ressourcen, bevor sie bereitgestellt werden. Wenn die Ressource nicht mit der Steuerung konform ist, wird sie nicht bereitgestellt. Weitere Informationen finden Sie im Referenzhandbuch zu Kontrollen in der AWS Control Tower Dokumentation und unter Proaktive Kontrollen unter Implementierung von Sicherheitskontrollen am AWS.

Produktlebenszyklusmanagement (PLM)

Das Management von Daten und Prozessen für ein Produkt während seines gesamten Lebenszyklus, vom Design, der Entwicklung und Markteinführung über Wachstum und Reife bis hin zur Markteinführung und Markteinführung.

Produktionsumgebung

Siehe Umgebung.

Speicherprogrammierbare Steuerung (SPS)

In der Fertigung ein äußerst zuverlässiger, anpassungsfähiger Computer, der Maschinen überwacht und Fertigungsprozesse automatisiert.

schnelle Verkettung

Verwendung der Ausgabe einer LLM-Eingabeaufforderung als Eingabe für die nächste Aufforderung, um bessere Antworten zu generieren. Diese Technik wird verwendet, um eine komplexe Aufgabe in Unteraufgaben zu unterteilen oder um eine vorläufige Antwort iterativ zu verfeinern oder zu erweitern. Sie trägt dazu bei, die Genauigkeit und Relevanz der Antworten eines Modells zu verbessern und ermöglicht detailliertere, personalisierte Ergebnisse.

Pseudonymisierung

Der Prozess, bei dem persönliche Identifikatoren in einem Datensatz durch Platzhalterwerte ersetzt werden. Pseudonymisierung kann zum Schutz der Privatsphäre beitragen. Pseudonymisierte Daten gelten weiterhin als personenbezogene Daten.

publish/subscribe (pub/sub)

Ein Muster, das asynchrone Kommunikation zwischen Microservices ermöglicht, um die Skalierbarkeit und Reaktionsfähigkeit zu verbessern. In einem auf Microservices basierenden MES kann ein Microservice beispielsweise Ereignismeldungen in einem Kanal veröffentlichen,

den andere Microservices abonnieren können. Das System kann neue Microservices hinzufügen, ohne den Veröffentlichungsservice zu ändern.

Q

Abfrageplan

Eine Reihe von Schritten, wie Anweisungen, die für den Zugriff auf die Daten in einem relationalen SQL-Datenbanksystem verwendet werden.

Abfrageplanregression

Wenn ein Datenbankserviceoptimierer einen weniger optimalen Plan wählt als vor einer bestimmten Änderung der Datenbankumgebung. Dies kann durch Änderungen an Statistiken, Beschränkungen, Umgebungseinstellungen, Abfrageparameter-Bindungen und Aktualisierungen der Datenbank-Engine verursacht werden.

R

RACI-Matrix

Siehe verantwortlich, rechenschaftspflichtig, konsultiert, informiert (RACI).

LAPPEN

Siehe Erweiterte Generierung beim Abrufen.

Ransomware

Eine bösartige Software, die entwickelt wurde, um den Zugriff auf ein Computersystem oder Daten zu blockieren, bis eine Zahlung erfolgt ist.

RASCI-Matrix

Siehe verantwortlich, rechenschaftspflichtig, konsultiert, informiert (RACI).

RCAC

Siehe Zugriffskontrolle für Zeilen und Spalten.

Read Replica

Eine Kopie einer Datenbank, die nur für Lesezwecke verwendet wird. Sie können Abfragen an das Lesereplikat weiterleiten, um die Belastung auf Ihrer Primärdatenbank zu reduzieren.

neu strukturieren

Siehe 7 Rs.

Recovery Point Objective (RPO)

Die maximal zulässige Zeitspanne seit dem letzten Datenwiederherstellungspunkt. Damit wird festgelegt, was als akzeptabler Datenverlust zwischen dem letzten Wiederherstellungspunkt und der Serviceunterbrechung gilt.

Wiederherstellungszeitziel (RTO)

Die maximal zulässige Verzögerung zwischen der Betriebsunterbrechung und der Wiederherstellung des Dienstes.

Refaktorierung

Siehe 7 Rs.

Region

Eine Sammlung von AWS Ressourcen in einem geografischen Gebiet. Jeder AWS-Region ist isoliert und unabhängig von den anderen, um Fehlertoleranz, Stabilität und Belastbarkeit zu gewährleisten. Weitere Informationen finden Sie unter Geben Sie an, was AWS-Regionen Ihr Konto verwenden kann.

Regression

Eine ML-Technik, die einen numerischen Wert vorhersagt. Zum Beispiel, um das Problem „Zu welchem Preis wird dieses Haus verkauft werden?“ zu lösen Ein ML-Modell könnte ein lineares Regressionsmodell verwenden, um den Verkaufspreis eines Hauses auf der Grundlage bekannter Fakten über das Haus (z. B. die Quadratmeterzahl) vorherzusagen.

rehosten

Siehe 7 Rs.

Veröffentlichung

In einem Bereitstellungsprozess der Akt der Förderung von Änderungen an einer Produktionsumgebung.

umziehen

Siehe 7 Rs.

neue Plattform

Siehe 7 Rs.

Rückkauf

Siehe 7 Rs.

Ausfallsicherheit

Die Fähigkeit einer Anwendung, Störungen zu widerstehen oder sich von ihnen zu erholen. Hochverfügbarkeit und Notfallwiederherstellung sind häufig Überlegungen bei der Planung der Ausfallsicherheit in der. AWS Cloud Weitere Informationen finden Sie unter AWS Cloud Resilienz.

Ressourcenbasierte Richtlinie

Eine mit einer Ressource verknüpfte Richtlinie, z. B. ein Amazon-S3-Bucket, ein Endpunkt oder ein Verschlüsselungsschlüssel. Diese Art von Richtlinie legt fest, welchen Prinzipalen der Zugriff gewährt wird, welche Aktionen unterstützt werden und welche anderen Bedingungen erfüllt sein müssen.

RACI-Matrix (verantwortlich, rechenschaftspflichtig, konsultiert, informiert)

Eine Matrix, die die Rollen und Verantwortlichkeiten aller an Migrationsaktivitäten und Cloud-Operationen beteiligten Parteien definiert. Der Matrixname leitet sich von den in der Matrix definierten Zuständigkeitstypen ab: verantwortlich (R), rechenschaftspflichtig (A), konsultiert (C) und informiert (I). Der Unterstützungstyp (S) ist optional. Wenn Sie Unterstützung einbeziehen, wird die Matrix als RASCI-Matrix bezeichnet, und wenn Sie sie ausschließen, wird sie als RACI-Matrix bezeichnet.

Reaktive Kontrolle

Eine Sicherheitskontrolle, die darauf ausgelegt ist, die Behebung unerwünschter Ereignisse oder Abweichungen von Ihren Sicherheitsstandards voranzutreiben. Weitere Informationen finden Sie unter Reaktive Kontrolle in Implementieren von Sicherheitskontrollen in AWS.

Beibehaltung

Siehe 7 Rs.

zurückziehen

Siehe 7 Rs.

Retrieval Augmented Generation (RAG)

Eine generative KI-Technologie, bei der ein LLM auf eine maßgebliche Datenquelle verweist, die sich außerhalb seiner Trainingsdatenquellen befindet, bevor eine Antwort generiert wird. Ein RAG-Modell könnte beispielsweise eine semantische Suche in der Wissensdatenbank oder in benutzerdefinierten Daten einer Organisation durchführen. Weitere Informationen finden Sie unter Was ist RAG.

Drehung

Der Vorgang, bei dem ein Geheimnis regelmäßig aktualisiert wird, um es einem Angreifer zu erschweren, auf die Anmeldeinformationen zuzugreifen.

Zugriffskontrolle für Zeilen und Spalten (RCAC)

Die Verwendung einfacher, flexibler SQL-Ausdrücke mit definierten Zugriffsregeln. RCAC besteht aus Zeilenberechtigungen und Spaltenmasken.

RPO

Siehe Recovery Point Objective.

RTO

Siehe Ziel der Wiederherstellungszeit.

Runbook

Eine Reihe manueller oder automatisierter Verfahren, die zur Ausführung einer bestimmten Aufgabe erforderlich sind. Diese sind in der Regel darauf ausgelegt, sich wiederholende Operationen oder Verfahren mit hohen Fehlerquoten zu rationalisieren.

S

SAML 2.0

Ein offener Standard, den viele Identitätsanbieter (IdPs) verwenden. Diese Funktion ermöglicht föderiertes Single Sign-On (SSO), sodass sich Benutzer bei den API-Vorgängen anmelden AWS Management Console oder die AWS API-Operationen aufrufen können, ohne dass Sie einen Benutzer in IAM für alle in Ihrer Organisation erstellen müssen. Weitere Informationen zum SAML-2.0.-basierten Verbund finden Sie unter Über den SAML-2.0-basierten Verbund in der IAM-Dokumentation.

SCADA

Siehe Aufsichtskontrolle und Datenerfassung.

SCP

Siehe Richtlinie zur Dienstkontrolle.

Secret

Interne AWS Secrets Manager, vertrauliche oder eingeschränkte Informationen, wie z. B. ein Passwort oder Benutzeranmeldedaten, die Sie in verschlüsselter Form speichern. Es besteht aus dem geheimen Wert und seinen Metadaten. Der geheime Wert kann binär, eine einzelne Zeichenfolge oder mehrere Zeichenketten sein. Weitere Informationen finden Sie unter Was ist in einem Secrets Manager Manager-Geheimnis? in der Secrets Manager Manager-Dokumentation.

Sicherheit durch Design

Ein systemtechnischer Ansatz, der die Sicherheit während des gesamten Entwicklungsprozesses berücksichtigt.

Sicherheitskontrolle

Ein technischer oder administrativer Integritätsschutz, der die Fähigkeit eines Bedrohungsakteurs, eine Schwachstelle auszunutzen, verhindert, erkennt oder einschränkt. Es gibt vier Haupttypen von Sicherheitskontrollen: präventiv, detektiv, reaktionsschnell und proaktiv.

Härtung der Sicherheit

Der Prozess, bei dem die Angriffsfläche reduziert wird, um sie widerstandsfähiger gegen Angriffe zu machen. Dies kann Aktionen wie das Entfernen von Ressourcen, die nicht mehr benötigt werden, die Implementierung der bewährten Sicherheitsmethode der Gewährung geringster Berechtigungen oder die Deaktivierung unnötiger Feature in Konfigurationsdateien umfassen.

System zur Verwaltung von Sicherheitsinformationen und Ereignissen (security information and event management – SIEM)

Tools und Services, die Systeme für das Sicherheitsinformationsmanagement (SIM) und das Management von Sicherheitsereignissen (SEM) kombinieren. Ein SIEM-System sammelt, überwacht und analysiert Daten von Servern, Netzwerken, Geräten und anderen Quellen, um Bedrohungen und Sicherheitsverletzungen zu erkennen und Warnmeldungen zu generieren.

Automatisierung von Sicherheitsreaktionen

Eine vordefinierte und programmierte Aktion, die darauf ausgelegt ist, automatisch auf ein Sicherheitsereignis zu reagieren oder es zu beheben. Diese Automatisierungen dienen als

detektive oder reaktionsschnelle Sicherheitskontrollen, die Sie bei der Implementierung bewährter AWS Sicherheitsmethoden unterstützen. Beispiele für automatisierte Antwortaktionen sind das Ändern einer VPC-Sicherheitsgruppe, das Patchen einer EC2 Amazon-Instance oder das Rotieren von Anmeldeinformationen.

Serverseitige Verschlüsselung

Verschlüsselung von Daten am Zielort durch denjenigen AWS-Service , der sie empfängt.

Service-Kontrollrichtlinie (SCP)

Eine Richtlinie, die eine zentrale Steuerung der Berechtigungen für alle Konten in einer Organisation in ermöglicht AWS Organizations. SCPs Definieren Sie Leitplanken oder legen Sie Grenzwerte für Aktionen fest, die ein Administrator an Benutzer oder Rollen delegieren kann. Sie können sie SCPs als Zulassungs- oder Ablehnungslisten verwenden, um festzulegen, welche Dienste oder Aktionen zulässig oder verboten sind. Weitere Informationen finden Sie in der AWS Organizations Dokumentation unter Richtlinien zur Dienststeuerung.

Service-Endpunkt

Die URL des Einstiegspunkts für einen AWS-Service. Sie können den Endpunkt verwenden, um programmgesteuert eine Verbindung zum Zielservice herzustellen. Weitere Informationen finden Sie unter AWS-Service -Endpunkte in der Allgemeine AWS-Referenz.

Service Level Agreement (SLA)

Eine Vereinbarung, in der klargestellt wird, was ein IT-Team seinen Kunden zu bieten verspricht, z. B. in Bezug auf Verfügbarkeit und Leistung der Services.

Service-Level-Indikator (SLI)

Eine Messung eines Leistungsaspekts eines Dienstes, z. B. seiner Fehlerrate, Verfügbarkeit oder Durchsatz.

Service-Level-Ziel (SLO)

Eine Zielkennzahl, die den Zustand eines Dienstes darstellt, gemessen anhand eines Service-Level-Indikators.

Modell der geteilten Verantwortung

Ein Modell, das die Verantwortung beschreibt, mit der Sie gemeinsam AWS für Cloud-Sicherheit und Compliance verantwortlich sind. AWS ist für die Sicherheit der Cloud verantwortlich, während Sie für die Sicherheit in der Cloud verantwortlich sind. Weitere Informationen finden Sie unter Modell der geteilten Verantwortung.

SIEM

Siehe Sicherheitsinformations- und Event-Management-System.

Single Point of Failure (SPOF)

Ein Fehler in einer einzelnen, kritischen Komponente einer Anwendung, der das System stören kann.

SLA

Siehe Service Level Agreement.

SLI

Siehe Service-Level-Indikator.

ALSO

Siehe Service-Level-Ziel.

split-and-seed Modell

Ein Muster für die Skalierung und Beschleunigung von Modernisierungsprojekten. Sobald neue Features und Produktversionen definiert werden, teilt sich das Kernteam auf, um neue Produktteams zu bilden. Dies trägt zur Skalierung der Fähigkeiten und Services Ihrer Organisation bei, verbessert die Produktivität der Entwickler und unterstützt schnelle Innovationen. Weitere Informationen finden Sie unter Schrittweiser Ansatz zur Modernisierung von Anwendungen in der. AWS Cloud

SPOTTEN

Siehe Single Point of Failure.

Sternschema

Eine Datenbank-Organisationsstruktur, die eine große Faktentabelle zum Speichern von Transaktions- oder Messdaten und eine oder mehrere kleinere dimensionale Tabellen zum Speichern von Datenattributen verwendet. Diese Struktur ist für die Verwendung in einem Data Warehouse oder für Business Intelligence-Zwecke konzipiert.

Strangler-Fig-Muster

Ein Ansatz zur Modernisierung monolithischer Systeme, bei dem die Systemfunktionen schrittweise umgeschrieben und ersetzt werden, bis das Legacy-System außer Betrieb

genommen werden kann. Dieses Muster verwendet die Analogie einer Feigenrebe, die zu einem etablierten Baum heranwächst und schließlich ihren Wirt überwindet und ersetzt. Das Muster wurde eingeführt von Martin Fowler als Möglichkeit, Risiken beim Umschreiben monolithischer Systeme zu managen. Ein Beispiel für die Anwendung dieses Musters finden Sie unter Schrittweises Modernisieren älterer Microsoft ASP.NET (ASMX)-Webservices mithilfe von Containern und Amazon API Gateway.

Subnetz

Ein Bereich von IP-Adressen in Ihrer VPC. Ein Subnetz muss sich in einer einzigen Availability Zone befinden.

Aufsichtskontrolle und Datenerfassung (SCADA)

In der Fertigung ein System, das Hardware und Software zur Überwachung von Sachanlagen und Produktionsabläufen verwendet.

Symmetrische Verschlüsselung

Ein Verschlüsselungsalgorithmus, der denselben Schlüssel zum Verschlüsseln und Entschlüsseln der Daten verwendet.

synthetisches Testen

Testen eines Systems auf eine Weise, die Benutzerinteraktionen simuliert, um potenzielle Probleme zu erkennen oder die Leistung zu überwachen. Sie können Amazon CloudWatch Synthetics verwenden, um diese Tests zu erstellen.

Systemaufforderung

Eine Technik, mit der einem LLM Kontext, Anweisungen oder Richtlinien zur Verfügung gestellt werden, um sein Verhalten zu steuern. Systemaufforderungen helfen dabei, den Kontext festzulegen und Regeln für Interaktionen mit Benutzern festzulegen.

T

tags

Schlüssel-Wert-Paare, die als Metadaten für die Organisation Ihrer Ressourcen dienen. AWS Mit Tags können Sie Ressourcen verwalten, identifizieren, organisieren, suchen und filtern. Weitere Informationen finden Sie unter Markieren Ihrer AWS -Ressourcen.

Zielvariable

Der Wert, den Sie in überwachtem ML vorhersagen möchten. Dies wird auch als Ergebnisvariable bezeichnet. In einer Fertigungsumgebung könnte die Zielvariable beispielsweise ein Produktfehler sein.

Aufgabenliste

Ein Tool, das verwendet wird, um den Fortschritt anhand eines Runbooks zu verfolgen. Eine Aufgabenliste enthält eine Übersicht über das Runbook und eine Liste mit allgemeinen Aufgaben, die erledigt werden müssen. Für jede allgemeine Aufgabe werden der geschätzte Zeitaufwand, der Eigentümer und der Fortschritt angegeben.

Testumgebungen

Siehe Umgebung.

Training

Daten für Ihr ML-Modell bereitstellen, aus denen es lernen kann. Die Trainingsdaten müssen die richtige Antwort enthalten. Der Lernalgorithmus findet Muster in den Trainingsdaten, die die Attribute der Input-Daten dem Ziel (die Antwort, die Sie voraussagen möchten) zuordnen. Es gibt ein ML-Modell aus, das diese Muster erfasst. Sie können dann das ML-Modell verwenden, um Voraussagen für neue Daten zu erhalten, bei denen Sie das Ziel nicht kennen.

Transit-Gateway

Ein Netzwerk-Transit-Hub, über den Sie Ihre Netzwerke VPCs und Ihre lokalen Netzwerke miteinander verbinden können. Weitere Informationen finden Sie in der Dokumentation unter Was ist ein Transit-Gateway. AWS Transit Gateway

Stammbasierter Workflow

Ein Ansatz, bei dem Entwickler Feature lokal in einem Feature-Zweig erstellen und testen und diese Änderungen dann im Hauptzweig zusammenführen. Der Hauptzweig wird dann sequentiell für die Entwicklungs-, Vorproduktions- und Produktionsumgebungen erstellt.

Vertrauenswürdiger Zugriff

Gewährung von Berechtigungen für einen Dienst, den Sie angeben, um Aufgaben in Ihrer Organisation AWS Organizations und in deren Konten in Ihrem Namen auszuführen. Der vertrauenswürdige Service erstellt in jedem Konto eine mit dem Service verknüpfte Rolle, wenn diese Rolle benötigt wird, um Verwaltungsaufgaben für Sie auszuführen. Weitere Informationen

finden Sie in der AWS Organizations Dokumentation unter Verwendung AWS Organizations mit anderen AWS Diensten.

Optimieren

Aspekte Ihres Trainingsprozesses ändern, um die Genauigkeit des ML-Modells zu verbessern. Sie können das ML-Modell z. B. trainieren, indem Sie einen Beschriftungssatz generieren, Beschriftungen hinzufügen und diese Schritte dann mehrmals unter verschiedenen Einstellungen wiederholen, um das Modell zu optimieren.

Zwei-Pizzen-Team

Ein kleines DevOps Team, das Sie mit zwei Pizzen ernähren können. Eine Teamgröße von zwei Pizzen gewährleistet die bestmögliche Gelegenheit zur Zusammenarbeit bei der Softwareentwicklung.

U

Unsicherheit

Ein Konzept, das sich auf ungenaue, unvollständige oder unbekannte Informationen bezieht, die die Zuverlässigkeit von prädiktiven ML-Modellen untergraben können. Es gibt zwei Arten von Unsicherheit: Epistemische Unsicherheit wird durch begrenzte, unvollständige Daten verursacht, wohingegen aleatorische Unsicherheit durch Rauschen und Randomisierung verursacht wird, die in den Daten liegt. Weitere Informationen finden Sie im Leitfaden Quantifizieren der Unsicherheit in Deep-Learning-Systemen.

undifferenzierte Aufgaben

Diese Arbeit wird auch als Schwerstarbeit bezeichnet. Dabei handelt es sich um Arbeiten, die zwar für die Erstellung und den Betrieb einer Anwendung erforderlich sind, aber dem Endbenutzer keinen direkten Mehrwert bieten oder keinen Wettbewerbsvorteil bieten. Beispiele für undifferenzierte Aufgaben sind Beschaffung, Wartung und Kapazitätsplanung.

höhere Umgebungen

Siehe Umgebung.

V

Vacuuming

Ein Vorgang zur Datenbankwartung, bei dem die Datenbank nach inkrementellen Aktualisierungen bereinigt wird, um Speicherplatz zurückzugewinnen und die Leistung zu verbessern.

Versionskontrolle

Prozesse und Tools zur Nachverfolgung von Änderungen, z. B. Änderungen am Quellcode in einem Repository.

VPC-Peering

Eine Verbindung zwischen zwei VPCs , die es Ihnen ermöglicht, den Verkehr mithilfe privater IP-Adressen weiterzuleiten. Weitere Informationen finden Sie unter Was ist VPC-Peering? in der Amazon-VPC-Dokumentation.

Schwachstelle

Ein Software- oder Hardwarefehler, der die Sicherheit des Systems beeinträchtigt.

W

Warmer Cache

Ein Puffer-Cache, der aktuelle, relevante Daten enthält, auf die häufig zugegriffen wird. Die Datenbank-Instance kann aus dem Puffer-Cache lesen, was schneller ist als das Lesen aus dem Hauptspeicher oder von der Festplatte.

warme Daten

Daten, auf die selten zugegriffen wird. Bei der Abfrage dieser Art von Daten sind mäßig langsame Abfragen in der Regel akzeptabel.

Fensterfunktion

Eine SQL-Funktion, die eine Berechnung für eine Gruppe von Zeilen durchführt, die sich in irgendeiner Weise auf den aktuellen Datensatz beziehen. Fensterfunktionen sind nützlich für die Verarbeitung von Aufgaben wie die Berechnung eines gleitenden Durchschnitts oder für den Zugriff auf den Wert von Zeilen auf der Grundlage der relativen Position der aktuellen Zeile.

Workload

Ein Workload ist eine Sammlung von Ressourcen und Code, die einen Unternehmenswert bietet, wie z. B. eine kundenorientierte Anwendung oder ein Backend-Prozess.

Workstream

Funktionsgruppen in einem Migrationsprojekt, die für eine bestimmte Reihe von Aufgaben verantwortlich sind. Jeder Workstream ist unabhängig, unterstützt aber die anderen Workstreams im Projekt. Der Portfolio-Workstream ist beispielsweise für die Priorisierung von Anwendungen, die Wellenplanung und die Erfassung von Migrationsmetadaten verantwortlich. Der Portfolio-Workstream liefert diese Komponenten an den Migrations-Workstream, der dann die Server und Anwendungen migriert.

WURM

Sehen Sie einmal schreiben, viele lesen.

WQF

Siehe AWS Workload-Qualifizierungsrahmen.

einmal schreiben, viele lesen (WORM)

Ein Speichermodell, das Daten ein einziges Mal schreibt und verhindert, dass die Daten gelöscht oder geändert werden. Autorisierte Benutzer können die Daten so oft wie nötig lesen, aber sie können sie nicht ändern. Diese Datenspeicherinfrastruktur gilt als unveränderlich.

Z

Zero-Day-Exploit

Ein Angriff, in der Regel Malware, der eine Zero-Day-Sicherheitslücke ausnutzt.

Zero-Day-Sicherheitslücke

Ein unfehlbarer Fehler oder eine Sicherheitslücke in einem Produktionssystem. Bedrohungsakteure können diese Art von Sicherheitslücke nutzen, um das System anzugreifen. Entwickler werden aufgrund des Angriffs häufig auf die Sicherheitsanfälligkeit aufmerksam.

Zero-Shot-Aufforderung

Bereitstellung von Anweisungen für die Ausführung einer Aufgabe an einen LLM, jedoch ohne Beispiele (Schnappschüsse), die ihm als Orientierungshilfe dienen könnten. Der LLM muss sein

vortrainiertes Wissen einsetzen, um die Aufgabe zu bewältigen. Die Effektivität von Zero-Shot Prompting hängt von der Komplexität der Aufgabe und der Qualität der Aufforderung ab. Siehe auch Few-Shot-Prompting.

Zombie-Anwendung

Eine Anwendung, deren durchschnittliche CPU- und Arbeitsspeichernutzung unter 5 Prozent liegt. In einem Migrationsprojekt ist es üblich, diese Anwendungen außer Betrieb zu nehmen.

Die vorliegende Übersetzung wurde maschinell erstellt. Im Falle eines Konflikts oder eines Widerspruchs zwischen dieser übersetzten Fassung und der englischen Fassung (einschließlich infolge von Verzögerungen bei der Übersetzung) ist die englische Fassung maßgeblich.