



Anwendung des AWS Well-Architected Frameworks für Amazon Neptune Analytics

# AWS Präskriptive Leitlinien



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

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 irreführt 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 .....	2
Ziele .....	2
Säule „Operational Excellence“ .....	3
Automatisieren Sie die Bereitstellung mithilfe eines IaC-Ansatzes .....	3
Design für den Betrieb .....	4
Nehmen Sie häufige, kleine, umkehrbare Änderungen vor .....	5
Implementieren Sie Observability für umsetzbare Erkenntnisse .....	5
Lernen Sie aus allen Betriebsausfällen .....	6
Verwenden Sie Protokollierungsfunktionen, um unbefugte oder ungewöhnliche Aktivitäten zu überwachen .....	6
Säule der Sicherheit .....	8
Implementieren Sie Datensicherheit .....	8
Schützen Sie Ihre Netzwerke .....	9
Implementieren Sie Authentifizierung und Autorisierung .....	10
Säule der 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
Säule der Leistungseffizienz .....	16
Verstehen Sie die Graphmodellierung für Analysen .....	16
Optimieren Sie Abfragen .....	19
Schreibvorgänge optimieren .....	20
Diagramme in der richtigen Größe .....	21
Säule der Kostenoptimierung .....	23
Verstehen Sie die Nutzungsmuster und die benötigten Dienste .....	23
Wählen Sie Ressourcen unter Berücksichtigung der Kosten aus .....	25
Säule der Nachhaltigkeit .....	27
Überlegen Sie sich Ihre AWS-Region Auswahl .....	27
Optimieren Sie den Verbrauch .....	28
Optimieren Sie Softwareentwicklungs- und Architekturmuster .....	28
Ressourcen .....	30
Referenzen .....	30

---

Blogbeiträge und Videos .....	30
Training .....	30
Dokumentverlauf .....	31
Glossar .....	32
# .....	32
A .....	33
B .....	36
C .....	38
D .....	42
E .....	46
F .....	48
G .....	50
H .....	51
I .....	53
L .....	56
M .....	57
O .....	61
P .....	64
Q .....	67
R .....	68
S .....	71
T .....	75
U .....	77
V .....	77
W .....	78
Z .....	79
.....	lxxx

# Anwendung des AWS Well-Architected Frameworks für Amazon Neptune Analytics

Michael Havey, Amazon Web Services (AWS)

Dezember 2024 ([Geschichte der Dokumente](#))

Sie können graphbasierte Lösungen auf Amazon Web Services (AWS) mithilfe von [Amazon Neptune](#) erstellen. Neptune beinhaltet [Neptune Analytics](#), eine speicheroptimierte Graphanalyse-Engine, mit der große Mengen an Grafikdaten schnell analysiert werden können, um Erkenntnisse zu gewinnen und Trends zu finden. Es kann Analysen von Daten in Ihrem vorhandenen [Neptune-Datenbankcluster](#) durchführen, oder Sie können Daten aus externen Datensätzen laden und analysieren. Dieses Handbuch enthält Anleitungen zur Anwendung der [AWS Well-Architected Framework-Prinzipien](#) bei der Planung Ihrer Neptune Analytics-Bereitstellung. [Die Anwendung des AWS Well-Architected Framework für Amazon Neptune behandelt dasselbe Thema für eine Neptune-Datenbank.](#)

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 Analytics 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 aus den unterstützten Bereitstellungsoptionen.
- Folgen Sie den AWS Well-Architected-Entwurfsmustern, die Ihnen helfen, Resilienz und Sicherheit zu verbessern.
- Entwerfen Sie Ihre Abfragen im Hinblick auf optimale Leistung und Kosteneinsparungen.
- Erfahren Sie, wie Sie Ihr Neptune Analytics-Diagramm in der Produktion betrieblich effizient verwalten können.

## Säule „Operational Excellence“

Die [Säule Operational Excellence](#) des AWS Well-Architected Framework konzentriert sich auf den Betrieb und die Überwachung von Systemen sowie die kontinuierliche 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 Best Practices befolgen und Amazon Neptune Analytics-Metriken und -Mechanismen verwenden 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 IaC an, um Neptune Analytics-Diagramme und verwandte Ressourcen bereitzustellen. Verwenden Sie für eine konsistente Umgebungskonfiguration die [Unterstützung für Neptune Analytics](#), die von [AWS CloudFormation](#) zur Bereitstellung von Graphen und privaten Endpunkten bereitgestellt wird.
- Wird verwendet CloudFormation , um [Neptune-Notebook-Instances auf Amazon SageMaker AI bereitzustellen](#). Sie können Notizbücher verwenden, um Daten in einem Neptune Analytics-Diagramm abzufragen und zu visualisieren.

- [Wenn Sie ein Neptune Analytics-Diagramm aus einer vorhandenen Quelle erstellen, z. B. einem Neptune-Datenbank-Cluster oder -Snapshot, oder Datendateien, die in Amazon Simple Storage Service \(Amazon S3\) bereitgestellt werden, überwachen Sie die Massenimportaufgabe.](#)
- Automatisieren Sie Betriebsabläufe in Neptune Analytics, z. B. [die Größenänderung des Diagramms, das Löschen und Erstellen eines Snapshots des Diagramms, das Wiederherstellen des Diagramms aus einem Snapshot sowie das Zurücksetzen und Neuladen des Diagramms.](#) Verwenden Sie die [Neptune Analytics-API](#), die über die [AWS Command Line Interface \(AWS CLI\)](#) oder [SDK s](#) verfügbar ist.
- Beurteilen Sie die erforderliche Verfügbarkeit Ihres Diagramms. Analysen sind oft kurzlebig; das Diagramm wird nur für die Zeit benötigt, die Sie für die Ausführung von Algorithmen benötigen. Wenn dies der Fall ist, verwenden Sie AWS CLI oder, um einen [Snapshot des Diagramms SDKs zu erstellen und es zu löschen](#), wenn es nicht mehr benötigt wird. Sie können [es dann später, falls erforderlich, aus einem Snapshot wiederherstellen](#).
- Speichern Sie Verbindungszeichenfolgen extern von Ihrem Client aus. Sie können Verbindungszeichenfolgen in [AWS Secrets Manager](#) oder an einem beliebigen Ort speichern, wo sie dynamisch geändert werden können.
- [Verwenden Sie Tags, um Metadaten zu Ihren Neptune Analytics-Ressourcen hinzuzufügen](#), und verfolgen Sie die Nutzung anhand von Tags. Mithilfe von Tags können Sie Ihre Ressourcen besser organisieren. Sie können beispielsweise ein allgemeines Tag auf Ressourcen in einer bestimmten Umgebung oder Anwendung anwenden. Sie können Tags auch verwenden, um die Abrechnung der Ressourcennutzung zu analysieren. Weitere Informationen finden Sie unter [Organisieren und Nachverfolgen von Kosten mithilfe von AWS Kostenzuordnungs-Tags](#) im AWS Billing User Guide. Darüber hinaus können Sie Bedingungen in Ihren AWS Identity and Access Management (IAM-) Richtlinien verwenden, um den Zugriff auf AWS Ressourcen anhand der für diese Ressource verwendeten Tags zu steuern. Hierzu können Sie den globalen Bedingungsschlüssel `aws:ResourceTag/tag-key` verwenden. Weitere Informationen finden Sie im IAM-Benutzerhandbuch unter [Steuern des Zugriffs auf AWS Ressourcen](#).

## Design für den Betrieb

Nutzen Sie Ansätze, um die Bedienung von Neptune Analytics-Diagrammen zu verbessern:

- Pflegen Sie separate Neptune Analytics-Diagramme für Entwicklung, Test und Produktion. Diese Diagramme können unterschiedliche Datensätze, Benutzer und Betriebskontrollen haben.

- Pflegen Sie separate Neptune Analytics-Diagramme für verschiedene Zwecke. Wenn beispielsweise zwei Gruppen von analytischen Benutzern separate Diagramme mit unterschiedlichen Zeitplänen, Modellen, Leistung und Verfügbarkeit sowie Nutzungsmustern benötigen SLAs, sollten Sie separate Diagramme für jede Gruppe verwalten.
- Bereiten Sie Benutzer und Betriebspersonal auf [Wartungsupdates](#) für Neptune Analytics vor.

## Nehmen Sie häufige, kleine, umkehrbare Änderungen vor

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

- Speichern Sie IaC-Vorlagen und -Skripts in einem Quellcodeverwaltungsdienst wie GitHub oder GitLab.

### Important

Speichern Sie keine AWS Anmeldeinformationen in der Quellcodeverwaltung.

- Erfordern Sie, dass IaC-Bereitstellungen einen CI/CD-Dienst (Continuous Integration and Continuous Delivery) wie oder verwenden. [AWS CodeDeploy](#)[AWS CodeBuild](#) Kompilieren, testen und implementieren Sie Code in einer Neptune Analytics-Umgebung außerhalb der Produktionsumgebung, bevor Sie ihn in ein Produktionsdiagramm hochstufen.

## Implementieren Sie Observability für umsetzbare Erkenntnisse

Verschaffen Sie sich ein umfassendes Verständnis von Verhalten, Leistung, Zuverlässigkeit, Kosten und Zustand von Workloads. Die folgenden Empfehlungen helfen Ihnen dabei, dieses Verständnis von Neptune Analytics zu erlangen:

- Überwachen Sie CloudWatch Amazon-Metriken für Neptune Analytics. Anhand dieser Metriken können Sie die Größe eines Diagramms (Anzahl der Knoten, Kanten und Vektoren plus Gesamtbytegröße), die CPU-Auslastung sowie die Anforderungs- und Fehlerraten bei Abfragen bestimmen.
- Erstellen Sie CloudWatch Dashboards und Alarmer für wichtige Kennzahlen wie `NumQueuedRequestsPerSec`, `NumOpenCypherRequestsPerSec`

GraphStorageUsagePercentGraphSizeBytes, und CPUUtilization sowie für Neptune-Kundenantworten, die Sie in Ihren Anwendungsprotokollen finden.

- Richten Sie Benachrichtigungen ein, um den Zustand des Neptune Analytics-Diagramms zu überwachen, z. B. wenn die Grafikgröße, die Anforderungsrate oder die CPU-Auslastung Ihren Schwellenwert überschreiten. Wenn beispielsweise in einer Grafik, die Sie deutlich erhöhen möchten, auf 90 Prozent GraphStorageUsagePercent gestiegen ist, entscheiden Sie, ob Sie die speicheroptimierte Kapazität der Neptune Capacity Unit (m-NCU) erhöhen möchten. Wenn die aktuelle m-NCU 128 ist, wird durch eine Erhöhung auf 256 der Speicherplatz um etwa 45 Prozent reduziert. Wenn der Wert häufig größer als Null NumQueuedRequestsPerSec ist, sollten Sie eine Erhöhung der m-NCU-Kapazität in Betracht ziehen, um mehr Rechenkapazität bereitzustellen. Alternativ können Sie die clientseitige Parallelität reduzieren.

## 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 in Ihrem Unternehmen Neptune verwenden, erstellen Sie einen gemeinsamen Chatroom oder eine Benutzergruppe, um Erfahrungen und bewährte Verfahren auszutauschen.

## Verwenden Sie Protokollierungsfunktionen, um unbefugte oder ungewöhnliche Aktivitäten zu überwachen

Verwenden Sie die Protokollierung, um ungewöhnliche Leistungs- und Aktivitätsmuster zu beobachten. Beachten Sie die folgenden bewährten Methoden:

- Neptune Analytics unterstützt die Protokollierung von Aktionen auf der Kontrollebene mithilfe von AWS CloudTrail Weitere Informationen finden Sie unter [Protokollieren von Neptune Analytics-API-Aufrufen mithilfe](#) von AWS CloudTrail Mit dieser Funktion können Sie die Erstellung, Aktualisierung und Löschung von Neptune Analytics-Ressourcen verfolgen. Für eine zuverlässige Überwachung und Alarmierung können Sie CloudTrail Ereignisse auch in [Amazon CloudWatch Logs](#) integrieren.

Um Ihre Analyse der Neptune Analytics-Serviceaktivitäten zu verbessern und Änderungen der Aktivitäten für einen zu identifizieren AWS-Konto, können Sie [CloudTrailProtokolle mithilfe von Amazon Athena abfragen](#). Beispielsweise können Sie mithilfe von Abfragen Trends ermitteln und Vorgänge nach Attributen (z. B. Quell-IP-Adresse oder Benutzer) trennen.

- Sie können es auch verwenden CloudTrail , um die [Protokollierung von Neptune Analytics-Datenebenenaktivitäten](#) wie Abfrageausführungen zu aktivieren. Sie können sehen, welche Abfragen ausgeführt werden, wie häufig sie ausgeführt werden und aus welcher Quelle sie stammen. Protokolliert standardmäßig CloudTrail keine Datenereignisse. Für Datenereignisse werden zusätzliche Gebühren fällig. Weitere Informationen finden Sie unter [AWS CloudTrail – Preise](#).
- Sie können Ihre Anwendungsaufrufe bei Neptune Analytics auch entweder in der Steuerungsebene oder in der Datenebene protokollieren. Wenn Sie beispielsweise die für Abfragen verwenden, können Sie [AWS SDK für Python \(Boto3\)](#) die [Protokollierung auf Debug-Ebene aktivieren](#), um eine Ablaufverfolgung von Anfragen an die Konsole oder Datei zu erhalten. Dies ist während der Entwicklung nützlich. Wir empfehlen Ihnen außerdem, Ausnahmen aus Ihrer Anwendung abzufangen und zu protokollieren.

# Säule der Sicherheit

Cloud-Sicherheit hat bei höchste Priorität. AWS 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 und. AWS Das [Modell der geteilten Verantwortung](#) beschreibt dies als Sicherheit der Cloud und Sicherheit in der Cloud:

- 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 von [AWS Compliance-Programmen](#). Weitere Informationen zu den Compliance-Programmen, die für Neptune gelten, finden Sie unter [AWS Services in Scope by Compliance](#) Program.
- 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 im [Datenschutz FAQs](#). Informationen zum Datenschutz in Europa finden Sie im Blogbeitrag [AWS Shared Responsibility Model und GDPR](#).

Die [Sicherheitssäule](#) des AWS Well-Architected Framework hilft Ihnen zu verstehen, wie Sie das Modell der gemeinsamen Verantwortung anwenden können, wenn Sie Neptune Analytics verwenden. In den folgenden Themen wird erklärt, wie Sie Neptune Analytics konfigurieren, um Ihre Sicherheits- und Compliance-Ziele zu erreichen. Sie lernen auch, wie Sie andere verwenden können AWS-Services , die Ihnen helfen, Ihre Neptune Analytics-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:

- Grafiknamen, Tags, IAM-Rollen und andere Metadaten sollten keine vertraulichen oder sensiblen Informationen enthalten, da diese Daten möglicherweise in Abrechnungs- oder Diagnoseprotokollen auftauchen.
- URIs oder Links zu externen Servern, die als Daten in Neptune gespeichert sind, sollten keine Anmeldeinformationen zur Validierung von Anfragen enthalten.
- Ein Neptune Analytics-Diagramm ist im Ruhezustand verschlüsselt. Sie können den Standardschlüssel oder einen Schlüssel AWS Key Management Service (AWS KMS) Ihrer Wahl verwenden, um das Diagramm zu verschlüsseln. Sie können auch Snapshots und Daten verschlüsseln, die beim Massenimport nach Amazon S3 exportiert werden. Sie können die Verschlüsselung entfernen, wenn der Import abgeschlossen ist.
- Wenn Sie die OpenCypher-Sprache 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 in der Neptune-Dokumentation](#).

## Schützen Sie Ihre Netzwerke

Sie können ein Neptune Analytics-Diagramm für öffentliche Konnektivität aktivieren, sodass es von außerhalb einer Virtual Private Cloud (VPC) erreicht werden kann. Diese Konnektivität ist standardmäßig deaktiviert. Für das Diagramm ist eine IAM-Authentifizierung erforderlich. Der Aufrufer muss eine Identität erhalten und über Berechtigungen zur Verwendung des Diagramms verfügen. Um beispielsweise [eine OpenCypher-Abfrage auszuführen, müsste der Aufrufer Lese-, Schreib- oder Löschberechtigungen für das jeweilige Diagramm haben](#).

Sie können auch [private Endpunkte für das Diagramm erstellen](#), um von einer VPC aus auf das Diagramm zuzugreifen. Wenn Sie den Endpunkt erstellen, geben Sie die VPC, Subnetze und Sicherheitsgruppen an, um den Zugriff auf den Graphen einzuschränken.

Um Ihre Daten während der Übertragung zu schützen, erzwingt Neptune Analytics SSL-Verbindungen über HTTPS zum Diagramm. Weitere Informationen finden Sie unter [Datenschutz in Neptune Analytics in der Neptune Analytics-Dokumentation](#).

## Implementieren Sie Authentifizierung und Autorisierung

Für Aufrufe eines Neptune Analytics-Graphen ist eine IAM-Authentifizierung erforderlich. Der Aufrufer muss eine Identität erhalten und über ausreichende Berechtigungen verfügen, um die Aktion im Diagramm ausführen zu können. Eine Beschreibung der API-Aktionen und der erforderlichen Berechtigungen finden Sie in der [Neptune Analytics-API-Dokumentation](#). Sie können [Bedingungsprüfungen erzwingen](#), um den Zugriff anhand von Tags einzuschränken.

Die IAM-Authentifizierung verwendet das [AWS Signature Version 4 \(Sigv4\)](#) -Protokoll. [Um die Nutzung in Ihrer Anwendung zu vereinfachen, empfehlen wir Ihnen, ein AWS SDK zu verwenden.](#) Verwenden Sie in Python beispielsweise den [Boto3-Client für Neptune Graph](#), der SigV4 abstrahiert.

Wenn Sie Daten in das Diagramm laden, verwendet das [Batch-Laden](#) die IAM-Anmeldeinformationen des Aufrufers. Der Anrufer muss über Berechtigungen zum Herunterladen von Daten von Amazon S3 verfügen, wobei die Vertrauensbeziehung eingerichtet ist, sodass Neptune Analytics die Rolle übernehmen kann, die Daten aus Amazon S3 S3-Dateien in das Diagramm zu laden.

Der [Massenimport](#) kann entweder während der Diagrammerstellung (durch das Infrastrukturteam) oder für ein vorhandenes, leeres Diagramm (von dem Datenentwicklungsteam, das über die Berechtigungen zum Starten von Importaufgaben verfügt) durchgeführt werden. In beiden Fällen übernimmt Neptune Analytics die IAM-Rolle, die der Aufrufer als Eingabe bereitstellt. Diese Rolle gibt ihr die Berechtigung, den Inhalt des Amazon S3 S3-Ordners, in dem die Eingabedaten gespeichert sind, zu lesen und aufzulisten.

# Säule der Zuverlässigkeit

Die [Zuverlässigkeitssäule](#) von AWS Well-Architected Framework umfasst die Fähigkeit eines Workloads, seine vorgesehene Funktion korrekt und konsistent auszuführen, wenn dies erwartet wird. Dazu gehört auch die Fähigkeit, den Workload während seines gesamten Lebenszyklus zu betreiben und zu testen.

Ausgangspunkt für eine zuverlässige Workload sind vorab getroffene Designentscheidungen für Software und Infrastruktur. Ihre Architekturentscheidungen wirken sich auf Ihr Workload-Verhalten in allen Well-Architected-Säulen aus. Aus Gründen der Zuverlässigkeit müssen Sie bestimmten Mustern folgen, wie in diesem Abschnitt beschrieben.

Die Säule Zuverlässigkeit konzentriert sich auf die folgenden Schlüsselbereiche:

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

## Neptune-Dienstkontingente verstehen

Ihr AWS Konto verfügt über Standardkontingente (früher als Limits bezeichnet) für jedes Konto. AWS-Service Wenn nicht anders angegeben, gilt jedes Kontingent spezifisch für eine Region. Sie können Erhöhungen für einige, aber nicht für alle Kontingente beantragen.

Um Kontingente für Neptune Analytics zu finden, öffnen Sie die [Service Quotas Console](#). Wählen Sie im Navigationsbereich Amazon Neptune AWS-ServicesAnalytics aus und wählen Sie es dann aus. Achten Sie auf die Kontingente für die Anzahl der Graphen und Snapshots, den maximal bereitgestellten Speicher für ein Diagramm und die API-Anforderungsraten.

Wenn der maximal bereitgestellte Speicher für Ihren Datensatz nicht ausreicht, sollten Sie abwägen, welche Knoten- und Edge-Typen für Ihre beabsichtigte analytische Nutzung unerlässlich sind. Laden Sie eine Teilmenge der Daten, sodass Analysen innerhalb der zulässigen bereitgestellten Kapazität möglich sind. Viele Analytics-Workloads, insbesondere solche, die Graph-Algorithmen ausführen, benötigen statt des vollständigen Transaktionsgraphen nur die Topologie mit einer begrenzten Anzahl von Eigenschaften. ([Eine Erläuterung der Unterschiede zwischen transaktionalen und analytischen Workloads finden Sie im Abschnitt Leistungseffizienz.](#))

Wenn die maximale Anzahl von Grafiken für Ihren Verwendungszweck nicht ausreicht:

- Erwägen Sie, Grafiken zu kombinieren, die ähnliche Verwendungszwecke haben.
- Beurteilen Sie, wie viele Grafiken gleichzeitig ausgeführt werden müssen. Wenn Sie einen Anwendungsfall für kurzlebige Analysen haben, erstellen Sie einen Snapshot von einem Diagramm und löschen Sie es, wenn es nicht mehr benötigt wird. Dadurch wird die Anzahl der Grafiken im Vergleich zum Kontingent reduziert.
- Erwägen Sie die Bereitstellung von Diagrammen in verschiedenen AWS-Konten Varianten.

## Verstehen Sie die Einsatzmuster von Neptune

Machen Sie sich mit den folgenden Entscheidungspunkten vertraut, wenn Sie planen, ein Neptune Analytics-Diagramm bereitzustellen:

- Seeding: Entscheiden Sie, ob Sie ein leeres Diagramm erstellen oder bei der Erstellung Daten mit Daten aus Amazon S3, einem vorhandenen Neptune-Datenbank-Cluster oder einem vorhandenen Neptune-Datenbank-Snapshot in dieses laden möchten.

Empfehlung: Wenn es sich bei der Quelle um einen Neptun-Cluster oder -Snapshot handelt, müssen Sie dessen Daten bei der Erstellung des Diagramms laden. Wenn es sich bei der Quelle um Amazon S3 handelt, laden Sie die Daten zum Zeitpunkt der Erstellung, wenn der Ladeaufwand erheblich ist und sich am besten als Infrastrukturbereitstellungsaktivität eignet. Wenn Sie es vorziehen, Daten als Datentechnik- oder Anwendungsaktivität zu laden, erstellen Sie ein leeres Diagramm und laden Sie später Daten aus Amazon S3.

- Kapazität: Schätzen Sie die erforderliche bereitgestellte Kapazität für ein Diagramm unter Berücksichtigung der Datengröße und der erwarteten Anwendungsnutzung.

Empfehlung: [Geben Sie bei der Erstellung den maximal bereitgestellten Speicher an](#), um die Diagrammgröße zu begrenzen. Diese Einstellung ist verpflichtend. Sie können die Kapazität bei Bedarf später ändern.

- Verfügbarkeit und Fehlertoleranz: Entscheiden Sie, ob Replikate für die Verfügbarkeit erforderlich sind. Ein Replikat fungiert als Warm-Standby für die Wiederherstellung im Falle eines Graphausfalls. Ein Diagramm mit Replikaten wird schneller wiederhergestellt als ein Diagramm ohne Replikate. Überlegen Sie auch, wie lange das Diagramm benötigt wird, ob es nur für kurzlebige Analysen bestimmt ist und, falls ja, wann es entfernt wird.

Empfehlung: Ermitteln Sie die Verfügbarkeitsanforderungen, z. B. wie lange das Diagramm nicht verfügbar sein kann und wann es entfernt werden kann, bevor Sie ein Diagramm erstellen.

- Netzwerk und Sicherheit: Stellen Sie fest, ob Sie öffentliche Konnektivität, private Konnektivität oder beides benötigen und ob Sie Ihre Daten verschlüsseln möchten.

Empfehlung: Machen Sie sich mit den organisatorischen Anforderungen vertraut, z. B. ob öffentliche Konnektivität zulässig ist und wo Graph-Client-Anwendungen bereitgestellt werden, bevor Sie ein Diagramm erstellen.

- Backups und Wiederherstellung: Ermitteln Sie, ob Snapshots erstellt werden sollen, und wenn ja, wann oder unter welchen Bedingungen. Überlegen Sie, ob Ihr Unternehmen Anforderungen an die Notfallwiederherstellung (DR) stellt.

Empfehlung: Das Erstellen von Snapshots ist eine manuelle Aktivität. Entscheiden Sie, wann Snapshots erstellt werden sollen, und berücksichtigen Sie Ihre DR-Anforderungen, bevor Sie ein Diagramm erstellen.

## Neptun-Cluster verwalten und skalieren

Ein Neptune Analytics-Diagramm besteht aus einer einzigen, speicheroptimierten Instanz. Die Kapazität (m-NCU) der Instanz wird zum Zeitpunkt der Erstellung festgelegt. Die Instanz kann vertikal skaliert werden, indem die bereitgestellte Kapazität durch eine [administrative Maßnahme](#) erhöht wird. Die bereitgestellte Kapazität kann auch verringert werden. Replikate sind passive Failover-Ziele, sodass sie den Maßstab eines Graphen nicht erhöhen. In dieser Hinsicht unterscheidet sich eine Graphreplik von einer [Neptune-Datenbank-Read-Replica](#), bei der es sich um eine aktive Instanz in einem Neptune-Cluster handelt, die Lesevorgänge von Anwendungen verarbeiten kann.

Replikate sind mit Kosten verbunden. Der Preis für das Replikat entspricht dem in der Abbildung angegebenen m-NCU-Kurs. Wenn ein Diagramm beispielsweise für 128 m-NCU bereitgestellt wird und über ein einzelnes Replikat verfügt, sind die Kosten doppelt so hoch wie bei einem gleichwertigen Diagramm ohne Replikate.

In der Analytik gibt es zwei Hauptgründe für eine Skalierung:

- Um mehr Speicher und CPU für analytische Abfragen und Algorithmen zur Verfügung zu stellen, weil die einzelne Abfrage teuer ist, der auszuführende Graph-Algorithmus von Natur aus komplex ist und aufgrund seiner Eingabe mehr Ressourcen benötigt, oder die Rate der gleichzeitigen

Anfragen ist hoch. Wenn bei solchen Abfragen out-of-memory Fehler auftreten, ist eine Skalierung eine vernünftige Abhilfe.

- Um eine größere Grafikgröße als geplant zu unterstützen. Wenn die derzeit bereitgestellte Kapazität beispielsweise 128 m-NCU zur Unterstützung von 60 GB an Quelldaten beträgt und Sie weitere 40 GB an Quelldaten benötigen, ist eine Erhöhung auf 256 m-NCU gerechtfertigt.

Überwachen Sie CloudWatch Metriken für Neptune Analytics, wie `NumQueuedRequestsPerSec`, `NumOpenCypherRequestsPerSec`, und `GraphStorageUsagePercentGraphSizeBytes`, um festzustellen `CPUUtilization`, ob eine Skalierung erforderlich ist. Sie können die Konfiguration eines Diagramms über die Konsole, AWS CLI, oder aktualisieren. SDKs (Beispiele und bewährte Verfahren finden Sie im Abschnitt [Operational Excellence](#).)

## Backups und Failover-Ereignisse verwalten

Verwenden Sie Replikate, um sicherzustellen, dass im Falle eines Fehlers ein Diagramm verfügbar bleibt. Ein Diagramm verwendet protokollbasierte Persistenz, um Änderungen in allen Availability Zones in einem zu übernehmen. AWS-Region Das Replikate fungiert als Warm-Standby und hat Zugriff auf diese Daten. Wenn ein Fehler auftritt, nimmt der Graph den Betrieb des Replikats wieder auf. Die Anwendung verwendet weiterhin denselben Endpunkt, um eine Verbindung zum Diagramm herzustellen. Anfragen während des Fehlers während des Fehlers führen zu Fehlern mit einer Ausnahme, dass der Dienst nicht verfügbar ist. Erwägen Sie, einen [Wiederholungsversuch mit Backoff-Muster](#) im Anwendungscode zu verwenden, um catch Fehler zu erkennen, und versuchen Sie es nach einem kurzen Intervall erneut. Neue Anfragen, die während des Failovers gestellt werden, werden in die Warteschlange gestellt und es kann zu einer längeren Latenz kommen.

Wenn kein Replikate konfiguriert ist und das Diagramm fehlschlägt, stellt Neptune Analytics die Daten von einem dauerhaften Speicher wieder her. Die Wiederherstellung dauert jedoch länger, da Neptune Ressourcen neu initialisieren muss.

Erstellen Sie Schnappschüsse des Graphen. (Neptune Analytics macht keine automatischen Schnappschüsse.) Wenn das Diagramm nach der Erstellung regelmäßig geändert wird, machen Sie regelmäßig Schnappschüsse, um seinen aktuellen Status zu erfassen. Löschen Sie ältere Schnappschüsse, wenn eine Wiederherstellung zu einem früheren Zeitpunkt nicht erforderlich ist.

Sie können Schnappschüsse mit anderen Konten und anderen Nutzern teilen. AWS-Regionen Wenn Sie DR-Anforderungen haben, sollten Sie überlegen, ob die Wiederherstellung des Diagramms aus

---

einem Snapshot in einer anderen Region Ihren Anforderungen an das Recovery Time Objective (RTO) und das Recovery Point Objective (RPO) entspricht.

## 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, anhand derer Sie das richtige Grafikdatenmodell und die richtigen Abfragesprachen finden können. Es enthält auch bewährte Verfahren, die bei der Aufnahme von Daten in Neptune Analytics und der Nutzung von Daten aus Neptune Analytics befolgt werden sollten.

Der Schwerpunkt der Leistungseffizienz konzentriert sich auf die folgenden Schlüsselbereiche:

- Graphmodellierung
- Optimierung von Abfragen
- Richtige Größe des Diagramms
- Optimierung schreiben

## Verstehen Sie die Graphmodellierung für Analysen

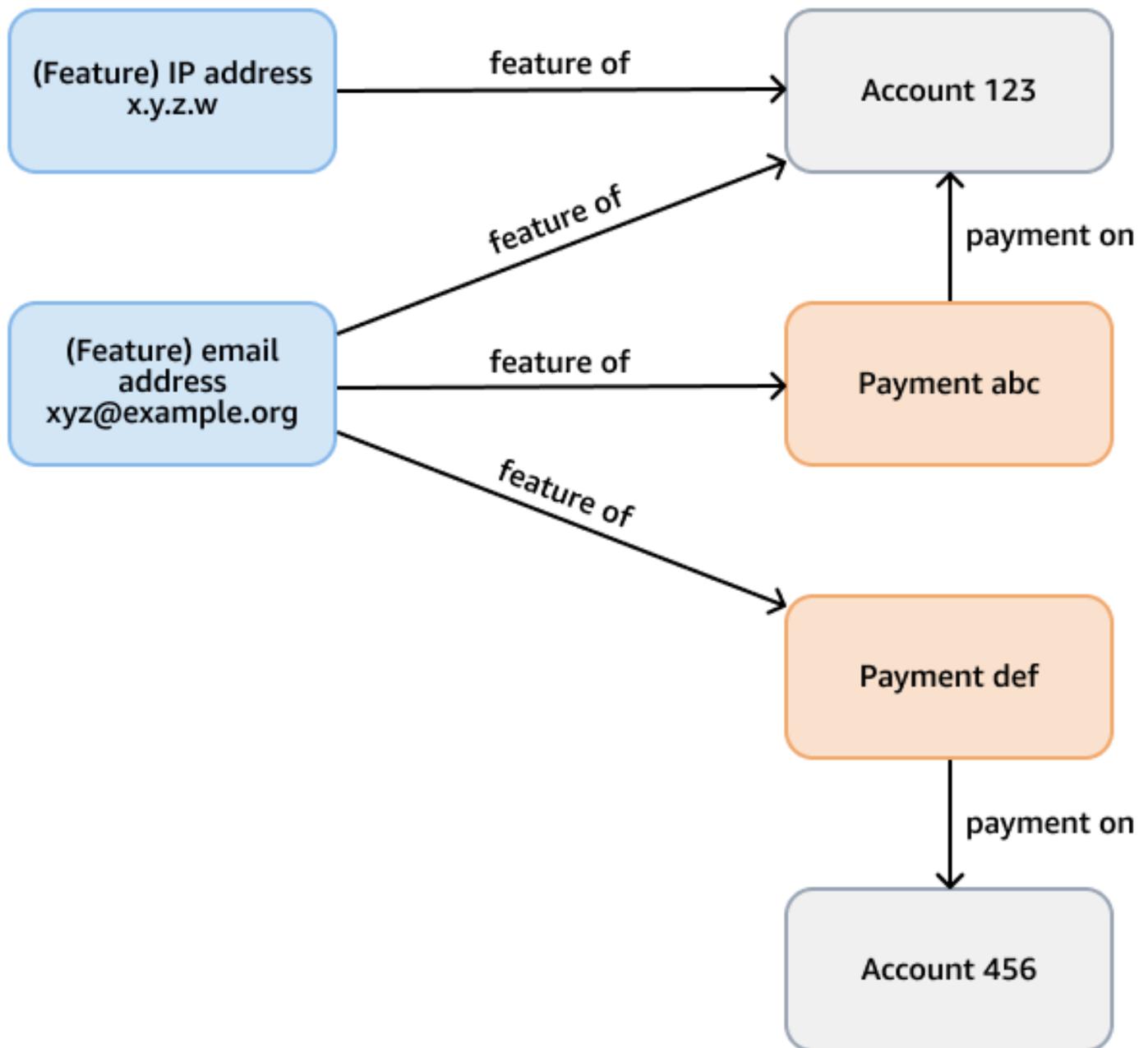
Der Leitfaden Applying the AWS Well-Architected Framework for Amazon Neptune befasst sich mit der [Graphmodellierung](#) zur Steigerung der Leistungseffizienz. Zu den Modellierungsentscheidungen, die sich auf die Leistung auswirken, gehören die Auswahl der erforderlichen Knoten und Kanten IDs, ihrer Labels und Eigenschaften, die Richtung der Kanten, ob Beschriftungen generisch oder spezifisch sein sollen und allgemein, wie effizient die Abfrage-Engine im Diagramm navigieren kann, um häufig auftretende Abfragen zu verarbeiten.

Diese Überlegungen gelten auch für Neptune Analytics. Es ist jedoch wichtig, zwischen transaktionalen und analytischen Nutzungsmustern zu unterscheiden. Ein Graphmodell, das

---

für Abfragen in einer Transaktionsdatenbank wie einer Neptune-Datenbank effizient ist, muss möglicherweise für Analysen neu gestaltet werden.

Stellen Sie sich zum Beispiel eine Betrugsgrafik in einer Neptune-Datenbank vor, deren Zweck darin besteht, Kreditkartenzahlungen auf betrügerische Muster zu überprüfen. Dieses Diagramm kann Knoten enthalten, die Konten, Zahlungen und Funktionen (wie E-Mail-Adresse, IP-Adresse, Telefonnummer) sowohl des Kontos als auch der Zahlung darstellen. Dieses verknüpfte Diagramm unterstützt Abfragen wie das Durchlaufen eines Pfads mit variabler Länge, der von einer bestimmten Zahlung ausgeht und mehrere Sprünge benötigt, um verwandte Funktionen und Konten zu finden. Die folgende Abbildung zeigt ein solches Diagramm.



Die Analyseanforderungen könnten spezifischer sein, z. B. das Auffinden von Kontengemeinschaften, die durch ein Feature miteinander verknüpft sind. Zu diesem Zweck können Sie den [WCC-Algorithmus \(Weakly Connected Components\)](#) verwenden. Es ist ineffizient, ihn mit dem Modell aus dem vorherigen Beispiel zu vergleichen, da er mehrere verschiedene Arten von Knoten und Kanten durchqueren muss. Das Modell im nächsten Diagramm ist effizienter. Es verbindet account Knoten mit einer shares feature Kante, wenn die Konten selbst — oder Zahlungen von den Konten — ein gemeinsames Merkmal aufweisen. Account 123 hat zum Beispiel eine E-Mail-Funktion

xyz@example.org und Account 456 verwendet dieselbe E-Mail für eine Zahlung (). Payment def



Die Rechenkomplexität von WCC ergibt sich  $O(|E| \log D)$  aus der Anzahl der Kanten im Graphen und  $D$  dem Durchmesser (der Länge des längsten Pfades), der die Knoten verbindet.  $|E|$  Da das Transaktionsmodell unwichtige Knoten und Kanten auslässt, werden sowohl die Anzahl der Kanten als auch der Durchmesser optimiert und die Komplexität des WCC-Algorithmus reduziert.

Wenn Sie Neptune Analytics verwenden, gehen Sie von den erforderlichen Algorithmen und analytischen Abfragen aus. Falls erforderlich, gestalten Sie das Modell neu, um diese Abfragen zu optimieren. Sie können das Modell umformen, bevor Sie Daten in das Diagramm laden, oder Abfragen schreiben, die vorhandene Daten im Diagramm ändern.

## Optimieren Sie Abfragen

Folgen Sie diesen Empfehlungen, um Neptune Analytics-Abfragen zu optimieren:

- Verwenden Sie [parametrisierte Abfragen](#) und den [Abfrageplan-Cache](#), der standardmäßig aktiviert ist. Wenn Sie den Plan-Cache verwenden, bereitet die Engine die Abfrage für die spätere Verwendung vor — vorausgesetzt, die Abfrage wird in 100 Millisekunden oder weniger abgeschlossen —, wodurch Zeit bei nachfolgenden Aufrufen gespart wird.
- Führen Sie bei langsamen Abfragen einen [Erklärungsplan aus, um Engpässe zu erkennen und entsprechende Verbesserungen vorzunehmen](#).
- Wenn Sie die [Vektorähnlichkeitssuche](#) verwenden, entscheiden Sie, ob kleinere Einbettungen zu genaueren Ähnlichkeitsergebnissen führen. Sie können kleinere Einbettungen effizienter erstellen, speichern und durchsuchen.
- Folgen Sie den [dokumentierten Best Practices für die Verwendung von OpenCypher](#) in Neptune Analytics. [Verwenden Sie beispielsweise abgeflachte Maps in einer UNWIND-Klausel und geben Sie nach Möglichkeit Kantenbeschriftungen an](#).

- Wenn Sie einen [Graphalgorithmus verwenden, sollten](#) Sie sich mit den Eingaben und Ausgaben des Algorithmus, seiner Rechenkomplexität und seiner Funktionsweise im Großen und Ganzen vertraut machen.
  - Bevor Sie einen Graphalgorithmus aufrufen, verwenden Sie eine MATCH Klausel, um den Eingabeknotensatz zu minimieren. Folgen Sie beispielsweise den [Beispielen](#) in der Neptune Analytics-Dokumentation, um Knoten einzuschränken, von denen aus eine [Breadth-First-Suche \(BFS\)](#) durchgeführt werden kann.
  - Filtern Sie nach Möglichkeit nach Knoten- und Kantenbeschriftungen. BFS verfügt beispielsweise über Eingabeparameter, um die Durchquerung nach einer bestimmten Knotenbeschriftung (`vertexLabel`) oder bestimmten Kantenbeschriftungen (`()`) zu filtern. `edgeLabels`
  - Verwenden Sie Begrenzungsparameter, um beispielsweise Ergebnisse einzuschränken `maxDepth`.
  - Experimentieren Sie mit dem `concurrency` Parameter. Versuchen Sie es mit einem Wert von 0, der alle verfügbaren Algorithmus-Threads verwendet, um die Verarbeitung zu parallelisieren. Vergleichen Sie das mit der Single-Thread-Ausführung, indem Sie den Parameter auf 1 setzen. Ein Algorithmus kann in einem einzigen Thread schneller abgeschlossen werden, insbesondere bei kleineren Eingaben, wie z. B. bei Suchanfragen mit oberflächlicher Breite, bei denen Parallelität keine messbare Verkürzung der Ausführungszeit bewirkt und zu Mehraufwand führen kann.
  - Wählen Sie zwischen ähnlichen Arten von Algorithmen. [Bellman-Ford](#) und [Delta-Stepping](#) sind beispielsweise beide Algorithmen mit kürzestem Pfad aus einer einzigen Quelle. Wenn Sie mit Ihrem eigenen Datensatz testen, probieren Sie beide Algorithmen aus und vergleichen Sie die Ergebnisse. Delta-Stepping ist aufgrund seiner geringeren Rechenkomplexität oft schneller als Bellman-Ford. Die Leistung hängt jedoch vom Datensatz und den Eingabeparametern ab, insbesondere vom Parameter `delta`

## Schreibvorgänge optimieren

Gehen Sie wie folgt vor, um Schreibvorgänge in Neptune Analytics zu optimieren:

- Suchen Sie nach dem effizientesten Weg, Daten in ein Diagramm zu laden. Wenn Sie Daten aus Amazon S3 laden, verwenden Sie den [Massenimport](#), wenn die Daten größer als 50 GB sind. Verwenden Sie für kleinere Daten das [Batch-Laden](#). Wenn beim Batch-Load out-of-memory Fehler auftreten, sollten Sie erwägen, den m-NCU-Wert zu erhöhen oder die Last in mehrere Anfragen

- aufzuteilen. Eine Möglichkeit, dies zu erreichen, besteht darin, Dateien auf mehrere Präfixe im S3-Bucket aufzuteilen. Rufen Sie in diesem Fall Batch Load für jedes Präfix separat auf.
- Verwenden Sie den Massenimport oder den Batch-Loader, um den ersten Satz von Grafikdaten aufzufüllen. Verwenden Sie transaktionale OpenCypher-Operationen zum Erstellen, Aktualisieren und Löschen nur für kleine Änderungen.
  - Verwenden Sie entweder den Massenimport oder den Batch-Loader mit einer Parallelität von 1 (Single-Thread), um Einbettungen in das Diagramm aufzunehmen. Versuchen Sie, Einbettungen mit einer dieser Methoden im Voraus zu laden.
  - Beurteilen Sie die Größe der Vektoreinbettungen, die für eine genaue Ähnlichkeitssuche in Algorithmen zur Suche nach Vektorähnlichkeit erforderlich sind. Verwenden Sie nach Möglichkeit eine kleinere Dimension. Dies führt zu einer schnelleren Ladegeschwindigkeit für Einbettungen.
  - Verwenden Sie bei Bedarf mutierte Algorithmen, um sich algorithmische Ergebnisse zu merken. Der [Algorithmus Degree Mutate Centrality](#) ermittelt beispielsweise den Grad jedes Eingabeknotens und schreibt diesen Wert als Eigenschaft des Knotens. Wenn sich die Verbindungen, die diese Knoten umgeben, anschließend nicht ändern, enthält die Eigenschaft das richtige Ergebnis. Der Algorithmus muss nicht erneut ausgeführt werden.
  - Verwenden Sie die [administrative Aktion zum Zurücksetzen des Graphen](#), um alle Knoten, Kanten und Einbettungen zu löschen, falls Sie von vorne beginnen müssen. Das Löschen aller Knoten, Kanten und Einbettungen mithilfe einer OpenCypher-Abfrage ist nicht möglich, wenn Ihr Diagramm groß ist. Eine einzelne Drop-Abfrage für einen großen Datensatz kann zu einer Zeitüberschreitung führen. Mit zunehmender Größe dauert es länger, bis der Datensatz entfernt wird, und die Transaktionsgröße nimmt zu. Im Gegensatz dazu ist die Zeit bis zum Zurücksetzen eines Diagramms in etwa konstant, und die Aktion bietet die Möglichkeit, vor der Ausführung einen Snapshot zu erstellen.

## Diagramme in der richtigen Größe

Die Gesamtleistung hängt von der bereitgestellten Kapazität eines Neptune Analytics-Diagramms ab. Die Kapazität wird in Einheiten gemessen, die als speicheroptimierte Neptun-Kapazitätseinheiten (m-) bezeichnet werden. NCUs Stellen Sie sicher, dass Ihr Diagramm ausreichend groß ist, um Ihre Grafikgröße und Abfragen zu unterstützen. Beachten Sie, dass eine höhere Kapazität nicht unbedingt die Leistung einer einzelnen Abfrage verbessert.

Wenn möglich, erstellen Sie das Diagramm, indem Sie Daten aus einer vorhandenen Quelle wie Amazon S3 oder einem vorhandenen Neptune-Cluster oder -Snapshot importieren. [Sie können](#)

---

[Grenzen für die Mindest- und Höchstkapazität festlegen](#). Sie können die [bereitgestellte Kapazität auch in einem vorhandenen Diagramm ändern](#).

Überwachen Sie CloudWatch Kennzahlen wie `NumQueuedRequestsPerSec`, `CPUUtilization`, `NumOpenCypherRequestsPerSec`, `GraphStorageUsagePercent`, `GraphSize`. Stellen Sie fest, ob mehr Kapazität benötigt wird, um die Größe und Auslastung Ihres Diagramms zu unterstützen. Weitere Informationen zur Interpretation einiger dieser Kennzahlen finden Sie im Abschnitt [Operationelle Exzellenz](#).

# Säule der Kostenoptimierung

Die [Säule der Kostenoptimierung](#) des AWS Well-Architected Framework konzentriert sich auf die Vermeidung unnötiger Kosten. Die folgenden Empfehlungen können Ihnen dabei helfen, die Entwurfsprinzipien zur Kostenoptimierung und die architektonischen Best Practices für Neptune Analytics 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

Bevor Sie Neptune Analytics einsetzen, sollten Sie prüfen, ob Ihr Anwendungsfall gut für Graph Analytics geeignet ist.

- **Graphdatenbanken:** Eine Graphdatenbank wie 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 sich in diesem Anwendungsfall, ob Ihre Daten besser als Attribute eines Objekts dargestellt werden könnten.
  - JSON- oder Blob-Daten (Binary Large Object), die als Eigenschaften gespeichert werden.
- **Graphanalyse:** Neptune Analytics ist eine Datenbank-Engine für Graphenanalysen, mit der große Mengen an Grafikdaten im Speicher schnell analysiert werden können, um Erkenntnisse zu gewinnen und Trends zu finden. Sie können Graphdaten sowohl in einer Neptune-Datenbank als auch in einem Neptune Analytics-Diagramm speichern und abfragen. Eine Neptune-Datenbank eignet sich am besten für skalierbare Anforderungen an die Online-Transaktionsverarbeitung (OLTP). Neptune Analytics eignet sich am besten für kurzlebige Analytics-Workloads. Sie können die beiden in Kombination verwenden, indem Sie Daten aus Ihrer transaktionsorientierten Neptune-Datenbank in ein Neptune Analytics-Diagramm laden, um Analysen dieser Daten durchzuführen. Wenn die Analyse abgeschlossen ist, können Sie das Neptune Analytics-Diagramm entfernen.

Einen detaillierteren Vergleich finden Sie in der Neptune Analytics-Dokumentation unter [Wann sollten Sie Neptune Analytics verwenden und wann sollten Sie Neptune Database verwenden](#).

Bestimmen Sie unter Berücksichtigung der Kosten, wie Sie Ihr Neptune Analytics-Diagramm am besten füllen können.

- [Importieren Sie Grafikdaten](#), die in einem S3-Bucket gespeichert sind, in großen Mengen. Wir empfehlen diese Option, wenn Ihre Daten zuvor für das Masseladen in eine Neptune-Datenbank bereitgestellt wurden, oder wenn Sie die zu analysierenden Daten bereits in [CSV oder anderen unterstützten Formaten, die für den Massenimport erforderlich sind, haben oder](#) ohne Weiteres erstellen können. Sie können den Massenimport als Teil der Diagrammerstellung ausführen. [Sie können der Mindest- und Höchstkapazität Grenzen setzen](#). Sie können [den Import auch für ein zuvor erstelltes leeres Diagramm ausführen](#) und [die Importaufgabe überwachen](#), während sie ausgeführt wird.
- [Sie können ein leeres Diagramm erstellen und es dann mithilfe einer OpenCypher-Abfrage mithilfe von Batch-Load auffüllen](#). Diese Option ist ideal, wenn die zu ladenden Daten in Amazon S3 bereitgestellt werden und kleiner als 50 GB sind.
- Sie können [das Diagramm aus Daten in Ihrem Neptune-Datenbank-Cluster füllen \(unterstützt in Neptune Database Version 1.3.0 oder höher\)](#). Die Absicht dieses Musters besteht darin, Analysen für Daten durchzuführen, die sich derzeit in Ihrer Graphdatenbank befinden. Selbst wenn die Datenbank ursprünglich per Bulk-Load aufgefüllt wurde, könnte sie sich seitdem erheblich verändert haben. Um aus der Datenbank zu importieren, kloniert Neptune Analytics Ihre Datenbank und exportiert Daten aus dem Klon in einen S3-Bucket. Dieses Verfahren verursacht Kosten: insbesondere Neptune-Datenbankkosten für den Betrieb des Klons und Amazon S3 S3-Kosten für die Speicherung und Nutzung der exportierten Daten. Der Klon wird entfernt, wenn der Export abgeschlossen ist. Sie können die exportierten Daten in Amazon S3 löschen.
- Sie können [das Diagramm aus dem Snapshot eines Neptune-Datenbank-Clusters auffüllen](#). Dies ähnelt der vorherigen Option, außer dass es sich bei der Quelle um einen Datenbank-Snapshot handelt. Um aus einem Snapshot zu importieren, stellt Neptune Analytics den Snapshot zunächst in einem neuen Datenbank-Cluster wieder her und exportiert dann die Daten in einen S3-Bucket. Dieses Verfahren verursacht Kosten: insbesondere Neptune-Datenbankkosten für den Betrieb des wiederhergestellten Clusters und Amazon S3 S3-Kosten für die Speicherung und Nutzung der exportierten Daten.
- Sie können auch OpenCypher-Abfragen durchführen, um Daten zu erstellen, zu aktualisieren oder zu löschen, indem Sie ACID-konforme Transaktionen (Atomicity, Consistency, Isolation, Durability)

im Diagramm verwenden. Wir empfehlen diesen Ansatz, um kleine Aktualisierungen vorzunehmen, aber nicht, um das Diagramm zu erstellen.

Wenn die für Analysen benötigten Daten bereits in Amazon S3 bereitgestellt wurden, empfehlen wir den Massenimport oder das Batch-Load. Diese sind kostengünstiger als das Auffüllen des Graphen aus einem Neptune-Datenbank-Cluster oder -Snapshot.

## Wählen Sie Ressourcen unter Berücksichtigung der Kosten aus

Bei der [Preisgestaltung von Neptune Analytics](#) wird eine Einheit verwendet, die als speicheroptimierte Neptune Capacity Unit (m-NCU) bekannt ist. Für die Ausführung eines Diagramms mit einer bestimmten m-NCU fallen feste Kosten pro Stunde an. Ein Diagramm kann Replikate für den Failover enthalten, und für diese Replikate fallen auch stündliche m-NCU-Kosten an.

Wir empfehlen die folgenden bewährten Methoden, um die Kapazität abzuschätzen, die Kosten zu begrenzen und die Kosten anhand der Leistung zu überwachen:

- Wenn möglich, erstellen Sie das Diagramm, indem Sie Daten aus einer vorhandenen Quelle importieren: Daten, die in Amazon S3 bereitgestellt wurden, oder aus einem vorhandenen Neptune-Cluster oder -Snapshot. Dies erspart Ihnen Mühe, da Neptune Analytics die Schwerarbeit beim Auslesen des Diagramms übernimmt und [Sie eine begrenzte maximale Kapazität angeben können](#).
- Sie können die [bereitgestellte Kapazität in einem vorhandenen Diagramm ändern](#).
- Wenn das Diagramm nicht mehr benötigt wird, können Sie [einen Snapshot erstellen und das Diagramm löschen](#). Wenn Sie es erneut verwenden müssen, können Sie das Diagramm aus dem Snapshot wiederherstellen.
- Sie können die Anzahl der Replikate wählen, wenn Sie das Diagramm erstellen. Legen Sie den Wert entsprechend Ihren Anforderungen an die Verfügbarkeit von Analysen fest. Sparen Sie Kosten, indem Sie diese Einstellung minimieren. Der Höchstwert von 2 erlaubt zwei Replikatinstanzen in separaten Availability Zones. Der Mindestwert von 0 bedeutet, dass Neptune Analytics kein Replikat ausführt. Die Wiederherstellung ist jedoch schneller, wenn ein Replikat verfügbar ist. Eine Erläuterung des Ausfalls und der Wiederherstellung in Diagrammen finden Sie im Abschnitt „[Zuverlässigkeit](#)“.
- Überwachen Sie die Neptune Analytics-Ausgaben für aktuelle und vergangene Abrechnungszeiträume mithilfe von [AWS Fakturierung und Kostenmanagement](#)

- Überwachen Sie die Neptune Analytics-Metriken insbesondere auf `NumQueuedRequestsPerSec`, `NumOpenCyperRequestsPerSec`, `GraphStorageUsagePercent`, und `GraphSizeBytes`, um zu beurteilen `CPUUtilization`, ob die bereitgestellte Kapazität für das Diagramm angemessen dimensioniert ist. CloudWatch Stellen Sie fest, ob eine kleinere Kapazität der beobachteten Anforderungsrate, der CPU-Auslastung und der Größe des Diagramms gerecht werden kann.
- Wenn Sie einen privaten Endpunkt für Ihr Diagramm benötigen, achten Sie auf die Kosten für elastische IPs VPC-Endpunkte (Virtual Private Cloud), NAT-Gateways oder andere VPC-bezogene Kosten. Weitere Informationen finden Sie unter [Amazon VPC-Preise](#) und [EC2Amazon-Preise](#).
- Möglicherweise möchten Sie eine oder mehrere Neptune-Notebook-Instances ausführen, um eine Client-Oberfläche bereitzustellen, mit der Entwickler und Analysten das Diagramm abfragen und visualisieren können (siehe Preise für [Neptune](#) Workbench). Um die Kosten zu minimieren, teilen Sie sich die Instanz gemeinsam mit anderen Benutzern und erstellen Sie separate Notizbuchordner für jeden Benutzer. Fahren Sie die Instanz herunter, wenn sie nicht verwendet wird. Einen Ansatz zur Automatisierung des Herunterfahrens finden Sie im AWS Blogbeitrag [Automatisieren Sie das Stoppen und Starten von Amazon Neptune Neptune-Umgebungsressourcen mithilfe von Ressourcen-Tags](#).

# Säule der Nachhaltigkeit

Die [Nachhaltigkeitssäule](#) des AWS Well-Architected Framework konzentriert sich auf die 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
- Antizipation und Einführung neuer, effizienterer Softwareangebote
- Nutzung von Managed Services
- Reduzierung der nachgelagerten Auswirkungen

Dieser Leitfaden konzentriert sich darauf, Ihre Auswirkungen zu verstehen. 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 Lösungen mit geringerer 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 Überlegungen erörtert, die sich positiv auf die Umwelt auswirken können, wenn sie bei der Planung Ihrer Arbeitslast und Ihrem laufenden Betrieb berücksichtigt werden

## Überlegen Sie sich Ihre AWS-Region Auswahl

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 Ihren Workload geeignet sein könnten, und vergleichen Sie Ihre Liste mit den [Regionen, in denen Neptune Analytics verfügbar](#) ist.

## Optimieren Sie den Verbrauch

Minimiere den Verbrauch von Neptune Analytics, indem du wie folgt vorgehst:

- Analytik ist oft kurzlebig. Das Diagramm wird nur für die Zeit benötigt, um Algorithmen auszuführen und die Ergebnisse aufzuzeichnen. Wenn dies der Fall ist, [erstellen Sie einen Snapshot des Diagramms und löschen Sie es](#), wenn es nicht mehr benötigt wird. Sie können [es bei Bedarf später aus einem Snapshot wiederherstellen](#).
- Wenn die Arbeitslast kurzlebig ist und Sie flexibel entscheiden können, wann die Analyse ausgeführt werden soll, sollten Sie day-to-day Trends beim Stromverbrauch berücksichtigen. Die Nachfrage nach Elektrizität ist zu bestimmten Zeiten höher. Wenn Sie sich in den Vereinigte Staaten befinden, finden Sie die [Kennzahlen zum täglichen Stromverbrauch](#) auf der Website der U.S. Energy Information Administration (EIA). Führen Sie Workloads in Ihrer Region nach Möglichkeit außerhalb der Spitzenzeiten aus.
- Wenn der Workload nicht kurzlebig ist, sondern nur für begrenzte Zeiträume verfügbar sein muss, löschen Sie das Diagramm und stellen Sie es bei Bedarf aus einem Snapshot wieder her. Wenn die Verfügbarkeit einem Zeitplan folgt, automatisieren Sie den Wiederherstellungsprozess mithilfe von Skripten, sodass das Diagramm zur geplanten Zeit fertig ist.
- Wenn die Daten schreibgeschützt sind oder sich seit dem letzten Snapshot nicht geändert haben, erstellen Sie vor dem Löschen keinen weiteren Snapshot.
- Stoppen Sie Neptune-Notizbücher, wenn sie nicht verwendet werden.
- Überwachen Sie CloudWatch Metriken wie `NumQueuedRequestsPerSec`, `CPUUtilization`, `NumOpenCypherRequestsPerSec`, `GraphStorageUsagePercent`, `GraphSize`. Stellen Sie fest, ob eine kleinere Instance-Kapazität der beobachteten Anforderungsrate, CPU-Auslastung und Grafikgröße gerecht werden kann.

## Optimieren Sie Softwareentwicklungs- 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 bewährten Methoden gehören:

- Optimieren Sie Abfragen und grafische Algorithmus-Aufrufe. Verwenden Sie parametrisierte Abfragen und [den Abfrageplan-Cache](#), der standardmäßig aktiviert ist. Führen Sie bei langsamen Abfragen einen [Explain-Plan](#) aus, um Verbesserungen vorzunehmen. Wenn Sie

die [Vektorähnlichkeitssuche](#) verwenden, sollten Sie entscheiden, ob kleinere Einbettungen zu genaueren Ähnlichkeitsergebnissen führen, da kleinere Einbettungen effizienter erstellt, gespeichert und durchsucht werden können. Bevor Sie einen [Graphalgorithmus](#) aufrufen, verwenden Sie eine MATCH Klausel, um den Eingabeknotensatz zu minimieren. Filtern Sie nach Knoten- und Kantenbeschriftungen, wenn möglich.

- Suchen Sie nach dem effizientesten Weg, Daten in das Diagramm zu laden. Wenn Sie Daten aus Amazon S3 laden, verwenden Sie den [Massenimport](#), wenn die Daten größer als 50 GB sind. Verwenden Sie [Batch-Load](#) für kleinere Daten.
- Bitten Sie die Entwickler, Neptune-Notebook-Instanzen gemeinsam zu nutzen, anstatt dass jede ihre eigene Instanz erstellt. Erstellen Sie separate Notizbuchordner für jeden Entwickler auf einer einzigen Jupyter-Instanz. Fahren Sie die Instanz herunter, wenn sie nicht verwendet wird.

# Ressourcen

## Referenzen

- [AWS Well-Architected](#)
- [AWS Well-Architected-Framework-Dokumentation](#)
- [Anwendung des AWS Well-Architected Frameworks für Amazon Neptune](#)
- [Bewährte Methoden von Neptune Analytics](#)

## Blogbeiträge und Videos

- [Neptun-Blogbeiträge \(AWS Datenbank-Blog\)](#)
- [Automatisieren Sie das Stoppen und Starten von Amazon Neptune Neptune-Umgebungsressourcen mithilfe von Ressourcen-Tags](#) (AWS Datenbank-Blog)
- [Amazon Neptune Snackables](#) (kurze Videos) YouTube

## Training

- [Erste Schritte mit Amazon Neptune](#)
- [Mit Amazon Neptune bauen](#)
- [Datenmodellierung für Amazon Neptune](#)
- [Workshop zu Amazon Neptune Analytics](#)

# 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
<a href="#">Erste Veröffentlichung</a>	—	20. Dezember 2024

# 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 den erfolgreichen Umstieg auf die Cloud unterstützt. 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 oder Bitbucket 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/CD kann 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 unter [Kontinuierliche 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.

## betreffene 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, 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 Kerngeschäftsziele 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-Berechtigungsgruppen 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 Transaktionsdaten. 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) Ilo

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 von [Modellen 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-Service management)

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-Service management](#).

## 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 Infrastrukturmgebung 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 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.

## Refaktorisierung

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.