



Bewährte Methoden für Abfragen für Amazon Redshift

# AWS Präskriptive Leitlinien



# AWS Präskriptive Leitlinien: Bewährte Methoden für Abfragen für Amazon Redshift

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

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

# Table of Contents

Einführung .....	1
Übersicht .....	1
Zielgruppe .....	1
Ziele .....	1
Architekturkomponenten .....	2
Leistungsfaktoren für Abfragen .....	7
Tabelleneigenschaften .....	7
Schlüssel sortieren .....	7
Datenkompression .....	8
Datenverteilung .....	8
Wartung von Tabellen .....	8
Cluster-Konfiguration .....	10
Knotentyp .....	10
Knotengröße, Anzahl der Knoten und Bereiche .....	10
Workload-Management .....	10
Short Query Acceleration .....	11
SQL-Abfrage .....	11
Struktur der Abfrage .....	11
Code-Kompilierung .....	12
Bewährte Methoden für Tabellen .....	13
Verstehen Sie, wie Sortierschlüssel funktionieren .....	13
Tipps zur Abfrageoptimierung .....	13
Bewerten Sie die Effektivität der Sortierschlüssel .....	14
Kenne deine Tabelle .....	15
Wählen Sie den richtigen Tabellenverteilungsstil .....	15
Bewährte Methoden für Abfragen .....	17
Vermeiden Sie die Verwendung der SELECT * FROM-Anweisung .....	17
Identifizieren Sie Probleme mit Abfragen .....	17
Rufen Sie zusammenfassende Informationen zu Ihrer Abfrage ab .....	17
Vermeiden Sie Kreuzverknüpfungen .....	18
Vermeiden Sie Funktionen in Abfrageprädikaten .....	18
Vermeiden Sie unnötige Cast-Konvertierungen .....	19
Verwenden Sie CASE-Ausdrücke für komplexe Aggregationen .....	19
Verwenden Sie Unterabfragen .....	20

Verwenden Sie Prädikate .....	20
Fügen Sie Prädikate hinzu, um Tabellen mit Verknüpfungen zu filtern .....	21
Verwenden Sie die kostengünstigsten Operatoren für Prädikate .....	21
Verwenden Sie Sortierschlüssel in GROUP BY-Klauseln .....	21
Nutzen Sie die Vorteile materialisierter Ansichten .....	22
Seien Sie vorsichtig mit Spalten in den Klauseln GROUP BY und ORDER BY .....	22
Bewährte Methoden für Redshift Spectrum .....	23
Prädikats-Pushdown in Redshift Spectrum .....	24
Tipps zur Abfrageoptimierung für Redshift Spectrum .....	25
Ressourcen .....	26
Dokumentverlauf .....	27
Glossar .....	28
# .....	28
A .....	29
B .....	32
C .....	34
D .....	37
E .....	42
F .....	44
G .....	46
H .....	47
I .....	49
L .....	51
M .....	52
O .....	57
P .....	60
Q .....	63
R .....	63
S .....	66
T .....	70
U .....	72
V .....	73
W .....	73
Z .....	74
.....	lxxvi

# Bewährte Methoden für Abfragen für Amazon Redshift

Ethan Stark, Amazon Web Services (AWS)

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

## Übersicht

Dieses Handbuch enthält Empfehlungen und bewährte Methoden zur Optimierung der Abfrage- und Tabellenleistung in [Amazon Redshift](#). Sie können Amazon Redshift verwenden, um Petabyte an strukturierten und halbstrukturierten Daten in Ihrem Data Warehouse und Ihrem Data Lake mithilfe von Standard-SQL abzufragen. Dieses Handbuch bietet auch einen Überblick über die Kernarchitekturkomponenten eines Amazon Redshift Data Warehouse. Dieses Wissen — zusammen mit einem Verständnis von Abfrageleistungsfaktoren wie Tabelleneigenschaften, Cluster-Konfiguration und Abfragestruktur — kann Ihnen helfen, effiziente und effektive Tabellen und Abfragen für Ihr Amazon Redshift Data Warehouse zu entwerfen.

## Zielgruppe

Dieses Handbuch richtet sich an Dateningenieure, Datenarchitekten und Datenanalysten, die Tabellen und Abfragen in Amazon Redshift entwerfen oder verwenden.

## Ziele

Dieser Leitfaden kann Ihnen und Ihrer Organisation dabei helfen, die folgenden Ziele zu erreichen:

- Entwerfen Sie Tabellen für optimale Datenspeicher- und Abrufvorgänge
- Entwerfen Sie Abfragen für optimale Leistung und Kosteneinsparungen
- Optimieren Sie die Leistung von [Amazon Redshift Spectrum](#), um Daten direkt aus Dateien auf [Amazon Simple Storage Service \(Amazon S3\)](#) abzufragen

# Architekturkomponenten eines Amazon Redshift Data Warehouse

Wir empfehlen, dass Sie über ein grundlegendes Verständnis der Kernarchitekturkomponenten in einem Amazon Redshift Data Warehouse verfügen. Dieses Wissen kann Ihnen helfen, besser zu verstehen, wie Sie Ihre Abfragen und Tabellen für eine optimale Leistung entwerfen können.

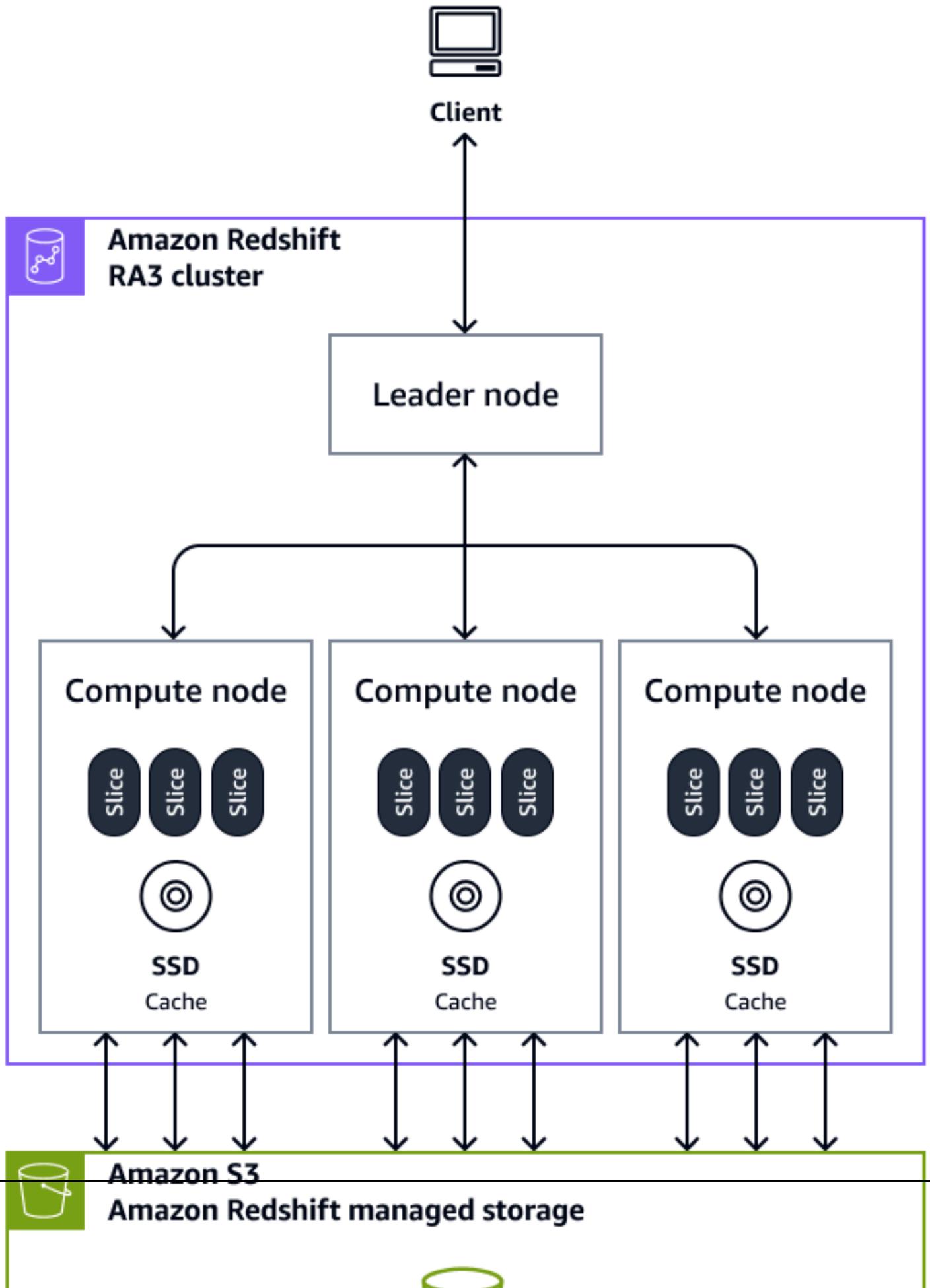
Ein Data Warehouse in Amazon Redshift besteht aus den folgenden Kernarchitekturkomponenten:

- **Cluster** — Ein Cluster, der aus einem oder mehreren Rechenknoten besteht, ist die zentrale Infrastrukturkomponente eines Amazon Redshift Data Warehouse. Rechenknoten sind für externe Anwendungen transparent, aber Ihre Client-Anwendung interagiert nur direkt mit dem Leader-Knoten. Ein typischer Cluster hat zwei oder mehr Rechenknoten. Die Rechenknoten werden über den Leader-Knoten koordiniert.
- **Leader-Node** — Ein Leader-Node verwaltet die Kommunikation für Client-Programme und alle Rechenknoten. Ein Leader-Node bereitet auch die Pläne für die Ausführung einer Abfrage vor, wann immer eine Anfrage an einen Cluster gesendet wird. Wenn die Pläne fertig sind, kompiliert der Leader-Knoten den Code, verteilt den kompilierten Code an die Rechenknoten und weist dann jedem Rechenknoten Datensegmente zu, um die Abfrageergebnisse zu verarbeiten.
- **Rechenknoten** — Ein Rechenknoten führt eine Abfrage aus. Der Leader-Knoten kompiliert Code für einzelne Elemente des Plans zur Ausführung der Abfrage und weist den Code einzelnen Rechenknoten zu. Die Datenverarbeitungsknoten führen den kompilierten Code aus und senden Zwischenergebnisse zur endgültigen Aggregation an den Führungsknoten zurück. Jeder Rechenknoten hat seine eigene dedizierte CPU, seinen eigenen Arbeitsspeicher und seinen eigenen Festplattenspeicher. Bei zunehmendem Workload können Sie die Rechenkapazität und Speicherkapazität eines Clusters steigern, indem sie die Anzahl der Knoten erhöhen, ein Upgrade des Knotentyps ausführen oder beides.
- **Node Slice** — Ein Rechenknoten ist in Einheiten unterteilt, die als Slices bezeichnet werden. Jedem Slice in einem Rechenknoten wird ein Teil des Arbeitsspeicher- und Festplattenspeichers des Knotens zugewiesen, wo er einen Teil der dem Knoten zugewiesenen Arbeitslast verarbeitet. Die Slices arbeiten dann parallel, um die Operation abzuschließen. Die Daten werden auf der Grundlage des [Verteilungsstils und des Verteilungsschlüssels](#) einer bestimmten Tabelle auf die Bereiche verteilt. Eine gleichmäßige Verteilung der Daten ermöglicht es Amazon Redshift, Workloads gleichmäßig Slices zuzuweisen und die Vorteile der Parallelverarbeitung zu maximieren. Die Anzahl der Slices pro Rechenknoten wird auf der Grundlage des Knotentyps festgelegt.

Weitere Informationen finden Sie unter [Cluster und Knoten in Amazon Redshift in der Amazon Redshift](#) Redshift-Dokumentation.

- Massive Parallelverarbeitung (MPP) — Amazon Redshift verwendet die MPP-Architektur, um Daten schnell zu verarbeiten, selbst komplexe Abfragen und riesige Datenmengen. Mehrere Rechenknoten führen denselben Abfragecode für Teile von Daten aus, um die Parallelverarbeitung zu maximieren.
- Client-Anwendung — Amazon Redshift lässt sich in verschiedene Tools zum Laden, Extrahieren, Transformieren und Laden (ETL), Business Intelligence (BI), Data Mining und Analysetools integrieren. Alle Client-Anwendungen kommunizieren nur über den Leader-Knoten mit dem Cluster.

Das folgende Diagramm zeigt, wie die Architekturkomponenten eines Amazon Redshift Data Warehouse zusammenarbeiten, um Abfragen zu beschleunigen.



Es gibt sieben Phasen des Abfragelebenszyklus:

#### 1. Empfang und Analyse von Abfragen:

- Die Abfrage wird an den Führungsknoten übermittelt und dort geparkt.
- Der Parser erzeugt einen anfänglichen Abfragebaum, der die logische Struktur der ursprünglichen Abfrage darstellt.
- Amazon Redshift speist diesen Abfragebaum in den Abfrageoptimierer ein.

#### 2. Abfrageoptimierung:

- Der Optimierer wertet die Abfrage aus und schreibt sie gegebenenfalls neu, um die Effizienz zu maximieren.
- Dieser Optimierungsprozess kann das Erstellen mehrerer verwandter Abfragen beinhalten, um eine einzelne zu ersetzen.

#### 3. Generierung des Abfrageplans:

- Der Optimierer generiert einen Abfrageplan (oder mehrere Pläne, falls erforderlich) zur Ausführung.
- Der Abfrageplan spezifiziert Ausführungsoptionen wie Join-Typen, Join-Reihenfolge, Aggregationsmethoden und Anforderungen an die Datenverteilung.

#### 4. Übersetzung der Ausführungs-Engine:

- Die Ausführungs-Engine übersetzt den Abfrageplan in einzelne Schritte, Segmente und Streams:
  - Schritt — Stellt einen einzelnen Vorgang dar, der während der Abfrageausführung erforderlich ist. Die Schritte können kombiniert werden, sodass Rechenknoten Abfragen, Verknüpfungen oder andere Datenbankoperationen ausführen können.
  - Segment — Kombiniert mehrere Schritte, die ein einzelner Prozess ausführen kann. Es ist die kleinste Kompilierungs-Einheit, die von einem Compute-Node-Slice ausgeführt werden kann. (Ein Slice ist die Einheit der Parallelverarbeitung in Amazon Redshift.)
  - Stream — Eine Sammlung von Segmenten, die auf verfügbare Rechenknotenbereiche verteilt sind.
- Die Execution Engine generiert kompilierten Code auf der Grundlage dieser Schritte, Segmente und Streams. Kompilierter Code wird schneller ausgeführt als interpretierter Code und verbraucht weniger Rechenkapazität.
- Der Leader-Knoten überträgt den kompilierten Code an die Rechenknoten.

#### 5. Parallele Ausführung:

- Dieser Schritt erfolgt einmal für jeden Stream.
  - Compute Node Slices führt Abfragesegmente parallel aus.
  - Während dieses Vorgangs optimiert Amazon Redshift die Netzwerkkommunikation, die Speichernutzung und die Festplattenverwaltung, um Zwischenergebnisse von einem Abfrageplanschritt zum nächsten weiterzuleiten.
  - Diese Optimierung trägt zu einer schnelleren Abfrageausführung bei.
6. Stream-Verarbeitung:
- Dieser Schritt erfolgt einmal für jeden Stream.
  - Die Engine erstellt für jeden Stream ausführbare Segmente für eine effiziente Parallelverarbeitung.
7. Endgültige Sortierung und Aggregation:
- Der Leader-Knoten befasst sich mit allen abschließenden Sortierungs- oder Aggregationsvorgängen, die für die Abfrage erforderlich sind.
  - Nach Abschluss des Vorgangs gibt der Leader-Knoten die Ergebnisse an den Client zurück.

Informationen zu Architekturkomponenten finden Sie unter [Data Warehouse-Systemarchitektur](#) in der Amazon Redshift Redshift-Dokumentation.

# Abfrageleistungsfaktoren für Amazon Redshift

Die Abfrageleistung kann durch eine Reihe von Faktoren beeinflusst werden. Die folgenden Aspekte Ihrer Daten-, Cluster- und Datenbankoperationen spielen alle eine Rolle dabei, wie schnell Ihre Abfragen verarbeitet werden:

- [Tabelleneigenschaften](#)
  - [Schlüssel sortieren](#)(Amazon Redshift Redshift-Berater)
  - [Datenkompression](#)(automatisiert)
  - [Datenverteilung](#)(automatisiert)
  - [Wartung von Tabellen](#)(automatisiert)
- [Cluster-Konfiguration](#)
  - [Knotentyp](#)
  - [Knotengröße, Anzahl der Knoten und Bereiche](#)
  - [Workload-Management](#)(automatisiert)
  - [Short Query Acceleration](#)(automatisiert)
- [SQL-Abfrage](#)
  - [Struktur der Abfrage](#)
  - [Code-Kompilierung](#)

## Tabelleneigenschaften

Amazon Redshift-Tabellen sind die grundlegenden Einheiten für das Speichern von Daten in Amazon Redshift, und jede Tabelle hat eine Reihe von Eigenschaften, die ihr Verhalten und ihre Zugänglichkeit bestimmen. Zu diesen Eigenschaften gehören Sortierung, Verteilungsstil, Komprimierungskodierung und viele andere. Das Verständnis dieser Eigenschaften ist entscheidend für die Optimierung der Leistung, Sicherheit und Kosteneffektivität von Amazon Redshift Redshift-Tabellen.

## Schlüssel sortieren

Amazon Redshift speichert Daten auf der Festplatte in sortierter Reihenfolge gemäß den Sortierschlüsseln einer Tabelle. Der Abfrageoptimierer und der Abfrageprozessor verwenden die Informationen darüber, wo sich die Daten innerhalb eines Rechenknotens befinden, um die Anzahl

der Blöcke zu reduzieren, die gescannt werden müssen. Dies verbessert die Abfragegeschwindigkeit erheblich, da die zu verarbeitende Datenmenge reduziert wird. Es wird empfohlen, Sortierschlüssel zu verwenden, um Filter in der WHERE Klausel zu vereinfachen. Weitere Informationen finden Sie unter [Arbeiten mit Sortierschlüsseln](#) in der Amazon Redshift Redshift-Dokumentation.

## Datenkompression

Durch die Datenkomprimierung werden die Speicheranforderungen reduziert, wodurch Festplatten-E/A reduziert und die Abfrageleistung verbessert wird. Wenn Sie eine Abfrage ausführen, werden die komprimierten Daten in den Speicher gelesen und dann dekomprimiert, wenn die Abfrage ausgeführt wird. Indem weniger Daten in den Speicher geladen werden, kann Amazon Redshift mehr Speicher für die Analyse der Daten zuweisen. Da der spaltenorientierte Speicher ähnliche Daten sequentiell speichert, kann Amazon Redshift adaptive Komprimierungskodierungen anwenden, die speziell an spaltenartige Datentypen gebunden sind. Die beste Methode, die Datenkomprimierung für Tabellenspalten zu aktivieren, besteht darin, die AUTO Option in Amazon Redshift zu verwenden, um optimale Komprimierungskodierungen anzuwenden, wenn Sie die Tabelle mit Daten laden. Weitere Informationen zur automatischen Datenkomprimierung finden Sie unter [Laden von Tabellen mit automatischer Komprimierung](#) in der Amazon Redshift Redshift-Dokumentation.

## Datenverteilung

Amazon Redshift speichert Daten auf den Rechenknoten entsprechend dem Verteilungsstil einer Tabelle. Wenn sie eine Abfrage ausführen, führt der Abfrageoptimierer nach Bedarf eine Neuverteilung der Daten auf die Datenverarbeitungsknoten durch, um Join- oder Aggregierungsoperationen durchführen zu können. Durch die Auswahl eines geeigneten Tabellenverteilungsstils können die Auswirkungen des Neuverteilungsschritts dadurch minimiert werden, dass die Daten dort platziert sind, wo sie benötigt werden, bevor die Join-Operationen ausgeführt werden. Wir empfehlen die Verwendung von Verteilungsschlüsseln, um die gängigsten Verknüpfungen zu ermöglichen. Weitere Informationen finden Sie unter [Arbeiten mit Datenverteilungsstilen](#) in der Amazon Redshift Redshift-Dokumentation.

## Wartung von Tabellen

Obwohl Amazon Redshift für die meisten Workloads sofort eine branchenführende Leistung bietet, ist für den reibungslosen Betrieb der Amazon Redshift Redshift-Cluster Wartung erforderlich. Beim Aktualisieren und Löschen von Daten entstehen tote Zeilen, die entfernt werden müssen, und selbst Tabellen, die nur Anhängen, müssen neu sortiert werden, wenn die Reihenfolge der Anfügungen nicht mit dem Sortierschlüssel übereinstimmt.

## Vacuum

Der Vakuurvorgang in Amazon Redshift ist für den Zustand und die Wartung Ihres Amazon Redshift Redshift-Clusters unerlässlich. Er wirkt sich auch auf die Leistung von Abfragen aus. Da sowohl Löschungen als auch Aktualisierungen die alten Daten kennzeichnen, sie aber nicht wirklich entfernen, müssen Sie den Speicherplatz zurückgewinnen, der durch Tabellenzeilen belegt war, die durch die vorherigen UND-Operationen zum Löschen markiert wurden. UPDATE DELETE Amazon Redshift kann Tabellen im Hintergrund automatisch sortieren und einen VACUUM DELETE Vorgang ausführen.

Um Tabellen nach einem Laden oder einer Reihe von inkrementellen Updates zu bereinigen, können Sie auch den VACUUM-Befehl ausführen (entweder gegen die gesamte Datenbank oder gegen einzelne Tabellen). Wenn Tabellen Sortierschlüssel haben und das Laden von Tabellen nicht so optimiert ist, dass sie beim Einfügen sortiert werden, müssen Sie die Daten mithilfe von Leerzeichen sortieren (was für die Leistung entscheidend sein kann). Weitere Informationen finden Sie unter [Vacuuming-Tabellen](#) in der Amazon Redshift Redshift-Dokumentation.

## Analysieren

Der ANALYZE Vorgang aktualisiert statistische Metadaten in den Tabellen in einer Amazon Redshift Redshift-Datenbank. Wenn die Statistiken aktuell sind, kann dies die Leistung bei Abfragen verbessern, weil die Abfrageplanung dann optimale Pläne auswählen kann. Amazon Redshift überwacht fortlaufend Ihre Datenbank und führt automatisch Analyseoperationen im Hintergrund durch. Um die Auswirkungen auf Ihre Systemleistung so gering wie möglich zu halten, wird der ANALYZE Vorgang in Zeiten mit geringer Auslastung automatisch ausgeführt. Wenn Sie sich für die explizite Ausführung entscheidenANALYZE, gehen Sie wie folgt vor:

- Führen Sie den ANALYZE Befehl aus, bevor Sie Abfragen ausführen.
- Führen Sie den ANALYZE Befehl routinemäßig am Ende jedes regulären Lade- oder Aktualisierungszyklus in der Datenbank aus.
- Führen Sie den ANALYZE Befehl für neue Tabellen aus, die Sie erstellen, und für bestehende Tabellen oder Spalten, die erheblich geändert wurden.
- Erwägen Sie, ANALYZE Operationen nach unterschiedlichen Zeitplänen für verschiedene Tabellen- und Spaltentypen auszuführen, je nachdem, wie sie in Abfragen verwendet werden und wie häufig sie geändert werden.
- Verwenden Sie die PREDICATE COLUMNS Klausel, wenn Sie den Befehl ausführen, um Zeit zu sparen und Ressourcen zu bündeln. ANALYZE

# Cluster-Konfiguration

Ein Cluster ist eine Sammlung von Knoten, die das eigentliche Speichern und Verarbeiten von Daten übernehmen. Die richtige Einrichtung Ihres Amazon Redshift Redshift-Clusters ist entscheidend, wenn Sie Folgendes erreichen möchten:

- Hohe Skalierbarkeit und Parallelität
- Effizienter Einsatz von Amazon Redshift
- Bessere Leistung
- Niedrigere Kosten

## Knotentyp

Ein Amazon Redshift Redshift-Cluster kann einen von mehreren Knotentypen (RA3 DC2, und DS2) verwenden. Die verschiedenen Knotentypen sind in verschiedenen Größen und mit unterschiedlichen Grenzwerten verfügbar, was eine geeignete Skalierung Ihres Clusters ermöglicht. Die Knotengröße bestimmt die Speicherkapazität, den Arbeitsspeicher, die CPU und den Preis für die einzelnen Knoten in dem Cluster. Die Kosten- und Leistungsoptimierung beginnt mit der Auswahl des richtigen Knotentyps und der richtigen Größe. Weitere Informationen zu Knotentypen finden Sie unter [Überblick über Amazon Redshift Redshift-Cluster in der Amazon Redshift](#) Redshift-Dokumentation.

## Knotengröße, Anzahl der Knoten und Bereiche

Ein Datenverarbeitungsknoten ist in Slices aufgeteilt. Mehr Knoten bedeuten mehr Prozessoren und Bereiche, wodurch Ihre Abfragen schneller verarbeitet werden können, indem Teile der Abfrage gleichzeitig über die Bereiche hinweg ausgeführt werden. Mehr Knoten bedeuten jedoch auch höhere Kosten. Das bedeutet, dass Sie das richtige Verhältnis zwischen Kosten und Leistung für Ihr System finden müssen. Weitere Informationen zur Amazon Redshift Redshift-Clusterarchitektur finden Sie unter [Data Warehouse-Systemarchitektur](#) in der Amazon Redshift Redshift-Dokumentation.

## Workload-Management

Amazon Redshift Workload Management (WLM) ermöglicht es Benutzern, Workload-Warteschlangen flexibel mit Prioritäten zu verwalten, sodass kurze, schnell laufende Abfragen nicht in Warteschlangen hinter lang andauernden Abfragen stecken bleiben. Automatic WLM verwendet Algorithmen für maschinelles Lernen (ML), um Abfragen zu profilieren und sie mit den entsprechenden Ressourcen

in die entsprechende Warteschlange zu stellen und gleichzeitig die Parallelität der Abfragen und die Speicherzuweisung zu verwalten. Weitere Informationen zu WLM finden Sie unter [Implementieren des Workload-Managements](#) in der Amazon Redshift Redshift-Dokumentation.

## Short Query Acceleration

Short Query Acceleration (SQA) priorisiert Abfragen mit kurzer Laufzeit vor Abfragen mit langer Laufzeit. SQA führt Abfragen in einem dafür vorgesehenen Bereich aus, sodass SQA-Abfragen nicht gezwungen sind, hinter längeren Abfragen in Warteschlangen zu warten. SQA priorisiert nur Warteschlangen, die über eine kurze Zeit ausgeführt werden und sich in einer benutzerdefinierten Warteschlange befinden. Wenn Sie SQA verwenden, werden Abfragen mit kurzer Laufzeit schneller ausgeführt, und Sie können schneller Ergebnisse sehen. Wenn Sie SQA aktivieren, können Sie WLM-Warteschlangen, die für Abfragen mit kurzer Laufzeit vorgesehen sind, reduzieren oder eliminieren. Darüber hinaus müssen Abfragen mit langer Laufzeit nicht um Slots in einer WLM-Warteschlange kämpfen. Das bedeutet, dass Sie Ihre WLM-Warteschlangen so konfigurieren können, dass weniger Abfrageslots verwendet werden. Wenn Sie weniger Parallelität verwenden, wird der Abfragedurchsatz erhöht und die Gesamtsystemleistung für die meisten Workloads verbessert. Weitere Informationen zu SQA finden Sie unter [Arbeiten mit Short Query Acceleration](#) in der Amazon Redshift Redshift-Dokumentation.

## SQL-Abfrage

Eine Datenbankabfrage ist eine Anforderung von Daten aus einer Datenbank. Die Anfrage sollte in einem Amazon Redshift Redshift-Cluster mit SQL eingehen. Amazon Redshift unterstützt SQL-Client-Tools, die über Java Database Connectivity (JDBC) und Open Database Connectivity (ODBC) eine Verbindung herstellen. Sie können die meisten SQL-Client-Tools verwenden, die JDBC- oder ODBC-Treiber unterstützen.

## Struktur der Abfrage

Die Art und Weise, wie Ihre Abfrage geschrieben ist, wirkt sich stark auf ihre Leistung aus. Wir empfehlen Ihnen, Abfragen zu schreiben, um so wenig Daten wie nötig zu verarbeiten und zurückzugeben, um Ihre Anforderungen zu erfüllen. Weitere Informationen zur Strukturierung Ihrer Abfragen finden Sie im Abschnitt [Bewährte Methoden zum Entwerfen von Amazon Redshift Redshift-Abfragen](#) in diesem Handbuch.

## Code-Kompilierung

Amazon Redshift generiert und kompiliert Code für jeden Abfrageausführungsplan. Der kompilierte Code wird schneller ausgeführt, weil der Verwaltungsaufwand für den Interpreter entfällt. In der Regel fallen bei der ersten Generierung und Kompilierung des Codes einige Gemeinkosten an. Aus diesem Grund kann die Abfrageleistung bei der ersten Ausführung signifikant von nachfolgenden Ausführungen abweichen. Die Gemeinkosten können sich besonders bemerkbar machen, wenn Sie einmalige Abfragen ausführen. Es wird empfohlen, die Abfrage ein zweites Mal auszuführen, um ihre typische Leistung zu ermitteln.

Amazon Redshift verwendet einen Serverless-Kompilierungsservice, um Abfragekompilierungen über die Rechenressourcen eines Amazon-Redshift-Clusters hinaus zu skalieren. Die kompilierten Codesegmente werden lokal auf dem Cluster und in einem praktisch unbegrenzten Cache gespeichert. Dieser Cache bleibt nach dem Neustart des Clusters bestehen. Nachfolgende Aufrufe derselben Abfrage werden schneller ausgeführt, da sie die Kompilierungsphase überspringen können. Der Cache ist nicht versionsübergreifend mit Amazon Redshift kompatibel, daher wird der Code neu kompiliert, wenn Abfragen nach einem Versions-Upgrade ausgeführt werden. Durch die Verwendung eines skalierbaren Kompilierungsservice ist Amazon Redshift in der Lage, Code parallel zu kompilieren, um eine konstant schnelle Leistung zu bieten. Die Ausdehnung der Workload hängt von der Komplexität und Nebenläufigkeit von Abfragen ab.

# Bewährte Methoden für das Entwerfen von Amazon Redshift Redshift-Tabellen

Dieser Abschnitt bietet einen Überblick über bewährte Methoden für das Entwerfen von Datenbanktabellen. Wir empfehlen Ihnen, diese bewährten Methoden zu befolgen, um eine optimale Abfrageleistung und Effizienz zu erzielen.

## Verstehen Sie, wie Sortierschlüssel funktionieren

Amazon Redshift speichert die Daten auf der Festplatte in sortierter Form gemäß dem Sortierschlüssel. Der Amazon-Redshift-Abfrageoptimierer verwendet die Sortierfolge bei der Bestimmung optimaler Abfragepläne. Um Sortierschlüssel effektiv zu verwenden, empfehlen wir Ihnen, wie folgt vorzugehen:

- Sortieren Sie die Tabelle so weit wie möglich.
- Verwenden Sie `VACUUM sort`, um die optimale Leistung wiederherzustellen.
- Vermeiden Sie es, die Spalte mit den Sortierschlüsseln zu komprimieren.
- Wenn der Sortierschlüssel komprimiert ist und das `sortkey1_skew` Verhältnis sehr hoch ist, erstellen Sie die Tabelle neu, ohne die Komprimierung des Sortierschlüssels zu aktivieren.
- Vermeiden Sie es, eine Funktion auf die Sortierschlüsselspalten anzuwenden. In der folgenden Abfrage wird die `trans_dt` : `TIMESTAMPTZ` Sortierschlüsselspalte beispielsweise nicht verwendet, wenn Sie sie wie folgt umwandeln `DATE`:

```
select order_id, order_amt
from sales
where trans_dt::date = '2021-01-08'::date
```

- Führen Sie `INSERT` Operationen in der Reihenfolge der Sortierschlüssel aus.
- Verwenden Sie nach Möglichkeit Sortierschlüssel in der `GROUP BY` Klausel.

## Tipps zur Abfrageoptimierung

Wir empfehlen Ihnen, Ihre Abfragen wie folgt zu optimieren:

- Ordnen Sie zusammengesetzte Sortierschlüssel immer von der niedrigsten Kardinalität zur höchsten Kardinalität an, um eine optimale Effektivität zu erzielen.

- Wenn der führende Schlüssel in einem zusammengesetzten Sortierschlüssel relativ eindeutig ist (d. h. er hat eine hohe Kardinalität), vermeiden Sie es, Ihrem Sortierschlüssel zusätzliche Spalten hinzuzufügen. Das Hinzufügen zusätzlicher Spalten hat nur geringe Auswirkungen auf die Abfrageleistung, erhöht jedoch die Wartungskosten.

## Bewerten Sie die Effektivität der Sortierschlüssel

Um Ihre Abfragen zu optimieren, müssen Sie in der Lage sein, die Effektivität Ihrer Abfragen zu bewerten. Es wird empfohlen, die Ansicht [SVL\\_QUERY\\_SUMMARY](#) zu verwenden, um allgemeine Informationen über die Ausführung einer Abfrage zu finden. In dieser Ansicht können Sie das Attribut `IS_RRSCAN` verwenden, um festzustellen, ob ein EXPLAIN Planschritt einen bereichsbeschränkten Scan verwendet. Sie können das Attribut auch verwenden `rows_pre_filter`, um die Selektivität eines Sortierschlüssels zu bestimmen.

Sie können auch eine Admin-Ansicht mit dem GitHub Namen [v\\_my\\_last\\_query\\_summary](#) verwenden. In der Ansicht werden Informationen für die letzte Abfrage angezeigt, die ausgeführt wurde.

Die folgende Anweisung zeigt, wie Sie allgemeine Informationen zur Ausführung einer Abfrage finden.

```
select lpad(' ',stm+seg+step) || label as label,
       rows,
       bytes,
       is_diskbased,
       is_rrscan,
       rows_pre_filter
from svl_query_summary
where query = pg_last_query_id()
order by stm, seg, step;
```

Die vorherige Abfrage gibt die folgende Beispielausgabe zurück.

label	<input type="checkbox"/> rows	bytes	is_diskbased	is_rrscan	rows_pre_filter
scan tbl=163860 name=orders	<input type="checkbox"/> 1500000	24000000	f	f	1500000
project	<input type="checkbox"/> 1500000	0	f	f	0
project	<input type="checkbox"/> 1500000	0	f	f	0
hash tbl=968	<input type="checkbox"/> 1500000	24000000	f	f	0
scan tbl=163852 name=lineitem	<input type="checkbox"/> 6001215	144029160	f	t	6001215
project	<input type="checkbox"/> 6001215	0	f	f	0
project	<input type="checkbox"/> 6001215	0	f	f	0
hjoin tbl=968	<input type="checkbox"/> 6001215	0	f	f	0
project	<input type="checkbox"/> 6001215	0	f	f	0
project	<input type="checkbox"/> 6001215	0	f	f	0

## Kenne deine Tabelle

Es ist wichtig, die kritischen Eigenschaften Ihrer Tabelle zu verstehen. Gehen Sie wie folgt vor, um mehr über Ihre Tabelle zu erfahren:

- Verwenden Sie [PG\\_TABLE\\_DEF, um Informationen zu Tabellenspalten](#) anzuzeigen.
- Verwenden Sie [SVV\\_TABLE\\_INFO](#), um umfassendere Informationen über eine Tabelle anzuzeigen, einschließlich Datenverteilungsverzerrung, Schräglage der Schlüsselverteilung, Tabellengröße und Statistiken.

## Wählen Sie den richtigen Tabellenverteilungsstil

Wenn sie eine Abfrage ausführen, führt der Abfrageoptimierer nach Bedarf eine Neuverteilung der Zeilen zu den Datenverarbeitungsknoten durch, um Join- oder Aggregierungsoperationen durchführen zu können. Das Ziel bei der Auswahl eines Tabellenverteilungsstils besteht darin, die Auswirkungen des Umverteilungsschritts zu minimieren, indem die Daten dort platziert werden, wo sie benötigt werden, bevor Sie die Abfrage ausführen.

Wir empfehlen den folgenden Ansatz zur Auswahl des richtigen Tabellenverteilungsstils:

- Vermeiden Sie Broadcasting und Redistribution in einem Abfrageausführungsplan, indem Sie die Zeilen innerhalb desselben Knotens zusammenlegen. Wenn Sie beispielsweise eine DISTKEY auswählen, können Sie die Faktentabelle und die eindimensionale Tabelle auf ihre gemeinsamen Spalten verteilen. Wählen Sie die größte Dimension auf der Grundlage der Größe des gefilterten Datasets. Nur die in dem Join verwendeten Zeilen müssen verteilt werden. Berücksichtigen Sie daher die Größe des Datensatzes nach der Filterung und nicht die Größe der Tabelle.

- Stellen Sie sicher, dass die Spalte, in der der Verteilungsschlüssel erstellt wird, nicht schief ist. Andernfalls könnte ein Rechenknoten mehr Arbeit leisten als andere. Wenn Sie eine Schiefe feststellen, sollten Sie in Erwägung ziehen, die Spalte mit dem Verteilungsschlüssel zu ändern. Eine Spalte kann als Kandidat für einen Verteilungsschlüssel in Betracht gezogen werden, wenn es sich bei ihren Werten um gleichmäßig verteilte Werte oder um hohe Kardinalwerte handelt.
- Wenn die in der Join-Bedingung verwendete Tabelle klein ist (weniger als 1 GB), sollten Sie den Verteilungsstil ALL berücksichtigen.
- Sie können den Verteilungsschlüssel komprimieren, müssen jedoch die Komprimierung der Sortierschlüsselspalte (insbesondere der ersten Spalte des Sortierschlüssels) vermeiden.

#### Note

Wenn Sie die automatische Tabellenoptimierung verwenden, müssen Sie den Verteilungsstil Ihrer Tabelle nicht wählen. Weitere Informationen finden Sie unter [Arbeiten mit automatischer Tabellenoptimierung](#) in der Amazon Redshift Redshift-Dokumentation. Wenn Amazon Redshift den geeigneten Verteilungsstil auswählt, geben Sie AUTO für den Verteilungsstil an.

# Bewährte Methoden für das Entwerfen von Amazon Redshift Redshift-Abfragen

Dieser Abschnitt bietet einen Überblick über bewährte Methoden für das Entwerfen von Abfragen. Wir empfehlen Ihnen, die bewährten Methoden in diesem Abschnitt zu befolgen, um eine optimale Abfrageleistung und Effizienz zu erzielen.

## Vermeiden Sie die Verwendung der SELECT \* FROM-Anweisung

Wir empfehlen, die SELECT \* FROM Anweisung nicht zu verwenden. Listen Sie stattdessen immer Spalten für die Analyse auf. Dies reduziert die Abfrageausführungszeit und die Scankosten für Amazon Redshift Spectrum Spectrum-Abfragen.

Beispiel dafür, was vermieden werden sollte

```
select *  
from sales;
```

Beispiel für bewährte Verfahren

```
select sales_date, sales_amt  
from sales;
```

## Identifizieren Sie Probleme mit Abfragen

Wir empfehlen Ihnen, die Ansicht [STL\\_ALERT\\_EVENT\\_LOG](#) zu überprüfen, um mögliche Probleme mit Ihrer Abfrage zu identifizieren und zu beheben.

## Rufen Sie zusammenfassende Informationen zu Ihrer Abfrage ab

Wir empfehlen, die Ansichten SVL\_QUERY\_SUMMARY und [SVL\\_QUERY\\_REPORT](#) zu verwenden, um zusammenfassende Informationen zu Ihren Abfragen zu erhalten. Sie können diese Informationen verwenden, um Ihre Abfragen zu optimieren.

## Vermeiden Sie Kreuzverknüpfungen

Wir empfehlen, Cross-Joins zu vermeiden, sofern dies nicht unbedingt erforderlich ist. Ohne eine Verbindungsbedingung ergeben Kreuzverknüpfungen das kartesische Produkt zweier Tabellen. Kreuzverknüpfungen werden in der Regel als Nested-Loop-Joins ausgeführt (der langsamste aller möglichen Join-Typen).

Beispiel dafür, was vermieden werden sollte

```
select c.c_name,  
       n.n_name  
from tpch.customer c,  
     tpch.nation n;
```

Beispiel für bewährte Verfahren

```
select c.c_name,  
       n.n_name  
from tpch.customer c,  
join tpch.nation n  
  on n.n_nationkey = c.c_nationkey;
```

## Vermeiden Sie Funktionen in Abfrageprädikaten

Es wird empfohlen, die Verwendung von Funktionen in Abfrageprädikaten zu vermeiden. Die Verwendung von Funktionen in Abfrageprädikaten kann sich negativ auf die Leistung auswirken, da Funktionen in der Regel für jede Zeile zusätzlichen Verarbeitungsaufwand bedeuten und die Gesamtausführung der Abfrage verlangsamen.

Beispiel dafür, was vermieden werden sollte

```
select sum(o_totalprice)  
from tpch.orders  
where datepart(year, o_orderdate) = 1992;
```

Beispiel für bewährte Verfahren

```
select sum(o_totalprice)  
from tpch.orders
```

```
where o_orderdate between '1992-01-01' and '1992-12-31';
```

## Vermeiden Sie unnötige Cast-Konvertierungen

Es wird empfohlen, unnötige Umwandlungskonvertierungen für die Abfragen zu vermeiden, da das Umwandeln von Datentypen Zeit und Ressourcen beansprucht und die Abfrageausführung verlangsamt.

Beispiel dafür, was vermieden werden sollte

```
select sum(o_totalprice)
from tpch.orders
where o_ordertime::date = '1992-01-01';
```

Beispiel für bewährte Verfahren

```
select sum(o_totalprice)
from tpch.orders
where o_ordertime between '1992-01-01 00:00:00' and '1992-12-31 23:59:59';
```

## Verwenden Sie CASE-Ausdrücke für komplexe Aggregationen

Es wird empfohlen, einen [CASE-Ausdruck](#) zu verwenden, um komplexe Aggregationen durchzuführen, anstatt mehrmals aus derselben Tabelle auszuwählen.

Beispiel dafür, was vermieden werden sollte

```
select sum(sales_amt) as us_sales
from sales
where country = 'US';

select sum(sales_amt) as ca_sales
from sales
where country = 'CA';
```

Beispiel für bewährte Verfahren

```
select sum(case when country = 'US' then sales_amt end) as us_sales,
```

```
sum(case when country = 'CA' then sales_amt end) as ca_sales
from sales;
```

## Verwenden Sie Unterabfragen

Es wird empfohlen, Unterabfragen in Fällen zu verwenden, in denen eine Tabelle in der Abfrage nur für Prädikatbedingungen verwendet wird und die Unterabfrage eine geringe Anzahl von Zeilen zurückgibt (weniger als etwa 200).

Beispiel dafür, was vermieden werden sollte

Wenn eine Unterabfrage weniger als 200 Zeilen zurückgibt:

```
select sum(order_amt) as total_sales
from sales
where region_key IN
      (select region_key
       from regions
       where state = 'CA');
```

Beispiel für ein bewährtes Verfahren

Wenn eine Unterabfrage mehr als oder gleich 200 Zeilen zurückgibt:

```
select sum(o.order_amt) as total_sales
from sales o
join regions r
  on r.region_key = o.region_key
  and r.state = 'CA';
```

## Verwenden Sie Prädikate

Wir empfehlen, Prädikate zu verwenden, um den Datensatz so weit wie möglich einzuschränken. Prädikate werden in SQL verwendet, um die in einer Abfrage zurückgegebenen Daten zu filtern und einzuschränken. Durch die Angabe von Bedingungen in einem Prädikat können Sie auf der Grundlage bestimmter Bedingungen angeben, welche Zeilen in den Abfrageergebnissen enthalten sein müssen. Auf diese Weise können Sie nur die Daten abrufen, an denen Sie interessiert sind, und die Effizienz und Genauigkeit Ihrer Abfragen wird verbessert. Weitere Informationen finden Sie unter [Bedingungen](#) in der Amazon Redshift Redshift-Dokumentation.

## Fügen Sie Prädikate hinzu, um Tabellen mit Verknüpfungen zu filtern

Es wird empfohlen, Prädikate hinzuzufügen, um Tabellen zu filtern, die an Verknüpfungen beteiligt sind, auch wenn die Prädikate dieselben Filter anwenden. Die Verwendung von Prädikaten zum Filtern von Tabellen mit Verknüpfungen in SQL kann die Abfrageleistung verbessern, indem die Menge der zu verarbeitenden Daten und die Größe der Zwischenergebnismenge reduziert werden. Durch die Angabe der Bedingungen für den Join-Vorgang in der WHERE Klausel kann das Modul zur Abfrageausführung Zeilen entfernen, die die Bedingungen nicht erfüllen, bevor sie verknüpft werden. Dies führt zu einer kleineren Ergebnismenge und einer schnelleren Abfrageausführung.

Beispiel dafür, was vermieden werden sollte

```
select p.product_name, sum(o.order_amt)
from sales o
join product p
  on r.product_key = o.product_key
where o.order_date > '2022-01-01';
```

Beispiel für bewährte Verfahren

```
select p.product_name, sum(o.order_amt)
from sales o
join product p
  on p.product_key = o.product_key
  and p.added_date > '2022-01-01'
where o.order_date > '2022-01-01';
```

## Verwenden Sie die kostengünstigsten Operatoren für Prädikate

Verwenden Sie im Prädikat die kostengünstigsten Operatoren, die Ihnen zur Verfügung stehen. Operatoren für [Vergleichsbedingungen](#) sind LIKE-Operatoren [vorzuziehen](#). LIKE-Operatoren sind immer noch [SIMILAR TO](#) - oder [POSIX-Operatoren](#) vorzuziehen.

## Verwenden Sie Sortierschlüssel in GROUP BY-Klauseln

Verwenden Sie Sortierschlüssel in der GROUP BY Klausel, damit der Abfrageplaner eine effizientere Aggregation verwenden kann. Eine Abfrage kann für eine einphasige Aggregation in Frage

kommen, wenn ihre GROUP BY Liste nur Sortierschlüsselspalten enthält, von denen eine auch der Verteilungsschlüssel ist. Die Sortierschlüsselspalten in der GROUP BY Liste müssen den ersten Sortierschlüssel enthalten, gefolgt von anderen Sortierschlüsseln, die Sie in der Reihenfolge der Sortierschlüssel verwenden möchten.

## Nutzen Sie die Vorteile materialisierter Ansichten

Wenn möglich, schreiben Sie die Abfrage neu, indem Sie den komplexen Code durch eine materialisierte Ansicht ersetzen, wodurch die Leistung der Abfrage erheblich verbessert wird. Weitere Informationen finden Sie unter [Erstellen materialisierter Ansichten in Amazon Redshift in der Amazon Redshift Redshift-Dokumentation](#).

## Seien Sie vorsichtig mit Spalten in den Klauseln GROUP BY und ORDER BY

Wenn Sie sowohl GROUP BY - als auch ORDER BY Klauseln verwenden, stellen Sie sicher, dass Sie die Spalten in den Klauseln GROUP BY sowohl ORDER BY als auch in derselben Reihenfolge anordnen. GROUP BY erfordert implizit, dass die Daten sortiert werden. Wenn Ihre ORDER BY Klausel anders ist, müssen die Daten zweimal sortiert werden.

Beispiel dafür, was zu vermeiden ist

```
select a, b, c, sum(d)
from a_table
group by b, c, a
order by a, b, c
```

Beispiel für bewährte Verfahren

```
select a, b, c, sum(d)
from a_table
group by a, b, c
order by a, b, c
```

# Bewährte Methoden für die Verwendung von Amazon Redshift Spectrum

Dieser Abschnitt bietet einen Überblick über bewährte Methoden für die Verwendung von [Amazon Redshift Spectrum](#). Wir empfehlen Ihnen, die folgenden bewährten Methoden zu befolgen, um eine optimale Leistung bei der Verwendung von Redshift Spectrum zu erzielen:

- Bedenken Sie, dass Dateitypen einen erheblichen Einfluss auf die Redshift Spectrum-Abfrageleistung haben. Verwenden Sie zur Verbesserung der Leistung spaltencodierte Dateien wie ORC oder Parquet und das CSV-Format nur für Tabellen mit sehr kleinen Dimensionen.
- Verwenden Sie präfixbasierte Partitionierung, um die Vorteile der Partitionsbereinigung zu nutzen. Das bedeutet, Filter zu verwenden, die den Partitionen in Ihrem Data Lake zugeordnet sind.
- Redshift Spectrum skaliert automatisch, um große Anfragen zu verarbeiten, also tun Sie so viel wie möglich in Redshift Spectrum (z. B. Predicate Pushdown).
- Achten Sie darauf, Dateien in häufig gefilterten Spalten zu partitionieren. Wenn Daten nach einer oder mehreren gefilterten Spalten partitioniert sind, kann Redshift Spectrum die Vorteile der Partitionsbereinigung nutzen und das Scannen nicht benötigter Partitionen und Dateien überspringen. Häufig werden die Daten nach der Zeit partitioniert.
- Sie können die Effektivität Ihrer Partitionen und die Effizienz Ihrer Redshift Spectrum-Abfrage überprüfen, indem Sie die folgende Abfrage verwenden.

```
Select query,  
       segment,  
       max(assigned_partitions) as total_partitions,  
       max(qualified_partitions) as qualified_partitions  
From svl_s3partition  
Where query=pg_last_query_id()  
Group by 1,2;
```

Die vorherige Abfrage zeigt Folgendes:

- `total_partitions` — Die Anzahl der Partitionen, die von der erkannt wurden AWS Glue Data Catalog
- `qualified_partitions` — Die Anzahl der Präfixe auf Amazon Simple Storage Service (Amazon S3), auf die für die Redshift Spectrum-Abfrage zugegriffen wird

- Sie können auch die SVL\_S3QUERY\_SUMMARY Systemtabelle überprüfen, um mehr über die Effektivität Ihrer Partitionen und die Effizienz Ihrer Redshift Spectrum-Abfrage zu erfahren. Verwenden Sie dazu die folgende Anweisung.

```
Select *  
From svl_s3query_summary  
Where query=pg_last_query_id();
```

Die vorherige Abfrage gibt noch mehr Informationen zurück, einschließlich `is_partitioned`, und `s3_returned_rows/bytes` Wertes `s3_scanned_rows/bytes`, zusätzlich zu Dateien, die die Effizienz der Partitionsbereinigung belegen.

## Prädikats-Pushdown in Redshift Spectrum

Durch die Verwendung von Predicate Pushdown wird der Ressourcenverbrauch im Amazon Redshift Redshift-Cluster vermieden. Sie können viele SQL-Operationen auf die Redshift Spectrum-Ebene übertragen. Wir empfehlen, dies wo immer möglich zu nutzen.

Beachten Sie Folgendes:

- Sie können einige Arten von SQL-Operationen vollständig innerhalb der Redshift Spectrum-Ebene auswerten, darunter die folgenden:
  - GROUP BY Klauseln
  - Bedingungen für Vergleich und Musterabgleich (zum Beispiel) LIKE
  - Aggregatfunktionen (z. B., COUNT, SUM, AVG, MIN, und) MAX
  - `regex_replace`, `to_upperdate_trunc`, und andere Funktionen
- Sie können einige Operationen nicht auf die Redshift Spectrum-Ebene übertragen, einschließlich DISTINCT und ORDER BY. Führen Sie möglichst ORDER BY nur auf der obersten Ebene der Abfrage aus, da die Sortierung im Leader-Knoten erfolgt.
- Prüfen Sie Ihren EXPLAIN Abfrageplan, um zu überprüfen, ob das Prädikat Pushdown wirksam ist. Gehen Sie wie folgt vor, um Redshift Spectrum-Teile in einem EXPLAIN Befehl zu finden:
  - S3 Seq Scan
  - S3 HashAggregate
  - S3 Query Scan
  - Seq Scan PartitionInfo

- **Partition Loop**
- Verwenden Sie die geringste Anzahl von Spalten in Ihrer Abfrage. Redshift Spectrum kann Spalten für das Scannen eliminieren, wenn Daten im Parquet- oder ORC-Format vorliegen.
- Verwenden Sie Partitionen ausgiebig für die Parallelverarbeitung und die Eliminierung von Partitionen und halten Sie die Dateigrößen nach Möglichkeit auf mindestens 64 MB.
- Stellen Sie das ein `TABLE PROPERTIES ' numRows ' = ' nnn '`, wenn Sie `CREATE EXTERNAL TABLE` oder verwenden `ALTER TABLE`. Amazon Redshift analysiert keine externen Tabellen, um Tabellenstatistiken zu generieren, die der Abfrageoptimierer zur Generierung eines Abfrageplans verwendet. Wenn die Statistiken nicht festgelegt sind, geht Amazon Redshift davon aus, dass externe Tabellen die größeren Tabellen und lokale Tabellen die kleineren Tabellen sind.

## Tipps zur Abfrageoptimierung für Redshift Spectrum

Wir empfehlen Ihnen, bei der Optimierung Ihrer Abfragen Folgendes zu beachten:

- Die Anzahl der Redshift Spectrum-Knoten, die Ihr Amazon Redshift Redshift-Cluster für eine Abfrage verwenden kann, ist an die Anzahl der Slices in Ihrem Cluster gebunden.
- Die Größe Ihres Clusters nach oben kann den lokalen Rechenprofilen, Speicherprofilen und den Abfragefunktionen der Amazon S3 S3-Data-Lake-Abfrage zugute kommen.
- Der Amazon-Redshift-Abfrageplaner pusht Prädikate und Aggregationen nach Möglichkeit auf die Redshift-Spectrum-Abfrageebene.
- Wenn große Mengen von Daten von Amazon S3 zurückgegeben werden, ist die Verarbeitung durch die Ressourcen Ihres Clusters begrenzt.
- Da Redshift Spectrum automatisch skaliert, um große Anfragen zu verarbeiten, verbessert sich Ihre Gesamtleistung, wenn Sie die Verarbeitung auf die Redshift Spectrum-Ebene übertragen können.

# Ressourcen

- [Bewährte Methoden für Amazon Redshift](#) (Amazon Redshift Redshift-Dokumentation)
- [Bewährte Methoden für Amazon Redshift zum Entwerfen von Abfragen](#) (Amazon Redshift Redshift-Dokumentation)
- [Optimieren der Abfrageleistung](#) (Amazon Redshift Redshift-Dokumentation)
- [Abfrageplan](#) (Amazon Redshift Redshift-Dokumentation)
- [Verbesserung der Abfrageleistung von Amazon Redshift Spectrum](#) (Amazon Redshift Redshift-Dokumentation)
- [Den Abfragelebenszyklus in Amazon Redshift verstehen](#) (AWS Prescriptive Guidance)

# 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="#">AQUA wurde entfernt</a>	Wir haben Informationen über Advanced Query Accelerator (AQUA) entfernt.	14. Juni 2024
<a href="#">Erste Veröffentlichung</a>	—	3. Februar 2023

# AWS Glossar zu präskriptiven Leitlinien

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

## Zahlen

### 7 Rs

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

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

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

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

## A

### ABAC

Siehe [attributbasierte](#) Zugriffskontrolle.

### abstrahierte Dienste

Weitere Informationen finden Sie unter [Managed Services](#).

### ACID

Siehe [Atomarität, Konsistenz, Isolierung und Haltbarkeit](#).

### Aktiv-Aktiv-Migration

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

### Aktiv-Passiv-Migration

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

### Aggregatfunktion

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

## AI

Siehe [künstliche Intelligenz](#).

## AIOps

Siehe [Operationen im Bereich künstliche Intelligenz](#).

## Anonymisierung

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

## Anti-Muster

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

## Anwendungssteuerung

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

## Anwendungsportfolio

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

## künstliche Intelligenz (KI)

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

## Operationen mit künstlicher Intelligenz (AIOps)

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

## Asymmetrische Verschlüsselung

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

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

### Atomizität, Konsistenz, Isolierung, Haltbarkeit (ACID)

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

### Attributbasierte Zugriffskontrolle (ABAC)

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

### autoritative Datenquelle

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

### Availability Zone

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

### AWS Framework für die Cloud-Einführung (AWS CAF)

Ein Framework mit Richtlinien und bewährten Verfahren, das Unternehmen bei der Entwicklung eines effizienten und effektiven Plans für die erfolgreiche Umstellung auf die Cloud unterstützt. AWS 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 stressen, 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.

### betroffene Person

Eine Person, deren Daten gesammelt und verarbeitet werden.

## Data Warehouse

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

### Datenbankdefinitionssprache (DDL)

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

### Datenbankmanipulationssprache (DML)

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

### DDL

Siehe [Datenbankdefinitionssprache](#).

### Deep-Ensemble

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

### Deep Learning

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

### defense-in-depth

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

### delegierter Administrator

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

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

## Bereitstellung

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

## Entwicklungsumgebung

Siehe [Umgebung](#).

## Detektivische Kontrolle

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

## Abbildung des Wertstroms in der Entwicklung (DVSM)

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

## digitaler Zwilling

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

## Maßtabelle

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

## Katastrophe

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

### Disaster Recovery (DR)

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

### DML

Siehe Sprache zur [Datenbankmanipulation](#).

### Domainorientiertes Design

Ein Ansatz zur Entwicklung eines komplexen Softwaresystems, bei dem seine Komponenten mit sich entwickelnden Domains oder 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 Endpunkt-Service verbinden, indem sie Schnittstellen-VPC-Endpunkte erstellen. Weitere Informationen finden Sie unter [Einen Endpunkt-Service erstellen](#) in der Amazon Virtual Private Cloud (Amazon VPC)-Dokumentation.

## Unternehmensressourcenplanung (ERP)

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

## Envelope-Verschlüsselung

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

## Umgebung

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

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

## Epics

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

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

## ERP

Siehe [Enterprise Resource Planning](#).

## Explorative Datenanalyse (EDA)

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

## F

### Faktentabelle

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

### schnell scheitern

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

### Grenze zur Fehlerisolierung

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

### Feature-Zweig

Siehe [Zweig](#).

### Features

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

## Bedeutung der Feature

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

## Featuretransformation

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

## Eingabeaufforderung mit wenigen Klicks

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

## FGAC

Siehe [detaillierte Zugriffskontrolle](#).

## Feinkörnige Zugriffskontrolle (FGAC)

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

## Flash-Cut-Migration

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

## FM

Siehe [Fundamentmodell](#).

## Fundamentmodell (FM)

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

## G

### generative KI

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

### Geoblocking

Siehe [geografische Einschränkungen](#).

### Geografische Einschränkungen (Geoblocking)

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

### Gitflow-Workflow

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

### goldenes Bild

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

## Greenfield-Strategie

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

## Integritätsschutz

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

# H

## HEKTAR

Siehe [Hochverfügbarkeit](#).

## Heterogene Datenbankmigration

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

## hohe Verfügbarkeit (HA)

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

## historische Modernisierung

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

## Holdout-Daten

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

## Homogene Datenbankmigration

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

## heiße Daten

Daten, auf die häufig zugegriffen wird, z. B. Echtzeitdaten oder aktuelle 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.

|

## 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 des [Modells für maschinelles Lernen](#) mit AWS

## IoT

Siehe [Internet der Dinge](#).

## IT information library (ITIL, IT-Informationsbibliothek)

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

## T service management (ITSM, IT-Servicemanagement)

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

## BIS

Weitere Informationen finden Sie in der [IT-Informationsbibliothek](#).

## ITSM

Siehe [IT-Servicemanagement](#).

## L

### Labelbasierte Zugangskontrolle (LBAC)

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

### Landing Zone

Eine landing zone ist eine gut strukturierte AWS Umgebung mit mehreren Konten, die skalierbar und sicher ist. Dies ist ein Ausgangspunkt, von dem aus Ihre Organisationen Workloads und Anwendungen schnell und mit Vertrauen in ihre Sicherheits- und 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 in sind. AWS Organizations Ein Konto kann jeweils nur einer Organisation angehören.

## DURCHEINANDER

Siehe [Manufacturing Execution System](#).

## Message Queuing-Telemetrietransport (MQTT)

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

## Microservice

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

## Microservices-Architekturen

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

## Migration Acceleration Program (MAP)

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

## Migration in großem Maßstab

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

## Migrationsfabrik

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

## Migrationsmetadaten

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

## Migrationsmuster

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

## Migration Portfolio Assessment (MPA)

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

## Migration Readiness Assessment (MRA)

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

## Migrationsstrategie

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

## ML

Siehe [maschinelles Lernen](#).

## Modernisierung

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

## Bewertung der Modernisierungsfähigkeit

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

## Monolithische Anwendungen (Monolithen)

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

## MPA

Siehe [Bewertung des Migrationsportfolios](#).

## MQTT

Siehe [Message Queuing-Telemetrietransport](#).

## Mehrklassen-Klassifizierung

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

## veränderbare Infrastruktur

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

## O

### OAC

[Siehe Origin Access Control.](#)

### OAI

Siehe [Zugriffsidentität von Origin.](#)

### COM

Siehe [organisatorisches Change-Management.](#)

## Offline-Migration

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

## OI

Siehe [Betriebsintegration.](#)

## OLA

Siehe Vereinbarung auf [operativer Ebene.](#)

## Online-Migration

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

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

## OPC-UA

Siehe [Open Process Communications — Unified Architecture](#).

## Offene Prozesskommunikation — Einheitliche Architektur (OPC-UA)

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

## Vereinbarung auf Betriebsebene (OLA)

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

## Überprüfung der Betriebsbereitschaft (ORR)

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

## Betriebstechnologie (OT)

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

## Betriebsintegration (OI)

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

## Organisationspfad

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

## Organisatorisches Veränderungsmanagement (OCM)

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

## Ursprungszugriffskontrolle (OAC)

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

## Ursprungszugriffsidentität (OAI)

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

## ORR

Weitere Informationen finden Sie unter [Überprüfung der Betriebsbereitschaft](#).

## NICHT

Siehe [Betriebstechnologie](#).

## Ausgehende (egress) VPC

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

# P

## Berechtigungsgrenze

Eine IAM-Verwaltungsrichtlinie, die den IAM-Prinzipalen zugeordnet ist, um die maximalen Berechtigungen festzulegen, die der Benutzer oder die Rolle haben kann. Weitere Informationen finden Sie unter [Berechtigungsgrenzen](#) für IAM-Entitäts 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.