



Modernisierung Ihrer Anwendung durch Migration von einem RDBMS zu Amazon DynamoDB

AWS Präskriptive Leitlinien



AWS Präskriptive Leitlinien: Modernisierung Ihrer Anwendung durch Migration von einem RDBMS zu Amazon DynamoDB

Copyright © 2024 Amazon Web Services, Inc. and/or its affiliates. All rights reserved.

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

Table of Contents

Einführung	1
Übersicht über DynamoDB	2
Funktionen und Vorteile	2
Partitionsschlüssel	3
Indizes	4
Zeit zum Leben	4
Preismodelle	4
Transaktionen	7
Große Komponenten	8
Backup und Backup	8
Sprach- und SDK-Unterstützung	9
Beispielanwendung	10
Altes Datenzugriffsmuster	10
Neues Datenzugriffsmuster	11
RDBMS-Tabellenschema und Zuordnung	12
Design für einzelne Tische	14
Globale sekundäre Indizes	15
Zugriffsmuster	16
Objektpersistenzschnittstelle	16
Schnittstelle für Dokumente	17
Low-Level-API	18
Konverter	19
Bewährte Methoden	21
Verwenden Sie das Objektpersistenzzugriffsmuster	21
Wählen Sie den richtigen Kapazitätsbereitstellungsmodus	21
Caching verwenden	21
Verwenden Sie Abfragen statt Scans	21
Validieren	22
Häufig gestellte Fragen	23
Was ist die maximale Tabellengröße, die ich in DynamoDB erstellen kann?	23
Wie viele Tabellen kann ich pro Konto erstellen?	23
Wie viele globale sekundäre Indizes kann ich für eine DynamoDB-Tabelle erstellen?	23
Wie viele Artikel kann ich pro Transaktion hinzufügen oder ändern?	23
Die nächsten Schritte und Ressourcen	24

Dokumentverlauf	25
Glossar	26
#	26
A	27
B	30
C	32
D	36
E	40
F	42
G	44
H	45
I	46
L	49
M	50
O	54
P	57
Q	60
R	60
S	63
T	67
U	69
V	69
W	70
Z	71
.....	lxxii

Modernisieren Sie Ihre Anwendung durch die Migration von einem RDBMS zu Amazon DynamoDB

Ramkumar Ramanujam und Mahesh Kumar Vemula, Amazon Web Services (AWS)

Februar 2023 ([Dokumentengeschichte](#))

Wenn ein Unternehmen sein Geschäft ausweitet, steigt die Belastung seiner Informationssysteme erheblich. Maßnahmen zur Leistungsoptimierung können dazu beitragen, dieser gestiegenen Nachfrage nur bis zu einem gewissen Grad gerecht zu werden. Oft führt die erhöhte Nachfrage dazu, dass die Datenbank die Last nicht mehr bewältigen kann. Dieses Problem tritt besonders häufig bei Anwendungen auf, die auf einem Verwaltungssystem für relationale Datenbanken (Relational Database Management System, RDBMS) basieren. Große RDBMS-Anbieter bieten zwar Möglichkeiten, die Systemlast durch Read Replicas, Datenbankspiegelung, primärsekundäre Konfigurationen und ähnliche Funktionen zu bewältigen, aber die Kosten und Lizenzanforderungen für diese Funktionen könnten zu Problemen führen. Für Organisationen, die nach Alternativen für die Datenspeicherung suchen, konzentriert sich dieser Leitfaden auf die Migration von einem RDBMS zu Amazon DynamoDB.

Der erste Teil dieses Handbuchs bietet einen Überblick über die Funktionen und Vorteile von DynamoDB. Der zweite Teil des Handbuchs basiert auf einer Fallstudie einer Anwendung, die von einem RDBMS (Microsoft SQL Server) zu DynamoDB migriert wurde. Es enthält Codebeispiele zur Bewältigung zweier Migrationsherausforderungen: die Zuordnung relationaler Daten zur DynamoDB-Dokumentstruktur und zur Schlüsselwerterfassung sowie die Änderung der Datenzugriffsebene der Anwendung, um die Erstellung-, Lese-, Aktualisierungs- und Löschvorgänge (CRUD) in DynamoDB zu unterstützen.

Dieses Handbuch richtet sich an Programm- oder Projektmanager, Datenbankadministratoren und Datenbankarchitekten, die planen, ihre RDBMS-Systeme zu DynamoDB zu migrieren, um ihren RAD-Anforderungen (Rapid Application Development) und hohen Leistungsanforderungen gerecht zu werden. Das Handbuch setzt ein grundlegendes Verständnis von relationalen Datenbanken und NoSQL-Konzepten voraus, erfordert jedoch keine DynamoDB-Kenntnisse oder -Erfahrung.

Übersicht über DynamoDB

Amazon DynamoDB ist eine NoSQL-Schlüsselwert- und Dokumentdatenbank, die schnelle und planbare Leistung nahtlos skalierbar bereitstellt. Es ist eine vollständig verwaltete, regionenübergreifende, multiaktive und langlebige Datenbank. (Die globale Tabellenfunktion in DynamoDB synchronisiert Änderungen, die in einer AWS Region vorgenommen wurden, automatisch mit allen anderen ausgewählten Regionen, um multiaktive Unterstützung zu bieten.) DynamoDB verfügt über integrierte Sicherheitsfunktionen, Sicherungs- und Wiederherstellungsoptionen sowie In-Memory-Caching für internetfähige Anwendungen.

Die schemalose Natur von NoSQL-Datenbanken trägt dazu bei, die Zeit und den Prozess zu reduzieren, die erforderlich sind, um eine Datenbankänderung (Leseschema) in die Produktion zu bringen, und ermöglicht so eine schnelle Anwendungsentwicklung (RAD). NoSQL-Datenbanken wie DynamoDB sind für leistungsstarke Lese-/Schreibvorgänge konzipiert.

Funktionen und Vorteile

DynamoDB bietet die folgenden Funktionen und Vorteile:

- Keine zu verwaltenden Server — DynamoDB ist ein vollständig verwalteter NoSQL-Datenbankdienst, was bedeutet, dass kein Serverwartungsaufwand anfällt.
- Schemaless — DynamoDB unterstützt die schnelle Entwicklung und Bereitstellung von Anwendungen.
- Skalierbare Leistung — DynamoDB bietet schnelle und vorhersehbare Leistung mit nahtloser Skalierbarkeit.
- ACID-Unterstützung — DynamoDB unterstützt Atomicity, Konsistenz, Isolation und Durability (Atomicity, Isolation, Isolation, Isolation, Consistency, Isolation, Isolation, Isolation, Isolation, Durability, Atomicity, Isolation)
- Hohe Verfügbarkeit und Beständigkeit — Ihre Daten werden auf Solid State Disks (SSDs) gespeichert und automatisch über mehrere Availability Zones in einer AWS -Region repliziert, um integrierte Hochverfügbarkeit und Datenhaltbarkeit bereitzustellen.
- Automatische Skalierung — DynamoDB verwendet den AWS Application Auto Scaling Service, um die bereitgestellte Durchsatzkapazität als Reaktion auf Datenverkehrsmuster dynamisch anzupassen.

- Flexible Preisoptionen — DynamoDB bietet zwei Kapazitätsmodi mit spezifischen Abrechnungsoptionen: den Modus der On-Demand-Kapazität und den Modus bereitgestellter Kapazität.
- point-in-time P-Wiederherstellung — Sie können kontinuierliche Backups mithilfe von point-in-time Recovery aktivieren, um Ihre Tabellen vor versehentlichen Schreib- oder Löschvorgängen zu schützen und Ihre Tabellen zu einem beliebigen Zeitpunkt der letzten 35 Tage wiederherzustellen.
- Time to Live (TTL) — Sie können Elemente nach einer bestimmten Dauer automatisch aus einer DynamoDB-Tabelle löschen.
- Globale Tabellen — Sie können mehrere Replikate regionsübergreifend AWS bereitstellen, ohne Ihre eigene Replikationslösung erstellen zu müssen.
- Globale sekundäre Indizes — Sie können Ihre DynamoDB-Tabelle abfragen, indem Sie Partitions- und Sortierschlüssel verwenden, die sich von den eigenen Partitions- und Sortierschlüsseln der Tabelle unterscheiden.
- DAX — Der Caching-Dienst DynamoDB Accelerator (DAX) bietet Reaktionszeiten unter einer Millisekunde für Leseoperationen.
- DynamoDB Streams — Diese Funktion stellt eine zeitlich geordnete Abfolge von Änderungen auf Elementebene in einem Protokoll bereit, um die Nachverfolgung und Benachrichtigung über Änderungen in DynamoDB-Tabellen in Echtzeit zu unterstützen.

Weitere Informationen zu diesen Vorteilen finden Sie auf der AWS Website unter [Amazon DynamoDB DynamoDB-Funktionen](#). In den folgenden Abschnitten werden einige der Funktionen beschrieben, die für die Modernisierung Ihrer Datenbank-Workloads durch die Migration von einem RDBMS zu DynamoDB relevant sind.

Partitionsschlüssel

DynamoDB ist schemaloss, daher müssen nicht alle Attribute für eine Tabelle definiert werden. Der Partitionsschlüsselattribut ist erforderlich und der Sortierschlüssel optional. Die restlichen Attribute sind beliebig und können je nach Artikel variieren. Wir empfehlen, einen Partitionsschlüssel mit hoher Kardinalität zu wählen, damit sich häufig aufgerufene Elemente nicht auf derselben Partition befinden. Diese Vorgehensweise hilft Ihnen, Ungleichgewichte beim Datenzugriff und heiße Partitionen zu vermeiden. Weitere Informationen finden Sie unter [Bewährte Methoden für das Entwerfen und die effektive Verwendung von Partitionsschlüsseln](#) in der DynamoDB-Dokumentation.

Indizes

Indexe ermöglichen Ihnen den Zugriff auf alternative Abfragemuster und können Abfragen beschleunigen. Sie sollten Indizes sorgfältig erstellen, unabhängig davon, ob Sie eine relationale Datenbank oder DynamoDB verwenden. Bei jedem Schreibvorgang in einer Tabelle müssen alle Indexe für diese Tabelle aktualisiert werden.

Ein globaler sekundärer Index enthält eine Auswahl von Attributen aus der Basistabelle. Diese sind jedoch nach einem Primärschlüssel organisiert, der sich vom Primärschlüssel der Tabelle unterscheidet. In DynamoDB sind globale sekundäre Indizes standardmäßig spärlich. Das heißt, der Sortierschlüssel ist optional und erscheint nicht in jedem Tabellenelement. Um diese Funktion zu nutzen, können Sie globale sekundäre Indizes erstellen, die nur die erforderlichen Attribute speichern und projizieren. Sie können bis zu 20 globale sekundäre Indexe in einer DynamoDB-Tabelle speichern. Weitere Informationen zu diesem Feature finden Sie unter [Verwenden von globalen sekundären Indizes in DynamoDB](#) in der DynamoDB-Dokumentation.

Zeit zum Leben

Sie können eine Time to Live (TTL) -Eigenschaft für eine DynamoDB-Tabelle festlegen, um einen Zeitstempel pro Element (Datensatz) zu definieren, um anzugeben, wann ein Element nicht mehr benötigt wird. Kurz nach dem angegebenen Zeitstempel löscht DynamoDB das Element aus der Tabelle, ohne zusätzliche Kapazitätseinheiten zu verbrauchen. Weitere Informationen zu dieser Funktion finden Sie unter [Expiring items by using DynamoDB Time to Live](#) in der DynamoDB-Dokumentation.

Preismodelle

DynamoDB bietet zwei Preismodelle zur Auswahl: bereitgestellte Kapazität und On-Demand-Kapazität. Ihre Wahl des Preismodells hängt von Ihren prognostizierten Workloads ab.

Preismodell	Art der Arbeitslast	Kosten	Lese-/Schreibdurchsatz
Bereitgestellte Kapazität	Vorhersagbar	Niedriger	Sie geben die Anzahl der Lese-/Schreibvorgänge pro Sekunde als Lesekapazitätseinheit

Preismodell	Art der Arbeitslast	Kosten	Lese-/Schreibdurchsatz
			<p>eiten (RCUs) und als Schreibkapazitätseinheiten (WCUs) an. Beispiel:</p> <ul style="list-style-type: none"> • Bei einem Element mit einer Größe von bis zu 4 KB kann 1 RCU 2 Eventually Consistent-Lesevorgänge pro Sekunde ausführen. • Bei einem Element mit einer Größe von bis zu 1 KB kann 1 WCU 2 Eventually Consistent-Lesevorgänge pro Sekunde ausführen. <p>Sie können die automatische Skalierung aktivieren, um die Kapazität an Verkehrsänderungen anzupassen.</p>

Preismodell	Art der Arbeitslast	Kosten	Lese-/Schreibdurchsatz
Kapazität auf Abruf	Dynamisch	Höher	<p>Sie geben keine Durchsatzanforderungen an. DynamoDB passt sich Ihren Workloads automatisch an.</p> <p>Ihnen werden die Lese- und Schreibvorgänge, die Ihre Anwendung in Ihren Tabellen durchführt, als Leseanforderungseinheiten und als Schreibanforderungseinheiten in Rechnung gestellt. Beispiel:</p> <ul style="list-style-type: none"> Ein 8-KB-Objekt benötigt 1 Leseanforderungseinheit für eine letztendlich konsistente Lese- und 8 Schreibanforderungseinheiten für eine nicht transaktionale Schreiboperation.

Weitere Informationen zu diesen beiden Modellen finden Sie unter [Lese-/Schreibkapazitätsmodus](#) in der DynamoDB-Dokumentation.

Transaktionen

DynamoDB unterstützt ACID (Atomicity, Consistency, Isolation, Isolation, Isolation, Isolation, Isolation, Isolation, Consistency, Isolation, Isolation, Isolation, Isolation, Isolation, AWS Isolation, Isolation, AWS, Isolation)

Um Änderungen an mehreren Elementen innerhalb und zwischen Tabellen zu verwalten, können Sie die Transaktions `TransactWriteItems` - und `TransactGetItems` APIs von DynamoDB verwenden.

- `TransactWriteItems` ist eine Batch-Operation, die einen Schreibrucksatz mit einer oder mehreren `PutItem`, `UpdateItem`, und `DeleteItem` Aktionen enthält. `TransactWriteItems` kann optional überprüfen, ob die Voraussetzungen erfüllt sein müssen, bevor Aktualisierungen vorgenommen werden. Diese Bedingungen können dieselben Elemente wie die im Schreibrucksatz oder verschiedene Elemente beinhalten. Wenn eine Bedingung nicht erfüllt ist, wird die Transaktion abgelehnt.
- `TransactGetItems` ist eine Batch-Operation, die einen Lesesatz mit einer oder mehreren `GetItem` Aktionen enthält. Wenn Sie eine `TransactGetItems` Anfrage für ein Element stellen, das Teil einer aktiven Schreibtransaktion ist, wird die Lesetransaktion storniert. Um den zuvor festgelegten Wert abzurufen, können Sie einen Standard-Lesevorgang verwenden.

Weitere Informationen zu diesen APIs finden Sie in der [DynamoDB-Dokumentation unter Amazon DynamoDB-Transaktionen: So funktioniert es](#).

Einschränkungen

Für transaktionale DynamoDB-API-Operationen gelten die folgenden Einschränkungen:

- Eine Transaktion kann nicht mehr als 100 eindeutige Elemente aktualisieren.
- Eine Transaktion darf nicht mehr als 4 MB Daten enthalten.
- Zwei Aktionen gleichzeitig dürfen nicht auf dasselbe Element in der derselben Tabelle einwirken. Beispielsweise können Sie in einer Transaktion nicht `ConditionCheck` sowohl `Update` AND-Aktionen für denselben Artikel ausführen.
- Eine Transaktion kann nicht auf Tabellen in mehr als einem AWS-Konto oder mehr als einer Region einwirken.

- Transaktionsoperationen bieten ACID-Garantien nur innerhalb der AWS Region, in der der Schreibvorgang ursprünglich stattfindet. Regionsübergreifende Transaktionen werden in globalen Tabellen nicht unterstützt.
- Das Objektpersistenzmodell unterstützt keine Transaktionen. Um die Transaktionsfunktion verwenden zu können, müssen Sie mithilfe der [DynamoDB-Low-Level-API](#) auf die Datenbank und die Tabellen zugreifen.

Große Komponenten

DynamoDB hat eine Größenbeschränkung von 400 KB für jedes Element. Dieses Limit umfasst sowohl den Attributnamen (binäre Länge mit UTF-8-Codierung) als auch den Attributwert (ebenfalls binäre Länge). Der Attributname wird bei der Größenbeschränkung berücksichtigt. Stellen Sie sich beispielsweise ein Element mit zwei Attributen vor: ein Attribut mit dem Namen „Country-Code“ mit dem Wert „IN“ und ein weiteres Attribut namens country-phone-prefix "" mit dem Wert „91“. Die Gesamtgröße dieses Elements beträgt 36 Byte.

Problemumgehung

Wenn ein Element mit vielen Attributen und Eigenschaften oder einer großen Datenmenge verknüpft ist, kann seine Größe 400 KB überschreiten. In diesem Fall können Sie das serialisierte Element in Amazon Simple Storage Service (Amazon S3) im JSON-Format speichern und die Amazon S3-Position als Attribut (S3Location) in dem Element speichern. Lese- und Schreibvorgänge für dieses Element rufen das S3-Objekt ab und aktualisieren die JSON-Zeichenfolge. Der Primärschlüssel, der Sortierschlüssel und alle Attribute, die von lokalen Indizes und globalen Sekundärindizes verwendet werden, sollten zusammen mit dem S3Location Attribut in der Tabelle gespeichert werden. Dies erfordert zusätzliche Logik in der Anwendung (Datenzugriffsebene), um nach dem S3Location Attribut zu suchen und die vollständigen Artikeldaten von Amazon S3 abzurufen.

Backup und Backup

Die Unterstützung von Backup- und Wiederherstellungsfunktionen wird in jeder Datenbank häufig erwartet. DynamoDB unterstützt von Haus aus Sicherungs- und Wiederherstellungsvorgänge innerhalb desselben Kontos, Sie können jedoch mithilfe anderer Optionen oder Prozesse Tabellen für mehrere Konten kopieren. Diese Prozesse verbrauchen keine Lese-/Schreibkapazitätseinheiten. Weitere Informationen finden Sie im Leitfaden [Kontübergreifende Optionen zum Kopieren vollständiger Tabellen für Amazon DynamoDB](#) auf der AWS Prescriptive Guidance-Website.

Einschränkungen

DynamoDB unterstützt derzeit die kontoübergreifende Sicherung und Wiederherstellung mithilfe [AWS Backup](#), das Konto muss jedoch Teil derselben Organisation sein. Sie können diese Einschränkung beheben, indem Sie eine der folgenden Lösungen anwenden:

- Benutzerdefinierte Implementierung in einer Programmiersprache Ihrer Wahl (z. B. .NET, Java oder Python) mithilfe eines [AWSSDK](#). Sie können Elemente aus der Quelltable in Konto A scannen und items (`BatchWrite`) in eine Tabelle in Konto B schreiben. Sie können diesen Code auf dem Server, auf einem lokalen Computer oder in AWS Lambda (wenn die Datenbank klein ist und die Ausführung des Skripts weniger als 15 Minuten dauert) ausführen. Weitere Informationen finden Sie auf der Website [AWS Prescriptive Guidance](#) unter dem Muster [Amazon DynamoDB-Tabellen kontoübergreifend mit einer benutzerdefinierten Implementierung kopieren](#).
- Verwenden von AWS Data Pipeline Sie können aus der Quelltable exportieren (scannen) und in die Zieltabelle importieren (schreiben). Weitere Informationen finden Sie [AWS Data Pipeline](#) in der [DynamoDB-Dokumentation unter Exportieren und Importieren von DynamoDB-Daten mithilfe](#).
- Verwenden von AWS Glue Informationen zu dieser Option finden Sie im Leitfaden [Kontoübergreifende Optionen zum Kopieren vollständiger Tabellen für Amazon DynamoDB](#) auf der [AWS Prescriptive Guidance-Website](#).

Sprach- und SDK-Unterstützung

Die [AWSSDKs](#) bieten eine vereinfachte Programmierschnittstelle für AWS Dienste und Unterstützung für .NET, Java JavaScript, Node.js, Python, PHP und Ruby.

Sie können aus drei Mustern wählen, um mithilfe eines AWS SDK auf DynamoDB-Tabellen zuzugreifen: Objektpersistenzmodell (High-Level-Schnittstelle), Dokumentschnittstellen und Low-Level-Schnittstellen. Weitere Informationen finden Sie unter [Zugriffsmuster weiter](#) unten in diesem Leitfaden.

Beispielanwendung

Dieser Abschnitt enthält Anleitungen für Teams, die eine Migration von ihrem relationalen Datenbankmanagementsystem (RDBMS) zu einer NoSQL-Datenbank evaluieren, und konzentriert sich auf Amazon DynamoDB als NoSQL-Zieldatenbank. Es befasst sich mit den folgenden zwei Herausforderungen, basierend auf einer Fallstudie einer Anwendung, die von Microsoft SQL Server zu DynamoDB migriert wurde:

- Zuordnung relationaler Daten aus mehreren Tabellen im RDBMS zu einer Dokumentstruktur und einer Schlüsselwertsammlung in DynamoDB
- Ändern der Datenzugriffsschicht in der Anwendung für die Ausführung von CRUD-Operationen (Create,,,,,,,,,,,,, aktualisieren/aktualisieren/aktualisieren/aktualisieren/aktualisieren/aktualisieren/aktualisieren/les)

Die Diskussion und Anleitung beinhalten Codebeispiele, die in C# geschrieben wurden und das AWS SDK for .NET verwenden.

Die Beispiel-Webanwendung verwaltet die Konfiguration für Hunderte von Anwendungen, die in einer Organisation verwendet werden, einschließlich zulässiger Benutzer und Hosts (Web, Handy, Desktop) für jede Anwendung, Metadaten, Suchschlüsselwörter usw. Die Anwendung bietet Konfigurations-, Wartungs- und Suchfunktionen für verschiedene Versionen verschiedener Anwendungen, die in der Organisation verwendet werden. Konfigurationsänderungen werden mithilfe von Audit-Tabellen nachverfolgt. Hier ist ein typischer Arbeitsablauf für die Beispielanwendung:

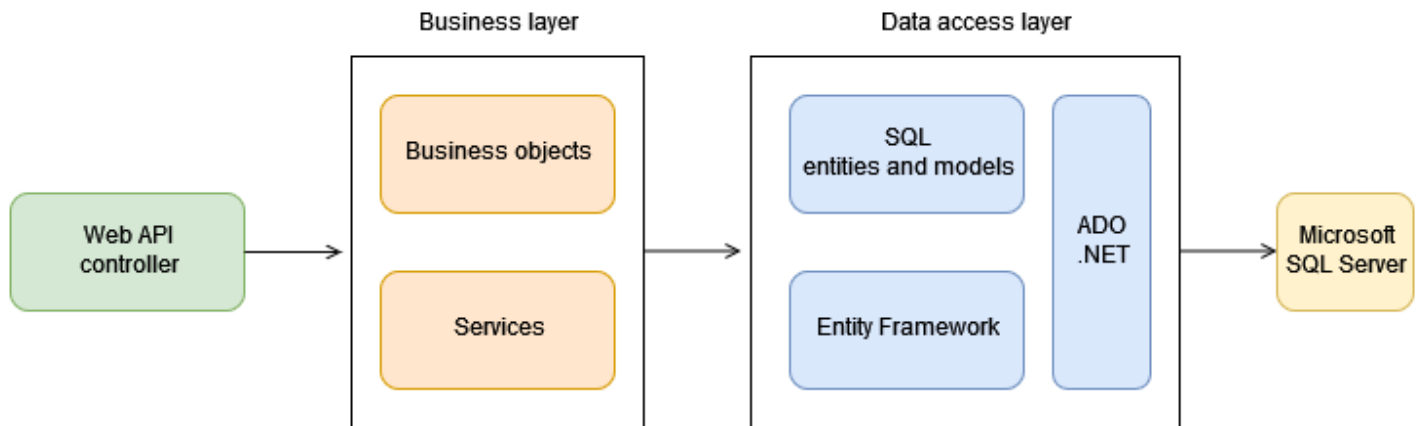
1. Erstellen Sie eine Konfiguration für die Testanwendung.
2. Führen Sie die Konfiguration der Testanwendung in die Produktionsumgebung über (d. h. erstellen Sie eine Produktionsanwendungskonfiguration).
3. Änderungen aktualisieren und prüfen (Audit-Datensatz erstellen, geänderte Anwendungskonfiguration aufrufen).

Altes Datenzugriffsmuster

Der Quelltechnologiestapel bestand aus Folgendem:

- ASP.NET-Web-API-Controller

- Geschäftsobjekte
- ASP.NET-Entitätsframework (EF)
- ADO.NET-Datendienste
- Microsoft SQL Server 2016

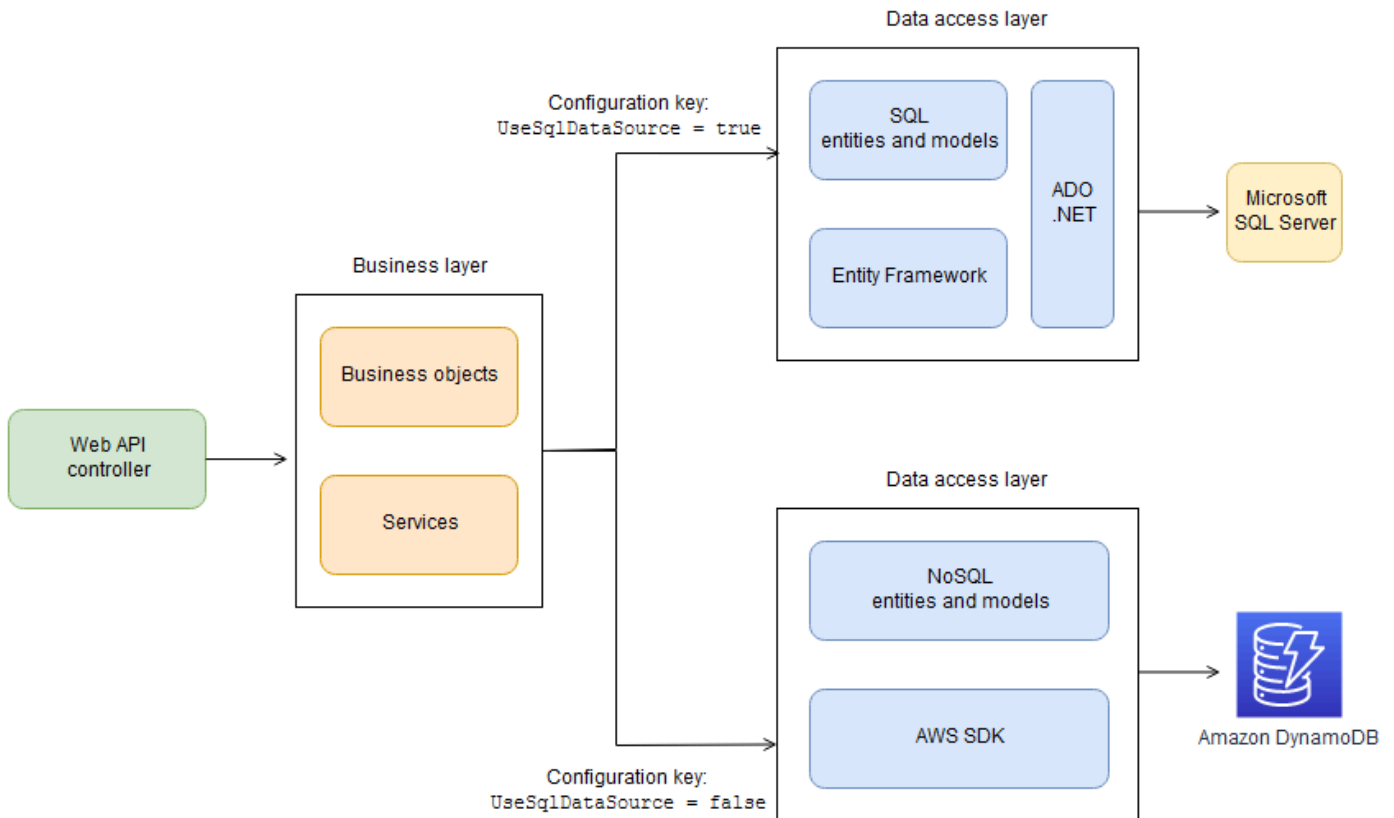


Neues Datenzugriffsmuster

Die migrierte Anwendung unterstützt sowohl SQL Server als auch DynamoDB auf der Grundlage des in der Konfigurationsdatei bereitgestellten Konfigurationsschlüssels (`UseSqlDataSource`). Wie im folgenden Diagramm dargestellt, stellt die Anwendung eine Verbindung zu SQL Server her, wenn der Wert von `UseSqlDataSource` ist. Wenn der Wert lautet `false`, stellt die Anwendung eine Verbindung zu DynamoDB her.

Der neue Technologie-Stack besteht aus Folgendem:

- ASP.NET-Web-API-Controller — Akzeptiert HTTP-Anfragen über verschiedene API-Endpunkte.
- Geschäftsobjekte und Dienste — Klassen und Objekte, die über die Geschäftslogik verfügen, um aus der Datenbank abgerufene Eingaben und Daten zu verarbeiten.
- NoSQL-Entitäten und -Modelle — Klassen, die in DynamoDB gespeicherten Elementen zugeordnet sind.
- AWSSDK — Bietet programmatischen Zugriff auf DynamoDB und andere AWS Dienste.
- DynamoDB — Datenbank zum Speichern von Anwendungsdaten.

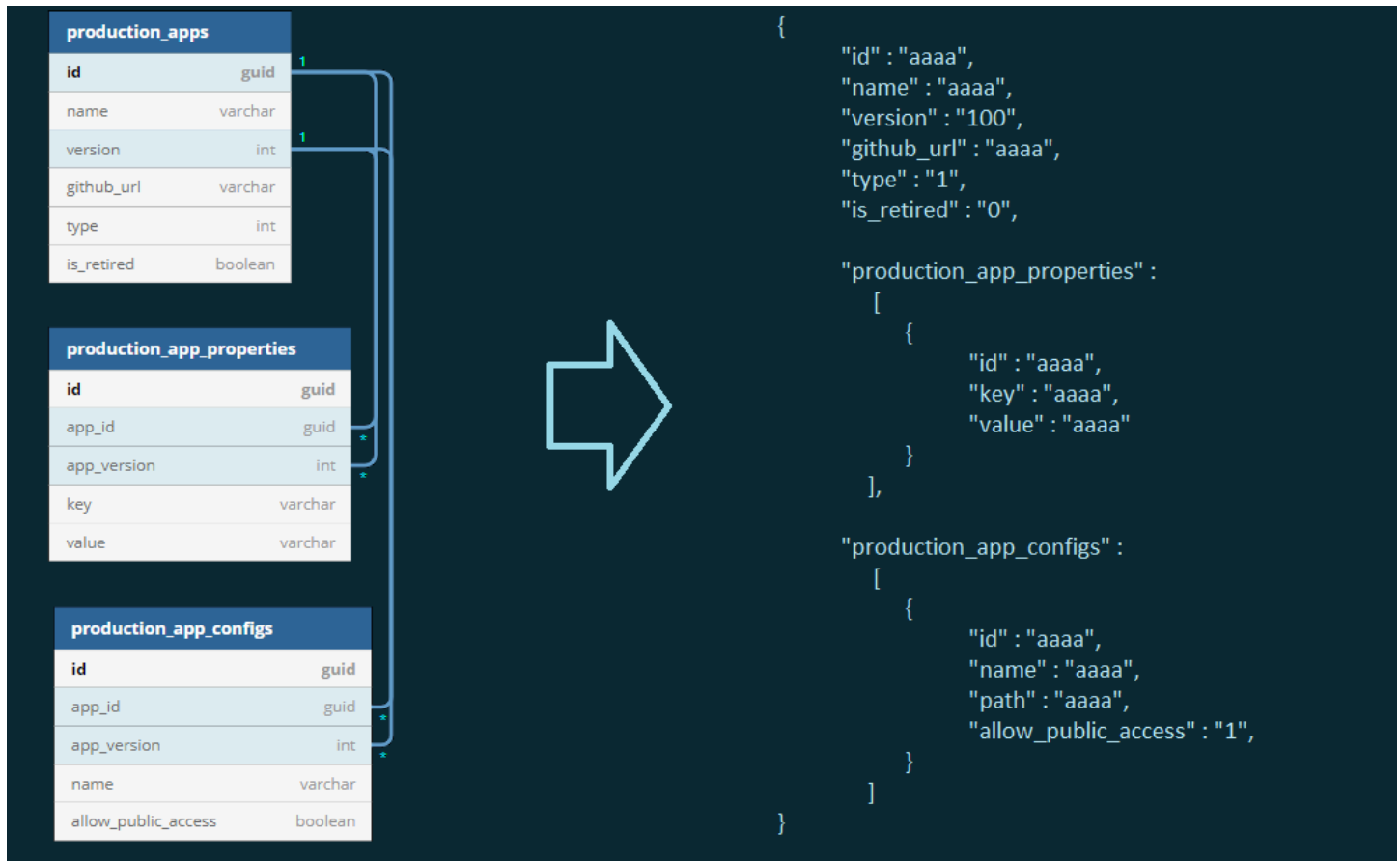


RDBMS-Tabellenschema und Zuordnung

Das folgende Diagramm zeigt die Tabellen und Beziehungen im Quell-RDBMS-Schema.



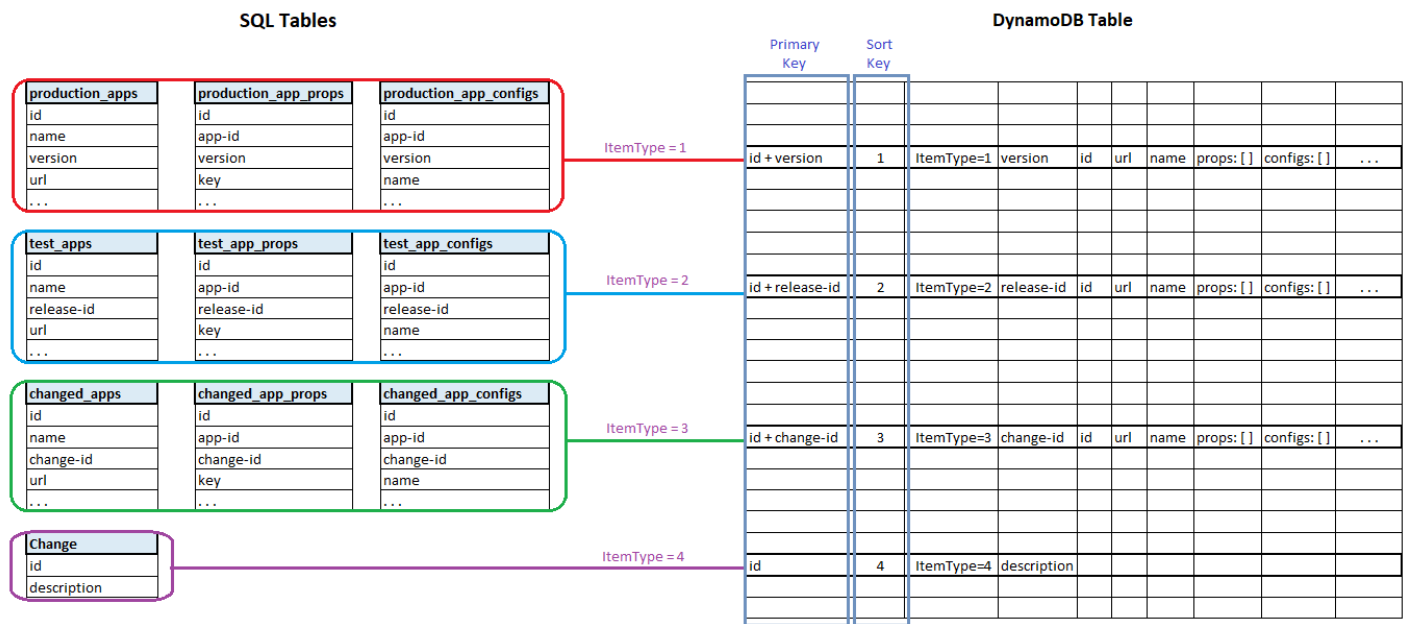
Wie im vorherigen Diagramm gezeigt, hat die `production_apps` Tabelle `id` und `version` Spalten, die in einer one-to-many Beziehung zu den `production_app_configs` Tabellen `production_app_properties` und stehen. Daher sind im DynamoDB-Design die `production_app_configs` Tabellen `production_app_properties` und in die eingebettet `production_app` item, wie im folgenden JSON-Code gezeigt. Weil `production_app_properties` diese Tabellen mehrere Werte haben `production_app_configs` können, werden sie als Arrays im JSON-Code hinzugefügt. Die `test_apps` Tabellen `changed_apps` und sind auf ähnliche Weise zugeordnet.



Design für einzelne Tische

DynamoDB verwaltet keine Beziehungen; es unterstützt ein festes Tabellenschema. Daher können Sie verschiedene Elementtypen (z. B. SQL-Tabellen) in einer einzigen DynamoDB-Tabelle speichern, mit einem Attribut (`ItemType`), das den Elementtyp identifiziert.

In DynamoDB muss die Kombination aus Partitionsschlüsseln (PK) und Sortierschlüsseln (SK) eindeutig sein, sodass diese Schlüssel je nach Elementtyp variieren.



Globale sekundäre Indizes

Indizes helfen dabei, Daten schneller abzurufen und die Anwendungsleistung zu verbessern. Für die Beispielanwendung wurden die folgenden Indizes erstellt. Die PKs und SKs wurden danach ausgewählt, wie unterschiedliche Elemente identifiziert werden konnten.

Name des Indexes	Beschreibung	Partitionsschlüssel (PK)	Sortierschlüssel (SK)	Projected Attribute
Version-index	Ruft alle Produktionsanwendungen für einen bestimmten Wert abversion.	version		id, name
Release-index	Ruft alle Testanwendungen für einen bestimmten Wert abrelease-id.	release-id		id, name

Name des Indexes	Beschreibung	Partitionsschlüssel (PK)	Sortierschlüssel (SK)	Projected _Attribute
Change-index	Ruft alle (geänderten) Anwendungen ab, die einem zugeordnet sindchange-id	change-id		id, modified-by, date

Zugriffsmuster

Wie bereits in diesem Handbuch erwähnt, können Sie aus drei Zugriffsmustern für DynamoDB-Tabellen wählen: Objektpersistenzschnittstelle, Dokument und API-Schnittstelle für DynamoDB-Tabellen ausführen: Objektpersistenzschnittstelle, Dokumentschnittstellen und Low-Level-API-Schnittstelle. In den folgenden Abschnitten wird jede Schnittstelle beschrieben. Für unseren Anwendungsfall SQL Server to DynamoDB haben wir aus Gründen der Einfachheit, Lesbarkeit und Wartungsfreundlichkeit die Object Persistence-Schnittstelle gewählt.

Objektpersistenzschnittstelle

Die Objektpersistenzschnittstelle bietet einen abstrakten Zugriffsmechanismus auf hoher Ebene für die Ausführung von CRUD-Operationen an DynamoDB-Elementen unter Verwendung von .NET-Modellen, ähnlich Entity Framework-Entitäten. Die Schnittstelleneigenschaften sind den Attributen von DynamoDB-Elementen zugeordnet. DasAWS SDK for .NET unterstützt benutzerdefinierte Eigenschaftsattribute in diesem Modell, um die Serialisierung und Deserialisierung einzelner Eigenschaften anzupassen, Nullwerte zu verarbeiten und Typkonvertierungen vorzunehmen.

In der Anwendung verwendetes Beispielmodell:

```
[DynamoDBTable("AppLibrary")]
public class ProdApp
{
    [DynamoDBHashKey]
    public string PK { get; set; }    //Partition key
```

```
[DynamoDBRangeKey]
public string SK { get; set; }    //Sort key

[DynamoDBGlobalSecondaryIndexRangeKey("Version-index")]
[DynamoDBProperty]
public int Version { get; set; }
. . .
[DynamoDBProperty]
public Int64 TTL { get; set; }
}
```

Zugriff auf Artikel:

```
var _dynamoDbClient = new AmazonDynamoDBClient(AWSCredentials);
var _context = new DynamoDBContext(_dynamoDbClient);

public ProdApp GetProdAppById (Guid id, int version)
{
    var pk = $"{id}-{version}";
    return _context.Load<ProdApp>(pk, ItemType.ProductionApplication);
}
```

Weitere Informationen finden Sie unter [Object Persistence Interface](#) in der DynamoDB-Dokumentation.

Schnittstelle für Dokumente

Das Dokumentenschnittstellenmodell bietet dokumentenbasierten Zugriff (ähnlich wieXMLDocument in .NET) auf ein DynamoDB-Element. Dieses Modell bietet eine übergeordnete Programmierschnittstelle, übersetzt seine Aufrufe jedoch an APIs auf niedriger Ebene, um den Vorgang auszuführen.

```
var _dynamoDbClient = new AmazonDynamoDBClient(AWSCredentials);
var _table = Table.LoadTable(_dynamoDbClient, "AppLibrary");

public ProdApp GetProdAppById (Guid id, int version)
{
    var pk = $"{id}-{version}";
    var doc = _table.GetItem(pk, ItemType.ProductionApplication);
    var app = new ProdApp {
        PK = doc["PK"],
```

```
        SK = doc["SK"],
        Version = doc["Version"],
        . . .
    };
    return app;
}
```

Weitere Informationen finden Sie unter [Document Interfaces](#) in der DynamoDB-Dokumentation.

Low-Level-API

Das AWS SDK für DynamoDB bietet auch API-Zugriff auf niedriger Ebene, um CRUD-Operationen mithilfe der `DeleteItem`, `PutItem`, `GetItem`, `UpdateItem`, und `BatchWriteItem` auszuführen. Dieses Modell bietet die vollständige Kontrolle über die Attributzuordnung und die Typkonvertierung. Die Antwort für diese Aufrufe ist ein Wörterbuch von Schlüssel-Wert-Paaren.

```
[DynamoDBTable("AppLibrary")]
public class ProdApp
{
    [DynamoDBHashKey]
    public string PK { get; set; }    //Partition key

    [DynamoDBRangeKey]
    public string SK { get; set; }    //Sort key

    [DynamoDBGlobalSecondaryIndexRangeKey("Version-index")]
    [DynamoDBProperty]
    public int Version { get; set; }

    . . .

    [DynamoDBProperty]
    public ProdConfig Config { get; set; }
}

var _dynamoDbClient = new AmazonDynamoDBClient(AWSCredentials);

public ProdApp GetProdAppById (Guid id, int version)
{
    var pk = $"{id}-{version}";
    var resp = _dynamoDbClient.Query(queryRequest);
    var item = resp.Items[0];
    var app = new ProdApp {
        PK = item["PK"].S,
```

```
SK = item["SK"].S,  
Version = Convert.ToInt32(item["Version"].S),  
. . .  
Config = new ProdConfig {  
    Name = item["Config"].M["Name"].S,  
    Id = Conver.ToInt32(item["Config"].M["Id"].S)  
}  
};  
return app;  
}
```

Weitere Informationen finden Sie unter [Low-Level-Schnittstellen](#) in der DynamoDB-Dokumentation.

Konverter

In einigen Fällen müssen Sie möglicherweise Daten ändern oder transformieren, während Sie sie aus der DynamoDB-Datenbank speichern oder daraus lesen. In diesen Szenarien können Sie die [PropertyConverterI-Schnittstelle](#) von [Amazon.DynamoDBv2 verwenden. DataModelNamespace](#), indem Sie Code ähnlich dem folgenden verwenden:

```
// Converts the null values of a string property to a valid string and vice versa.  
public class NullOrStringConverter : IPropertyConverter  
{  
    // Called when creating the JSON / DynamoDB item from the model  
    public DynamoDBEntry ToEntry(object value)  
    {  
        var entry = new Primitive  
        {  
            value = new DynamoDBNull()  
        };  
        if(value != null)  
        {  
            entry.Value = value.ToString();  
        }  
        return entry;  
    }  
    // Called when populating the model from the JSON / DynamoDB item  
    public object FromEntry(DynamoDBEntry entry)  
    {  
        if(entry is DynamoDBNull)  
        {
```

```
        return string.Empty;
    }
    else
    {
        return entry.ToString();
    }
}
}
```

Verwendung des Konverters im Modell:

```
[DynamoDBTable("AppLibrary")]
public class ProdApp
{
    . . .

    [DynamoDBProperty (typeof(NullOrString))]
    public string AppConfigId { get; set; }
    . . .
}
```


Bewährte Methoden

In diesem Abschnitt werden zusätzlich zu den bewährten Methoden, die in den vorherigen Abschnitten erörtert wurden, Empfehlungen zusammengefasst (z. B. das Speichern von Elementen mit mehr als 400 KB in Amazon S3, Verwendung von Indizes, Einzeltabellendesign und Transaktionen).

Verwenden Sie das Objektpersistenzzugriffsmuster

Wie bereits in diesem Handbuch beschrieben, bietet Amazon DynamoDB drei Zugriffsmuster. Die [Object Persistence-Schnittstelle](#) bietet Clean und easy-to-maintain Code. Sofern Sie bei Lese-/Schreibvorgängen keine Modelleigenschaftswerte anpassen oder transformieren müssen, empfehlen wir, die Objektpersistenzschnittstelle zu verwenden.

Wählen Sie den richtigen Kapazitätsbereitstellungsmodus

Durch die bedarfsgesteuerte Kapazitätsbereitstellung werden Lese- und Schreibvorgänge automatisch skaliert, wenn Ihre Workloads steigen und fallen. Wir empfehlen, dass Sie diesen Modus nutzen, wenn Ihre Arbeitslast nicht vorhersehbar ist. Die Kosten sind in der Regel höher als im Modus mit bereitgestellter Kapazität und werden auf einer bestimmten pay-as-you-use Grundlage berechnet. Wir empfehlen, den Modus für bereitgestellte Kapazität zu verwenden, wenn Ihr Workload vorhersehbar ist und Sie Ihre Kapazitätsanforderungen prognostizieren können. Weitere Informationen finden Sie im Abschnitt [Preismodelle](#).

Caching verwenden

Wir empfehlen, dass Sie bei der Arbeit mit DynamoDB Caching verwenden, um die mit jedem Lese-/Schreibvorgang verbundenen Kosten zu reduzieren. Verwenden Sie die entsprechende Invalidierungslogik, um Elemente aus dem Cache zu entfernen, wenn zwischengespeicherte Daten veraltet sind. Identifizieren Sie die am häufigsten verwendeten Endpunkte, um Caching zu implementieren.

Verwenden Sie Abfragen statt Scans

Vermeiden Sie DynamoDB-Scans so weit wie möglich. Eine DynamoDB-Abfrage ist effizienter und kostengünstiger als ein Scanvorgang. Eine Abfrage filtert Elemente auf der Grundlage

von Partitionsschlüsselwerten (PK) und Sortierschlüsselwerten (SK), wohingegen ein Scan alle Datensätze lesen muss, um Elemente anhand der angegebenen Parameter zu filtern. Die DynamoDB-Preise basieren auf dem Umfang der Datenlese/Schreibvorgänge, sodass Scans mehr Kosten verursachen als Abfragen. Abfragen sind auch schneller, was letztendlich die Leistung der Anwendung verbessert.

Validieren

DynamoDB ist eine NoSQL-Datenbank, sie verwaltet also keine Beziehungsdaten und enthält keine Datenintegritätsbeschränkungen. Es erfordert, dass jedes Element eine eindeutige Kombination aus einem Primärschlüssel und nur einem Sortierschlüssel hat. Um die Datenintegrität zwischen verwandten Elementen in der DynamoDB-Tabelle zu gewährleisten, empfehlen wir Ihnen, strenge Validierungen auf der Anwendungs- oder Geschäftsebene Ihres Systems durchzuführen.

Häufig gestellte Fragen

Dieser Abschnitt enthält Antworten auf häufig gestellte Fragen zur Verwendung von DynamoDB.

Was ist die maximale Tabellengröße, die ich in DynamoDB erstellen kann?

Die Größe der Tabelle oder die Anzahl der Spalten, die Sie erstellen können, ist unbegrenzt.

Wie viele Tabellen kann ich pro Konto erstellen?

Sie können bis zu 2 500 Tabellen für jede AWS-Region pro Konto erstellen. Wenn Sie mehr Tabellen erstellen möchten, können Sie [unter https://aws.amazon.com/support](https://aws.amazon.com/support) eine Erhöhung des Servicekontingents beantragen.

Wie viele globale sekundäre Indizes kann ich für eine DynamoDB-Tabelle erstellen?

Es gibt eine anfängliche Beschränkung von 20 globalen sekundären Indizes pro Tabelle. Wenn Sie mehr Indizes erstellen möchten, können Sie unter <https://aws.amazon.com/support> eine Erhöhung des Servicekontingents beantragen.

Wie viele Artikel kann ich pro Transaktion hinzufügen oder ändern?

Sie können bis zu 100 Elemente (oder 4 MB Daten) pro Transaktion hinzufügen oder ändern. Wenn Sie mehr als 100 Datensätze in eine Tabelle schreiben möchten, können Sie stapelweise Schreiboperationen verwenden.

Eine vollständige Liste der Kontingente finden Sie unter [Service-, Konto- und Tabellenkontingente in Amazon DynamoDB](#) in der DynamoDB-Dokumentation.

Die nächsten Schritte und Ressourcen

Amazon DynamoDB wurde als leistungsstarke NoSQL-Datenbank konzipiert. Aufgrund seiner geringen Kosten, hohen Leistung, automatischen Skalierung und anderer Funktionen ist es eine gute Alternative zu einem relationalen Datenbanksystem (RDBMS). Wenn Sie bewährte Methoden befolgen, Indizes verwenden, die richtigen Partitionsschlüssel auswählen und die Tabellenstruktur sorgfältig entwerfen, können Sie das Beste aus DynamoDB herausholen. Sie können DynamoDB Accelerator (DAX) für hochfrequente Lesevorgänge verwenden und von einer schnellen In-Memory-Leistung profitieren. Wenn Sie nach einer Alternative zu einem RDBMS suchen, sollten Sie DynamoDB aufgrund seiner Kosten- und Leistungsvorteile in Betracht ziehen.

Lesen Sie zum Einstieg die Verwendung von DynamoDB die folgenden Links.

DynamoDB-Dokumentation

- [NoSQL-Design](#)
- [Globale sekundäre Indizes](#)
- [Transaktionen](#)
- [Datenzugriff — Low-Level-API](#)
- [Datenzugriff — Dokumentenmodell \(mittlere Ebene\)](#)
- [Datenzugriff — Objektpersistenzmodell \(auf hoher Ebene\)](#)
- [So funktioniert's — Konsistenz lesen](#)
- [So funktioniert es — Lese-/Schreibkapazitätsmodus](#)
- [Mapping beliebiger Daten](#)
- [Bewährte Methoden](#)

AWSVeröffentlichungen zu präskriptiven Leitlinien

- [Modellieren von Daten mit Amazon DynamoDB \(Anleitung\)](#)
- [Kontoübergreifende Optionen zum Kopieren vollständiger Tabellen für Amazon DynamoDB \(Anleitung\)](#)
- [Kopieren Sie Amazon DynamoDB-Tabellen mithilfe einer benutzerdefinierten Implementierung \(Muster\) kontenübergreifend](#)

Dokumentverlauf

In der folgenden Tabelle werden die wichtigsten Änderungen an diesem Handbuch beschrieben. Wenn Sie über future Updates informiert werden möchten, können Sie einen [RSS-Feed](#) abonnieren.

Änderung	Beschreibung	Datum
Aktualisierte Informationen	Die Abschnitte zu transaktionalen API-Vorgängen , Sicherung und Wiederherstellung sowie die häufig gestellten Fragen wurden aktualisiert.	24. Februar 2023
Erste Veröffentlichung	—	30. September 2021

AWS Glossar zu präskriptiven Leitlinien

Im Folgenden finden Sie häufig verwendete Begriffe in Strategien, Leitfäden und Mustern, die von Prescriptive Guidance bereitgestellt AWS werden. 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 On-Premises-Oracle-Datenbank zu der PostgreSQL-kompatible Amazon-Aurora-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 auf einer EC2-Instance in der Cloud zu Oracle. AWS
- 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. Dieses Migrationsszenario ist spezifisch für VMware Cloud on AWS, das die Kompatibilität mit virtuellen Maschinen (VM) und die Workload-Portabilität zwischen Ihrer lokalen Umgebung und unterstützt. AWS Sie können die VMware-Cloud-Foundation-Technologien von Ihren On-Premises-Rechenzentren aus verwenden, wenn Sie

Ihre Infrastruktur zu VMware Cloud in AWS migrieren. Beispiel: Verlagern Sie den Hypervisor, der Ihre Oracle-Datenbank hostet, zu VMware Cloud on. 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

Siehe [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 mit künstlicher 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 von AIOps in der AWS - Migrationsstrategie finden Sie im [Leitfaden zur Betriebsintegration](#).

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 Einführung der Cloud (AWS CAF)

Ein Framework mit Richtlinien und bewährten Verfahren, das Unternehmen bei der Entwicklung eines effizienten und effektiven Plans für den erfolgreichen Umstieg auf die Cloud unterstützt. AWS CAF unterteilt die Leitlinien in sechs Schwerpunktbereiche, die als Perspektiven bezeichnet werden: Unternehmen, Mitarbeiter, Unternehmensführung, Plattform, Sicherheit und

Betrieb. Die Perspektiven Geschäft, Mitarbeiter und Unternehmensführung konzentrieren sich auf Geschäftskompetenzen und -prozesse, während sich die Perspektiven Plattform, Sicherheit und Betriebsabläufe auf technische Fähigkeiten und Prozesse konzentrieren. Die Personalperspektive zielt beispielsweise auf Stakeholder ab, die sich mit Personalwesen (HR), Personalfunktionen und Personalmanagement befassen. Aus dieser Perspektive bietet AWS CAF Leitlinien für Personalentwicklung, Schulung und Kommunikation, um das Unternehmen auf eine erfolgreiche Cloud-Einführung vorzubereiten. Weitere Informationen finden Sie auf der [AWS -CAF-Webseite](#) und dem [AWS -CAF-Whitepaper](#).

AWS Workload-Qualifizierungsrahmen (AWS WQF)

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

B

schlechter Bot

Ein [Bot](#), der Einzelpersonen oder Organisationen stören oder ihnen Schaden zufügen soll.

BCP

Siehe [Planung der Geschäftskontinuität](#).

Verhaltensdiagramm

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

Big-Endian-System

Ein System, welches das höchstwertige Byte zuerst speichert. Siehe auch [Endianness](#).

Binäre Klassifikation

Ein Prozess, der ein binäres Ergebnis vorhersagt (eine von zwei möglichen Klassen). Beispielsweise könnte Ihr ML-Modell möglicherweise Probleme wie „Handelt es sich bei dieser

E-Mail um Spam oder nicht?“ vorhersagen müssen oder „Ist dieses Produkt ein Buch oder ein Auto?“

Bloom-Filter

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

Blau/Grün-Bereitstellung

Eine Bereitstellungsstrategie, bei der Sie zwei separate, aber identische Umgebungen erstellen. Sie führen die aktuelle Anwendungsversion in einer Umgebung (blau) und die neue Anwendungsversion in der anderen Umgebung (grün) aus. Mit dieser Strategie können Sie schnell und mit minimalen Auswirkungen ein Rollback durchführen.

Bot

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

Botnetz

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

branch

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

Zugang durch Glasbruch

Unter außergewöhnlichen Umständen und im Rahmen eines genehmigten Verfahrens ist dies eine schnelle Methode für einen Benutzer, auf einen Bereich zuzugreifen AWS-Konto , für den

er in der Regel keine Zugriffsrechte besitzt. Weitere Informationen finden Sie unter dem Indikator [Implementation break-glass procedures](#) in den AWS Well-Architected-Leitlinien.

Brownfield-Strategie

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

Puffer-Cache

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

Geschäftsfähigkeit

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

Planung der Geschäftskontinuität (BCP)

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

C

CAF

Weitere Informationen finden Sie unter [Framework für die AWS Cloud-Einführung](#).

Bereitstellung auf Kanaren

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

CCoE

Weitere Informationen finden Sie [im Cloud Center of Excellence](#).

CDC

Siehe [Erfassung von Änderungsdaten](#).

Erfassung von Datenänderungen (CDC)

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

Chaos-Technik

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

CI/CD

Siehe [Continuous Integration und Continuous Delivery](#).

Klassifizierung

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

clientseitige Verschlüsselung

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

Cloud-Kompetenzzentrum (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 normalerweise durchlaufen, wenn sie zur AWS Cloud migrieren:

- Projekt – Durchführung einiger Cloud-bezogener Projekte zu Machbarkeitsnachweisen und zu Lernzwecken
- Fundament – Grundlegende Investitionen tätigen, um Ihre Cloud-Einführung zu skalieren (z. B. Einrichtung einer Landing Zone, Definition eines CCoE, 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-Repositories gehören GitHub oder AWS CodeCommit. 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. AWS Panorama Bietet beispielsweise Geräte an, die CV zu lokalen Kameranetzwerken hinzufügen, und Amazon SageMaker stellt Bildverarbeitungsalgorithmen für CV bereit.

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 wird allgemein als Pipeline beschrieben. 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, mit denen sichergestellt werden kann, dass nur vertrauenswürdige Identitäten auf vertrauenswürdige Ressourcen von erwarteten Netzwerken zugreifen. Weitere Informationen finden Sie unter [Aufbau eines Datenperimeters](#) auf AWS

Vorverarbeitung der Daten

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

Herkunft der Daten

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

betreffene Person

Eine Person, deren Daten gesammelt und verarbeitet werden.

Data Warehouse

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

Datenbankdefinitionssprache (DDL)

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

Datenbankmanipulationssprache (DML)

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

DDL

Siehe [Datenbankdefinitionssprache](#).

Deep-Ensemble

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

Deep Learning

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

defense-in-depth

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

delegierter Administrator

Ein kompatibler Dienst kann ein AWS Mitgliedskonto registrieren AWS Organizations, um die Konten der Organisation zu verwalten und die Berechtigungen für diesen Dienst zu verwalten. Dieses Konto wird als delegierter Administrator für diesen Service bezeichnet. Weitere Informationen und eine Liste kompatibler Services finden Sie unter [Services, die mit AWS Organizations funktionieren](#) in der AWS Organizations -Dokumentation.

Bereitstellung

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

Entwicklungsumgebung

Siehe [Umgebung](#).

Detektivische Kontrolle

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

Abbildung des Wertstroms in der Entwicklung (DVSM)

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

digitaler Zwilling

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

Maßtabelle

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

Katastrophe

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

Disaster Recovery (DR)

Die Strategie und der Prozess, mit denen Sie Ausfallzeiten und Datenverluste aufgrund einer [Katastrophe](#) 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 der Wertströme in der Entwicklung](#).

E

EDA

Siehe [explorative Datenanalyse](#).

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.

Verschlüsselung

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

Verschlüsselungsschlüssel

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

Endianismus

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

Endpunkt

[Siehe](#) Service-Endpunkt.

Endpunkt-Services

Ein Service, den Sie in einer Virtual Private Cloud (VPC) hosten können, um ihn mit anderen Benutzern zu teilen. Sie können einen Endpunktdienst mit anderen AWS-Konten oder AWS Identity and Access Management (IAM AWS PrivateLink -) Prinzipalen erstellen und diesen Berechtigungen gewähren. Diese Konten oder Prinzipale können sich privat mit Ihrem Endpunktservice verbinden, indem sie Schnittstellen-VPC-Endpunkte erstellen. Weitere Informationen finden Sie unter [Einen Endpunkt-Service erstellen](#) in der Amazon Virtual Private Cloud (Amazon VPC)-Dokumentation.

Unternehmensressourcenplanung (ERP)

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

Envelope-Verschlüsselung

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

Umgebung

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

- **Entwicklungs Umgebung** – Eine Instance einer laufenden Anwendung, die nur dem Kernteam zur Verfügung steht, das für die Wartung der Anwendung verantwortlich ist. Entwicklungs Umgebungen 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 Entwicklungs Umgebungen 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.

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.

G

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.

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 dabei hilft, Ressourcen, Richtlinien und die Einhaltung von Vorschriften in allen Organisationseinheiten (OUs) zu regeln. 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, eine 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.

Homogene Datenbankmigration

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

heiße Daten

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

Hotfix

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

Hypercare-Phase

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

I

IaC

Sehen Sie [Infrastruktur als Code](#).

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.

IloT

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 -Referenzarchitektur für die Sicherheit](#) empfiehlt, Ihr Netzwerkkonto mit eingehenden und ausgehenden VPCs und Inspektions-VPCs einzurichten, um die bidirektionale Schnittstelle zwischen Ihrer Anwendung und dem Internet 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 (IIoT)

Einsatz von mit dem Internet verbundenen Sensoren und Geräten in Industriesektoren wie Fertigung, Energie, Automobilindustrie, Gesundheitswesen, Biowissenschaften und Landwirtschaft. Mehr 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 derselben oder unterschiedlichen AWS-Regionen), dem Internet und lokalen Netzwerken verwaltet. Die [AWS -Referenzarchitektur für die Sicherheit](#) empfiehlt, Ihr Netzwerkkonto mit eingehenden und ausgehenden VPCs und Inspektions-VPCs einzurichten, um die bidirektionale Schnittstelle zwischen Ihrer Anwendung und dem Internet zu schützen.

Internet of Things (IoT)

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

Interpretierbarkeit

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

IoT

[Siehe Internet der Dinge.](#)

IT information library (ITIL, IT-Informationsbibliothek)

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

T service management (ITSM, IT-Service-Management)

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

BIS

Siehe [IT-Informationsbibliothek](#).

ITSM

Siehe [IT-Service-Management](#).

L

Labelbasierte Zugangskontrolle (LBAC)

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

Landing Zone

Eine landing zone ist eine gut strukturierte AWS Umgebung mit mehreren Konten, die skalierbar und sicher ist. Dies ist ein Ausgangspunkt, von dem aus Ihre Organisationen Workloads und Anwendungen schnell und mit Vertrauen in ihre Sicherheits- und Infrastrukturmgebung starten und bereitstellen können. Weitere Informationen zu Landing Zones finden Sie unter [Einrichtung einer sicheren und skalierbaren AWS -Umgebung mit mehreren Konten.](#)

Große Migration

Eine Migration von 300 oder mehr Servern.

SCHWARZ

Siehe [Labelbasierte 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](#).

Niedrigere Umgebungen

[Siehe Umwelt](#).

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 Service, der über klar definierte APIs kommuniziert 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 über eine klar definierte Schnittstelle mithilfe einfacher 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 [Implementieren 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 in die Cloud bereitstellt. AWS 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 wird, um einen Workload in die AWS Cloud zu migrieren. Weitere Informationen finden Sie im Eintrag [7 Rs](#) in diesem Glossar und unter [Mobilisieren Sie Ihr Unternehmen, um groß angelegte 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 [Bewertung 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](#).

EICHE

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.

ODER

Siehe [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 -Referenzarchitektur für die Sicherheit](#) empfiehlt, Ihr Netzwerkkonto mit eingehenden und ausgehenden VPCs und Inspektions-VPCs einzurichten, um die bidirektionale Schnittstelle zwischen Ihrer Anwendung und dem Internet zu schützen.

P

Berechtigungsgrenze

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

persönlich identifizierbare Informationen (PII)

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

Personenbezogene Daten

Siehe [persönlich identifizierbare Informationen](#).

Playbook

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

PLC

Siehe [programmierbare Logiksteuerung](#).

PLM

Siehe [Produktlebenszyklusmanagement](#).

policy

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

Polyglotte Beharrlichkeit

Unabhängige Auswahl der Datenspeichertechnologie eines Microservices auf der Grundlage von Datenzugriffsmustern und anderen Anforderungen. Wenn Ihre Microservices über dieselbe Datenspeichertechnologie verfügen, kann dies zu Implementierungsproblemen oder zu Leistungseinbußen führen. Microservices lassen sich leichter implementieren und erzielen eine bessere Leistung und Skalierbarkeit, wenn sie den Datenspeicher verwenden, der ihren Anforderungen am besten entspricht. Weitere Informationen finden Sie unter [Datenpersistenz in Microservices aktivieren](#).

Portfoliobewertung

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

predicate

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

Prädikat Pushdown

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

Präventive Kontrolle

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

Prinzipal

Eine Entität AWS, die Aktionen ausführen und auf Ressourcen zugreifen kann. Bei dieser Entität handelt es sich in der Regel um einen Root-Benutzer für eine AWS-Konto, eine IAM-Rolle oder

einen Benutzer. Weitere Informationen finden Sie unter Prinzipal in [Rollenbegriffe und -konzepte](#) in der IAM-Dokumentation.

Datenschutz durch Design

Ein Ansatz in der Systemtechnik, der den Datenschutz während des gesamten Engineering-Prozesses berücksichtigt.

Privat gehostete Zonen

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

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.

veröffentlichen/abonnieren (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 \(RACI\)](#).

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. Dies bestimmt, was als akzeptabler Datenverlust zwischen dem letzten Wiederherstellungspunkt und der Betriebsunterbrechung angesehen wird.

Ziel der Wiederherstellungszeit (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 für alle Parteien definiert, die an Migrationsaktivitäten und Cloud-Vorgängen beteiligt sind. 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.](#)

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 [Secret](#) in der Secrets Manager-Dokumentation.

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 Amazon EC2 EC2-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 Kontrolle über die Berechtigungen für alle Konten in einer Organisation in AWS Organizations ermöglicht. SCPs definieren Integritätsschutz oder legen Grenzwerte für Aktionen fest, die ein Administrator an Benutzer oder Rollen delegieren kann. Sie können SCPs als Zulassungs- oder Ablehnungslisten verwenden, um festzulegen, welche Services 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 klargelegt 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, wohingegen 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.

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 Transit-Gateway ist ein Netzwerk-Transit-Hub, mit dem Sie Ihre VPCs und On-Premises-Netzwerke miteinander verbinden können. Weitere Informationen finden Sie in der AWS Transit Gateway Dokumentation unter [Was ist ein 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, mit der Sie den Datenverkehr mithilfe von privaten IP-Adressen weiterleiten können. Weitere Informationen finden Sie unter [Was ist VPC-Peering?](#) in der Amazon-VPC-Dokumentation.

Schwachstelle

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

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

[Mal schreiben, viele lesen.](#)

WQF

Weitere Informationen finden Sie unter [AWS Workload Qualification Framework.](#)

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 wird als [unveränderlich](#) angesehen.

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.

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.