



Migrieren von Microsoft SQL Server-Datenbanken in die Cloud AWS

# AWS Präskriptive Leitlinien



# AWS Präskriptive Leitlinien: Migrieren von Microsoft SQL Server-Datenbanken in die Cloud AWS

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 irregeführt 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 .....	1
Migrationsstrategien .....	4
Auswahl der richtigen Migrationsstrategie .....	5
Online- und Offline-Migration .....	7
Methoden der Migration .....	8
Native SQL Server-Sicherung/Wiederherstellung .....	16
Versand protokollieren .....	18
Datenbankspiegelung .....	19
AlwaysOn-Verfügbarkeitsgruppen .....	20
Verteilte Verfügbarkeitsgruppen .....	21
Transaktionsreplikation .....	23
AWS Migration Hub Orchestrator .....	24
AWS Snowball Edge .....	26
Homogene Datenbankmigration .....	27
Amazon RDS für SQL Server .....	28
Wann sollten Sie sich für Amazon RDS entscheiden .....	29
Hohe Verfügbarkeit .....	30
Read Replicas .....	32
Notfallwiederherstellung .....	34
Amazon RDS Custom für SQL Server .....	35
Wann sollten Sie Amazon RDS Custom für SQL Server wählen .....	35
Funktionsweise .....	35
Amazon EC2 für SQL Server .....	39
Wann sollten Sie sich für Amazon EC2 entscheiden .....	39
Hohe Verfügbarkeit .....	40
Notfallwiederherstellung .....	49
VMware Cloud on AWS für SQL Server .....	50
Wann sollten Sie sich für VMware Cloud entscheiden AWS .....	51
Heterogene Datenbankmigration .....	52
Tools .....	54
AWS SCT .....	54
AWS DMS .....	55
Babelfish .....	55

Hybride Migration Szenarien .....	58
Sichern Sie Ihre Datenbanken in der Cloud .....	58
Erweitern der Lösungen für hohe Verfügbarkeit und Notfallwiederherstellung .....	58
Storage Gateway .....	59
benutzenAWS DMSundAWS SCT .....	60
Modernisierung der SQL Server-Datenbank .....	61
Migration der SQL Server-Workloads von Windows nach Linux .....	61
Hohe Verfügbarkeit unter Linux .....	62
AWSStartassistent .....	63
Bewährte Methoden für die Migration zu Amazon RDS für SQL Server .....	67
Bereitstellung Ihrer Zieldatenbank .....	67
Aus Ihrer Quelldatenbank sichern .....	68
Übertragung von Datendump-Dateien an AWS .....	68
Daten in Ihrer Zieldatenbank wiederherstellen .....	68
Schritte nach der Migration .....	69
Die Migration testen .....	69
Betrieb und Optimierung Ihrer Amazon RDS-Datenbank .....	70
Auswählen zwischen Amazon EC2 und Amazon RDS .....	71
Entscheidungsmatrix .....	71
Gemeinsame Verantwortlichkeit .....	101
Migrationsmuster der SQL Server-Datenbank .....	103
Partner .....	104
Weitere Ressourcen .....	105
Danksagungen .....	106
Anhang: Fragebogen zur SQL Server-Datenbank-Migration .....	107
Allgemeine Informationen .....	107
Infrastruktur .....	108
Datenbank-Backups .....	108
Funktionen der Datenbank .....	108
Datenbanksicherheit .....	108
Hohe Verfügbarkeit der Datenbank und Notfallwiederherstellung .....	109
Dokumentverlauf .....	110
Glossar .....	113
# .....	113
A .....	114
B .....	117

---

C .....	119
D .....	123
E .....	127
F .....	129
G .....	131
H .....	132
I .....	133
L .....	136
M .....	137
O .....	141
P .....	144
Q .....	147
R .....	147
S .....	150
T .....	154
U .....	155
V .....	156
W .....	157
Z .....	158
.....	clix

# Migrieren von Microsoft SQL Server-Datenbanken in die Cloud AWS

Sagar Patel, Amazon Web Services (AWS)

April 2024 (Verlauf der [Dokumente](#))

Amazon Web Services (AWS) bietet eine umfassende Palette von Services und Tools für die Bereitstellung von Microsoft SQL Server-Datenbanken in der zuverlässigen und sicheren AWS Cloud-Infrastruktur. Zu den Vorteilen der Ausführung von SQL Server AWS gehören Kosteneinsparungen, Skalierbarkeit, hohe Verfügbarkeit und Notfallwiederherstellung, bessere Leistung und einfache Verwaltung. Weitere Informationen finden [Sie im AWS Compute-Blog unter Erfahren Sie, warum die beste Cloud für die Ausführung von Microsoft Windows Server- und SQL Server-Workloads AWS ist.](#)

In diesem Handbuch werden die verfügbaren Optionen für die Migration von SQL Server-Datenbanken von lokalen Datenbanken in die AWS Cloud, zu Amazon Relational Database Service (Amazon RDS), Amazon Elastic Compute Cloud (Amazon EC2) oder VMware Cloud on beschrieben. Es befasst sich mit den bewährten Methoden und Empfehlungen für die Verwendung dieser Migrationsoptionen. Es enthält auch Informationen zur Einrichtung einer Hochverfügbarkeits- und Notfallwiederherstellungslösung zwischen einer lokalen SQL Server-Umgebung und AWS mithilfe systemeigener SQL Server-Funktionen wie Protokollversand, Replikation und AlwaysOn-Verfügbarkeitsgruppen.

Dieses Handbuch richtet sich an Programm- oder Projektmanager, Produktbesitzer, Datenbankadministratoren, Datenbankingenieure sowie Betriebs- oder Infrastrukturmanager, die planen, ihre lokalen SQL Server-Datenbanken zu migrieren. AWS

## Übersicht

Bevor Sie Ihre SQL Server-Datenbanken zu migrieren AWS, sollten Sie Ihre Migrationsstrategie verstehen und bewerten, indem Sie das unter [Migrationsstrategie für relationale](#) Datenbanken beschriebene Framework verwenden.

Der erste Schritt besteht darin, eine Analyse Ihrer Anwendungs- und SQL Server-Datenbank-Workloads durchzuführen, indem Sie sich mit der Komplexität, Kompatibilität und den Kosten der Migration vertraut machen. Hier sind einige der wichtigsten Punkte, die Sie bei der Planung einer Migration berücksichtigen sollten:

- **Datenbankgröße** — Überprüfen Sie die aktuelle Größe und das allgemeine Kapazitätswachstum Ihrer Datenbank. Wenn Sie beispielsweise planen, Ihre SQL Server-Datenbank zu Amazon RDS oder Amazon RDS Custom zu migrieren, können Sie DB-Instances mit bis zu 16 TiB Speicher erstellen. Sie können mehr Speicherplatz anfordern, indem Sie [ein Support-Ticket beim AWS Support öffnen](#). Aktuelle Informationen finden Sie unter [Amazon RDS-DB-Instance-Speicher](#) in der Amazon RDS-Dokumentation.
- **IOPS** — Ermitteln Sie die IOPS und den Durchsatz Ihrer Datenbanken. Wenn Sie eine Migration zu Amazon RDS planen, sollten Sie die [I/O-Leistung der Amazon RDS-DB-Instances](#) berücksichtigen.
- **Abhängigkeiten** — Überprüfen Sie die aktuellen Datenbankabhängigkeiten. Wenn Ihre Datenbank von anderen Datenbanken abhängig ist, können Sie sie entweder zusammen migrieren oder nach der Migration Ihrer Hauptdatenbank Abhängigkeiten erstellen.

Wenn Ihre Datenbank ältere, benutzerdefinierte oder verpackte Anwendungen unterstützt, ist Amazon RDS Custom for SQL Server möglicherweise eine gute Wahl. Mit diesem Service behalten Sie die Kontrolle über Datenbankkonfigurationen, gemeinsam genutzte Dateisysteme und Betriebssystem-Patches.

Inventarisieren Sie alle SQL Server-Abhängigkeiten. Finden Sie heraus, welche Webserver (z. B. Berichtsserver oder Business Intelligence-Server) eine Schnittstelle zu SQL Server haben. Wenn es Zeit für die Migration ist, können Sie anhand dieser Informationen bestimmen, welche Auswirkungen davon betroffen sein werden und wie Sie die Auswirkungen minimieren können.

- **Compliance** — Überprüfen Sie Ihre aktuellen Architektur- und Audit- oder Compliance-Anforderungen, um sicherzustellen, dass Sie diese Anforderungen auch nach der Umstellung auf Amazon RDS oder Amazon EC2 erfüllen können.
- **HA/DR** — Benötigen Sie Hochverfügbarkeit (HA) und automatisierte Failover-Funktionen? Wenn Sie einen Produktions-Workload ausführen, werden Hochverfügbarkeit und Disaster Recovery (DR) als bewährte Methoden empfohlen.

Machen Sie sich mit Ihren HA/DR-Anforderungen vertraut, um festzustellen, ob Sie eine Architektur mit mehreren Regionen benötigen. Wenn ja, migrieren Sie Ihre SQL Server-Datenbank zu Amazon EC2. Amazon RDS unterstützt keine Konfiguration mit mehreren Regionen.

- **Versionsunterstützung** — Überprüfen Sie die Version und Edition Ihrer SQL Server-Software, wenn Sie planen, zu Amazon RDS for SQL Server zu wechseln (siehe aktuell unterstützte Versionen für [Amazon RDS](#) und [Amazon RDS](#)).

- **Netzwerkonnektivität** — Überprüfen Sie die Netzwerkonnektivität zwischen Ihrer lokalen Umgebung und stellen Sie sicher AWS, dass sie genügend Bandbreite für schnelle Datenübertragungen zwischen Ihrer lokalen Umgebung und AWS bietet.
- **Ausfallzeit bei der Migration** — Ermitteln Sie, wie viele Ausfallzeiten für die Migration zur Verfügung stehen, damit Sie Ihren Migrationsansatz planen und entscheiden können, ob Sie die Online- oder Offline-Migration verwenden möchten.
- **RTO-, RPO- und SLA-Anforderungen** — Identifizieren Sie Ihre Anforderungen an das Recovery Time Objective (RTO), das Recovery Point Objective (RPO) und das Service Level Agreement (SLA) für Ihre vorhandenen Datenbank-Workloads.
- **Lizenzierung** — Machen Sie sich mit Ihren Lizenzoptionen vertraut. Sie können bei Amazon EC2 und Amazon RDS zwischen Optionen inklusive Lizenz wählen oder [Ihre eigene Lizenz \(BYOL\) für Amazon EC2 mitbringen](#).
- **Funktionsunterstützung** — Identifizieren Sie die Datenbankfunktionen und -funktionen, die Ihre Anwendung verwendet, unabhängig davon, ob sie intern entwickelt wurde oder ob es sich um Software (COTS) handelt. commercial-off-the-shelf Anhand dieser Informationen können Sie feststellen, ob Sie Ihre Lizenzkosten senken können, indem Sie von der SQL Server Enterprise Edition zur Standard Edition wechseln. Informieren Sie sich jedoch vor dem Wechsel über die Ressourcenbeschränkungen der Standard Edition. Die Standard Edition unterstützt beispielsweise nur 128 GB RAM.

Passt Ihr Workload zu den Funktionen und Fähigkeiten, die Amazon RDS for SQL Server bietet? Weitere Informationen finden Sie unter [SQL Server-Funktionen auf Amazon RDS](#). Wenn Sie Funktionen benötigen, die nicht unterstützt werden, ist die Migration zu Amazon EC2 eine Option.

# Strategien zur Migration von SQL Server-Datenbanken

Generell gibt es zwei Optionen für die Migration einer SQL Server-Datenbank von einer lokalen Datenbank in die AWS Cloud: Entweder bleiben Sie auf dem SQL Server ([homogene Migration](#)) oder Sie verlassen SQL Server ([heterogene Migration](#)). Bei einer homogenen Migration ändern Sie die Datenbank-Engine nicht. Das heißt, Ihre Zieldatenbank ist auch eine SQL Server-Datenbank. Bei einer heterogenen Migration stellen Sie Ihre SQL Server-Datenbanken entweder auf eine Open-Source-Datenbank-Engine wie MySQL, PostgreSQL oder MariaDB oder auf eine AWS cloudnative Datenbank wie Amazon Aurora, Amazon DynamoDB oder Amazon Redshift um.

Es gibt drei gängige Strategien für die Migration Ihrer SQL Server-Datenbanken AWS: Rehost, Replatform und Re-Architect (Refactor). Diese sind Teil der [7 Rs der Strategien für die Anwendungsmigration](#) und werden in der folgenden Tabelle beschrieben.

Strategie	Typ	Wann soll ich wählen	Beispiel
Erneut hosten	Homogen	Sie möchten Ihre SQL Server-Datenbank unverändert migrieren, mit oder ohne Änderung des Betriebssystems, der Datenbanksoftware oder der Konfiguration.	SQL Server zu Amazon EC2  (Durchsuchen Sie <a href="#">die Rehost-Muster</a> )
Wechsel der Plattform	Homogen	Sie möchten den Zeitaufwand für die Verwaltung von Datenbankinstanzen reduzieren, indem Sie ein vollständig verwaltetes Datenbankangebot verwenden.	SQL Server zu Amazon RDS für SQL Server  (Stöbern Sie in <a href="#">den Replatform-Mustern</a> )

Strategie	Typ	Wann soll ich wählen	Beispiel
Re-Architect (Refactoring)	Heterogen	Sie möchten Ihre Datenbank und Anwendung umstrukturieren, neu schreiben und neu gestalten, um die Vorteile von Open-Source- und cloudnativen Datenbankfunktionen nutzen zu können.	SQL Server zu Amazon Aurora PostgreSQL, MySQL oder MariaDB  Suchen Sie nach <a href="#">Re-Architect-Mustern</a> )

Wenn Sie versuchen, sich zwischen einem Rehosting oder einer Plattformneuerung Ihrer SQL Server-Datenbanken zu entscheiden, finden Sie unter [Wählen zwischen Amazon EC2 und Amazon RDS](#) weiter unten in diesem Handbuch einen side-by-side Vergleich der unterstützten Funktionen.

## Auswahl der richtigen Migrationsstrategie

Die Wahl der richtigen Strategie hängt von Ihren Geschäftsanforderungen, Ressourcenbeschränkungen, dem Migrationszeitrahmen und Kostenüberlegungen ab. Das folgende Diagramm zeigt den Aufwand und die Komplexität, die mit Migrationen verbunden sind, einschließlich aller sieben Strategien.



Wenn Sie Ihre SQL Server-Datenbank umgestalten und zu einer Open-Source-oder AWS Cloud-nativen Datenbank wie Amazon Aurora PostgreSQL-Compatible Edition oder Aurora MySQL-Compatible Edition migrieren, können Sie Ihre Datenbank modernisieren und optimieren. Durch die Umstellung auf eine Open-Source-Datenbank können Sie teure Lizenzen (die zu niedrigeren Kosten führen), Anbieterbindungsfristen und Audits vermeiden. Abhängig von der Komplexität Ihres Workloads kann das Refactoring Ihrer SQL Server-Datenbank jedoch ein komplizierter, zeitaufwändiger und ressourcenintensiver Aufwand sein.

Um die Komplexität zu reduzieren, sollten Sie, anstatt Ihre Datenbank in einem einzigen Schritt zu migrieren, einen schrittweisen Ansatz in Betracht ziehen. In der ersten Phase können Sie sich auf die Kernfunktionen der Datenbank konzentrieren. In der nächsten Phase können Sie zusätzliche AWS Dienste in Ihre Cloud-Umgebung integrieren, um die Kosten zu senken und Leistung, Produktivität und Compliance zu optimieren. Wenn Ihr Ziel beispielsweise darin besteht, Ihre lokale SQL Server-Datenbank durch eine Aurora MySQL-kompatible Datenbank zu ersetzen, könnten Sie erwägen, in der ersten Phase Ihre Datenbank auf Amazon EC2 neu zu hosten oder Ihre Datenbank auf Amazon RDS for SQL Server umzustellen und dann in einer nachfolgenden Phase auf Aurora MySQL-kompatibel umzustellen. Dieser Ansatz trägt zur Reduzierung von Kosten, Ressourcen und Risiken während der Migrationsphase bei und konzentriert sich in der zweiten Phase auf die Optimierung und Modernisierung.

## Online- und Offline-Migration

Sie können zwei Methoden verwenden, um Ihre SQL Server-Datenbank von einer lokalen oder einer anderen Cloud-Umgebung in die AWS Cloud zu migrieren, basierend auf Ihrem Migrationszeitplan und den zulässigen Ausfallzeiten: Offline-Migration oder Online-Migration.

- **Offline-Migration:** Diese Methode wird verwendet, wenn sich Ihre Anwendung eine geplante Ausfallzeit leisten kann. Bei der Offline-Migration ist die Quelldatenbank während des Migrationszeitraums offline. Während die Quelldatenbank offline ist, wird sie auf AWS die Zieldatenbank migriert. Nach Abschluss der Migration werden Validierungs- und Verifizierungsprüfungen durchgeführt, um die Datenkonsistenz mit der Quelldatenbank sicherzustellen. Wenn die Datenbank alle Validierungsprüfungen bestanden hat, führen Sie eine Umstellung auf AWS durch, indem Sie Ihre Anwendung mit der Zieldatenbank verbinden.
- **Online-Migration:** Diese Methode wird verwendet, wenn für Ihre Anwendung nahezu keine oder nur minimale Ausfallzeiten erforderlich sind. Bei der Online-Migration wird die Quelldatenbank in mehreren Schritten zu AWS migriert. In den ersten Schritten werden die Daten in der Quelldatenbank in die Zieldatenbank kopiert, während die Quelldatenbank noch läuft. In den nachfolgenden Schritten werden alle Änderungen aus der Quelldatenbank an die Zieldatenbank weitergegeben. Wenn die Quell- und Zieldatenbanken synchronisiert sind, sind sie für die Umstellung bereit. Während der Umstellung schaltet die Anwendung ihre Verbindungen zur Zieldatenbank ein, sodass keine Verbindungen zur Quelldatenbank bestehen. Sie können AWS Database Migration Service (AWS DMS) oder Tools verwenden, die von [AWS Marketplace](#) (wie Attunity) verfügbar sind, um die Quell- und Zieldatenbanken zu synchronisieren.

# Methoden für die SQL Server-Datenbankmigration

Es gibt verschiedene Methoden, um Ihre SQL Server-Datenbanken zu AWS migrieren. Sie können je nach Einschätzung und Anforderungen zwischen AWS Diensten und systemeigenen Funktionen von SQL Server wählen. In diesem Abschnitt werden einige der gängigsten Methoden beschrieben, die in den folgenden beiden Tabellen zusammengefasst sind. Ausführliche Erläuterungen einiger dieser Methoden finden Sie in den Abschnitten zu Amazon EC2 und Amazon RDS weiter unten in diesem Handbuch.

## AWS-Services

Methode der Migration	Ziel	Funktionen und Einschränkungen	Weitere Informationen
AWS DMS	Amazon EC2	<ul style="list-style-type: none"> <li>• Unterstützt Volllast und CDC</li> <li>• Unterstützt alle Datenbankgrößen</li> </ul>	<a href="#">AWS DMS</a> Abschnitt
	Amazon RDS		
	Amazon RDS Custom		
	Amazon Aurora		
AWS Migration Hub Orchestrator	Amazon EC2 Amazon RDS	<ul style="list-style-type: none"> <li>• Stellt vordefinierte step-by-step Workflow-Vorlagen bereit</li> <li>• Automatisiert systemeigene Backups und Wiederherstellungen</li> <li>• Unterstützt alle SQL Server-Editionen und -Versionen</li> <li>• Kann auf eine oder mehrere Datenbanken gleichzeitig</li> </ul>	<a href="#">AWS Migration Hub Orchestrator</a> Abschnitt

Methode der Migration	Ziel	Funktionen und Einschränkungen	Weitere Informationen
		angewendet werden <ul style="list-style-type: none"> <li>• Unterstützt alle Datenbankgrößen</li> </ul>	
AWS Application Migration Service	Amazon EC2	<ul style="list-style-type: none"> <li>• Hochautomatisierte lift-and-shift Lösung</li> <li>• Agentenbasierte Replikation auf Blockebene</li> </ul>	In diesem Handbuch nicht behandelt (siehe <a href="#">Dokumentation zum Application Migration Service</a> )
AWS Snowball Edge	Amazon EC2 Amazon RDS Amazon RDS Custom	<ul style="list-style-type: none"> <li>• Unterstützt sehr große Datenbanken (bis zu 210 TB)</li> <li>• Verwendet Amazon Simple Storage Service (Amazon S3) zum Speichern und Wiederherstellen von Daten</li> </ul>	<a href="#">Bereich Snowball Edge</a>

### Systemeigene Methoden von SQL Server

Methode der Migration	Ziel	Funktionen und Einschränkungen	Weitere Informationen
Systemeigene Sicherung und Wiederherstellung	Amazon EC2 Amazon RDS Amazon RDS Custom	<ul style="list-style-type: none"> <li>• Kann auf eine oder mehrere Datenbanken gleichzeitig angewendet werden</li> <li>• Erfordert Ausfallzeiten</li> </ul>	Abschnitt zum <a href="#">Sichern/Wiederherstellen von systemeigenem SQL Server</a> (Sie können ihn verwenden, um systemeigene

Methode der Migration	Ziel	Funktionen und Einschränkungen	Weitere Informationen
		<ul style="list-style-type: none"> <li>• Unterstützt alle Datenbankgrößen</li> </ul>	Sicherungen und Wiederherstellungen ( <a href="#">AWS Migration Hub Orchestrator</a> zu automatisieren)
Versand protokollieren	Amazon EC2 Amazon RDS Amazon RDS Custom	<ul style="list-style-type: none"> <li>• Wird pro Datenbank angewendet</li> <li>• Kann verzögert werden</li> </ul>	Abschnitt „ <a href="#">Versand protokollieren</a> “
Spiegelung von Datenbanken	Amazon EC2	<ul style="list-style-type: none"> <li>• Wird pro Datenbank angewendet</li> <li>• Kann synchron oder asynchron sein, basierend auf der SQL Server-Edition</li> <li>• Die sekundäre Datenbank ist nicht lesbar; sie fungiert als Standby-Datenbank</li> <li>• Unterstützt sowohl automatisches als auch manuelles Failover</li> </ul>	Abschnitt zur <a href="#">Datenbankspiegelung</a>

Methode der Migration	Ziel	Funktionen und Einschränkungen	Weitere Informationen
AlwaysOn-Verfügbarkeitsgruppen	Amazon EC2 Amazon RDS Custom	<ul style="list-style-type: none"><li>• Wird auf eine Reihe von Benutzerdatenbanken angewendet</li><li>• Kann synchron oder asynchron sein</li><li>• Die sekundäre Datenbank ist lesbar (nur SQL Server Enterprise Edition)</li><li>• Unterstützt sowohl automatisches als auch manuelles Failover</li><li>• Ein Failover kann für mehrere Datenbanken gleichzeitig auf Datenbankgruppenebene initiiert werden</li></ul>	Abschnitt „ <a href="#">AlwaysOn-Verfügbarkeitsgruppen</a> “

Methode der Migration	Ziel	Funktionen und Einschränkungen	Weitere Informationen
Grundlegende AlwaysOn-Verfügbarkeitsgruppen	Amazon EC2 Amazon RDS Custom	<ul style="list-style-type: none"><li>• Wird in der SQL Server Standard Edition unterstützt</li><li>• Wird auf eine einzelne Benutzerdatenbank pro Verfügbarkeitsgruppe angewendet</li><li>• Kann synchron oder asynchron sein</li><li>• Unterstützt sowohl automatisches als auch manuelles Failover</li><li>• Ein Failover kann auf der Ebene der Verfügbarkeitsgruppen initiiert werden</li><li>• Kann als Hybridumgebung zwischen lokalen und AWS</li></ul>	In diesem Handbuch nicht behandelt (siehe <a href="#">Basic Always On-Verfügbarkeitsgruppen für eine einzelne Datenbank</a> in der Microsoft-Dokumentation)

Methode der Migration	Ziel	Funktionen und Einschränkungen	Weitere Informationen
Verteilte Verfügbarkeitsgruppen	Amazon EC2 Amazon RDS Custom	<ul style="list-style-type: none"><li>• Kann für SQL Server-Bereitstellungen in mehreren Regionen verwendet werden</li><li>• Ein Failover auf eine neuere Version von SQL Server ist möglich</li><li>• Windows Server Failover Clustering (WSFC) muss nicht auf die Zielumgebung ausgedehnt werden AWS</li><li>• Kann zwischen Windows-basierten (Quell-) und Linux-basierten (Ziel-) SQL Server-Datenbanken verwendet werden</li><li>• Kann als hybride SQL Server-Bereitstellung zwischen lokalen und AWS</li></ul>	Abschnitt „ <a href="#">Verteilte Verfügbarkeitsgruppen</a> “

Methode der Migration	Ziel	Funktionen und Einschränkungen	Weitere Informationen
Transaktionsreplikation	Amazon EC2 Amazon RDS Amazon RDS Custom	<ul style="list-style-type: none"><li>• Unterstützt die Migration einer Reihe von Objekten (Tabellen, Ansichten, gespeicherte Prozeduren)</li><li>• Unterstützt asynchrone Replikation mit Daten nahezu in Echtzeit</li><li>• Die Abonnementdatenbank ist lesbar</li><li>• Erfordert eine genaue Überwachung der SQL Server-Replikation saufträge, die die Replikation durchführen</li></ul>	<a href="#">Abschnitt zur Transaktionsreplikation</a>

Methode der Migration	Ziel	Funktionen und Einschränkungen	Weitere Informationen
Programm zum Massenkopieren (bcp)	Amazon EC2 Amazon RDS Custom	<ul style="list-style-type: none"> <li>• Unterstützt kleine Datenbanken</li> <li>• Erfordert Ausfallzeiten</li> <li>• Das Schema ist am Ziel vorab erstellt</li> <li>• Wird für das Verschieben von Daten, aber nicht für Metadaten verwendet</li> </ul>	In diesem Handbuch nicht behandelt (siehe <a href="#">Import und Export von SQL Server-Daten mit anderen Methoden</a> , Abschnitt Massenkopie in der Amazon RDS-Dokumentation)
Trennen und anhängen	Amazon EC2 Amazon RDS Custom	<ul style="list-style-type: none"> <li>• Kein Backup erforderlich</li> <li>• Erfordert Ausfallzeiten</li> <li>• Beinhaltet das Stoppen, Trennen, Kopieren und Anhängen von Dateien an Amazon EC2</li> </ul>	In diesem Handbuch nicht behandelt (siehe <a href="#">Database Detach and Attach</a> in der Microsoft-Dokumentation)

Methode der Migration	Ziel	Funktionen und Einschränkungen	Weitere Informationen
Importieren/Exportieren	Amazon EC2 Amazon RDS Custom	<ul style="list-style-type: none"> <li>• Unterstützt kleine Datenbanken</li> <li>• Erfordert Ausfallzeiten</li> <li>• Das Schema ist am Ziel vorab erstellt</li> <li>• Wird für das Verschieben von Daten, aber nicht für Metadaten verwendet</li> </ul>	In diesem Handbuch nicht behandelt (siehe <a href="#">Importieren und Exportieren von SQL Server-Daten mit anderen Methoden</a> in der Amazon RDS-Dokumentation)

## Native SQL Server-Sicherung/Wiederherstellung

Amazon RDS unterstützt native Sicherungs- und Wiederherstellungsvorgänge für Microsoft SQL Server-Datenbanken mithilfe von vollständigen und differenziellen Sicherungsdateien (.bak). Es unterstützt auch differenzielle Rücksicherungs- und Protokollwiederherstellungsoptionen auf einer Amazon RDS for SQL Server-DB-Instance oder einer Amazon EC2 SQL Server-Instance, um Ausfallzeiten für Ihre Anwendung zu minimieren.

### Note

Sie können vollständige, differenzielle und Protokollwiederherstellungsvorgänge auf Amazon RDS for SQL Server durchführen. Derzeit können Sie jedoch nur vollständige und differenzielle Sicherungen (keine Protokollsicherungen) durchführen.

Die Verwendung systemeigener BAK-Dateien ist die einfachste Methode zum Sichern und Wiederherstellen von SQL Server-Datenbanken. Sie können diese Methode verwenden, um Datenbanken zu oder von Amazon RDS zu migrieren. Sie können einzelne Datenbanken anstelle ganzer DB-Instances sichern und wiederherstellen. Sie können Datenbanken auch zwischen Amazon RDS for SQL Server-DB-Instances verschieben.

Wenn Sie Amazon RDS verwenden, können Sie Sicherungsdateien in Amazon Simple Storage Service (Amazon S3) speichern und übertragen, um eine zusätzliche Schutzebene für die Notfallwiederherstellung zu schaffen. Beispiele:

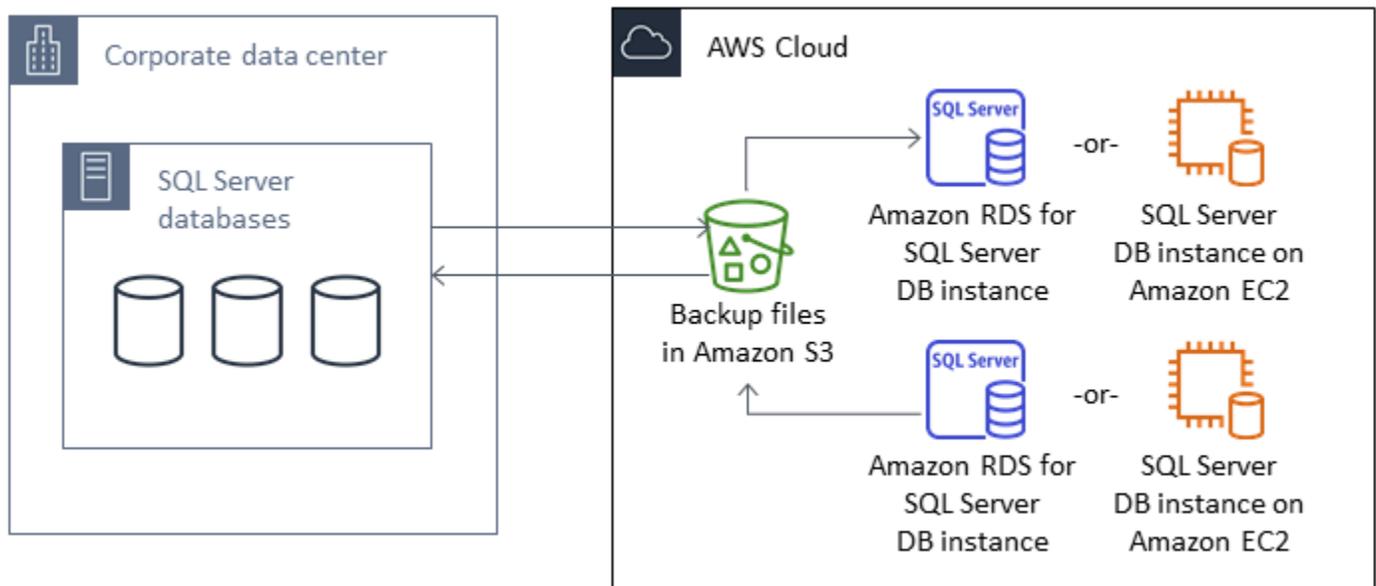
- Sie können ein vollständiges Backup Ihrer Datenbank von Ihrem lokalen Server aus erstellen, es in einen S3-Bucket kopieren und es dann auf einer vorhandenen Amazon RDS SQL Server-DB-Instance wiederherstellen.
- Sie können Backups von einer Amazon RDS for SQL Server-DB-Instance erstellen, sie in Amazon S3 speichern und sie dann wiederherstellen, wo immer Sie möchten.
- Sie können [Amazon S3 Lifecycle-Konfigurationsregeln](#) implementieren, um langfristige Backups zu archivieren oder zu löschen.

Amazon RDS for SQL Server unterstützt die Wiederherstellung nativer SQL Server-Backups auf SQL Server-DB-Instances, für die Read Replicas konfiguriert sind. Das bedeutet, dass Sie die Read Replica nicht entfernen müssen, bevor Sie die native Backup-Datei auf Ihrer Amazon RDS for SQL Server-DB-Instance wiederherstellen.

 Note

Sie können Migration Hub Orchestrator verwenden, um Ihre SQL Server-Datenbankmigrationen zu Amazon EC2 oder Amazon RDS mithilfe nativer Sicherung und Wiederherstellung zu automatisieren und zu orchestrieren. [Weitere Informationen finden Sie im Abschnitt. AWS Migration Hub Orchestrator](#)

Das folgende Diagramm zeigt den systemeigenen Sicherungs- und Wiederherstellungsprozess von SQL Server. Sie können den Migration Hub Orchestrator verwenden, um diesen Prozess zu automatisieren. Sie können diesen Prozess auch verwenden, um SQL Server-Datenbanken in Amazon EC2 zu sichern und wiederherzustellen.



Informationen zur Automatisierung von Backup und Wiederherstellung finden Sie in der [Migration Hub Orchestrator-Dokumentation](#).

Informationen zum Einrichten der systemeigenen Sicherung/Wiederherstellung mit Amazon S3 finden Sie in der [Amazon RDS-Dokumentation](#).

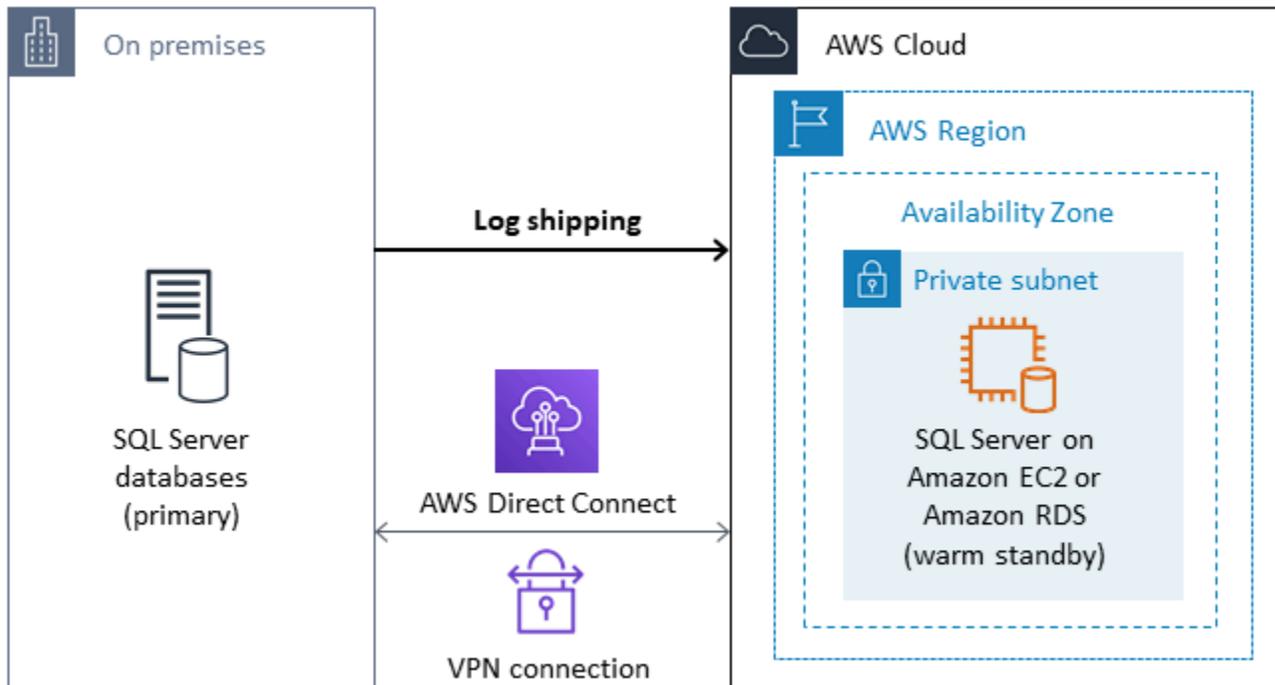
Informationen zu Einschränkungen bei der Verwendung der systemeigenen Sicherung und Wiederherstellung von SQL Server finden Sie unter [Einschränkungen und Empfehlungen](#) in der Amazon RDS-Dokumentation.

## Versand protokollieren

Sie können den Protokollversand verwenden, um Transaktionsprotokollsicherungen von Ihrer primären, lokalen SQL Server-Datenbank an eine oder mehrere sekundäre (Warm-Standby) SQL Server-Datenbanken zu senden, die auf EC2-Instances oder Amazon RDS for SQL Server Server-DB-Instances in der AWS Cloud bereitgestellt werden. Um den Protokollversand auf Amazon RDS for SQL Server einzurichten, müssen Sie Ihre eigenen benutzerdefinierten Skripts verwenden.

In diesem Szenario konfigurieren Sie eine Warm-Standby-SQL Server-Datenbank auf einer EC2-Instance oder einer Amazon RDS for SQL Server-DB-Instance und senden Transaktionsprotokoll-Backups asynchron zwischen Ihrer lokalen Datenbank und dem Warm-Standby-Server in der Cloud. Die Transaktionsprotokoll-Backups werden dann auf die Warm-Standby-Datenbank angewendet. Wenn alle Protokolle angewendet wurden, können Sie ein manuelles Failover durchführen und zur Cloud wechseln.

Diese Option unterstützt alle Versionen und Editionen von SQL Server. Nachdem Sie die Datenbank in die AWS Cloud migriert haben, können Sie ein sekundäres Replikat hinzufügen, indem Sie eine Always-On-Verfügbarkeitsgruppe verwenden, um hohe Verfügbarkeit und Ausfallsicherheit zu gewährleisten.



Weitere Informationen zur Verwendung dieser Methode, um Hochverfügbarkeit, Datenschutz und Notfallwiederherstellung für Ihre SQL Server-Datenbanken auf Amazon EC2 zu erreichen, finden Sie unter [Protokollversand](#) im Abschnitt Amazon EC2 für SQL Server.

## Datenbankspiegelung

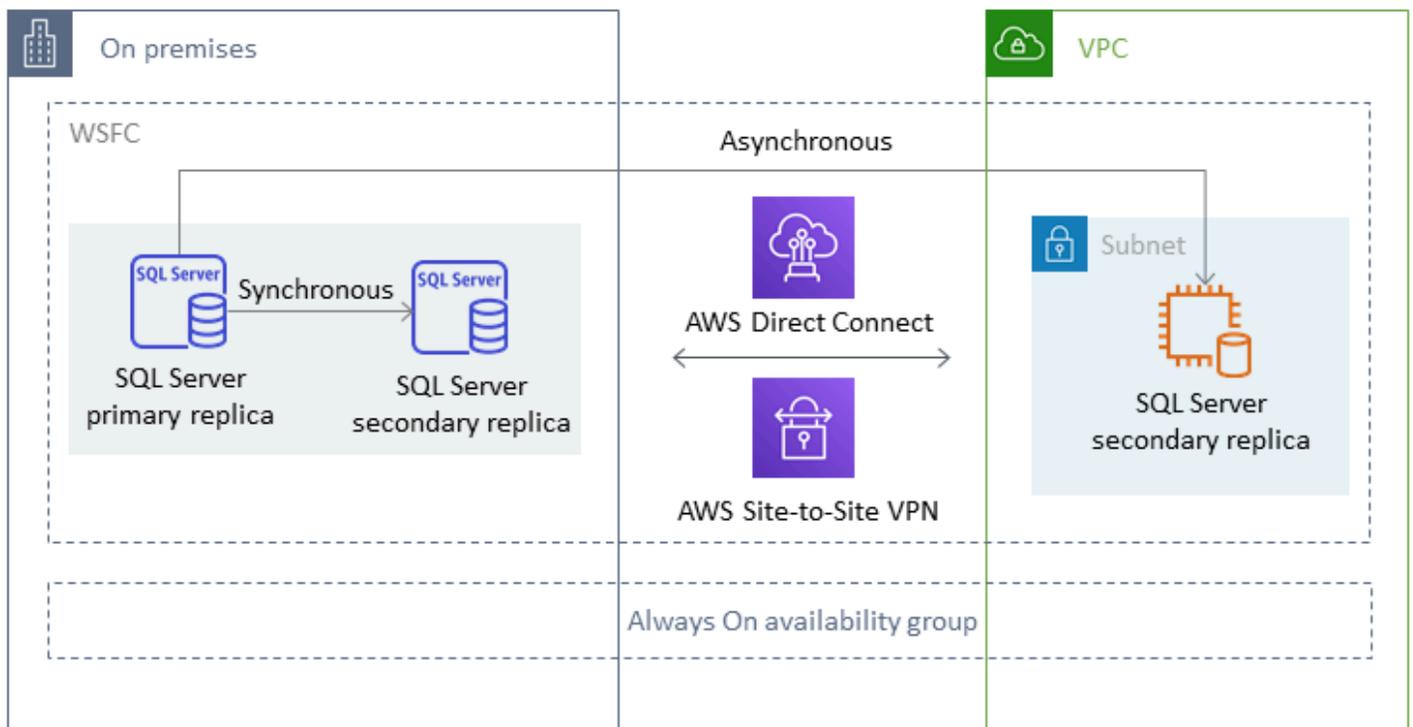
Sie können die Datenbankspiegelung verwenden, um eine Hybrid-Cloud-Umgebung für Ihre SQL Server-Datenbanken einzurichten. Für diese Option ist die SQL Server Enterprise Edition erforderlich. In diesem Szenario wird Ihre SQL Server-Prinzipaldatenbank lokal ausgeführt, und Sie erstellen eine Warm-Standby-Datenbank in der Cloud. Sie replizieren Ihre Daten asynchron und führen ein manuelles Failover durch, wenn Sie für die Umstellung bereit sind. Nachdem Sie die Datenbank in die AWS Cloud migriert haben, können Sie aus Gründen der Hochverfügbarkeit und Ausfallsicherheit ein sekundäres Replikat hinzufügen, indem Sie eine Always-On-Verfügbarkeitsgruppe verwenden.

Weitere Informationen zur Verwendung dieser Methode, um Hochverfügbarkeit, Datenschutz und Notfallwiederherstellung für Ihre SQL Server-Datenbanken auf Amazon EC2 zu erreichen, finden Sie unter [Datenbankspiegelung](#) im Abschnitt Amazon EC2 für SQL Server.

# AlwaysOn-Verfügbarkeitsgruppen

Bei SQL Server AlwaysOn-Verfügbarkeitsgruppen handelt es sich um ein erweitertes Feature auf Unternehmensebene, das Hochverfügbarkeits- und Notfallwiederherstellungslösungen bereitstellt. Diese Funktion ist verfügbar, wenn Sie SQL Server 2014 und höhere Versionen verwenden. Sie können auch eine Always-On-Verfügbarkeitsgruppe verwenden, um Ihre lokalen SQL Server-Datenbanken auf Amazon EC2 zu migrieren. AWS Dieser Ansatz ermöglicht es Ihnen, Ihre Datenbanken mit minimalen bis gar keinen Ausfallzeiten zu migrieren.

Wenn Sie bereits über eine lokale Bereitstellung von SQL Server AlwaysOn-Verfügbarkeitsgruppen verfügen, replizieren Ihr primäres Replikat und Ihr sekundäres Replikat synchron Daten innerhalb der Verfügbarkeitsgruppe. Um Ihre Datenbank in die AWS Cloud zu migrieren, können Sie also Ihren Windows Server Failover Clustering (WSFC) -Cluster auf die Cloud ausdehnen. Dies kann vorübergehend sein, nur zu Migrationszwecken. Anschließend erstellen Sie ein sekundäres Replikat in der AWS Cloud und verwenden die asynchrone Replikation, wie in der folgenden Abbildung dargestellt. Nachdem das sekundäre Replikat mit der primären lokalen Datenbank synchronisiert wurde, können Sie ein manuelles Failover durchführen, wann immer Sie für die Übernahme bereit sind.



Weitere Informationen zur Verwendung dieser Methode, um Hochverfügbarkeit, Datenschutz und Notfallwiederherstellung für Ihre SQL Server-Datenbanken auf Amazon EC2 zu erreichen, finden Sie unter [AlwaysOn-Verfügbarkeitsgruppen](#) im Abschnitt Amazon EC2 für SQL Server.

## Verteilte Verfügbarkeitsgruppen

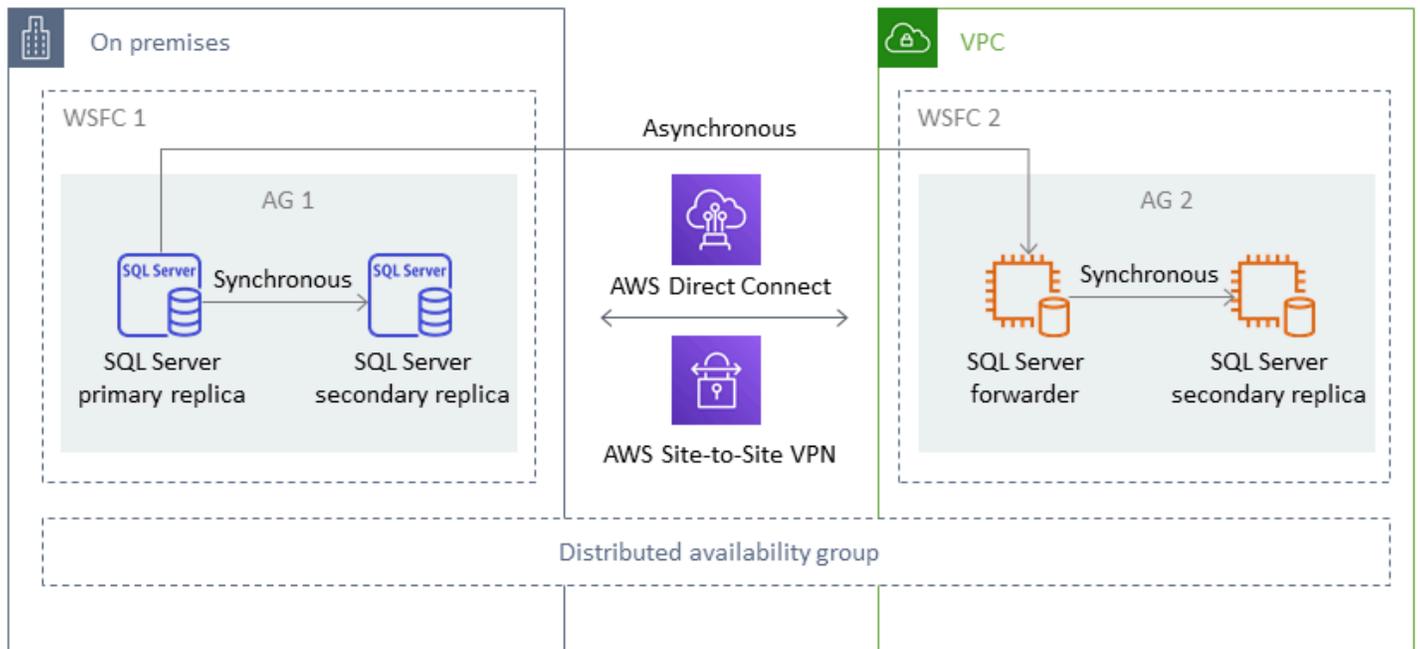
Eine verteilte Verfügbarkeitsgruppe umfasst zwei separate Verfügbarkeitsgruppen. Sie können sie sich als eine Verfügbarkeitsgruppe von Verfügbarkeitsgruppen vorstellen. Die zugrunde liegenden Verfügbarkeitsgruppen sind auf zwei verschiedenen WSFC-Clustern konfiguriert. Die Verfügbarkeitsgruppen, die Teil einer verteilten Verfügbarkeitsgruppe sind, müssen sich nicht denselben Standort teilen. Sie können physisch oder virtuell, lokal oder in der Public Cloud sein. Die Verfügbarkeitsgruppen in einer verteilten Verfügbarkeitsgruppe müssen nicht dieselbe Version von SQL Server ausführen. Die Ziel-DB-Instance kann eine neuere Version von SQL Server als die Quell-DB-Instance ausführen.

Eine verteilte Verfügbarkeitsgruppenarchitektur bietet Ihnen eine flexible Möglichkeit, eine unternehmenskritische SQL Server-Instanz oder Datenbank neu zu hosten. AWS bietet eine Hybridlösung für das Lifting and Shifting (oder Lifting und Transformation) Ihrer kritischen SQL Server-Datenbanken. AWS

Die Verwendung einer Architektur für verteilte Verfügbarkeitsgruppen ist effizienter als die Erweiterung vorhandener lokaler WSFC-Cluster. AWS Daten werden nur vom lokalen Primärserver an eines der AWS Replikate (den Forwarder) übertragen. Der Forwarder ist dafür verantwortlich, Daten an andere sekundäre Read-Replicas zu senden. AWS

In der folgenden Abbildung wird der erste WSFC-Cluster (WSFC 1) lokal gehostet und verfügt über eine lokale Verfügbarkeitsgruppe (AG 1). Der zweite WSFC-Cluster (WSFC 2) wird auf gehostet AWS und verfügt über eine AWS Verfügbarkeitsgruppe (AG 2). [AWS Direct Connect](#) wird als dedizierte Netzwerkverbindung zwischen der lokalen Umgebung und verwendet. AWS Die lokale Verfügbarkeitsgruppe (AG 1) hat zwei Replikate (Knoten). Die Datenübertragung zwischen den Knoten erfolgt synchron mit automatischem Failover. In ähnlicher Weise verfügt die AWS Availability Group (AG 2) ebenfalls über zwei Replikate, und die Datenübertragung zwischen ihnen erfolgt synchron mit automatischem Failover. Die verteilte Verfügbarkeitsgruppe sorgt dafür, dass die Datenbanken asynchron synchronisiert werden. Daten werden vom primären SQL Server-Replikat in AG 1 (das sich lokal befindet) zum primären Replikat (der Forwarder) in AG 2 (das aktiviert ist) übertragen. AWS Der Forwarder ist dafür verantwortlich, Daten an andere Read Replicas zu senden und diese auf dem AWS neuesten Stand zu halten. Nachdem die lokalen Daten und die AWS Datenbanken synchronisiert wurden, können Sie ein manuelles Failover der verteilten

Verfügbarkeitsgruppe auf durchführen. AWS Die AWS Datenbank wird zur primären Datenbank für den Lese-/Schreibzugriff von Anwendungen aus.



#### Note

Zu einem bestimmten Zeitpunkt gibt es nur eine Datenbank, die für Schreibvorgänge verfügbar ist. Sie können die verbleibenden sekundären Replikat für Lesevorgänge verwenden. Um Ihre Lese-Workloads zu skalieren, können Sie weitere Read Replicas in mehreren Availability Zones hinzufügen. AWS

Weitere Informationen zu verteilten Verfügbarkeitsgruppen finden Sie unter:

- [Dokumentation zu Microsoft SQL Server](#)
- [So entwerfen Sie eine hybride Microsoft SQL Server-Lösung mithilfe verteilter Verfügbarkeitsgruppen](#) im AWS Datenbank-Blog
- [Migrieren Sie SQL Server auf die AWS Verwendung verteilter Verfügbarkeitsgruppen](#) auf der AWS Prescriptive Guidance-Website

# Transaktionsreplikation

Die Transaktionsreplikation ist eine SQL Server-Technologie, die verwendet wird, um Änderungen zwischen zwei Datenbanken zu replizieren. Diese Änderungen können Datenbankobjekte wie Tabellen (Primärschlüssel ist erforderlich), gespeicherte Prozeduren, Ansichten usw. sowie Daten umfassen. Der Replikationsprozess umfasst einen Herausgeber (die primäre Datenbank, die Daten veröffentlicht), einen Abonnenten (eine sekundäre Datenbank, die replizierte Daten empfängt) und einen Verteiler (einen Server, der Metadaten und Transaktionen für die Transaktionsreplikation speichert). Sie können die Transaktionsreplikation für SQL Server auf Amazon EC2 und Amazon RDS for SQL Server Server-DB-Instances verwenden.

Bei der Transaktionsreplikation wird ein Snapshot der Objekte und Daten in Ihrer lokalen Datenbank (Publikations-) Datenbank erstellt und an die Abonnentendatenbank gesendet. Nachdem der Snapshot auf den Abonnenten angewendet wurde, werden alle nachfolgenden Daten- und Schemaänderungen, die beim Herausgeber vorgenommen wurden, sofort an den Abonnenten gesendet. Die Datenänderungen werden dann kontinuierlich in derselben Reihenfolge, in der sie beim Herausgeber vorgenommen wurden, auf den Abonnenten angewendet.

Nach Abschluss der Synchronisation führen Sie eine Überprüfung auf der SQL Server-DB-Zielinstanz durch. Wenn die beiden Datenbanken synchronisiert sind, beenden Sie die Aktivität in der lokalen Datenbank, stellen sicher, dass die Replikation abgeschlossen ist, und führen dann die Umstellung auf die SQL Server-DB-Zielinstanz durch. Anschließend können Sie das Push-Abonnement beenden, löschen und Amazon RDS for SQL Server verwenden.

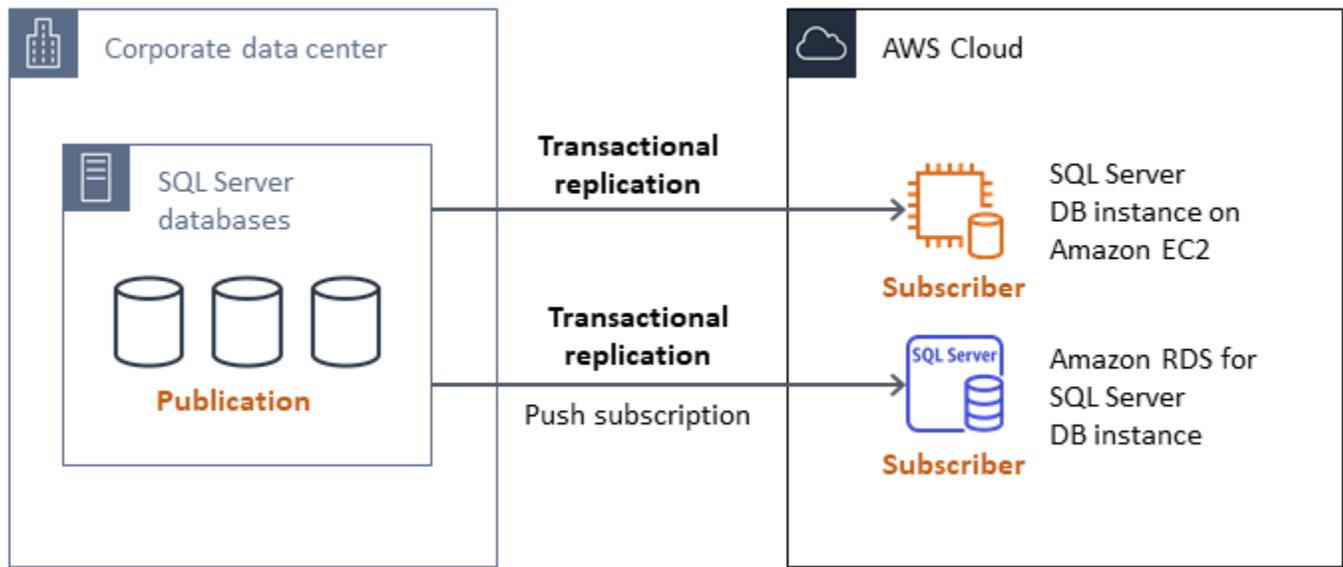
Abonentendatenbanken können auch als schreibgeschützte Datenbanken verwendet werden. Es wird empfohlen, dass sich der Verteiler, der Synchronisationsaufträge aufzeichnet, auf einem separaten Server befindet. Wenn sich Ihre Zieldatenbank auf Amazon RDS for SQL Server befindet, können Sie ein Push-Abonnement einrichten, um Änderungen an den Abonnenten weiterzugeben.

Wir empfehlen Ihnen, die Transaktionsreplikation zu verwenden, wenn Sie:

- Führen Sie eine einmalige Migration Ihrer Daten zu Amazon RDS oder Amazon EC2 durch.
- Migrieren Sie Objekte auf Schema- oder Tabellenebene zu. AWS
- Migrieren Sie einen Teil einer Datenbank zu. AWS
- Migrieren Sie mit minimalen Ausfallzeiten mithilfe vorhandener SQL Server-Replikationsstrategien, indem Sie zusätzliche Abonnenten hinzufügen.

Wenn Sie planen, die Transaktionsreplikation für die einmalige Migration Ihrer Daten zu Amazon RDS for SQL Server zu verwenden, empfehlen wir Ihnen, eine Single-AZ-Konfiguration für die Replikation einzurichten. Nach Abschluss des Replikationsvorgangs können Sie Ihre Umgebung für hohe Verfügbarkeit in eine Multi-AZ-Architektur umwandeln.

Das folgende Diagramm zeigt den Transaktionsreplikationsprozess für Datenbanken auf Amazon RDS und Amazon EC2.



Weitere Informationen zur Transaktionsreplikation finden Sie in der [Microsoft SQL Server-Dokumentation](#) und im Datenbank-Blog [How to Migration to Amazon RDS for SQL Server using Transactional Replication](#). AWS

## AWS Migration Hub Orchestrator

AWS Migration Hub Orchestrator hilft Ihnen bei der Orchestrierung und Automatisierung der Migration von SQL Server-Datenbanken zu Amazon EC2 oder Amazon RDS. Mit dieser Funktion von AWS Migration Hub können Sie schnell loslegen, indem vordefinierte Workflow-Vorlagen verwendet werden, die auf bewährten Methoden basieren. Migration Hub Orchestrator automatisiert fehleranfällige manuelle Aufgaben, die mit dem Migrationsprozess verbunden sind, wie z. B. die Überprüfung der Umgebungsbereitschaft und der Verbindungen. Sie können den Migration Hub Orchestrator auch verwenden, um Migrationen für .NET-Anwendungen, SAP-Workloads und VM-Images zusätzlich zu Ihren SQL Server-Datenbanken zu orchestrieren und zu beschleunigen. Sie können über die [Migration Hub Orchestrator-Konsole](#) auf dieses Tool zugreifen.

Für die SQL Server-Migration unterstützt Migration Hub Orchestrator drei Anwendungsfälle:

- Hosten Sie SQL Server auf Amazon EC2 neu. Sie können bestimmte SQL-Server auswählen und sie auf Amazon EC2 rehosten, indem Sie automatische native Sicherung und Wiederherstellung in Migration Hub Orchestrator verwenden. Weitere Informationen finden Sie unter [Rehost SQL Server on Amazon EC2](#) in der Migration Hub Orchestrator-Dokumentation.
- Führen Sie eine neue Plattform für SQL Server auf Amazon RDS ein. Sie können bestimmte SQL Server-Datenbanken auswählen und sie auf Amazon RDS neu plattformieren, indem Sie die automatische native Sicherung und Wiederherstellung im Migration Hub Orchestrator verwenden. Weitere Informationen finden Sie unter [Replatform SQL Server on Amazon RDS](#) in der Migration Hub Orchestrator-Dokumentation.
- Rehosten Sie Windows- und SQL Server-Anwendungen auf Amazon EC2. Mithilfe der Vorlage Rehost applications on Amazon EC2 können Sie Ihre Windows-Server, auf denen .NET und SQL Server ausgeführt werden, auf Amazon EC2 umstellen und auf Amazon EC2 umstellen. Weitere Informationen finden Sie unter [Rehost Applications on Amazon EC2](#) in der Migration Hub Orchestrator-Dokumentation.

Migration Hub Orchestrator hilft Ihnen, Zeitplan- und Budgetüberschreitungen bei Ihren SQL Server-Migrationen zu vermeiden. Zu den weiteren wichtigen Vorteilen gehören:

- Migrieren Sie Anwendungen mithilfe einer präskriptiven Methode. Mit den vordefinierten Workflow-Vorlagen, die auf bewährten Best Practices für die Migration basieren, können Sie schnell loslegen. Sie können Ihren Migrationsablauf auch anpassen, indem Sie Schritte je nach Bedarf hinzufügen, neu anordnen und entfernen. Sie können beispielsweise einen Schritt für die Genehmigung der Umstellung hinzufügen.
- Automatisieren Sie manuelle Schritte. Migration Hub Orchestrator automatisiert manuelle Aufgaben wie die Installation von Agenten, den Import von lokalen Images, die Bereitstellung Ihrer Zielumgebung auf und die Überprüfung von Quell AWS - und Zielumgebungen. Durch die Automatisierung sparen Sie Zeit und Kosten und reduzieren gleichzeitig Fehler.
- Orchestrieren Sie den Migrations-Workflow. Migration Hub Orchestrator orchestriert die in Migrationsschritten verwendeten Tools, indem er die Inventarmetadaten, die Konfigurationsspezifikation und den Umgebungskontext wiederverwendet, um die Anzahl der Eingaben zu minimieren, die diese Tools benötigen.

Weitere Informationen finden Sie in den folgenden Ressourcen:

- [Migration Hub Orchestrator-Konsole](#)

- [Rehosten von Anwendungen auf Amazon EC2](#) (Migration Hub Orchestrator-Benutzerhandbuch)
- [Neuplattformierung des SQL-Servers auf Amazon RDS](#) (Migration Hub Orchestrator-Benutzerhandbuch)
- [Migrationsworkflows](#) (Migration Hub Orchestrator-Benutzerhandbuch)
- [Verwendung von Migration Hub Orchestrator zur Vereinfachung und Beschleunigung von Microsoft SQL Server-Migrationen \(Blogbeitrag\)](#) AWS
- [Vereinfachen Sie die Migration Ihrer Windows Server-Images mit AWS Migration Hub Orchestrator \(Blogbeitrag\)](#) AWS

## AWS Snowball Edge

Sie können AWS Snowball Edge verwenden, um sehr große Datenbanken (bis zu 210 TB Größe) zu migrieren. Snowball verfügt über einen 10-Gbit-Ethernet-Port, den Sie an Ihren lokalen Server anschließen und alle Datenbanksicherungen oder Daten auf dem Snowball-Gerät ablegen. Nachdem die Daten nach Snowball kopiert wurden, senden Sie die Appliance zur AWS Platzierung in Ihrem dafür vorgesehenen S3-Bucket an. Anschließend können Sie die Backups von Amazon S3 herunterladen und auf SQL Server auf einer EC2-Instance wiederherstellen oder die `rds_restore_database` gespeicherte Prozedur ausführen, um die Datenbank in Amazon RDS wiederherzustellen. Sie können [AWSSnowcone](#) auch für Datenbanken mit einer Größe von bis zu 8 TB verwenden. Weitere Informationen finden Sie in der [AWS SnowballEdge-Dokumentation](#) und im Abschnitt [Importieren und Exportieren von SQL Server-Datenbanken](#) im Abschnitt Datenbank wiederherstellen in der Amazon RDS-Dokumentation.

# Homogene Datenbankmigration für SQL Server

AWS bietet Ihnen die Möglichkeit, SQL Server-Datenbanken in einer Cloud-Umgebung auszuführen. Für Entwickler und Datenbankadministratoren ist das Ausführen einer SQL Server-Datenbank in der AWS Cloud dem Ausführen einer SQL Server-Datenbank in einem Rechenzentrum sehr ähnlich. In diesem Abschnitt werden Optionen für die Migration Ihrer SQL Server-Datenbank von einer lokalen Umgebung oder einem Rechenzentrum in die AWS Cloud beschrieben.

AWS bietet drei Optionen für die Ausführung von SQL Server auf AWS, wie in der folgenden Tabelle beschrieben.

Option	Highlights	Weitere Informationen
SQL Server auf Amazon RDS	Managed Service bietet einfache Bereitstellung und Lizenzierung, ist kostengünstig, einfach einzurichten, zu verwalten und zu warten.	Abschnitt <a href="#">Amazon RDS for SQL Server</a>
SQL Server auf Amazon RDS Benutzerdefiniert	Verwalteter Service, aber Sie behalten die Administratorrechte für die Datenbank und das zugrunde liegende Betriebssystem.	Abschnitt <a href="#">Amazon RDS Benutzerdefiniert für SQL Server</a>
SQL Server auf Amazon EC2	Selbstverwaltet, bietet volle Kontrolle und Flexibilität.	<a href="#">Abschnitt Amazon EC2 für SQL Server</a>
SQL Server auf VMware Cloud auf AWS	Richten Sie Ihre SQL Server-Workloads auf VMware Cloud on ein, skalieren und betreiben Sie sie AWS und integrieren Sie sie mit AWS Directory Service Active Directory Connector und Amazon S3.	Abschnitt <a href="#">„VMware Cloud on AWS für SQL Server“</a>

**i** Notice (Hinweis)

Seit dem 30. April 2024 AWS wird VMware Cloud on nicht mehr von AWS oder seinen Channel-Partnern weiterverkauft. Der Service wird weiterhin über Broadcom verfügbar sein. Wir empfehlen Ihnen, sich für weitere Informationen an Ihren AWS Vertreter zu wenden.

Ihre Anwendungsanforderungen, Datenbankfunktionen, Funktionalität, Wachstumskapazität und die allgemeine Komplexität der Architektur bestimmen, welche Option Sie wählen sollten. Wenn Sie mehrere SQL Server-Datenbanken migrieren AWS, eignen sich einige davon möglicherweise hervorragend für Amazon RDS, während andere möglicherweise besser für die direkte Ausführung auf Amazon EC2 geeignet sind. Möglicherweise verfügen Sie über Datenbanken, die auf der SQL Server Enterprise Edition ausgeführt werden, sich aber gut für die SQL Server Standard Edition eignen. Möglicherweise möchten Sie auch Ihre unter Windows ausgeführte SQL Server-Datenbank so modernisieren, dass sie auf einem Linux-Betriebssystem ausgeführt wird, um Kosten und Lizenzen zu sparen. Viele AWS Kunden führen mehrere SQL Server-Datenbank-Workloads in Amazon RDS, Amazon EC2 und VMware Cloud auf AWS

**i** Note

Sie können Migration Hub Orchestrator verwenden, um Ihre SQL Server-Datenbankmigrationen zu Amazon EC2 oder Amazon RDS mithilfe nativer Sicherung und Wiederherstellung zu automatisieren und zu orchestrieren. [Weitere Informationen finden Sie im Abschnitt.AWS Migration Hub Orchestrator](#)

## Amazon RDS für SQL Server

Amazon RDS for SQL Server ist ein verwalteter Datenbankservice, der die Bereitstellung und Verwaltung von SQL Server auf AWS vereinfacht. Amazon RDS macht es einfach, SQL Server-Bereitstellungen in der Cloud einzurichten, zu betreiben und zu skalieren. Mit Amazon RDS können Sie innerhalb von Minuten mehrere Versionen von SQL Server (2014, 2016, 2017, 2019 und 2022) und Editionen (einschließlich Express, Web, Standard und Enterprise) mit kosteneffizienter und anpassbarer Rechenkapazität bereitstellen. Sie können Amazon RDS for SQL Server Server-DB-Instances entweder mit Allzweck-SSD oder bereitgestelltem IOPS-SSD-Speicher bereitstellen. (Einzelheiten finden Sie in der AWS Dokumentation unter [Amazon RDS-Speichertypen](#).) Die

bereitgestellte IOPS-SSD ist auf schnelle, vorhersehbare und konsistente I/O-Leistung ausgelegt und für I/O-intensive, transaktionale (OLTP) Datenbank-Workloads optimiert.

Mit Amazon RDS können Sie sich auf die Anwendungsentwicklung konzentrieren, da es zeitaufwändige Datenbankverwaltungsaufgaben wie Bereitstellung, Backups, Software-Patching, Überwachung und Hardwareskalierung übernimmt. Amazon RDS for SQL Server bietet auch Multi-AZ-Bereitstellungen und Read Replicas (für SQL Server Enterprise Edition), um hohe Verfügbarkeit, Leistung, Skalierbarkeit und Zuverlässigkeit für Produktionsworkloads zu gewährleisten.

Weitere Informationen zur Migration von SQL Server zu Amazon RDS finden Sie in den [Replatform Patterns](#) auf der AWS Prescriptive Guidance-Website.

## Wann sollten Sie sich für Amazon RDS entscheiden

Amazon RDS for SQL Server ist eine Migrationsoption, wenn:

- Sie möchten sich auf Ihr Geschäft und Ihre Anwendungen konzentrieren und sich AWS um undifferenzierte, schwere Aufgaben wie die Bereitstellung der Datenbank, die Verwaltung von Sicherheits- und Wiederherstellungsaufgaben, die Verwaltung von Sicherheitspatches, kleinere Upgrades der SQL Server-Version und die Speicherverwaltung kümmern.
- Sie benötigen eine hochverfügbare Datenbanklösung und möchten die Vorteile der von Amazon RDS angebotenen synchronen Multi-AZ-Replikation per Knopfdruck nutzen, ohne Datenbankspiegelung, Failover-Cluster oder Always-On-Verfügbarkeitsgruppen manuell einrichten und verwalten zu müssen.
- Sie möchten die SQL Server-Lizenz als Teil der Instanzkosten auf Stundenbasis bezahlen, anstatt eine große Vorabinvestition zu tätigen.
- Ihre Datenbankgröße und Ihre IOPS-Anforderungen werden von Amazon RDS for SQL Server unterstützt. Die aktuellen Höchstgrenzen finden Sie in der AWS Dokumentation unter [Amazon RDS DB Instance Storage](#).
- Sie möchten keine Backups oder point-in-time Wiederherstellungen Ihrer Datenbank verwalten.
- Sie möchten sich auf wichtige Aufgaben wie Leistungsoptimierung und Schemaoptimierung konzentrieren, anstatt sich auf die tägliche Verwaltung der Datenbank zu konzentrieren.
- Sie möchten den Instanztyp auf der Grundlage Ihrer Workload-Muster nach oben oder unten skalieren, ohne sich Gedanken über die Komplexität der Lizenzierung machen zu müssen.

Wenn Sie sich nach der Bewertung Ihrer Datenbank- und Projektanforderungen für eine Migration zu Amazon RDS for SQL Server entscheiden, lesen Sie sich die Details in den folgenden Abschnitten

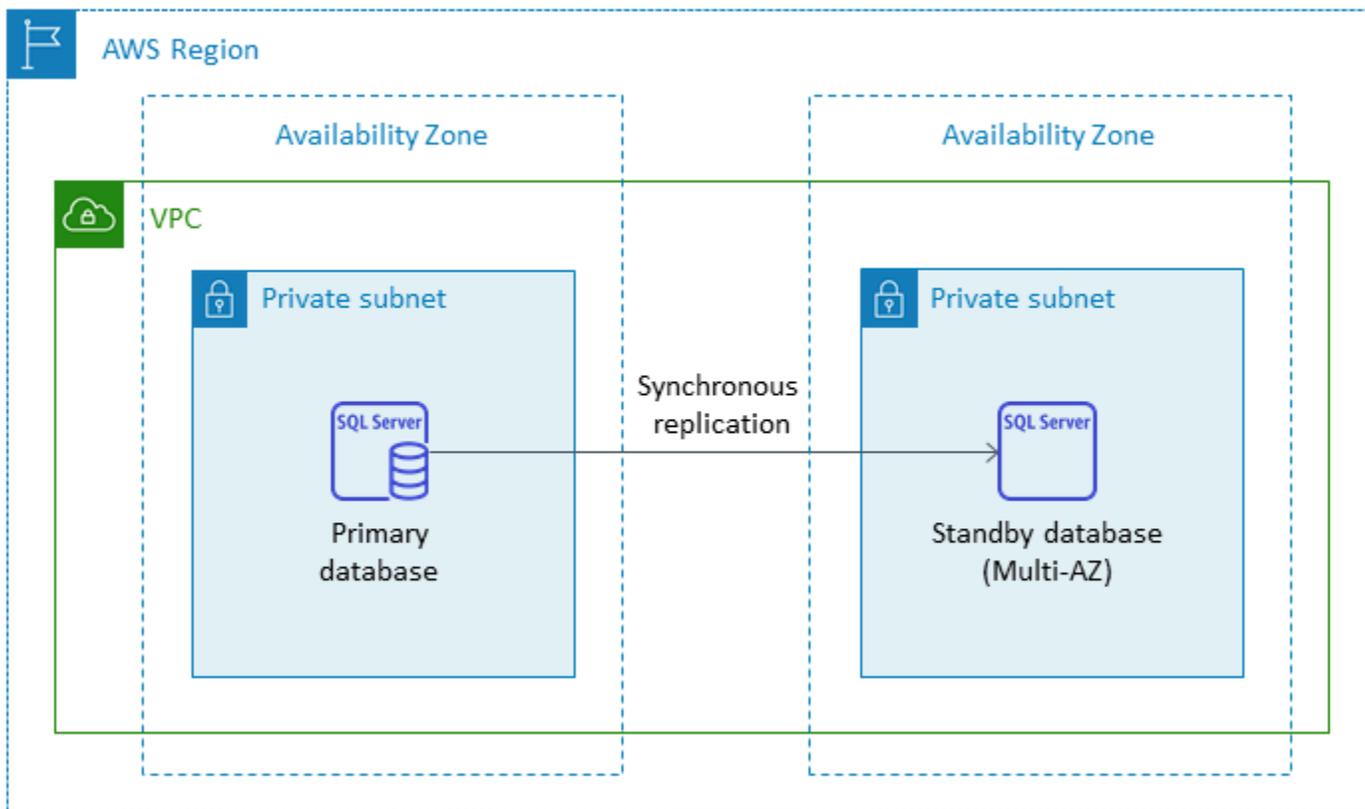
durch und lesen Sie sich die [bewährten Methoden für die Migration](#) durch, auf die wir weiter unten in diesem Leitfaden eingehen.

Informationen zu den derzeit unterstützten Funktionen, Versionen und Optionen von SQL Server finden Sie unter [Funktionen von Amazon RDS for SQL Server](#) auf der AWS Website, [Choosing between Amazon EC2 and Amazon RDS](#) weiter unten in diesem Handbuch und [Microsoft SQL Server on Amazon RDS](#) in der AWS Dokumentation. Wenn Sie zu Amazon RDS Custom wechseln, lesen Sie sich unbedingt die [Anforderungen und Einschränkungen für Amazon RDS Custom for SQL Server](#) durch.

## Hohe Verfügbarkeit

Amazon RDS bietet Hochverfügbarkeit und Failover-Unterstützung für Datenbanken, die mit der Multi-AZ-Option bereitgestellt werden. Wenn Sie Ihre Datenbank mit der Multi-AZ-Option bereitstellen, stellt Amazon RDS automatisch eine synchrone Standby-Instance in einer anderen Availability Zone bereit und verwaltet diese. Die Primärdatenbank repliziert die Daten synchron auf die Standby-Instance. Wenn Probleme auftreten, repariert Amazon RDS automatisch die fehlerhafte Instance und stellt die Synchronisation wieder her. Im Falle eines Infrastrukturausfalls oder einer Unterbrechung der Availability Zone führt Amazon RDS einen automatischen Failover zur Standby-Instance durch. Ein Failover erfolgt nur, wenn die Standby- und Primärdatenbanken vollständig synchronisiert sind. Da der Endpunkt für die Primär- und Standby-Instances derselbe bleibt, können Sie den Datenbankbetrieb wieder aufnehmen, sobald der Failover abgeschlossen ist, ohne dass ein manuelles Eingreifen erforderlich ist. Die Failover-Zeit hängt von der Zeit ab, die benötigt wird, um den Wiederherstellungsprozess abzuschließen. Große Transaktionen erhöhen die Failover-Zeit.

Das folgende Diagramm veranschaulicht die Multi-AZ-Bereitstellungsoption von Amazon RDS for SQL Server.



Wenn Sie SQL Server in einer Multi-AZ-Konfiguration einrichten, konfiguriert Amazon RDS automatisch die Standby-Datenbank-Instance mithilfe von Datenbankspiegelung oder AlwaysOn-Verfügbarkeitsgruppen, basierend auf der Version von SQL Server, die Sie bereitstellen. Die spezifischen Versionen und Editionen von SQL Server sind in der [Amazon RDS-Dokumentation](#) aufgeführt.

In Multi-AZ-Bereitstellungen werden Operationen wie die Skalierung von Instanzen oder System-Upgrades wie das Patchen von Betriebssystemen (OS) zunächst auf der Standby-Instance angewendet, bevor der automatische Failover der primären Instance durchgeführt wird, um die Verfügbarkeit zu erhöhen.

Aufgrund der Failover-Optimierung von SQL Server können bestimmte Workloads eine höhere I/O-Last auf der Standby-Instanz erzeugen als auf der Primärinstanz, insbesondere bei Bereitstellungen mit Datenbankspiegelung. Diese Funktionalität kann zu höheren IOPS auf der Standby-Instanz führen. Wir empfehlen, dass Sie bei der Bereitstellung des Speichertyps und der IOPS Ihrer Amazon RDS for SQL Server-DB-Instance die maximalen IOPS-Anforderungen sowohl der primären als auch der Standby-Instances berücksichtigen. Sie können auch angeben `MultiSubnetFailover=True`, ob Ihr Client-Treiber dies unterstützt, um die Failover-Zeit erheblich zu reduzieren.

## Einschränkungen

- Die Multi-AZ-Option ist für SQL Server Express und Web Editionen nicht verfügbar. Sie ist nur für die Editionen SQL Server Standard und Enterprise verfügbar.
- Sie können die Standby-DB-Instance nicht so konfigurieren, dass sie Datenbank-Leseaktivitäten akzeptiert.
- Regionsübergreifendes Multi-AZ wird nicht unterstützt.
- In Amazon RDS können Sie einen Stop-Befehl für eine eigenständige DB-Instance ausgeben und die Instance im gestoppten Zustand belassen, um Rechengebühren zu vermeiden. Sie können eine Amazon-RDS-for-SQL-Server-DB-Instance in einer Multi-AZ-Konfiguration nicht anhalten. Stattdessen können Sie die Instance beenden, vor der Kündigung einen letzten Snapshot erstellen und bei Bedarf aus dem Snapshot eine neue Amazon RDS-Instance erstellen. Oder Sie können zuerst die Multi-AZ-Konfiguration entfernen und dann die Instance beenden. Nach sieben Tagen wird Ihre gestoppte Instance neu gestartet, sodass alle ausstehenden Wartungsarbeiten durchgeführt werden können.

Weitere Einschränkungen finden Sie in den [Hinweisen und Empfehlungen zur Bereitstellung von Microsoft SQL Server Multi-AZ](#) in der Amazon RDS-Dokumentation.

## Read Replicas

Read Replicas bieten Skalierbarkeit und Lastenausgleich. Eine SQL Server-Lesereplik ist eine physische Kopie einer Amazon RDS for SQL Server-DB-Instance, die nur für Lesezwecke verwendet wird. Amazon RDS trägt dazu bei, die Belastung der primären DB-Instance zu reduzieren, indem schreibgeschützte Workloads auf die Read Replica-DB-Instance ausgelagert werden. An Ihrer primären DB-Instance vorgenommene Aktualisierungen werden asynchron auf die Read Replica-Instance kopiert.

Wenn Sie eine Read Replica anfordern, erstellt Amazon RDS einen Snapshot der Quell-DB-Instance, und dieser Snapshot wird zur Read Replica. Beim Erstellen und Löschen einer Read Replica tritt kein Ausfall auf. Amazon RDS for SQL Server aktualisiert die Primärdatenbank unmittelbar nach dem Upgrade der Read Replicas, unabhängig vom Wartungsfenster. Jede Read Replica verfügt über einen separaten Endpunkt, über den Sie eine Verbindung zur Read Replica-Datenbank herstellen.

Amazon RDS for SQL Server macht es einfach, Read Replicas zu erstellen, indem Always-On-Verfügbarkeitsgruppen konfiguriert und sichere Netzwerkverbindungen zwischen einer primären DB-Instance und ihren Read Replicas aufrechterhalten werden.

Sie können eine Read Replica in derselben AWS Region wie Ihre Primärdatenbank einrichten. Amazon RDS for SQL Server unterstützt keine regionsübergreifenden Read Replicas. Sie können bis zu fünf Read Replicas für eine Quell-DB-Instance erstellen.

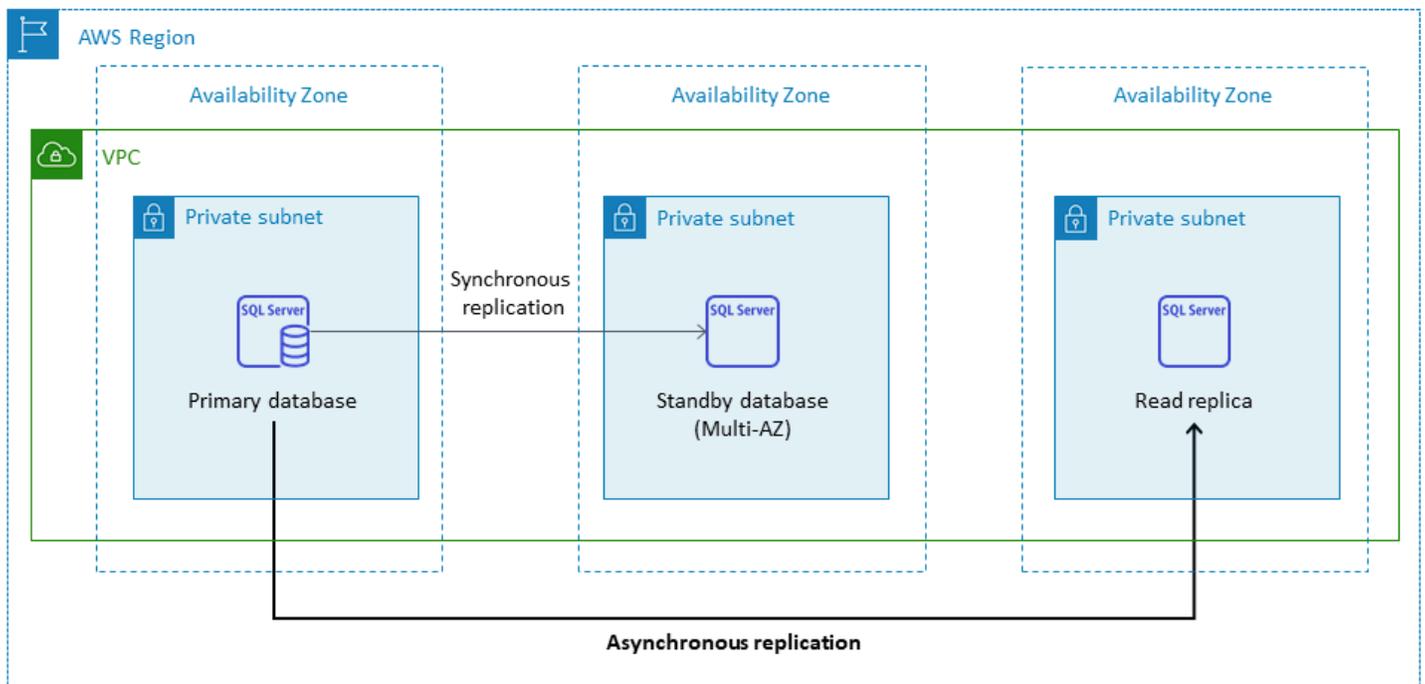
### Note

Read Replicas sind nur mit den folgenden SQL Server-Versionen und -Editionen verfügbar:

- SQL Server 2017 Enterprise Edition 14.00.3049.1 oder höher
- SQL Server 2016 Enterprise Edition 13.00.5216.0 oder höher

SQL Server-Versionen und -Editionen, die die Datenbankspiegelung für Multi-AZ-Umgebungen unterstützen, bieten keine Read Replicas.

Das folgende Diagramm zeigt eine Amazon RDS for SQL Server-DB-Instance in einer Multi-AZ-Umgebung mit einer Read Replica in einer anderen Availability Zone innerhalb derselben AWS Region. Nicht alle AWS Regionen bieten mehr als zwei Availability Zones. Sie sollten daher [überprüfen, welche Region](#) Sie verwenden möchten, bevor Sie diese Strategie anwenden.



Eine SQL Server-Lesereplik erlaubt keine Schreibvorgänge. Sie können das Lesereplikat jedoch heraufstufen, um es schreibbar zu machen. Nachdem Sie es heraufgestuft haben, können Sie

es nicht wieder in ein Read Replica umwandeln. Sie wird zu einer einzelnen, eigenständigen DB-Instance, die keine Beziehungen zu ihrer ursprünglichen primären Datenbank-Instance hat. Die Daten in der hochgestuften Read Replica stimmen mit den Daten in der Quell-DB-Instance bis zu dem Zeitpunkt überein, zu dem die Anfrage zur Heraufstufung gestellt wurde. Die Version der SQL Server-DB-Engine der Quell-DB-Instance und all ihrer Read Replicas wird identisch sein.

Für eine effiziente Replikation empfehlen wir Folgendes:

- Richten Sie jede Read Replica mit denselben Rechen- und Speicherressourcen ein wie die Quell-DB-Instance.
- Sie müssen automatische Backups auf der Quell-DB-Instance aktivieren, indem Sie den Aufbewahrungszeitraum für Backups auf einen anderen Wert als 0 (Null) setzen.
- Die Quell-DB-Instance muss in einer Multi-AZ-Umgebung mit AlwaysOn-Verfügbarkeitsgruppen bereitgestellt werden.

Informationen zur Unterstützung, Editionen und Einschränkungen von SQL Server-Versionen finden Sie unter [Einschränkungen von Read Replica with SQL Server](#) in der Amazon RDS-Dokumentation.

Weitere Informationen zur Verwendung von Read Replicas finden Sie in der Dokumentation unter [Working with Read Replicas](#) und [Working with SQL Server Read Replicas for Amazon RDS](#). AWS Weitere Informationen zu den Kosten von Datenübertragungen finden Sie unter [Amazon RDS – Preise](#).

## Notfallwiederherstellung

Mit Amazon RDS for SQL Server können Sie eine zuverlässige, regionsübergreifende Disaster Recovery-Strategie (DR) entwickeln. Die Hauptgründe für die Entwicklung einer DR-Lösung sind Geschäftskontinuität und Compliance:

- Eine effektive DR-Strategie hilft Ihnen dabei, Ihre Systeme bei einem katastrophalen Ereignis mit minimalen oder gar keinen Unterbrechungen am Laufen zu halten. Eine zuverlässige und effektive regionsübergreifende DR-Strategie sorgt dafür, dass Ihr Unternehmen auch dann weiterläuft, wenn eine ganze Region offline geht.
- Eine regionsübergreifende DR-Lösung hilft Ihnen dabei, Prüf- und Compliance-Anforderungen zu erfüllen.

Recovery Point Objective (RPO), Recovery Time Objective (RTO) und Kosten sind drei wichtige Kennzahlen, die Sie bei der Entwicklung Ihrer DR-Strategie berücksichtigen sollten. Weitere Optionen für die Bereitstellung regionsübergreifender Replikate finden Sie unter [AWS Marketplace](#). Weitere Informationen zu diesen Ansätzen finden Sie unter [Regionsübergreifende Notfallwiederherstellung von Amazon RDS for SQL Server](#) im AWS Datenbank-Blog.

## Amazon RDS Custom für SQL Server

Wenn Sie aufgrund von Anpassungsanforderungen für Drittanbieteranwendungen nicht zu einem vollständig verwalteten Service wie Amazon RDS wechseln können, können Sie zu Amazon RDS Custom for SQL Server migrieren. Mit Amazon RDS Custom können Sie Administratorrechte für die Datenbank und das zugrunde liegende Betriebssystem behalten, um abhängige Anwendungen zu aktivieren.

### Wann sollten Sie Amazon RDS Custom für SQL Server wählen

Amazon RDS Custom for SQL Server ist eine gute Migrationsoption, wenn:

- Sie haben ältere, benutzerdefinierte und verpackte Anwendungen, die Zugriff auf das zugrunde liegende Betriebssystem und die Datenbankumgebung benötigen.
- Sie benötigen Administratorzugriff, um die Anforderungen der herstellerspezifischen Anwendungsbereitstellung zu erfüllen.
- Sie benötigen Zugriff auf das zugrunde liegende Betriebssystem, um Einstellungen zu konfigurieren, Patches zu installieren und native Funktionen zu aktivieren, um die Anforderungen der abhängigen Anwendung zu erfüllen.
- Sie möchten auf die Datenbankumgebung zugreifen und diese anpassen (indem Sie benutzerdefinierte Datenbank-Patches anwenden oder Betriebssystempakete ändern), um sie an Ihre Datenbank- und Anwendungsanforderungen anzupassen.

### Funktionsweise

Um Amazon RDS Custom for SQL Server zu verwenden, lesen Sie sich die [Anforderungen](#) in der Dokumentation zu Amazon RDS Custom for SQL Server durch. Sie müssen zuerst Ihre Umgebung für Amazon RDS Custom for SQL Server einrichten, wie in der [Amazon RDS-Dokumentation](#) beschrieben. Nachdem die Umgebung eingerichtet ist, folgen Sie diesen Schritten, die in der folgenden Abbildung dargestellt sind:

1. Erstellen Sie eine Amazon RDS Custom for SQL Server-DB-Instance aus einer von Amazon RDS Custom angebotenen SQL Server-Engine-Version.

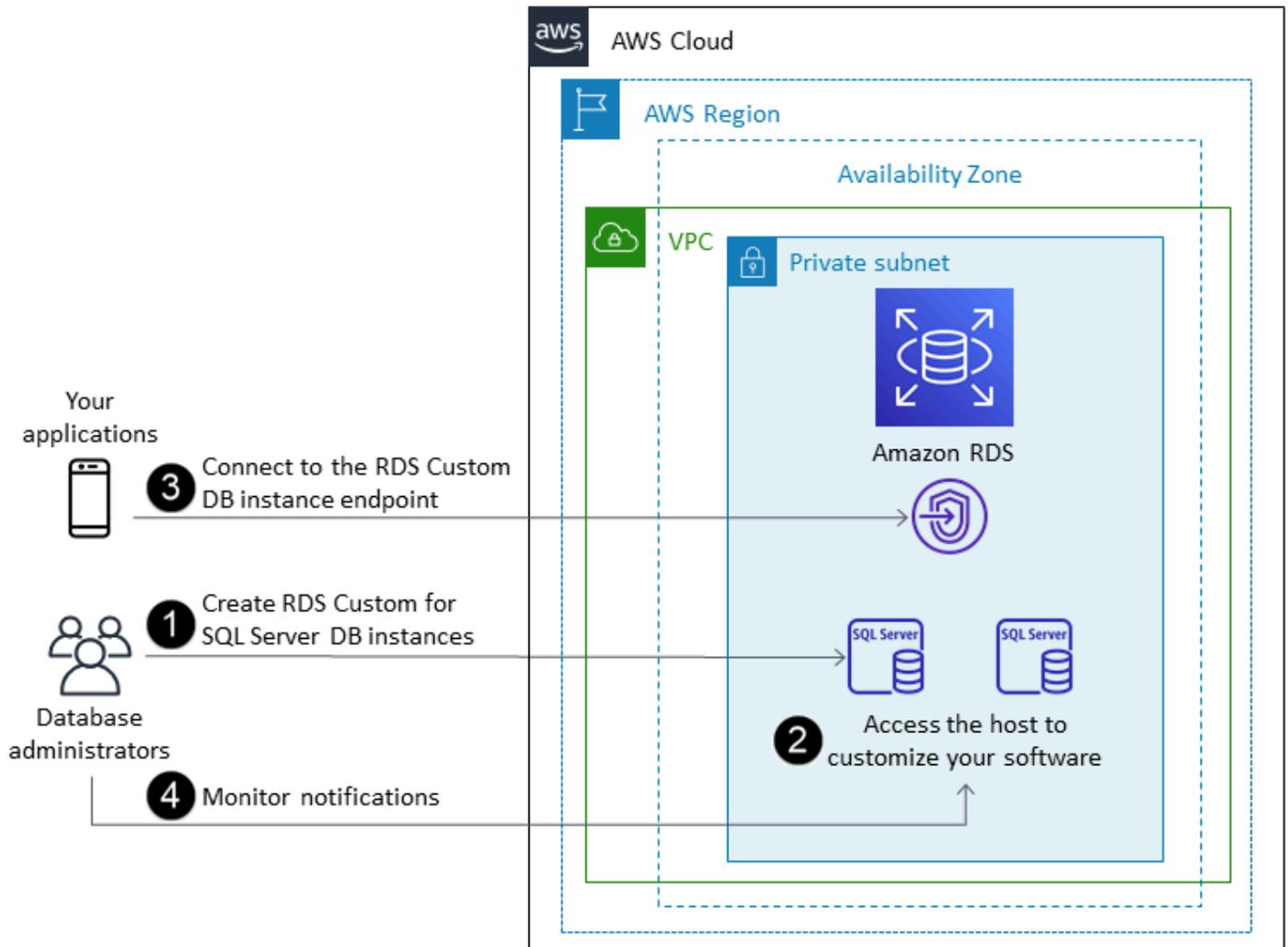
Amazon RDS Custom for SQL Server unterstützt derzeit SQL Server 2022 oder 2019 Enterprise, Standard oder Developer Edition mit den [unterstützten DB-Instance-Klassen](#), die in der Dokumentation aufgeführt sind. Weitere Informationen finden Sie unter [Erstellen einer benutzerdefinierten RDS-DB-Instance für SQL Server](#).

2. Connect Ihre Anwendung mit dem Amazon RDS Custom DB-Instance-Endpoint.

Weitere Informationen finden Sie unter [Herstellen einer Verbindung zu Ihrer benutzerdefinierten RDS-Instance mithilfe AWS Systems Manager](#) und [Herstellen einer Verbindung zu Ihrer benutzerdefinierten RDS-DB-Instance mithilfe von RDP](#).

3. (Optional) Greifen Sie auf den Host zu, um Ihre Software anzupassen.
4. Überwachen Sie Benachrichtigungen und Nachrichten, die von Amazon RDS Custom Automation generiert wurden.

Weitere Informationen zu diesen Schritten finden Sie in der [Amazon RDS Custom-Dokumentation](#).



In Amazon RDS Custom for SQL Server können Sie Software installieren, um benutzerdefinierte Anwendungen und Agenten auszuführen. Da Sie privilegierten Zugriff auf den Host haben, können Sie Dateisysteme ändern, um ältere Anwendungen zu unterstützen. Sie können auch benutzerdefinierte Datenbank-Patches anwenden oder Betriebssystempakete auf Ihren Amazon RDS Custom DB-Instances ändern.

Amazon RDS Custom bietet automatisch Überwachung, Backups und Instance-Wiederherstellung und stellt sicher, dass Ihre DB-Instance eine unterstützte AWS Infrastruktur, ein Betriebssystem und eine Datenbank verwendet. Wenn Sie Ihre Instance anpassen möchten, können Sie die Amazon RDS-Automatisierung für bis zu 24 Stunden pausieren und sie dann fortsetzen, wenn Ihre Anpassungsarbeiten abgeschlossen sind. Durch das Anhalten der Automatisierung wird verhindert, dass die Amazon RDS-Automatisierung Ihre Anpassungen direkt beeinträchtigt.

Wenn Sie die Automatisierung wieder aufnehmen, bestimmt der [Support-Perimeter](#), ob Ihre Anpassung der Datenbank- oder Betriebssystemumgebung die benutzerdefinierte Amazon RDS-Automatisierung beeinträchtigt oder unterbricht. Amazon RDS Custom unterstützt Ihre Anpassung der Host- und Datenbankumgebung, solange Ihre Änderungen die DB-Instance nicht außerhalb des Support-Perimeters platzieren. Die Support-Perimeterprüfungen werden standardmäßig alle 30 Minuten durchgeführt und treten auch nach Ereignissen wie dem Löschen von Snapshots oder der Deinstallation des Amazon RDS Custom Agents, der die DB-Instance überwacht, auf. Der Amazon RDS Custom Agent ist eine wichtige Komponente, um die Funktionalität von Amazon RDS Custom sicherzustellen. Wenn Sie den Agenten deinstallieren, führt Amazon RDS Custom nach einer Minute die Support-Perimeterprüfung durch und verschiebt die DB-Instance außerhalb des Support-Perimeters.

Wenn Sie eine Amazon RDS Custom for SQL Server-DB-Instance einrichten, ist die Softwarelizenz enthalten. Das heißt, Sie müssen SQL Server-Lizenzen nicht separat erwerben. Weitere Informationen zur Lizenzierung finden Sie in Abschnitt 10.5 in den [AWS Servicebedingungen](#). Wenn Sie über ein aktives AWS Premium Support-Konto verfügen, können Sie sich bei SQL Server-spezifischen Problemen an den AWS Premium-Support für Amazon RDS Custom wenden.

Amazon RDS Custom for SQL Server wird in einer begrenzten Auswahl von AWS-Regionen und mit begrenzten DB-Instance-Klassen unterstützt. Informationen zu diesen und anderen Einschränkungen finden Sie auf der Seite mit den [Anforderungen und Einschränkungen](#) in der Dokumentation zu Amazon RDS Custom for SQL Server.

Wenn Sie über eine lokale SQL Server-Datenbank verfügen, können Sie den in der [Amazon RDS-Dokumentation](#) beschriebenen Prozess befolgen, um sie mithilfe systemeigener Sicherungs- und Wiederherstellungsprogramme zu Amazon RDS Custom for SQL Server zu migrieren.

Weitere Informationen finden Sie in den folgenden Ressourcen:

- [Neu — Amazon RDS Custom for SQL Server ist allgemein verfügbar](#) (AWS News-Blog)
- [Konfiguration der SQL Server-Replikation zwischen Amazon RDS Custom for SQL Server und Amazon RDS for SQL Server](#) (AWS Datenbank-Blog)
- [Automatisieren Sie die lokale Migration oder die Migration von Amazon EC2 SQL Server zu Amazon RDS for SQL Server mithilfe von benutzerdefiniertem Protokollversand](#) (AWS Datenbank-Blog)
- [Konfiguration von Hochverfügbarkeit mit Always-On-Verfügbarkeitsgruppen auf Amazon RDS Custom for SQL Server](#) (AWS Datenbank-Blog)

- [Erste Schritte mit Amazon RDS Custom for SQL Server mithilfe einer AWS CloudFormation Vorlage \(Netzwerk-Setup\)](#) (AWS Datenbank-Blog)
- [Migrieren Sie lokale SQL Server-Workloads mithilfe verteilter Verfügbarkeitsgruppen zu Amazon RDS Custom for SQL Server](#) (AWS Datenbank-Blog)
- [Optimieren Sie Ihre SQL Server-Kosten, indem Sie Bring Your Own Media \(BYOM\) auf Amazon RDS Custom for SQL Server verwenden](#) (AWS Datenbank-Blog)

## Amazon EC2 für SQL Server

Amazon EC2 unterstützt eine selbstverwaltete SQL Server-Datenbank. Das heißt, Sie haben damit die volle Kontrolle über die Einrichtung der Infrastruktur und der Datenbankumgebung. Das Ausführen der Datenbank auf Amazon EC2 ist dem Ausführen der Datenbank auf Ihrem eigenen Server sehr ähnlich. Sie haben die volle Kontrolle über die Datenbank und den Zugriff auf Betriebssystemebene, sodass Sie die Tools Ihrer Wahl verwenden können, um das Betriebssystem, die Datenbanksoftware, Patches, Datenreplikation, Sicherung und Wiederherstellung zu verwalten. Für diese Migrationsoption müssen Sie alle Komponenten, einschließlich EC2-Instances, Speichervolumen, Skalierbarkeit, Netzwerk und Sicherheit, auf der Grundlage von bewährten AWS Architekturpraktiken einrichten, konfigurieren, verwalten und optimieren. Sie sind für die Datenreplikation und -wiederherstellung in Ihren Instances in derselben oder in verschiedenen AWS Regionen verantwortlich.

Weitere Informationen zur Migration von SQL Server zu Amazon EC2 finden Sie in den [Rehost-Mustern](#) auf der AWS Prescriptive Guidance-Website.

## Wann sollten Sie sich für Amazon EC2 entscheiden

Amazon EC2 ist eine gute Migrationsoption für Ihre SQL Server-Datenbank, wenn:

- Sie benötigen die volle Kontrolle über die Datenbank und Zugriff auf das zugrunde liegende Betriebssystem, die Datenbankinstallation und -konfiguration.
- Sie möchten Ihre Datenbank verwalten, einschließlich Backups und Wiederherstellung, Patches des Betriebssystems und der Datenbank, Optimieren der Betriebssystem- und Datenbankparameter, Sicherheitsmanagement und Konfiguration von Hochverfügbarkeit oder Replikation.
- Sie möchten Funktionen und Optionen verwenden, die derzeit nicht von Amazon RDS unterstützt werden. Einzelheiten finden Sie unter [Nicht unterstützte Funktionen und Funktionen mit eingeschränkter Unterstützung](#) in der Amazon RDS-Dokumentation.

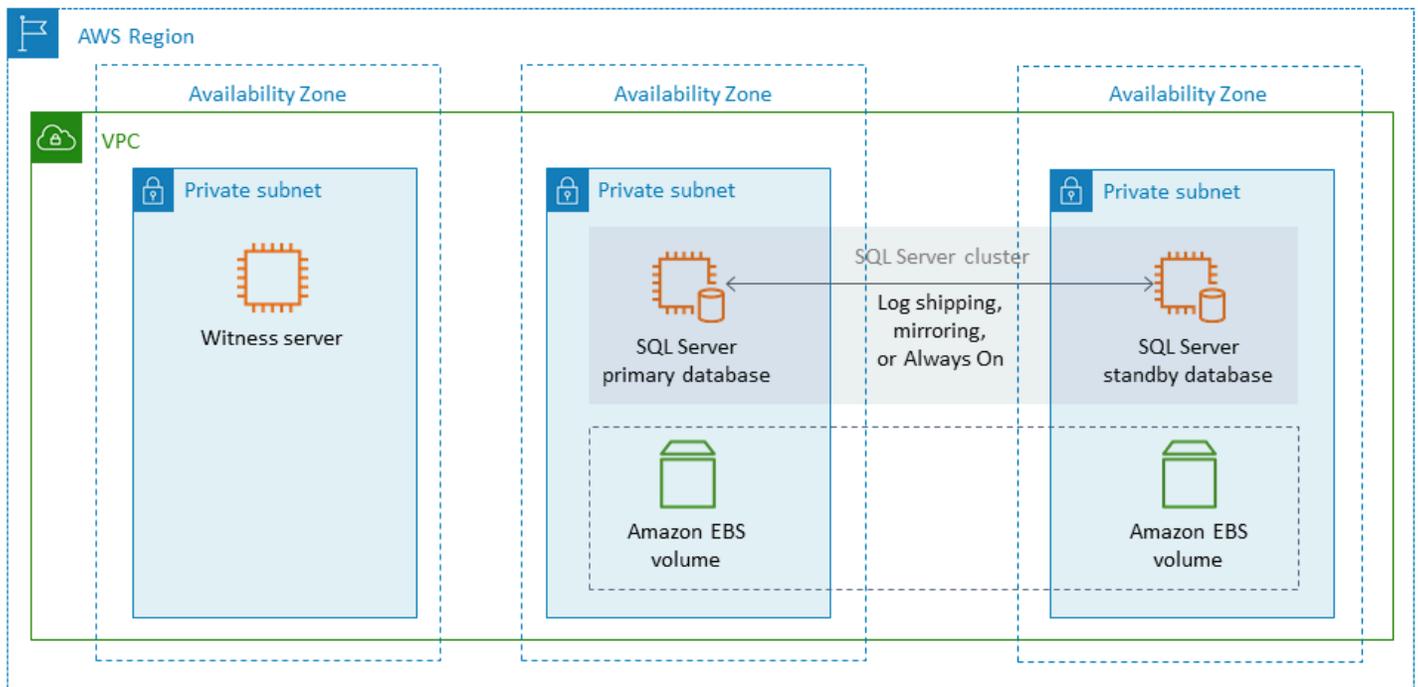
- Sie benötigen eine bestimmte SQL Server-Version, die von Amazon RDS nicht unterstützt wird. Eine Liste der unterstützten Versionen und Editionen finden Sie in der [Amazon RDS-Dokumentation unter SQL Server-Versionen auf Amazon RDS](#).
- Ihre Anforderungen an Datenbankgröße und Leistung übertreffen die aktuellen Angebote von Amazon RDS for SQL Server. Einzelheiten finden Sie unter [Amazon RDS-DB-Instance-Speicher](#) in der Amazon RDS-Dokumentation.
- Sie möchten automatische Softwarepatches vermeiden, die möglicherweise nicht mit Ihren Anwendungen kompatibel sind.
- Sie möchten Ihre eigene Lizenz mitbringen, anstatt das in der Lizenz enthaltene Modell von Amazon RDS for SQL Server zu verwenden.
- Sie möchten IOPS und Speicherkapazität erreichen, die über den aktuellen Grenzwerten liegen. Einzelheiten finden Sie unter [Amazon RDS-DB-Instance-Speicher](#) in der Amazon RDS-Dokumentation.

Eine Liste der derzeit auf Amazon EC2 unterstützten SQL Server-Funktionen und -Versionen finden Sie weiter unten in diesem [Handbuch unter Choosing between Amazon EC2 and Amazon RDS](#).

## Hohe Verfügbarkeit

Sie können jede von SQL Server unterstützte Replikationstechnologie mit Ihrer SQL Server-Datenbank auf Amazon EC2 verwenden, um Hochverfügbarkeit, Datenschutz und Notfallwiederherstellung zu erreichen. Einige der gängigen Lösungen sind Protokollversand, Datenbankspiegelung, AlwaysOn-Verfügbarkeitsgruppen und AlwaysOn-Failover-Cluster-Instances.

Das folgende Diagramm zeigt, wie Sie SQL Server auf Amazon EC2 in mehreren Availability Zones innerhalb einer einzigen AWS Region verwenden können. Die primäre Datenbank ist eine Datenbank mit Lese-/Schreibzugriff, und die sekundäre Datenbank ist mit Protokollversand, Datenbankspiegelung oder Always-On-Verfügbarkeitsgruppen für hohe Verfügbarkeit konfiguriert. Alle Transaktionsdaten aus der Primärdatenbank werden übertragen und können asynchron für den Protokollversand und asynchron für Always-On-Verfügbarkeitsgruppen und Spiegelung auf die sekundäre Datenbank angewendet werden.



## Versand protokollieren

Mit dem Protokollversand können Sie Transaktionsprotokoll-Backups automatisch von einer primären Datenbank-Instance an eine oder mehrere sekundäre Datenbanken (auch bekannt als Warm-Standby) auf separaten DB-Instances senden. Beim Protokollversand werden Aufträge des SQL Server-Agents verwendet, um das Sichern, Kopieren und Anwenden der Transaktionsprotokollsicherungen zu automatisieren. Obwohl der Protokollversand in der Regel als Disaster Recovery-Funktion betrachtet wird, kann er auch für hohe Verfügbarkeit sorgen, da sekundäre DB-Instances heraufgestuft werden können, falls die primäre DB-Instance ausfällt. Wenn Ihr RTO und RPO flexibel sind oder Ihre Datenbanken nicht als äußerst geschäftskritisch angesehen werden, sollten Sie den Protokollversand in Betracht ziehen, um eine bessere Verfügbarkeit Ihrer SQL Server-Datenbanken zu gewährleisten.

Der Protokollversand erhöht die Verfügbarkeit von Datenbanken, indem er den Zugriff auf sekundäre Datenbanken ermöglicht, die bei Bedarf als schreibgeschützte Kopien der Primärdatenbank verwendet werden können. Sie können eine Verzögerung (eine längere Verzögerungszeit) konfigurieren, während der Sie versehentlich geänderte Daten in der Primärdatenbank wiederherstellen können, bevor diese Änderungen an die Sekundärdatenbank gesendet werden.

Wir empfehlen, die primären und sekundären DB-Instances in separaten Availability Zones auszuführen und eine Monitor-Instance bereitzustellen, um alle Details des Protokollversands nachzuverfolgen. Backup-, Kopier-, Wiederherstellungs- und Fehlerereignisse für eine

Protokollversandgruppe sind in der Monitor-Instance verfügbar. Bei einer Konfiguration für den Protokollversand wird nicht automatisch ein Failover vom Primärserver auf den Sekundärserver durchgeführt. Jede der sekundären Datenbanken kann jedoch manuell online geschaltet werden, wenn die Primärdatenbank nicht mehr verfügbar ist.

Der Protokollversand wird häufig als Notfallwiederherstellungslösung verwendet, kann aber je nach Ihren Anwendungsanforderungen auch als Hochverfügbarkeitslösung verwendet werden. Verwenden Sie den Protokollversand in folgenden Fällen:

- Sie haben flexible RTO- und RPO-Anforderungen. Der Versand von Protokollen bietet ein RPO von Minuten und ein RTO von Minuten bis Stunden.
- Sie benötigen kein automatisches Failover auf die sekundäre Datenbank.
- Sie möchten aus der sekundären Datenbank lesen, benötigen aber keine Lesbarkeit während eines Wiederherstellungsvorgangs.

Weitere Informationen zum Protokollversand finden Sie in der [Microsoft SQL Server-Dokumentation](#).

## Datenbankspiegelung

Bei der Datenbankspiegelung wird eine Datenbank, die sich auf einer EC2-Instance befindet, auf einer separaten DB-Instance vollständig oder fast vollständig schreibgeschützt (Mirror) bereitgestellt. Amazon RDS verwendet Datenbankspiegelung, um Multi-AZ-Unterstützung für Amazon RDS for SQL Server bereitzustellen. Diese Funktion erhöht die Verfügbarkeit und den Schutz von Datenbanken und bietet einen Mechanismus, um Datenbanken während Upgrades verfügbar zu halten.

### Note

Laut der [Microsoft-Dokumentation](#) wird die Datenbankspiegelung in einer future Version von SQL Server entfernt. Sie sollten stattdessen planen, AlwaysOn-Verfügbarkeitsgruppen zu verwenden.

Bei der Datenbankspiegelung können SQL-Server eine von drei Rollen übernehmen:

- Der Prinzipalserver, der die primäre Lese-/Schreibversion der Datenbank hostet.
- Der Spiegelserver, der eine Kopie der Prinzipaldatenbank hostet.

- Ein optionaler Zeugenserver. Dieser Server ist nur im Hochsicherheitsmodus verfügbar. Er überwacht den Status des Datenbankspiegels und automatisiert den Failover von der Primärdatenbank zur Spiegeldatenbank.

Zwischen dem Principal- und dem Spiegelserver wird eine Spiegelungssitzung eingerichtet. Während der Spiegelung werden alle Datenbankänderungen, die in der Prinzipaldatenbank vorgenommen werden, auch in der Spiegeldatenbank ausgeführt. Die Datenbankspiegelung kann entweder ein synchroner oder ein asynchroner Vorgang sein. Dies wird durch zwei Betriebsmodi für die Spiegelung bestimmt: den Hochsicherheitsmodus und den Hochleistungsmodus.

- Hochsicherheitsmodus: Dieser Modus verwendet synchrone Operationen. In diesem Modus synchronisiert die Datenbankspiegelungssitzung die Einfüge-, Aktualisierungs- und Löschvorgänge aus der Prinzipaldatenbank so schnell wie möglich mit der Spiegeldatenbank. Sobald die Datenbank synchronisiert ist, wird die Transaktion sowohl in der Principal- als auch in der Spiegeldatenbank festgeschrieben. Wir empfehlen, diesen Betriebsmodus zu verwenden, wenn sich die Spiegeldatenbanken in derselben oder unterschiedlichen Availability Zones befinden, aber in derselben AWS Region gehostet werden.
- Hochleistungsmodus: In diesem Modus werden asynchrone Operationen verwendet. In diesem Modus synchronisiert die Datenbankspiegelungssitzung die Einfüge-, Aktualisierungs- und Löschvorgänge von der Prinzipaldatenbank mit der Spiegeldatenbank. Es kann jedoch zu einer Verzögerung zwischen dem Zeitpunkt, zu dem die Prinzipaldatenbank Transaktionen festschreibt, und dem Zeitpunkt, zu dem die Spiegeldatenbank Transaktionen festschreibt, auftreten. Wir empfehlen, diesen Modus zu verwenden, wenn sich die Spiegeldatenbanken in verschiedenen Regionen befinden. AWS

Verwenden Sie die Datenbankspiegelung, wenn:

- Sie haben strenge RTO- und RPO-Anforderungen und dürfen keine Verzögerungen zwischen der primären und der sekundären Datenbank haben. Die Datenbankspiegelung bietet ein RPO von null Sekunden (mit synchronem Commit) und ein RTO von Sekunden bis Minuten.
- Sie müssen nicht aus der sekundären Datenbank lesen.
- Sie möchten ein automatisches Failover durchführen, wenn Sie einen Zeugenserver im Synchronisierungsmodus konfiguriert haben.
- Sie können keine AlwaysOn-Verfügbarkeitsgruppen verwenden, was die bevorzugte Option ist.

## Einschränkungen:

- Nur one-to-one Failover wird unterstützt. Es ist nicht möglich, mehrere Datenbankziele mit der Primärdatenbank zu synchronisieren.

Weitere Informationen zur Spiegelung finden Sie in der [Microsoft SQL Server-Dokumentation](#).

## AlwaysOn-Verfügbarkeitsgruppen

SQL Server Always-On-Verfügbarkeitsgruppen bieten Hochverfügbarkeits- und Notfallwiederherstellungslösungen für SQL Server-Datenbanken. Eine Verfügbarkeitsgruppe besteht aus einer Reihe von Benutzerdatenbanken, für die gemeinsam ein Failover ausgeführt wird. Sie umfasst einen einzelnen Satz primärer Lese-/Schreibdatenbanken und mehrere (ein bis acht) Sätze verwandter, sekundärer Datenbanken. Sie können die sekundären Datenbanken der Anwendungsebene als schreibgeschützte Kopien der Primärdatenbanken zur Verfügung stellen (nur SQL Server Enterprise Edition), um eine Scale-Out-Architektur für Lese-Workloads bereitzustellen. Sie können die sekundären Datenbanken auch für Sicherungsvorgänge verwenden.

SQL Server Always On-Verfügbarkeitsgruppen unterstützen sowohl den synchronen als auch den asynchronen Commit-Modus. Im synchronen Modus schreibt das primäre Replikat Datenbanktransaktionen fest, nachdem die Änderungen festgeschrieben oder in das Protokoll des sekundären Replikats geschrieben wurden. In diesem Modus können Sie ein geplantes manuelles Failover und ein automatisches Failover durchführen, wenn die Replikate synchron sind. Sie können den synchronen Commit-Modus zwischen SQL Server-Instanzen innerhalb derselben Umgebung verwenden (z. B. wenn sich alle Instanzen lokal oder alle Instanzen in einer lokalen Umgebung befinden). AWS

Im asynchronen Commit-Modus schreibt das primäre Replikat Datenbanktransaktionen fest, ohne auf das sekundäre Replikat zu warten. Sie können den asynchronen Commit-Modus zwischen SQL Server-Instanzen verwenden, die sich in unterschiedlichen Umgebungen befinden (z. B. wenn Sie Instanzen vor Ort und in der Umgebung haben). AWS

Sie können AlwaysOn-Verfügbarkeitsgruppen für Hochverfügbarkeit oder Notfallwiederherstellung verwenden. Verwenden Sie diese Methode, wenn:

- Sie haben strenge RTO- und RPO-Anforderungen. AlwaysOn-Verfügbarkeitsgruppen bieten einen RPO-Wert von Sekunden und einen RTO-Wert von Sekunden bis Minuten.

- Sie möchten eine Gruppe von Datenbanken verwalten und ein Failover durchführen. AlwaysOn-Verfügbarkeitsgruppen unterstützen 0-4 sekundäre Replikate im synchronen Commit-Modus für SQL Server 2019.
- Sie möchten den automatischen Failover im synchronen Commit-Modus verwenden und benötigen keinen Zeugenserver.
- Sie möchten aus der sekundären Datenbank lesen.
- Sie möchten mehrere Datenbankziele mit Ihrer Primärdatenbank synchronisieren.

Ab SQL Server 2016 SP1 bietet die SQL Server Standard Edition grundlegende Hochverfügbarkeit für eine einzelne, nicht lesbare sekundäre Datenbank und einen Listener pro Verfügbarkeitsgruppe. Sie unterstützt außerdem maximal zwei Knoten pro Verfügbarkeitsgruppe.

## Immer aktive Failover-Clusterinstanzen

SQL Server Always On Failover Cluster Instances (FCIs) verwenden Windows Server Failover Clustering (WSFC), um Hochverfügbarkeit auf Serverinstanzebene bereitzustellen. Eine FCI ist eine einzelne Instanz von SQL Server, die auf allen WSFC-Knoten installiert wird, um eine hohe Verfügbarkeit für die gesamte Installation von SQL Server zu gewährleisten. Wenn auf dem zugrunde liegenden Knoten Hardware-, Betriebssystem-, Anwendungs- oder Dienstauffälle auftreten, wird alles innerhalb der SQL Server-Instanz auf einen anderen WSFC-Knoten verschoben. Dazu gehören Systemdatenbanken, SQL Server-Logins, SQL Server-Agent-Jobs und Zertifikate.

Eine FCI ist im Allgemeinen einer Always-On-Verfügbarkeitsgruppe vorzuziehen, wenn:

- Sie verwenden die SQL Server Standard Edition anstelle der Enterprise Edition.
- Sie haben eine große Anzahl kleiner Datenbanken pro Instanz.
- Sie ändern ständig Objekte auf Instanzebene, wie z. B. SQL Server-Agent-Jobs, Anmeldungen usw.

Es gibt vier Optionen für die Bereitstellung von FCIs auf: AWS

- Amazon EBS Multi-Attach mit dauerhaften Reservierungen
- Amazon FSx für Windows File Server
- Amazon FSx für ONTAP NetApp
- Lösungen von Partnern AWS

## Amazon EBS Multi-Attach mit dauerhaften Reservierungen verwenden

[Amazon EBS Multi-Attach mit NVMe-Reservierungen](#) unterstützt die Erstellung von SQL Server-FCIs mit Amazon `io2` EBS-Volumes als gemeinsam genutztem Speicher auf Windows Server-Failoverclustern. Diese Funktion vereinfacht den Einrichtungsprozess für Failover-Cluster, indem Sie mithilfe von Amazon `io2` EBS-Volumes einen Failover-Cluster erstellen können. Diese Volumes können nur Instances zugeordnet werden, die sich in derselben Availability Zone befinden. Um Windows Server-Failover-Cluster mithilfe von Amazon `io2` EBS-Volumes bereitzustellen, müssen Sie die neuesten AWS NVMe-Treiber verwenden.

Amazon-EBS- und -Instance-Speicher-Volumes werden als NVMe-Blockgeräte auf [Nitro-basierenden Instances](#) angezeigt. Sie müssen den [AWS NVMe-Treiber](#) installiert und die [persistente SCSI-Reservierungsfunktion](#) konfiguriert haben, wenn Sie Amazon `io2` EBS-Volumes verwenden, um WSFC- und SQL Server-FCIs zu bilden.

Weitere Informationen zu dieser Funktion finden Sie im AWS Blogbeitrag [So stellen Sie einen SQL Server-Failover-Cluster mit Amazon EBS Multi-Attach auf](#) Windows Server bereit.

## Verwenden von Amazon FSx for Windows File Server

[Amazon FSx for Windows File Server](#) bietet vollständig verwalteten gemeinsamen Dateispeicher. Es repliziert den Speicher automatisch synchron über zwei Availability Zones, um eine hohe Verfügbarkeit zu gewährleisten. Die Verwendung von FSx for Windows File Server für die Dateispeicherung hilft dabei, SQL Server-Hochverfügbarkeitsbereitstellungen auf Amazon EC2 zu vereinfachen und zu optimieren.

Bei Microsoft SQL Server wird Hochverfügbarkeit in der Regel über mehrere Datenbankknoten in einem WSFC bereitgestellt, und jeder Knoten hat Zugriff auf gemeinsam genutzten Dateispeicher. Sie können FSx for Windows File Server auf zwei Arten als gemeinsam genutzten Speicher für SQL Server-Hochverfügbarkeitsbereitstellungen verwenden: als Speicher für aktive Datendateien und als SMB-Dateifreigabezeuge.

Informationen darüber, wie Sie die Komplexität und die Kosten der Ausführung von SQL Server FCI-Bereitstellungen mithilfe von FSx für Windows File Server reduzieren können, finden Sie im Blogbeitrag [Simplify your Microsoft SQL Server-Hochverfügbarkeitsbereitstellungen mithilfe von Amazon FSx](#) for Windows File Server. Der Blogbeitrag enthält auch step-by-step Anweisungen für die Bereitstellung von SQL Server-FCIs mithilfe eines Amazon FSx Multi-AZ-Dateisystems als gemeinsam genutzte Speicherlösung. Weitere Informationen finden Sie in der Dokumentation zu [Amazon FSx for Windows File Server](#).

## Amazon FSx für NetApp ONTAP verwenden

Amazon FSx for NetApp ONTAP ist ein vollständig verwalteter Service, der äußerst zuverlässigen, skalierbaren, leistungsstarken und funktionsreichen Dateispeicher bietet, der auf dem ONTAP-Dateisystem basiert. NetApp FSx for ONTAP kombiniert die vertrauten Funktionen, Leistungen, Fähigkeiten und API-Operationen von NetApp Dateisystemen mit der Agilität, Skalierbarkeit und Einfachheit eines vollständig verwalteten AWS Dienstes.

FSx for ONTAP bietet Multiprotokollzugriff auf Daten über die NFS-, SMB- und iSCSI-Protokolle für Windows- und Linux-Systeme. Sie können eine hochverfügbare SQL Server Always On FCi-Architektur erstellen, wie im Blogbeitrag [SQL Server High Availability Deployments Using Amazon FSx for ONTAP](#) ausführlich erklärt. NetApp FSx for ONTAP bietet auch eine schnelle Möglichkeit, Ihre SQL Server-Umgebung auf eine andere umzuschalten, um die Anforderungen AWS-Region an Recovery Time Objective (RTO) und Recovery Point Objective (RPO) zu erfüllen. Weitere Informationen finden Sie im Blogbeitrag [Implementieren von HA und DR für SQL Server Always-On Failover Cluster Instance using FSx for ONTAP](#).

Sie können es auch für AWS Launch Wizard die Bereitstellung von SQL Server-Lösungen verwenden AWS, wobei Always-On-Verfügbarkeitsgruppen und Einzelknotenbereitstellungen unterstützt werden. Launch Wizard unterstützt die Bereitstellung für SQL Server Always on FCIs auf Amazon EC2 mit FSx for ONTAP als gemeinsam genutztem Speicher. Dieser Service spart Ihnen Zeit und Mühe, indem er einen komplexen manuellen Bereitstellungsprozess durch einen geführten, konsolenbasierten Assistenten ersetzt, der die Migration Ihrer lokalen SQL Server-Workloads beschleunigt, die auf gemeinsam genutzten Speicher angewiesen sind. Weitere Informationen darüber, wie Launch Wizard Ihnen helfen kann, SQL Server-FCIs innerhalb weniger Stunden bereitzustellen und zu konfigurieren, finden Sie im Blogbeitrag [Simplify SQL Server Always On-Bereitstellungen mit AWS Launch Wizard und Amazon FSx](#). Launch Wizard unterstützt auch die Bereitstellung für SQL Server Always On FCIs, indem [Amazon FSx for Windows File Server](#) als gemeinsam genutzte Speicherlösung verwendet wird.

## Verwendung von Lösungen von Partnern AWS

- [SIOS DataKeeper](#) bietet Hochverfügbarkeits-Cluster-Failover-Unterstützung für alle Availability AWS-Regionen Zones. SIOS DataKeeper ist verfügbar in [AWS Marketplace](#)
- [DxEnterprise](#) von DH2i ermöglicht ein vollautomatisches Failover von SQL Server-Verfügbarkeitsgruppen in Kubernetes und ein einheitliches Instanz-Failover für Windows und Linux. D2HI ist verfügbar in [AWS Marketplace](#)

## FSx für Windows File Server

FSx for Windows File Server bietet vollständig verwalteten, äußerst zuverlässigen und skalierbaren Dateispeicher, auf den über das Server Message Block (SMB) -Protokoll zugegriffen werden kann. Es basiert auf Windows Server und bietet eine Vielzahl von Verwaltungsfunktionen wie Benutzerkontingente, Dateiwiederherstellung für Endbenutzer und Microsoft Active Directory (AD) -Integration. Es bietet Single-AZ- und Multi-AZ-Bereitstellungsoptionen, vollständig verwaltete Backups und Verschlüsselung von Daten im Speicher und bei der Übertragung. Mit den Speicheroptionen Solid-State-Drives (SSD) und Festplattenlaufwerke (HDD) können Sie Kosten und Leistung für Ihre Workloads optimieren. Außerdem können Sie den Speicher skalieren und die Durchsatzleistung Ihres Dateisystems jederzeit ändern. Auf den Amazon FSx-Dateispeicher kann von Windows- und Linux-Compute-Instances aus zugegriffen werden AWS, die vor Ort und vor Ort ausgeführt werden.

Amazon FSx erleichtert die Bereitstellung von gemeinsam genutztem Windows-Speicher für SQL Server-Bereitstellungen mit hoher Verfügbarkeit, da es kontinuierlich verfügbare (CA) Dateifreigaben und kleinere Dateisysteme unterstützt. Diese Option eignet sich für die folgenden Anwendungsfälle:

- Als gemeinsam genutzter Speicher, der von SQL Server-Knoten in einer WSFC-Instanz verwendet wird.
- Als SMB-Dateifreigabezeuge, der mit jedem SQL Server-Cluster mit WSFC verwendet werden kann.

Amazon FSx bietet eine schnelle Leistung mit einem Basisdurchsatz von bis zu 2 GB/Sekunde pro Dateisystem, Hunderttausenden von IOPS und konsistenten Latenzen unter einer Millisekunde.

Um die richtige Leistung für Ihre SQL-Instances bereitzustellen, können Sie ein Durchsatzniveau wählen, das unabhängig von der Größe Ihres Dateisystems ist. Höhere Durchsatzkapazitäten bedeuten auch höhere IOPS-Werte, die der Dateiserver für die SQL Server-Instanzen bereitstellen kann, die auf ihn zugreifen.

Die Speicherkapazität bestimmt nicht nur, wie viele Daten Sie speichern können, sondern auch, wie viele IOPS Sie auf dem Speicher ausführen können. Jedes Gigabyte Speicher bietet 3 IOPS. Sie können für jedes Dateisystem eine Größe von bis zu 64 TB bereitstellen.

Informationen zur Konfiguration und Verwendung von Amazon FSx zur Reduzierung der Komplexität und der Kosten Ihrer SQL Server-Hochverfügbarkeitsbereitstellungen finden Sie unter [Vereinfachen Sie Ihre Microsoft SQL Server-Hochverfügbarkeitsbereitstellungen mithilfe von FSx for Windows File](#)

[Server](#) im Storage-Blog. AWS Weitere Informationen zum Erstellen einer neuen CA-Freigabe finden Sie in der [FSx for Windows File Server Server-Dokumentation](#).

## Notfallwiederherstellung

Viele Unternehmen implementieren Hochverfügbarkeit für ihre SQL Server-Datenbanken, aber das ist für Unternehmen, die eine echte IT-Resilienz benötigen, nicht ausreichend. Wir empfehlen Ihnen, eine Notfallwiederherstellungslösung zu implementieren, um Datenverlust und Ausfallzeiten unternehmenskritischer Datenbanken zu vermeiden. Die Einführung einer Disaster Recovery-Architektur für mehrere Regionen für Ihre SQL Server-Bereitstellungen hilft Ihnen:

- Sorgen Sie für Geschäftskontinuität
- Verbessern Sie die Latenz für Ihren geografisch verteilten Kundenstamm
- Erfüllen Sie Ihre Prüfungs- und behördlichen Anforderungen

Zu den Optionen für die Notfallwiederherstellung gehören [Protokollversand](#), [Always-On-Verfügbarkeitsgruppen](#), [Amazon EBS-Snapshots](#), die in Amazon S3 gespeichert und AWS regionsübergreifend repliziert werden, [Always-On-Failover-Cluster-Instances \(FCIs\)](#) in Kombination mit Always-On-Verfügbarkeitsgruppen und verteilte Verfügbarkeitsgruppen.

### Verteilte Verfügbarkeitsgruppen

Eine Architektur mit verteilten Verfügbarkeitsgruppen ist ein optimaler Ansatz für die Bereitstellung von SQL Server in mehreren Regionen. Eine verteilte Verfügbarkeitsgruppe ist ein besonderer Typ von Verfügbarkeitsgruppe, die sich über zwei separate Verfügbarkeitsgruppen erstreckt. Sie können sie sich als eine Verfügbarkeitsgruppe von Verfügbarkeitsgruppen vorstellen. Die zugrunde liegenden Verfügbarkeitsgruppen sind auf zwei verschiedenen WSFC-Clustern konfiguriert.

Verteilte Verfügbarkeitsgruppen sind lose miteinander verknüpft, was bedeutet, dass sie keinen einzigen WSFC-Cluster benötigen und von SQL Server verwaltet werden. Da die WSFC-Cluster einzeln verwaltet werden und Übertragungen hauptsächlich asynchron zwischen zwei Verfügbarkeitsgruppen erfolgen, ist es einfacher, die Notfallwiederherstellung an einem anderen Standort zu konfigurieren. Die primären Replikate in jeder Verfügbarkeitsgruppe synchronisieren ihre eigenen sekundären Replikate.

Verteilte Verfügbarkeitsgruppen unterstützen derzeit nur manuelles Failover. Um sicherzustellen, dass keine Daten verloren gehen, beenden Sie alle Transaktionen in den globalen

Primärdatenbanken (d. h. in den Datenbanken der primären Verfügbarkeitsgruppe). Stellen Sie dann die verteilte Verfügbarkeitsgruppe auf synchrones Commit ein.

## VMware Cloud on AWS für SQL Server

### Notice (Hinweis)

Seit dem 30. April 2024 AWS wird VMware Cloud on nicht mehr von AWS oder seinen Channel-Partnern weiterverkauft. Der Service wird weiterhin über Broadcom verfügbar sein. Wir empfehlen Ihnen, sich für weitere Informationen an Ihren AWS Vertreter zu wenden.

[VMware Cloud on AWS](#) ist ein integriertes Cloud-Angebot, das gemeinsam von VMware AWS und VMware entwickelt wurde. SQL Server lässt sich problemlos in VMware Cloud on integrieren AWS. Diese Migrationsoption ermöglicht es Ihnen, auf Ihren bestehenden Investitionen in Virtualisierung aufzubauen.

Sie können stündlich, AWS auf Abruf oder in Abonnementform auf VMware Cloud zugreifen. Es umfasst dieselben VMware-Kerntechnologien, die Sie in Ihren Rechenzentren ausführen, darunter vSphere Hypervisor (ESXi), Virtual SAN (vSAN) und die NSX-Netzwerkvirtualisierungsplattform, und wurde entwickelt, um eine effiziente, nahtlose Erfahrung bei der Verwaltung Ihrer SQL Server-Datenbanken zu bieten. Sie können den Speicher, die Rechenleistung und den Arbeitsspeicher Ihrer SQL Server-Datenbanken auf VMware Cloud On innerhalb weniger Minuten skalieren. AWS

VMware Cloud on AWS wird direkt auf der physischen Hardware ausgeführt, nutzt jedoch die Netzwerk- und Hardwarefunktionen, die zur Unterstützung des Infrastrukturmodells entwickelt wurden, bei dem AWS Sicherheit an erster Stelle steht. Das bedeutet, dass der VMware-Virtualisierungstapel auf der AWS Infrastruktur ausgeführt wird, ohne dass verschachtelte Virtualisierung verwendet werden muss.

AWS Mit VMware Cloud on ist es einfach, Ihre SQL Server-Datenbank-Workloads einzurichten, zu skalieren und zu betreiben. AWS Es bietet Hochverfügbarkeitslösungen, lässt sich in das lokale Active Directory integrieren und bietet Zugriff auf AWS Dienste wie AWS Directory Service for Microsoft Active Directory AD Connector, Amazon Route 53 CloudWatch, Amazon und Amazon S3. Sie können Ihre Backups in Amazon S3 speichern und Ihren Disaster Recovery-Prozess modernisieren und vereinfachen.

## Wann sollten Sie sich für VMware Cloud entscheiden AWS

VMware Cloud on AWS ist eine Option für Ihre SQL Server-Datenbank, wenn:

- Ihre SQL Server-Datenbanken werden bereits in einem lokalen Rechenzentrum in einer virtualisierten vSphere-Umgebung ausgeführt.
- Sie haben eine große Anzahl von Datenbanken und benötigen aus einem der folgenden Gründe eine schnelle Migration (z. B. nur wenige Stunden) zur Cloud, ohne dass das Migrationsteam zusätzliche Arbeit leisten muss:
  - Erweiterung des Rechenzentrums. Sie benötigen On-Demand-Kapazität, um virtualisierte Desktops auszuführen, Anwendungen zu veröffentlichen oder eine Entwicklungs-/Testumgebung bereitzustellen.
  - Wiederherstellung im Notfall. Sie möchten ein neues Disaster Recovery-System einrichten oder Ihr bestehendes System ersetzen.
  - Cloud-Migration. Sie möchten Ihr gesamtes Rechenzentrum in die Cloud migrieren oder Ihre Infrastruktur auffrischen.

Wenn Ihre SQL Server-Datenbank mehr als 80.000 IOPS benötigt, können Sie vSAN verwenden.

Weitere Informationen finden Sie [im AWS News-Blog unter In the Works — VMware Cloud AWS on](#) und [Deploy Microsoft SQL Server on VMware Cloud on VMware Cloud AWS](#) auf der AWS Website.

# Heterogene Datenbankmigration für SQL Server

Aufgrund der Innovationen und Verbesserungen bei Open-Source-Datenbanken und Cloud-Computing-Plattformen wechseln viele Unternehmen von proprietären Datenbank-Engines (Online Transaction Processing oder OLTP) wie SQL Server zu Open-Source-Engines. AWS SQL Server-Datenbanken sind unternehmenskritische Systeme für jedes Unternehmen, aber an einen bestimmten Anbieter gebunden zu sein, ist eine riskante und kostspielige Situation. Niedrige Betriebskosten und keine Lizenzgebühren sind überzeugende Gründe, die zugrundeliegende Datenbanktechnologie auf Open Source- oder AWS Cloud-native Datenbanken umzustellen.

Weitere Gründe für die Migration von SQL Server sind Anbieterbindungsfristen, Lizenzprüfungen, teure Lizenzierungen und Kosten. Aus diesem Grund entscheiden sich viele Unternehmen bei der Migration dafür, ihre SQL Server-Datenbanken entweder auf Open-Source-Datenbanken (wie PostgreSQL, MySQL oder MariaDB) oder AWS Cloud-native Datenbanken (wie Amazon Aurora oder Amazon DynamoDB) zu migrieren. AWS

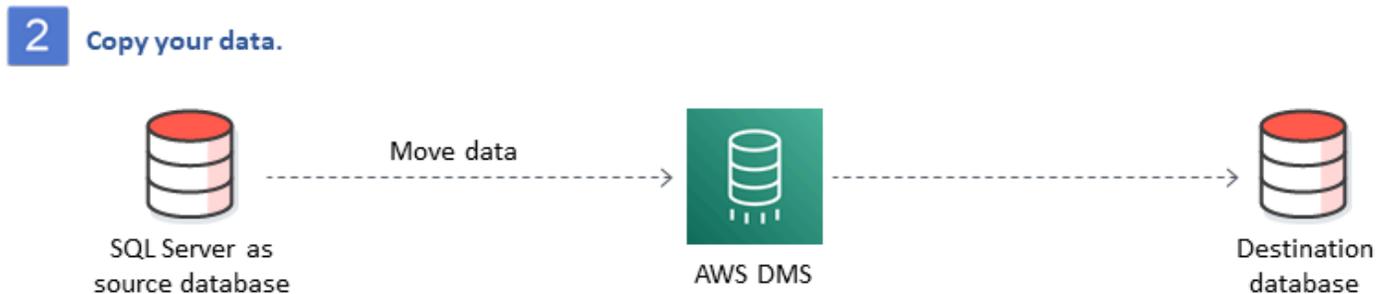
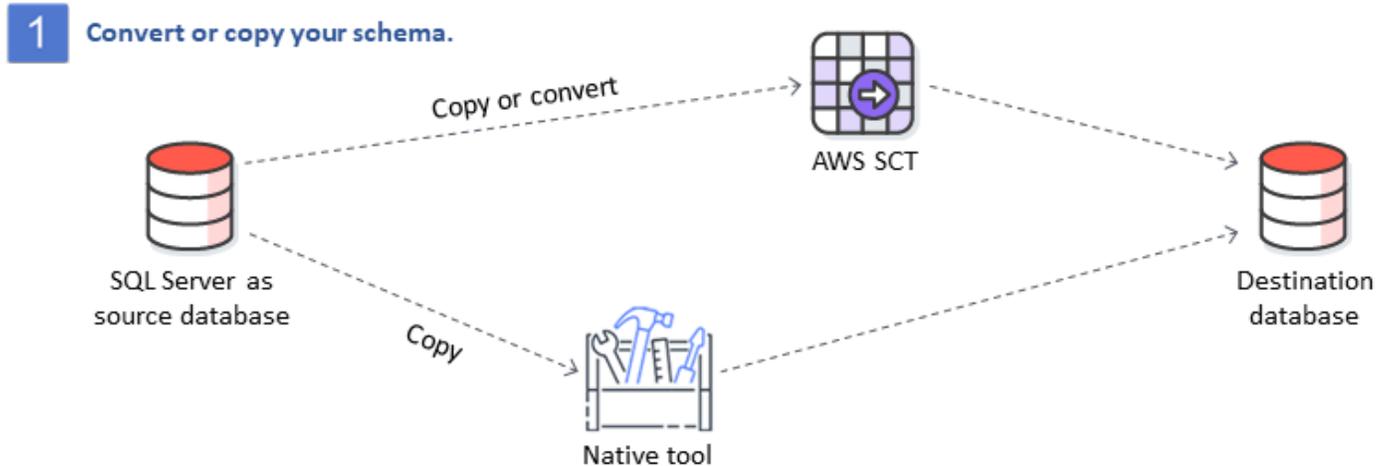
Sie können Ihre SQL Server Data Warehouse-Datenbank auch zu Amazon Redshift migrieren, einem schnellen, vollständig verwalteten Cloud-Data Warehouse. Amazon Redshift ist in Ihren Data Lake integriert, bietet eine bis zu dreimal schnellere Leistung als jedes andere Data Warehouse und kostet bis zu 75 Prozent weniger als jedes andere Cloud-Data Warehouse. Weitere Informationen finden Sie unter dem Muster [Migrieren einer lokalen Microsoft SQL Server-Datenbank zu Amazon Redshift AWS DMS auf der AWS Prescriptive](#) Guidance-Website.

Um zu einer Open-Source-oder AWS Cloud-nativen Datenbank zu migrieren, wählen Sie je nach Datentyp, Zugriffsmodell, Skalierbarkeit, Anwendungspraktiken und Komplexität die richtige Datenbank aus. Die Migration von SQL Server zu PostgreSQL und zu anderen Open-Source-Datenbanken war oft schwierig und zeitaufwändig und erfordert eine sorgfältige Bewertung, Planung und Tests.

Dieser Prozess wird mit Diensten wie AWS Database Migration Service (AWS DMS) und AWS Schema Conversion Tool (AWS SCT) einfacher, die Ihnen helfen, Ihre kommerzielle Datenbank AWS mit minimalen Ausfallzeiten auf eine Open-Source-Datenbank zu migrieren.

Bei heterogenen Datenbankmigrationen unterscheiden sich die Quell- und Zieldatenbank-Engines, wie bei Migrationen von SQL Server zu Aurora oder von SQL Server zu MariaDB. Die Schemastruktur, die Datentypen und der Datenbankcode in den Quell- und Zieldatenbanken können sehr unterschiedlich sein. Daher müssen Schema und Code transformiert werden, bevor die Datenmigration beginnt. Aus diesem Grund ist die heterogene Migration ein zweistufiger Prozess:

- Schritt 1. Konvertiert das Quellschema und den Quellcode so, dass sie mit denen der Zieldatenbank übereinstimmen. Sie können es AWS SCT für diese Konvertierung verwenden.
- Schritt 2. Migrieren Sie Daten von der Quelldatenbank zur Zieldatenbank. Sie können es AWS DMS für diesen Prozess verwenden.



AWS DMSwickelt die wichtigsten Datentypkonvertierungen während der Migration automatisch ab. Die Quelldatenbank kann sich in Ihren eigenen Räumlichkeiten außerhalb befindenAWS, es kann sich um eine Datenbank handeln, die auf einer EC2-Instance ausgeführt wird, oder es kann sich um eine Amazon RDS-Datenbank handeln (siehe [Quellen für die Datenmigration](#) in der AWS DMS Dokumentation). Das Ziel kann eine Datenbank in Amazon EC2, Amazon RDS oder Aurora sein. Informationen zur Verwendung von MySQL als Zieldatenbank finden Sie im [Datenbank-Blog unter Migrieren einer SQL Server-Datenbank zu einer MySQL-kompatiblen Datenbank-Engine](#). AWS

Weitere Informationen zum Refactoring Ihrer SQL Server-Datenbank finden Sie in den [AWS Architektur-Patterns](#) auf der AWS Prescriptive Guidance-Website.

## Tools für heterogene Datenbankmigrationen

Die folgende Tabelle enthält eine Liste von Tools, die Sie für die Migration von SQL Server zu einer anderen Datenbank-Engine verwenden können.

Tool zur Migration	Unterstützung für Zieldatenbanken	Wird verwendet für
<a href="#">AWS SCT</a>	Amazon RDS for MySQL Amazon RDS for PostgreSQL Amazon Aurora MySQL Amazon Aurora PostgreSQL	Schemakonvertierung
<a href="#">AWS DMS</a>	Amazon RDS for MySQL Amazon RDS for PostgreSQL Amazon Aurora MySQL Amazon Aurora PostgreSQL	Datenmigration
<a href="#">Babelfish</a>	Amazon Aurora PostgreSQL	Datenzugriff und Migration

In den folgenden Unterabschnitten finden Sie weitere Informationen zu den einzelnen Tools.

### AWS SCT

[AWS Schema Conversion Tool\(AWS SCT\)](#) konvertiert Ihre vorhandenen kommerziellen Datenbankschemas in eine Open-Source-Engine oder in eine AWS Cloud-native Datenbank. AWS SCT macht heterogene Datenbankmigrationen vorhersehbar, indem das Quelldatenbankschema und ein Großteil der Datenbankcode-Objekte, einschließlich Ansichten, gespeicherter Prozeduren und Funktionen, automatisch in ein Format konvertiert werden, das mit der Zieldatenbank kompatibel ist.

Wenn Sie Ihr Datenbankschema von einer Engine in eine andere konvertieren, müssen Sie auch den SQL-Code in Ihren Anwendungen aktualisieren, um mit der neuen Datenbank-Engine statt mit der alten zu interagieren. AWS SCT konvertiert auch den SQL-Code in C++, C#, Java oder anderen Anwendungscode. Alle Objekte, die nicht automatisch konvertiert werden können, sind deutlich für

die manuelle Konvertierung gekennzeichnet. AWS SCT kann den Quellcode Ihrer Anwendung auch nach eingebetteten SQL-Anweisungen durchsuchen und diese im Rahmen eines Datenbankschema-Konvertierungsprojekts konvertieren. Weitere Informationen finden Sie AWS SCT in der AWS Dokumentation [unter Verwenden von Microsoft SQL Server als Quelle für](#).

## AWS DMS

[AWS Database Migration Service \(AWS DMS\)](#) migriert Ihre Daten schnell und sicher zu. AWS Während der Migration bleibt die Quelldatenbank voll funktionsfähig, wodurch Ausfallzeiten der Anwendung minimiert werden. AWS DMS unterstützt homogene Migrationen wie die Migration von Daten von einer SQL Server-Datenbank zu einer anderen. Es unterstützt auch heterogene Migrationen zwischen verschiedenen Datenbankplattformen, z. B. die Migration Ihrer SQL Server-Datenbank zu einer Open-Source-Datenbank oder zu einer cloudnativen Datenbank. AWS DMS verwaltet die Komplexität des Migrationsprozesses, einschließlich der automatischen Replikation von Datenänderungen, die in der Quelldatenbank vorgenommen werden, in die Zieldatenbank. Nach Abschluss der Datenbankmigration bleibt die Zieldatenbank so lange mit der Quelldatenbank synchronisiert, wie Sie möchten, und Sie können zu einem geeigneten Zeitpunkt zur Zieldatenbank wechseln. Weitere Informationen finden Sie AWS DMS in der AWS Dokumentation [unter Verwenden einer Microsoft SQL Server-Datenbank als Quelle für](#).

## Babelfish

Babelfish ist eine integrierte Funktion von Amazon Aurora. Babelfish für Aurora PostgreSQL ermöglicht es Ihren Aurora PostgreSQL-kompatiblen Edition-Datenbanken, Befehle von Anwendungen zu verstehen, die für Microsoft SQL Server geschrieben wurden. Das Ändern von SQL Server-Anwendungen, deren SQL Server-Datenbankcode in Transact-SQL (T-SQL), dem firmeneigenen SQL-Dialekt von SQL Server, geschrieben ist, erfordert Aufwand und ist zeitaufwändig. [Babelfish für Aurora PostgreSQL](#) macht diesen Prozess immer einfacher. Mit Babelfish müssen Sie keine Änderungen an Ihrem Anwendungscode vornehmen. Stattdessen können Sie Babelfish for Aurora PostgreSQL verwenden, um eine SQL Server-Datenbank zu einem Aurora PostgreSQL-kompatiblen DB-Cluster zu migrieren.

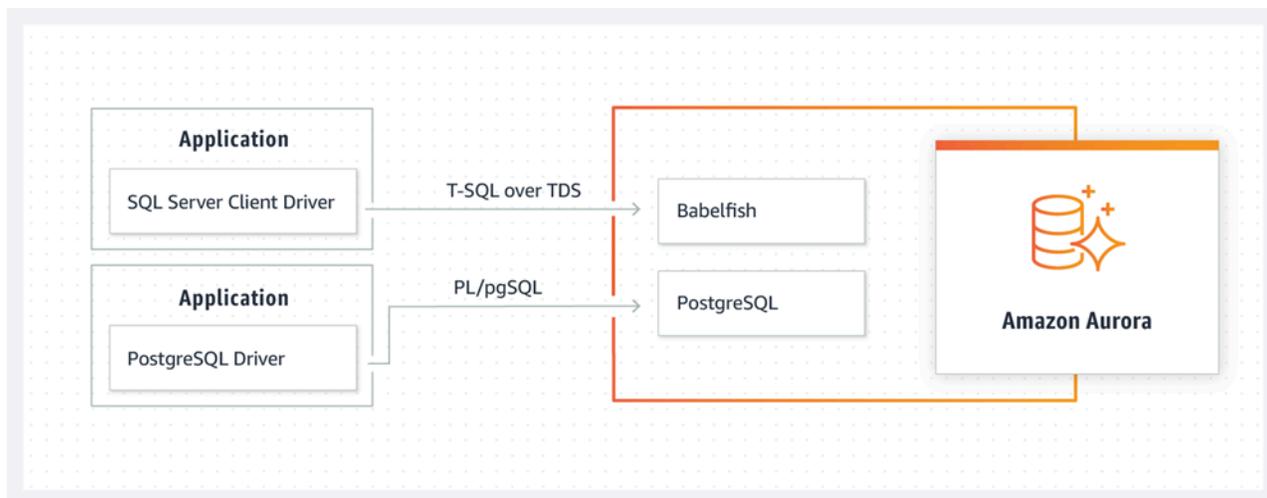
Mit Babelfish versteht Aurora PostgreSQL T-SQL und unterstützt dasselbe Kommunikationsprotokoll, sodass Sie die Datenbanktreiber nicht wechseln oder Ihre Anwendungsabfragen neu schreiben müssen. Ihre Anwendungen, die ursprünglich für SQL Server geschrieben wurden, können jetzt mit Aurora mit weniger Codeänderungen verwendet werden. Dies reduziert den Aufwand für die Änderung und Übertragung von Anwendungen, die auf SQL Server oder neuer ausgeführt werden, nach Aurora, was zu schnelleren, risikoärmeren und kostengünstigeren Migrationen führt.

Wenn Sie von älteren SQL Server-Datenbanken migrieren, können Sie BabelFish verwenden, um SQL Server-Code parallel zu neuen Funktionen auszuführen, die Sie mithilfe nativer PostgreSQL-APIs erstellt haben. BabelFish ermöglicht es Aurora PostgreSQL, mit häufig verwendeten SQL Server-Tools, -Befehlen und -Treibern zu arbeiten.

BabelFish bietet auch Zugriff auf Daten mithilfe der nativen PostgreSQL-Verbindung. Standardmäßig sind beide von BabelFish unterstützten SQL-Dialekte über ihre nativen Wire-Protokolle an den folgenden Ports verfügbar:

- Stellen Sie für den SQL Server-Dialekt (T-SQL) eine Verbindung zu Port 1433 her.
- Stellen Sie für den PostgreSQL-Dialekt (PL/pgSQL) eine Verbindung zu Port 5432 her.

BabelFish ermöglicht es Ihren älteren SQL Server-Anwendungen, ohne umfangreiche Code-Umschreibungen mit Aurora zu kommunizieren, indem Verbindungen vom SQL Server- oder PostgreSQL-Port aus bereitgestellt werden. Das folgende Diagramm veranschaulicht diese Architektur.



Sie können BabelFish auf Ihrem Aurora-Cluster über die Amazon RDS-Managementkonsole aktivieren. Anweisungen finden Sie unter [Erstellen eines BabelFish for Aurora PostgreSQL-DB-Clusters](#) in der Amazon RDS-Dokumentation.

Weitere Informationen zur Migration finden Sie unter [Migrieren einer SQL Server-Datenbank zu BabelFish for Aurora PostgreSQL](#) in der Aurora-Dokumentation.

Weitere Informationen finden Sie in den folgenden Ressourcen:

- [Beginnen Sie mit BabelFish für Aurora PostgreSQL](#) (Datenbank-Blog) AWS

- [Migrieren Sie mit Babelfish von SQL Server zu Amazon Aurora](#) (AWSDatenbank-Blog)
- [Migrieren Sie mit SSIS und Babelfish von SQL Server zu Aurora PostgreSQL](#) (Datenbank-Blog) AWS
- [Ändern Sie SSIS-Pakete von SQL Server zu Babelfish for Aurora PostgreSQL](#) (Datenbank-Blog) AWS
- [SQL Server Reporting Services-Berichte für Babelfish for Aurora PostgreSQL ausführen](#) (Datenbank-Blog) AWS
- [Bereiten Sie sich mit dem AWS SCT Bewertungsbericht auf die Babelfish-Migration](#) vor (AWSDatenbank-Blog)

## Hybride Migrationsszenarien für SQL Server

Sie können SQL Server-Workloads auch in einer Hybrid-Umgebung ausführen, die AWS aus. Beispielsweise führen Sie möglicherweise bereits SQL Server in Ihrem lokalen oder gemeinsam lokalisierten Rechenzentrum aus, möchten aber die AWS Cloud zur Verbesserung Ihrer Architektur, um eine Hochverfügbarkeits- oder Disaster Recovery-Lösung bereitzustellen. Sie können auch hybride Lösungen verwenden, um langfristige SQL Server-Backups zu speichern AWS, um Ihre Migration im Falle von Problemen rückgängig zu machen oder um ein sekundäres Replikat mit SQL Server Always On-Verfügbarkeitsgruppen im AWS Cloud. SQL Server verfügt über mehrere Replikationstechnologien, die Lösungen für hohe Verfügbarkeit und Disaster Recovery bieten.

## Sichern Sie Ihre SQL Server-Datenbanken im AWS Cloud

Amazon Simple Storage Service (Amazon S3) ermöglicht es Ihnen, die Flexibilität und Preisgestaltung des Cloud-Speichers zu nutzen. Sie haben die Möglichkeit, Ihre SQL Server-Datenbanken auf einem sicheren, hochverfügbaren, äußerst langlebigen und zuverlässigen Speichersystem zu sichern. Sie können Ihre SQL Server-Backups sicher in Amazon S3 speichern. Sie können auch Amazon S3 Lifecycle-Richtlinien verwenden, um Ihre Backups langfristig zu speichern. Mit Amazon S3 können Sie große Datenmengen zu sehr niedrigen Kosten speichern. Sie können [AWS DataSync](#) um Sicherungsdateien auf Amazon S3 zu übertragen.

Sie können Storage Gateway verwenden, um Ihre lokalen SQL Server-Backups zu speichern und Daten auf Amazon S3 oder Amazon S3 Glacier zu archivieren. Sie können zwischengespeicherte Speicher-Volumes erstellen und diese als Internet Small Computer System Interface (iSCSI) -Geräte über Ihre lokalen Backup-Anwendungsserver mounten. Alle Daten werden sicher an übertragen AWS über SSL und in verschlüsseltem Format in Amazon S3 gespeichert. Die Verwendung von gecachten Gateway-Volumes spart die Vorabkosten für die Wartung und Skalierung kostspieliger Speicherhardware vor Ort. Wenn Sie Ihre primären Daten oder Backups vor Ort aufbewahren möchten, können Sie gespeicherte Gateway-Volumes verwenden, um diese Daten lokal zu speichern und die Daten außerhalb des Standorts auf Amazon S3 zu sichern.

## Erweitern der Lösungen für hohe Verfügbarkeit und Notfallwiederherstellung

Sie können Ihre bestehenden lokalen Hochverfügbarkeitspraktiken erweitern und eine Disaster Recovery-Lösung in AWS mithilfe der Funktion für den nativen Protokollversand in SQL Server. Sie

können Ihre SQL Server-Transaktionsprotokolle von Ihren lokalen oder gemeinsam lokalisierten Rechenzentren auf eine SQL Server-Instanz übertragen, die auf einer EC2-Instanz oder einer Amazon RDS for SQL Server DB-Instanz in einer Virtual Private Cloud (VPC) ausgeführt wird. Sie können diese Daten sicher über eine dedizierte Netzwerkverbindung übertragen, indem Sie AWS Direct Connect oder über eine sichere VPN-Verbindung übertragen. Die Transaktionslog-Backups werden an die EC2-Instanz gesendet und auf sekundäre Datenbankinstanzen angewendet.

Sie können das AWS Cloud zur Bereitstellung eines höheren Niveaus an Hochverfügbarkeit und Disaster Recovery durch Verwendung von SQL Server Always On-Verfügbarkeitsgruppen zwischen Ihrem lokalen Rechenzentrum und Amazon EC2. Dies kann erreicht werden, indem Sie Ihr Rechenzentrum in eine VPC erweitern AWS durch Verwendung einer dedizierten Netzwerkverbindung wie AWS Direct Connect oder indem Sie einen sicheren VPN-Tunnel zwischen diesen beiden Umgebungen einrichten.

Bei der Planung einer hybriden Implementierung von SQL Server Always On-Verfügbarkeitsgruppen sollten Sie Folgendes beachten:

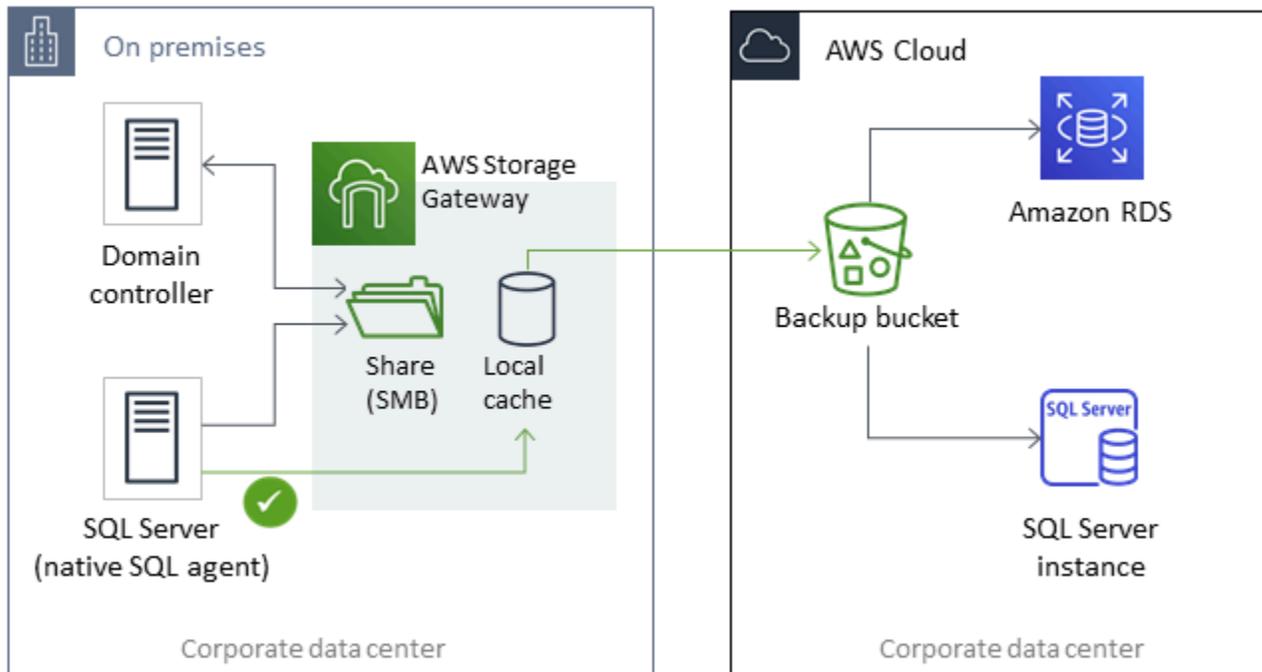
- Stellen Sie sichere, zuverlässige und konsistente Netzwerkverbindungen zwischen Ihrer lokalen Umgebung her und AWS durch AWS Direct Connect oder VPN.
- Erstellen Sie eine VPC mithilfe des Amazon Virtual Private Cloud Cloud-Dienstes. Verwenden Sie Amazon VPC-Routentabellen und Sicherheitsgruppen, um die entsprechende Kommunikation zwischen den beiden Umgebungen zu ermöglichen.
- Erweitern Sie Active Directory-Domänen in die VPC, indem Sie Domänencontroller als EC2-Instanzen bereitstellen oder AWS Directory Service for Microsoft Active Directory aus. Sie können auch AWS Managed Microsoft AD für Amazon RDS for SQL Server. Weitere Informationen finden Sie in der [Dokumentation zu Amazon RDS](#).

## Storage Gateway

Mit Storage Gateway können Sie Dateien mithilfe einer SMB-Freigabe (Server Message Block) für Windows speichern und abrufen. Sie können das Storage Gateway Ihrer lokalen Active Directory-Domäne beitreten. Wenn Sie Ihre SQL Server-Datenbank und Ihr Speicher-Gateway in derselben Domäne haben, können Sie die Backups direkt zur SMB-Netzwerkfreigabe bringen, anstatt sie lokal zu speichern und sie dann auf die Netzwerkfreigabe hochzuladen. Das Storage-Gateway ist so konfiguriert, dass es einen S3-Bucket verwendet, sodass alle Ihre Backups im S3-Bucket verfügbar sind AWS aus. Sie können Ihre Datenbank wiederherstellen, indem Sie die Sicherungsdateien

auf eine EC2-Instanz auf SQL Server heruntergeladen oder die Datenbank direkt auf Amazon RDS wiederherstellen.

Das folgende Diagramm zeigt, wie Sie Backups mithilfe von Storage Gateway und Amazon S3 speichern und darauf zugreifen können. Weitere Informationen finden Sie im [Storage Gateway Dokumentation](#) aus.



## benutzenAWS DMSundAWS SCT

Sie könnenAWS DMSin hybriden SQL Server-Umgebungen, um Daten von Ihrer lokalen Datenbank in die Cloud zu migrieren oder umgekehrt. Sie können Ihre SQL Server-Datenbank zu MySQL oder PostgreSQL migrieren, indem SieAWS DMSmitAWS SCTaus. Informationen zu Migrationsschritten finden Sie im[AWS SCTDokumentation](#) aus. Bevor Sie Ihre Daten migrieren, können Sie eine[Bewertungsbericht zur Migration](#)das markiert alle zusätzlichen manuellen Arbeiten, die möglicherweise erforderlich sind.

Sie können auchAWS DMSfür die fortlaufende Replikation (Change Data Capture oder CDC). Weitere Informationen finden Sie unter[Verwenden der fortlaufenden Replikation \(CDC\) von einer SQL Server-Quelle](#)imAWS DMS-Dokumentation.

# Modernisierung der SQL Server-Datenbank

In diesem Abschnitt wird beschrieben, wie Sie Ihre SQL Server-Workloads auf modernisieren können, indem Sie vom Windows-Betriebssystem auf Linux wechseln. Diese Änderung ermöglicht es Ihnen, Open-Source-Technologien zu nutzen und Windows-Lizenzkosten zu sparen, ohne Ihre Systemarchitektur drastisch zu verändern oder Ihre Benutzer umzuschulen.

## Migration der SQL Server-Workloads von Windows nach Linux

Ab SQL Server 2017 kann SQL Server auf Linux-Betriebssystemen ausgeführt werden. Das Verschieben Ihrer SQL Server-Workloads auf Linux bietet sowohl Kosteneinsparungen als auch Leistungsverbesserungen.

Fast alle SQL Server-Funktionen, Anwendungen, Anweisungen und Skripte, die Sie unter Microsoft Windows verwenden, werden auch unter Linux unterstützt. Sie können auch Tools wie SQL Server Management Studio (SSMS), SQL Server Data Tools (SSDT) und PowerShell-Modul (SQLPS) verwenden, um SQL Server unter Linux von einer Windows-Instanz aus zu verwalten.

Sie können eine dieser drei Optionen verwenden, um Ihre SQL Server-Workloads auf Linux zu migrieren:

- Native SQL Server-Backup- und Wiederherstellungsfunktion (siehe die [Microsoft SQL Server-Dokumentation](#))
- Verteilte Verfügbarkeitsgruppen (um Ihr Betriebssystem zu ändern, während Sie zu migrieren AWS)
- Die AWS Replatforming-Assistent, ein PowerShell-basiertes Skriptwerkzeug

Die AWS Replatforming-Assistent hilft Ihnen, von Ihren vorhandenen SQL Server-Workloads von Windows auf ein Linux-Betriebssystem zu migrieren. Wenn Sie das PowerShell-Skript für den Replatforming-Assistenten in einer Quelldatenbank von SQL Server ausführen, sichert die Windows-Instance die Datenbank in einem verschlüsselten Amazon S3 S3-Speicher-Bucket. Er stellt dann das Backup in einer neuen oder vorhandenen SQL Server-Datenbank auf einer EC2-Linux-Instance wieder her. Sie können Ihre Datenbank replizieren und Ihre Anwendungen testen, während Ihre SQL Server-Quelldatenbank online bleibt. Nach dem Testen können Sie die Ausfallzeit der Anwendung planen und das PowerShell-Backup-Skript erneut ausführen, um den endgültigen Cutover durchzuführen.

Weitere Informationen zur Verwendung des Replatforming-Assistenten finden Sie unter [Migrieren Sie Ihre lokalen SQL Server-Windows-Workloads zu Amazon EC2 Linux](#) auf der AWS Datenbank-Blog und das [Amazon EC2-Dokumentation](#) aus.

## Hohe Verfügbarkeit unter Linux

SQL Server 2017 unterstützt Always On-Verfügbarkeitsgruppen zwischen Windows und Linux, um Leseskalierungs-Workloads ohne hohe Verfügbarkeit zu erstellen. Leider können Sie keine hohe Verfügbarkeit zwischen Windows und Linux erreichen, da es keine Clusterlösung gibt, die diese plattformübergreifende Konfiguration verwalten kann.

Um Hochverfügbarkeit mit Always On-Verfügbarkeitsgruppen zu verwenden, sollten Sie einen Windows Server Failover Cluster (WSFC) oder Pacemaker unter Linux verwenden. Diese Lösung eignet sich für einen Migrationspfad von SQL Server unter Windows zu Linux und umgekehrt oder für Disaster Recovery mit manuellem Failover. Weitere Informationen zu diesem Szenario finden Sie unter [Bereitstellen von Always On-Verfügbarkeitsgruppen zwischen Amazon EC2 Windows und Amazon Linux 2-Instanzen](#) auf der AWS Datenbank-Blog.

# AWSLaunch Wizard für SQL Server

AWSLaunch Wizard ist ein Service, der Sie durch die Größe, Konfiguration und Bereitstellung von Microsoft SQL Server auf Amazon EC2 führt. Es unterstützt sowohl SQL Server-Einzelinstanz- als auch Hochverfügbarkeits-Bereitstellungen (HA) auf Amazon EC2.

Launch Wizard ist ein kostenloser Service. Sie zahlen nur für denAWS-Ressourcen, die für die Ausführung Ihrer Anwendung bereitgestellt werden, wie Amazon EC2-, Amazon EBS- und Amazon VPC-Ressourcen.

Sie geben Ihre Anwendungsanforderungen, einschließlich Leistung, Anzahl der Knoten und Konnektivität, in der Launch Wizard Wizard-Konsole ein. Launch Wizard identifiziert das RechtAWSRessourcen zum Bereitstellen und Ausführen Ihrer SQL Server-Anwendung. Es bietet auch geschätzte Bereitstellungskosten, und Sie können Ihre Ressourcen ändern und sofort die aktualisierte Kostenbewertung anzeigen. Wenn Sie Ihre Auswahl bestätigen und die Bereitstellung einleiten, stellt Launch Wizard die ausgewählten Ressourcen in wenigen Stunden bereit und konfiguriert sie, um eine voll funktionsfähige, produktionsfähige SQL Server-Anwendung zu erstellen. Sie können über die Amazon EC2 EC2-Konsole auf Ihre bereitgestellte SQL Server-Anwendung zugreifen.

Hier sind einige Vorteile der Verwendung von Launch Wizard für SQL Server:

- Einfache Bereitstellung— Sie können die Bereitstellung Ihrer SQL Server-Ressourcen unterAWSindem Sie Fragen beantworten, die auf Ihren Anforderungen basieren. Eine Bereitstellung des Launch Wizard ist schneller als eine manuelle Bereitstellung, sodass die Zeit für die Bereitstellung und Konfiguration Ihrer Anwendung auf AWS entfällt.
- Automatisierte Dimensionierung und Kostenschätzung— Launch Wizard bietet eine integrierte Instanzauswahl basierend auf Ihren Anforderungen. Es wählt den Instanztyp, EBS-Volumes und andere Ressourcen aus, die Ihren SQL Server-Anforderungen am besten entsprechen. Launch Wizard stellt Ihnen auch einen Kostenvoranschlag zur Verfügung, bevor er dieAWSRessourcen schätzen.
- Zeitersparnis mit wiederholbaren Automatisierungsvorlagen— Sie können SQL Server mit wiederverwendbarem erneut bereitstellenAWS CloudFormationVorlagen, die vom Launch Wizard erstellt werden. Diese Vorlagen dienen als Baseline und sparen Ihnen Zeit.

Launch Wizard unterstützt die folgenden Betriebssysteme, SQL Server-Versionen und Funktionen. Aktuelle Informationen finden Sie unter[AWSStartassisten-Dokumentation](#)aus.

Kategorie	Anwendungsfall oder Feature	Launch Wizard	<a href="#">Schnellstart</a> Unterstützung	Amazon EC2 EC2-Konsole Unterstützung
Bereitstellung unter Windows	Bereitstellung eines einzelnen SQL-Kno			
	HA-Bereitstellung: Immer On-Verfügbarkeitsgruppen			
	HA-Bereitstellung: Immer auf Failovercluster-Instanzen (FCIs) mit FSx for Windows File Server			
	HA-Bereitstellung: Dedicated Hosts			
	Wiederverwendbare Codevorlagen			
	Bereitstellung eines einzelnen SQL-Kno			
Bereitstellung unter Linux	HA-Bereitstellung auf Ubuntu			

Kategorie	Anwendungsfall oder Feature	Launch Wizard	<a href="#">Schnellstart</a> Unterstützung	Amazon EC2 EC2-Konsole Unterstützung
	Wiederverwendbare Codevorlagen			
Größenbestimmung	Empfehlung für Instance-Typen			
	Einschätzen der -Kosten			
Konfiguration	Automatisches Erstellen der AWS Systems Manager Ressourcengruppe			
	Amazon SNS SNS-Benachrichtigung mit einem Klick			
	Amazon CloudWatch CloudWatch-Überwachung mit einem Klick			

Kategorie	Anwendungsfall oder Feature	Launch Wizard	<u>Schnellstart</u> Unterstützung	Amazon EC2 EC2-Konsoleunterstützung
	Verbinden mit vorhandenem Active Directory (sowohl lokal als auch verwaltet)		Ja 	Nein 
	Vorzeitiges Eingabe-		Ja 	Nein 
	Verwaltete IAM-Richtlinie		Ja 	Nein 

Weitere Informationen zum Launch Wizard für SQL Server finden Sie unter:

- [AWSLaunch Wizard für SQL Server-Dokumentation](#)
- [Vereinfachen Sie SQL Server AlwaysOn-Bereitstellungen mitAWSLaunch Wizard und Amazon FSxBlog-Posts](#)
- [Beschleunigen Sie SQL Server AlwaysOn-Bereitstellungen mitAWSStartassistentBlog-Posts](#)

# Bewährte Methoden für die Migration zu Amazon RDS für SQL Server

Wenn Sie auf der Grundlage der Bewertung Ihrer Datenbank und Ihrer Projektanforderungen eine Migration zu Amazon RDS for SQL Server anstreben, befolgen Sie die bewährten Methoden in diesem Abschnitt, um Ihre Zieldatenbank bereitzustellen, die Migration durchzuführen und Ihre Amazon RDS for SQL Server-Datenbank zu testen, zu betreiben und zu optimieren.

## Important

Stellen Sie sicher, dass Sie über einen Rollback-Plan verfügen, bevor Sie Ihre Datenbank migrieren.

## Note

Sie können Migration Hub Orchestrator verwenden, um Ihre SQL Server-Datenbankmigrationen zu Amazon EC2 oder Amazon RDS zu automatisieren und zu orchestrieren, indem Sie native Backups und Wiederherstellungen verwenden. Weitere Informationen finden Sie im [AWS Migration Hub Orchestrator Abschnitt](#).

## Bereitstellung Ihrer Zieldatenbank

Nachdem Sie die Bewertung, Planung und Vorbereitung Ihrer Datenbankmigrationsstrategie abgeschlossen haben, befolgen Sie bei der Bereitstellung Ihrer Amazon RDS for SQL Server-Datenbank die folgenden bewährten Methoden:

- Passen Sie die Amazon RDS for SQL Server-DB-Instance entsprechend Ihren Anforderungen an CPU, Arbeitsspeicher, IOPS und Speichertyp an. (Wenn Sie die SQL Server Standard Edition verwenden, stellen Sie CPU und Arbeitsspeicher innerhalb der Einschränkungen der Standard Edition bereit.)
- Stellen Sie die richtige Zeitzone und Sortierung ein.
- Stellen Sie sicher, dass Sie Amazon RDS in der richtigen Virtual Private Cloud (VPC) starten.
- Erstellen Sie die Sicherheitsgruppen mit dem richtigen Port und den richtigen IP-Adressen.

- Stellen Sie Ihre Amazon RDS-Datenbank aus Sicherheitsgründen in einem privaten Subnetz bereit.
- Wenn möglich, stellen Sie die SQL Server-Instanz mit der neuesten Version von SQL Server bereit.
- Erstellen Sie für jede Amazon RDS-Datenbank eine separate Optionsgruppe und Parametergruppe.
- Sammeln und extrahieren Sie Logins, Benutzer und Rollen für die Migration.
- Überprüfen Sie die SQL Server-Agent-Jobs auf Wartung und Anwendungen, die migriert werden müssen.

## Aus Ihrer Quelldatenbank sichern

Es gibt viele Tools für die Migration einer SQL Server-Datenbank zu einer Amazon RDS for SQL Server-Datenbank. Die am häufigsten verwendete Methode ist die systemeigene Sicherung und Wiederherstellung von SQL Server, wenn Ihre Anforderungen Ausfallzeiten zulassen.

Wenn Sie nur begrenzte Ausfallzeiten haben, können Sie die native SQL Server-Sicherung/-Wiederherstellung mit differentieller Sicherung und Protokollsicherung verwenden. Oder Sie können es verwenden AWS DMS, das drei Optionen bietet: Volllast, Volllast und CDC oder nur CDC.

## Übertragung von Datendump-Dateien an AWS

- Wenn Sie eine Verbindung mit AWS Direct Connect hoher Bandbreite zwischen Ihrer lokalen Umgebung und Ihrer lokalen Umgebung verwenden AWS, können Sie Ihre SQL Server-Backups nach Amazon S3 kopieren und die [Amazon S3-Integration](#) einrichten.
- Wenn Sie nicht über eine hohe Bandbreite verfügen AWS Direct Connect, verwenden Sie diese, AWS Snowball um große Datenbank-Backup-Dateien zu übertragen. Sie können es auch verwenden AWS DMS, um die Daten zu übertragen, wenn eine Replikation erforderlich ist.

## Daten in Ihrer Zieldatenbank wiederherstellen

- Wenn Sie eine sehr große Datenbank migrieren, empfehlen wir Ihnen, für die Dauer der Migration zunächst einen größeren [Amazon RDS-Instance-Typ](#) bereitzustellen, um schnellere Datenladungen zu erzielen.
- Deaktivieren Sie Multi-AZ. (Dies kann nach der Migration wieder aktiviert werden.)
- Deaktivieren Sie die Aufbewahrung von Backups. (Dies kann nach der Migration wieder aktiviert werden.)

- Stellen Sie die Datenbank mithilfe des nativen SQL Server-Wiederherstellungsbefehls wieder her.
- Erstellen Sie Logins und Benutzer und beheben Sie bei Bedarf verwaiste Benutzer.
- Erstellen Sie SQL Server-Agent-Jobs und überprüfen Sie gegebenenfalls den Zeitplan.

## Schritte nach der Migration

Nach Abschluss der Migration können Sie:

- Ändern Sie die DB-Instance auf den Instance-Typ mit der richtigen Größe.
- Aktivieren Sie Multi-AZ und Backup-Aufbewahrung.
- Stellen Sie sicher, dass alle Jobs auf sekundären Knoten erstellt werden (für die Multi-AZ-Konfiguration).
- Veröffentlichen Sie SQL Server-Fehler- und Agentenprotokolle in Amazon CloudWatch Logs und verwenden Sie CloudWatch sie, um Metriken einzusehen und Alarmer zu erstellen. Weitere Informationen finden Sie in der [Dokumentation zu Amazon RDS](#).
- Aktivieren Sie die [erweiterte Überwachung](#), um Metriken für Ihre DB-Instance in Echtzeit abzurufen.
- Richten Sie Amazon Simple Notification Service (Amazon SNS) -Themen für Benachrichtigungen ein.

## Die Migration testen

Wir empfehlen die folgenden Tests, um Ihre Anwendung anhand Ihrer neuen Amazon RDS for SQL Server-Datenbank zu validieren:

- Führen Sie Funktionstests durch.
- Vergleichen Sie die Leistung von SQL-Abfragen in Ihren Quell- und Zieldatenbanken und passen Sie die Abfragen nach Bedarf an. Einige Abfragen werden in der Zieldatenbank möglicherweise langsamer ausgeführt. Wir empfehlen daher, die Basislinien der SQL-Abfragen in der Quelldatenbank zu erfassen.

Für eine zusätzliche Validierung während der proof-of-concept (POC-) Phase empfehlen wir die folgenden zusätzlichen Tests:

- Führen Sie Leistungstests durch, um sicherzustellen, dass sie Ihren Geschäftserwartungen entsprechen.

- Testen Sie Datenbank-Failover, Recovery und Wiederherstellung, um sicherzustellen, dass Sie die RPO- und RTO-Anforderungen erfüllen.
- Führen Sie alle kritischen Jobs und Berichte auf und führen Sie sie auf Amazon RDS aus, um ihre Leistung anhand Ihrer Service Level Agreements (SLAs) zu bewerten.

## Betrieb und Optimierung Ihrer Amazon RDS-Datenbank

Wenn Ihre Datenbank aktiviert ist AWS, stellen Sie sicher, dass Sie die bewährten Methoden in Bereichen wie Überwachung, Warnmeldungen, Backups und Hochverfügbarkeit in der Cloud befolgen. Beispiele:

- Richten Sie die CloudWatch Überwachung ein und aktivieren Sie eine detaillierte Überwachung.
- Verwenden Sie [Amazon RDS Performance Insights](#) und andere Überwachungslösungen von Drittanbietern wie [SentryOne](#) oder [Foglight for SQL Server](#), um Ihre Datenbank zu überwachen.
- Richten Sie Benachrichtigungen mithilfe von SNS-Themen ein.
- Richten Sie automatische Backups mithilfe unserer [AWS Backup](#) nativen SQL Server-Backups ein und kopieren Sie sie nach Amazon S3.
- Richten Sie für eine hohe Verfügbarkeit die Amazon RDS Multi-AZ-Funktion ein.
- Wenn Sie schreibgeschützte Datenbanken benötigen, [richten Sie je nach Bedarf ein Read Replica](#) innerhalb derselben oder AWS regionsübergreifend ein.

# Auswählen zwischen Amazon EC2 und Amazon RDS

Amazon EC2 und Amazon RDS bieten einzigartige Vorteile, die für Ihren spezifischen Anwendungsfall von Vorteil sein können. Sie haben je nach Bedarf die Flexibilität, einen oder beide Services für Ihre SQL Server-Datenbank zu verwenden. Dieser Abschnitt enthält detaillierte Informationen, die Ihnen bei Ihrer Auswahl helfen.

## Entscheidungsmatrix

Die folgende Tabelle enthält einen side-by-side Vergleich der SQL Server-Funktionen, die auf Amazon RDS, Amazon RDS Custom for SQL Server und Amazon EC2 unterstützt werden. Verwenden Sie diese Informationen, um ihre Unterschiede zu verstehen und den besten Ansatz für Ihren Anwendungsfall auszuwählen.

Aktuelle Informationen zu Amazon RDS finden Sie unter [Microsoft SQL Server auf Amazon RDS](#) in der - AWS Dokumentation.

### Development

Entwicklungsfunktion	Amazon RDS	Amazon RDS Custom	Amazon EC2	Hinweise
Pufferpool-Erweiterungen		N 	J; 	J; Wenn diese Funktion für Ihren Workload von entscheidender Bedeutung ist, sollten Sie Amazon RDS Custom oder Amazon EC2 wählen.
BULK INSERT		J; 	J; 	J; Weitere Informationen finden Sie unter

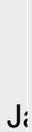
Entwicklungs-funktion	Amazon RDS	Amazon RDS Custom	Amazon EC2	Hinweise
				<a href="#">Integration einer Amazon RDS for SQL Server-DB-Instance in Amazon S3</a> in der Amazon RDS-Dokumentation.
Erfassung von Datenänderungen (Change Data Capture, CDC)	 (Enterprise Edition: alle Versionen ; Standard Edition: 2016 SP1 und höher)	Ja 	Ja 	Weitere Informationen finden Sie unter <a href="#">Verwenden der Änderungsdaten</a> in der Amazon-RDS-Dokumentation.
Änderungsnachverfolgung		Ja 	Ja 	Ja
Columnstore-Indizes	 (Enterprise Edition: 2014 und höher)	Ja  (Enterprise Edition: 2019)	Ja  (Enterprise Edition: 2014 und höher)	Ja

Entwicklungs-funktion	Amazon RDS	Amazon RDS Custom	Amazon EC2	Hinweise
Data Quality Services		N 	Ja 	Wenn diese Funktion für Ihren Workload von entscheidender Bedeutung ist, sollten Sie Amazon RDS Custom oder Amazon EC2 wählen.

Entwicklungs-funktion	Amazon RDS	Amazon RDS Custom	Amazon EC2	Hinweise
Datenbank-E-Mail		Ja 	Ja 	<p>Weitere Informationen finden Sie im Blogbeitrag <a href="#">Using Database Mail on Amazon RDS for SQL Server</a> .</p> <p>Wir empfehlen Ihnen, den <a href="#">Amazon Simple Email Service (Amazon SES)</a> zu verwenden, um ausgehende E-Mails von - AWS Ressourcen zu senden, um ein hohes Maß an Zustellbarkeit zu gewährleisten.</p>
Database Engine Tuning Advisor		Ja 	Ja 	Ja

Entwicklungsfunktion	Amazon RDS	Amazon RDS Custom	Amazon EC2	Hinweise
DB-Ereignisbenachrichtigungen		J: 	J:  (verfolgen und verwalten Sie DB-Ereignisse manuell)	N Weitere Informationen finden Sie unter <a href="#">Verwenden von Amazon-RDS-Ereignisbenachrichtigungen</a> in der Amazon-RDS-Dokumentation.
DDL-Ereignisbenachrichtigungen		N 	J: 	J: Wenn diese Funktion für Ihren Workload von entscheidender Bedeutung ist, sollten Sie Amazon RDS Custom oder Amazon EC2 wählen.
Verzögerte Transaktionsbeständigkeit (Lazy Commit)	 (SQL Server 2014 und höher)	J:  (SQL Server 2019)	J:  (SQL Server 2014 und höher)	J:

Entwicklungsfunktion	Amazon RDS	Amazon RDS Custom	Amazon EC2	Hinweise
Verteilte Abfragen	 (SQL-Server-Ziele)	Ja  (SQL-Server-Ziele)	Ja  (SQL-Server-Ziele)	Weitere Informationen finden Sie im Blogbeitrag <a href="#">Implementieren von verknüpften Servern mit Amazon RDS für SQL Server</a> .
Erweiterte Ereignisse		Ja 	Ja 	
Erweiterte gespeicherte Prozeduren, einschließlich xp_cmdshell		Nein 	Ja 	Wenn diese Funktion für Ihren Workload von entscheidender Bedeutung ist, sollten Sie Amazon RDS Custom oder Amazon EC2 wählen.

Entwicklungsfunktion	Amazon RDS	Amazon RDS Custom	Amazon EC2	Hinweise
Datenbanktabellen		N 	J: 	Wenn diese Funktion für Ihren Workload von entscheidender Bedeutung ist, sollten Sie Amazon RDS Custom oder Amazon EC2 wählen.
FILESTREAM		N 	J: 	FILESTREAM ist nicht mit Amazon RDS kompatibel. Sie können jedoch die In-Memory-Datenbank konfigurieren.
Volltextsuche	 (außer semantische Suche)	J: 	J: 	J: 
In-Memory-Datenbank	 (SQL Server 2014 und höher)	J:  (SQL Server 2019)	J:  (SQL Server 2014 und höher)	J: 

Entwicklungs-funktion	Amazon RDS	Amazon RDS Custom	Amazon EC2	Hinweise
Verknüpfte Server  (SQL Server- und Oracle-Ziele)	 (SQL Server- und Oracle-Ziele)	Ja 	Ja 	Weitere Informationen finden Sie im Blogbeitrag <a href="#">Implementierung verknüpfter Server mit Amazon RDS für SQL Server</a> und <a href="#">Unterstützung für verknüpfte Server mit Oracle OLEDB in Amazon RDS für SQL Server</a> in der Amazon-RDS-Dokumentation.

Entwicklungs-funktion	Amazon RDS	Amazon RDS Custom	Amazon EC2	Hinweise
Machine Learning Services (mit R-Skripten)		Ja 	Ja 	<p>Machine Learning Services müssen separat auf einem Windows- oder Linux-Computer installiert werden. Es wird auf einer <a href="#">Always On Failover Cluster Instance (FCI)</a> nur in SQL Server 2019 und höher unterstützt.</p> <p>Obwohl R auf Amazon RDS nicht unterstützt wird, können Sie es auf verwenden AWS (siehe <a href="#">Blogbeitrag Erste Schritte mit R auf AWS</a>).</p>

Entwicklungs-funktion	Amazon RDS	Amazon RDS Custom	Amazon EC2	Hinweise
Wartungspläne		N 	J; 	Amazon RDS bietet eine separate Reihe von Funktionen, um die Sicherung und Wiederherstellung von Datenbanken zu erleichtern. Für Backups können Sie automatisierte Backups konfigurieren.
Master Data Services		N 	J; 	Wenn diese Funktion für Ihren Workload von entscheidender Bedeutung ist, sollten Sie Amazon RDS Custom oder Amazon EC2 wählen.

Entwicklungsfunktion	Amazon RDS	Amazon RDS Custom	Amazon EC2	Hinweise
Microsoft Distributed Transaction Coordinator (MSDTC)		Ja 	Ja 	Weitere Informationen finden Sie im Blogbeitrag <a href="#">Enabling Distributed Transaction Support for Domain Joined Amazon RDS for SQL Server instances.</a>
OPENROWSET		Ja 	Ja 	
Teilweise enthaltene Datenbanken	 (SQL Server 2014 und höher)	Ja  (SQL Server 2019)	Ja  (SQL Server 2014 und höher)	

Entwicklungsfunktion	Amazon RDS	Amazon RDS Custom	Amazon EC2	Hinweise
Performance-Datenaufbilder		N 	J 	Auf Amazon RDS können Sie Amazon Performance Insights verwenden. CloudWatch AWS CloudTrail, um Ihre SQL Server-Leistung zu überwachen (siehe <a href="#">Übersicht über die Überwachung von Amazon RDS</a> in der Amazon RDS-Dokumentation).
Richtlinienbasierte Verwaltung		N 	J 	Wenn diese Funktion für Ihren Workload von entscheidender Bedeutung ist, sollten Sie Amazon RDS Custom oder Amazon EC2 wählen.

Entwicklungsfunktion	Amazon RDS	Amazon RDS Custom	Amazon EC2	Hinweise
PolyBase		N 	J: 	Wenn diese Funktion für Ihren Workload von entscheidender Bedeutung ist, sollten Sie Amazon RDS Custom oder Amazon EC2 wählen.
Vorkonfigurierte Parameter		J: 	N 	
Ressourcenkontrolle		N 	J: 	Wenn diese Funktion für Ihren Workload von entscheidender Bedeutung ist, sollten Sie Amazon RDS Custom oder Amazon EC2 wählen.
Sichere CLR	 (SQL Server 2014 und 2016)	J: 	J: 	J:

Entwicklungsfunktion	Amazon RDS	Amazon RDS Custom	Amazon EC2	Hinweise
Sequenzen	 (SQL Server 2014 und höher)	Ja  (SQL Server 2019)	Ja  (SQL Server 2014 und höher)	
Auslöser auf Serverebene		Nein 	Ja 	Wenn diese Funktion für Ihren Workload von entscheidender Bedeutung ist, sollten Sie Amazon RDS Custom oder Amazon EC2 wählen.
Service-Broker	 (außer - Endpunkte)	Ja 	Ja 	
Geodaten und Standortfunktionen		Ja 	Ja 	
SQL Server Agent		Ja 	Ja 	

Entwicklungsfunktion	Amazon RDS	Amazon RDS Custom	Amazon EC2	Hinweise
SQL Server Analysis Services (SSAS)	 (SQL Server 2016 und höher)	Ja 	Ja 	Siehe <a href="#">Unterstützung für SSAS in Amazon RDS für SQL Server</a> in der Amazon-RDS-Dokumentation.
SQL Server Integration Services (SSIS)	 (SQL Server 2016 und höher)	Ja 	Ja 	Siehe <a href="#">Unterstützung für SSIS in Amazon RDS für SQL Server</a> in der Amazon-RDS-Dokumentation.
SQL Server Management Studio (SSMS)		Ja 	Ja 	Ja
SQL Server Migration Assistant (SSMA)		Ja 	Ja 	Ja
SQL Server Profiler	 (serverseitige und clientseitige Ablaufverfolgungen)	Ja 	Ja 	Ja

Entwicklungsfunktion	Amazon RDS	Amazon RDS Custom	Amazon EC2	Hinweise
SQL Server Reporting Services (SSRS)	 (SQL Server 2016 und höher)	Ja 	Ja 	Siehe <a href="#">Unterstützung für SSRS in Amazon RDS für SQL Server</a> in der Amazon-RDS-Dokumentation.
sqlcmd		Ja 	Ja 	Ja
Stretch-Datenbank		Nein 	Ja 	Wenn diese Funktion für Ihren Workload von entscheidender Bedeutung ist, sollten Sie Amazon RDS Custom oder Amazon EC2 wählen.
THROW-Anweisung	 (SQL Server 2014 und höher)	Ja  (SQL Server 2019)	Ja  (SQL Server 2014 und höher)	Ja

Entwicklungsfunktion	Amazon RDS	Amazon RDS Custom	Amazon EC2	Hinweise
Transact-SQL-Endpunkte		N 	J 	Alle Operationen, die verwenden <code>CREATE ENDPOINT</code> sind in Amazon RDS nicht verfügbar. Wir empfehlen Ihnen, SQL Server für diese Operationen auf einer EC2-Instanz zu installieren.
UTF-16-Unterstützung	 (SQL Server 2014 und höher)	J 	J  (SQL Server 2014 und höher)	
WCF Data Service		N 	J 	Wenn diese Funktion für Ihren Workload von entscheidender Bedeutung ist, sollten Sie Amazon RDS Custom oder Amazon EC2 wählen.

HA/DR

Entwicklungsfunktion	Amazon RDS	Amazon RDS Custom	Amazon EC2	Hinweise
AlwaysOn-Verfügbarkeitsgruppen		J:  (sowohl synchron als auch asynchron)	J: 	J: Wenn Sie eine selbstverwaltete AlwaysOn-Verfügbarkeitsgruppe benötigen, empfehlen wir Ihnen, zu verwenden, AWS Launch Wizard um die SQL Server HA-Bereitstellung auf einer EC2-Instance zu vereinfachen. Siehe <a href="#">AWS Launch Wizard für SQL Server</a> in der - AWS Dokumentation.
Always On Failover Cluster Instances (FCIs)		N 	J: 	J: Sie können verwenden AWS Launch Wizard , um Ihre SQL Server-FCI-Bereitstellung auf Amazon EC2 zu vereinfachen. Siehe <a href="#">AWS</a>

Entwicklungs-funktion	Amazon RDS	Amazon RDS Custom	Amazon EC2	Hinweise
				<a href="#">Launch Wizard für SQL Server</a> in der - AWS Dokumentation.
Sichern in Amazon S3		Ja 	Ja 	Amazon RDS unterstützt die native Sicherung und Wiederherstellung für SQL Server-Datenbanken mithilfe von vollständigen Sicherungsdateien (BAK-Dateien) und Amazon S3 als Repository. Weitere Informationen finden Sie unter <a href="#">Importieren und Exportieren von SQL Server-Datenbanken</a> in der Amazon RDS-Dokumentation.

Entwicklungsfunktion	Amazon RDS	Amazon RDS Custom	Amazon EC2	Hinweise
BACKUP command		N 	J 	Weitere Informationen finden Sie unter <a href="#">Wie führe ich native Backups einer Amazon RDS-DB-Instance durch, auf der SQL Server ausgeführt wird?</a> im - AWS Wissenscenter.
Datenbank spiegellung	 (Multi-AZ)	J 	J 	
Datenbank replikation	 (begrenzt Push-Abonnement)	N 	J 	Wenn Sie eine einzelne Tabelle in Amazon RDS replizieren möchten, können Sie auch Lesereplikate verwenden <a href="#">AWS DMS</a> oder einrichten.

Entwicklungsfunktion	Amazon RDS	Amazon RDS Custom	Amazon EC2	Hinweise
Verteilte Verfügbarkeitsgruppen		N 	J: 	J: Wenn diese Funktion für Ihren Workload von entscheidender Bedeutung ist, sollten Sie Amazon RDS Custom oder Amazon EC2 wählen.
Versand protokollieren		N 	J: 	J: Für Notfallwiederherstellungszwecke können Sie Lesereplikate oder verwenden <a href="#">AWS DMS</a> .
Verwaltete automatisierte Backups		J: 	J:  (erfordert die Konfiguration und Verwaltung von Wartungsp länen oder die Verwendung von Drittanbieterlösungen)	N Weitere Informationen finden Sie unter <a href="#">Arbeiten mit Sicherungen</a> in der Amazon-RDS-Dokumentation.

Entwicklungsfunktion	Amazon RDS	Amazon RDS Custom	Amazon EC2	Hinweise
Multi-AZ mit automatisiertem Failover		J:  (mit manueller Konfiguration von AlwaysOn-Verfügbarkeitsgruppen)	J:  (nur Enterprise Edition, mit manueller Konfiguration von AlwaysOn-Verfügbarkeitsgruppen)	Siehe <a href="#">Multi-AZ-Bereitstellungen für Amazon RDS für SQL Server</a> in der Amazon-RDS-Dokumentation.
Read Replicas	 (SQL Server 2016 und höher)	J:  (mit manueller Konfiguration von AlwaysOn-Verfügbarkeitsgruppen)	J:  (mit manueller Konfiguration von AlwaysOn-Verfügbarkeitsgruppen)	
RESTORE command		J: 	J: 	Siehe <a href="#">AWS Knowledge Center</a> .

### Scalability

Entwicklungsfunktion	Amazon RDS	Amazon RDS Custom	Amazon EC2	Hinweise
Integrierte Instance- und Datenbank		J: 	N:  (Exportieren Sie	Weitere Informationen finden Sie im Blogbeitrag

Entwicklungsfunktion	Amazon RDS	Amazon RDS Custom	Amazon EC2	Hinweise
Überwachung und -metriken			Ihre eigenen Metriken in CloudWatch oder verwenden Sie eine Drittanbieterlösung)	<a href="#">Überwachen Ihrer SQL Server-Datenbank mithilfe von benutzerdefinierten Metriken mit Amazon CloudWatch und AWS Systems Manager</a> .
Konfigurierbare Speichergröße				
Maximale Anzahl von Datenbanken pro Instance	Hängt von der Instance-Größe und der Multi-AZ-Konfiguration ab	Maximaler SQL Server (5000)	 Einschränkung	Weitere Informationen finden Sie unter <a href="#">Maximale Kapazitätsspezifikationen für SQL Server</a> in der Microsoft SQL Server-Dokumentation.

Entwicklungs-funktion	Amazon RDS	Amazon RDS Custom	Amazon EC2	Hinweise
Maximale Speichergröße einer DB-Instan-ce	16 TiB	16 TiB	 Einschränkung	K Amazon RDS unterstützt auch tempdb-Datenbanken auf lokalen Datenträgern, indem es Non-Volatile Memory Express (NVMe)-Instance-Speicher verwendet. Weitere Informationen finden Sie unter <a href="#">Instance-Speicher-Support für die tempdb-Datenbank auf Amazon RDS für SQL Server</a> in der Amazon-RDS-Dokumen-tation.
Minimale Speichergröße einer DB-Instan-ce	20 GiB (Enterprise-, Standard-, Web- und Express-Editionen)	20 GiB (Enterprise-, Standard-, Web- und Express-Editionen)	 Einschränkung	K

Entwicklungsfunktion	Amazon RDS	Amazon RDS Custom	Amazon EC2	Hinweise
Neuer Abfrageoptimierer	 (SQL Server 2014 und höher)	Ja;  (SQL Server 2014 und höher)	Ja;  (SQL Server 2014 und höher)	
Read Replicas	 (SQL Server 2016 und höher)	Ja;  (mit manueller Konfiguration von AlwaysOn-Verfügbarkeitsgruppen)	Ja;  (mit manueller Konfiguration von AlwaysOn-Verfügbarkeitsgruppen)	

Security

Entwicklungsfunktion	Amazon RDS	Amazon RDS Custom	Amazon EC2	Hinweise
Automatisches Software-Patching		Ja; 	N; 	
Verschlüsselter Speicher mit AWS KMS	 (alle SQL Server-Editionen außer Express)	Ja; 	Ja; 	Weitere Informationen finden Sie im Blogbeitrag <a href="#">Sichern von Daten in Amazon RDS mithilfe von AWS KMS</a>

Entwicklungsfunktion	Amazon RDS	Amazon RDS Custom	Amazon EC2	Hinweise
				<a href="#">Verschlüsselung</a>
Flexible Serverrollen	 (SQL Server 2014 und höher)	 (SQL Server 2019)	 (SQL Server 2014 und höher)	
SQL-Authentifizierung	 Ja	 Ja	 Ja	
SQL Server-Audit	 Ja	 Ja	 Ja	
SSL (Verschlüsselung während der Übertragung)	 Ja	 Ja	 Ja	Weitere Informationen finden Sie unter <a href="#">Verwenden von SSL mit einer Microsoft SQL Server-DB-Instance</a> in der Amazon RDS-Dokumentation.

Entwicklungsfunktion	Amazon RDS	Amazon RDS Custom	Amazon EC2	Hinweise
Sysadmin-Rolle		N 	Ja 	<p>Nicht unterstützte Rollen auf Serverebene finden Sie unter <a href="#">Microsoft SQL Server-Sicherheit</a> in der Amazon RDS-Dokumentation.</p> <p>Wenn Sie eine neue RDS-DB-Instance erstellen, erhält der von Ihnen verwendete Standard-Hauptbenutzer bestimmte Berechtigungen für diese DB-Instance (siehe <a href="#">Kontoberechtigungen</a> in der Amazon RDS-Dokumentation).</p>

Entwicklungsfunktion	Amazon RDS	Amazon RDS Custom	Amazon EC2	Hinweise
TDE (Verschlüsselung im Ruhezustand)	 (Enterprise Edition: 2014-2019 ; Standard Edition: 2019)	Ja;  (SQL Server 2019 Enterprise, Standard, Web und Developer Editions)	Ja;  (Enterprise Edition: 2014-2019 ; Standard Edition: 2019)	Weitere Informationen zur TDE-Unterstützung finden Sie in der Dokumentation zu <a href="#">Amazon RDS</a> und <a href="#">Amazon RDS Custom</a> .
Windows-Authentifizierung		Ja; 	Ja; 	

Other features

Entwicklungsfunktion	Amazon RDS	Amazon RDS Custom	Amazon EC2	Hinweise
Möglichkeit, einen Drittanbieter-Agenten zu installieren		Nein; 	Ja; 	
Möglichkeit, vorhandene Datenbanken umzubenennen	 (nur Single-AZ)	Ja;  (nicht verfügbar für Datenbanken in Verfügbarkeitsgruppen oder für	Ja;  (nicht verfügbar für Datenbanken in Verfügbarkeitsgruppen oder für	Informationen zu Multi-AZ-Bereitstellungen in Amazon RDS finden Sie unter <a href="#">Umbenennen einer Microsoft SQL Server-Datenbank in</a>

Entwicklungsfunktion	Amazon RDS	Amazon RDS Custom	Amazon EC2	Hinweise
		Spiegelung aktiviert)	Spiegelung aktiviert)	<a href="#">einer Multi-AZ-Bereitstellung</a> in der Amazon RDS-Dokumentation.
Kontrolle über DB-Instance und Betriebssystem		N 	J; 	J; Wenn diese Funktion für Ihren Workload von entscheidender Bedeutung ist, sollten Sie Amazon RDS Custom oder Amazon EC2 wählen.
Benutzerdefinierte festgelegte Zeitzonen		J; 	J; 	J;
Verteilte Wiedergabe		N 	J; 	J; Der SQL Server Distributed Replay-Client-Service <a href="#">erfordert Sysadmin-Berechtigungen</a> , daher wird er in Amazon RDS nicht unterstützt.

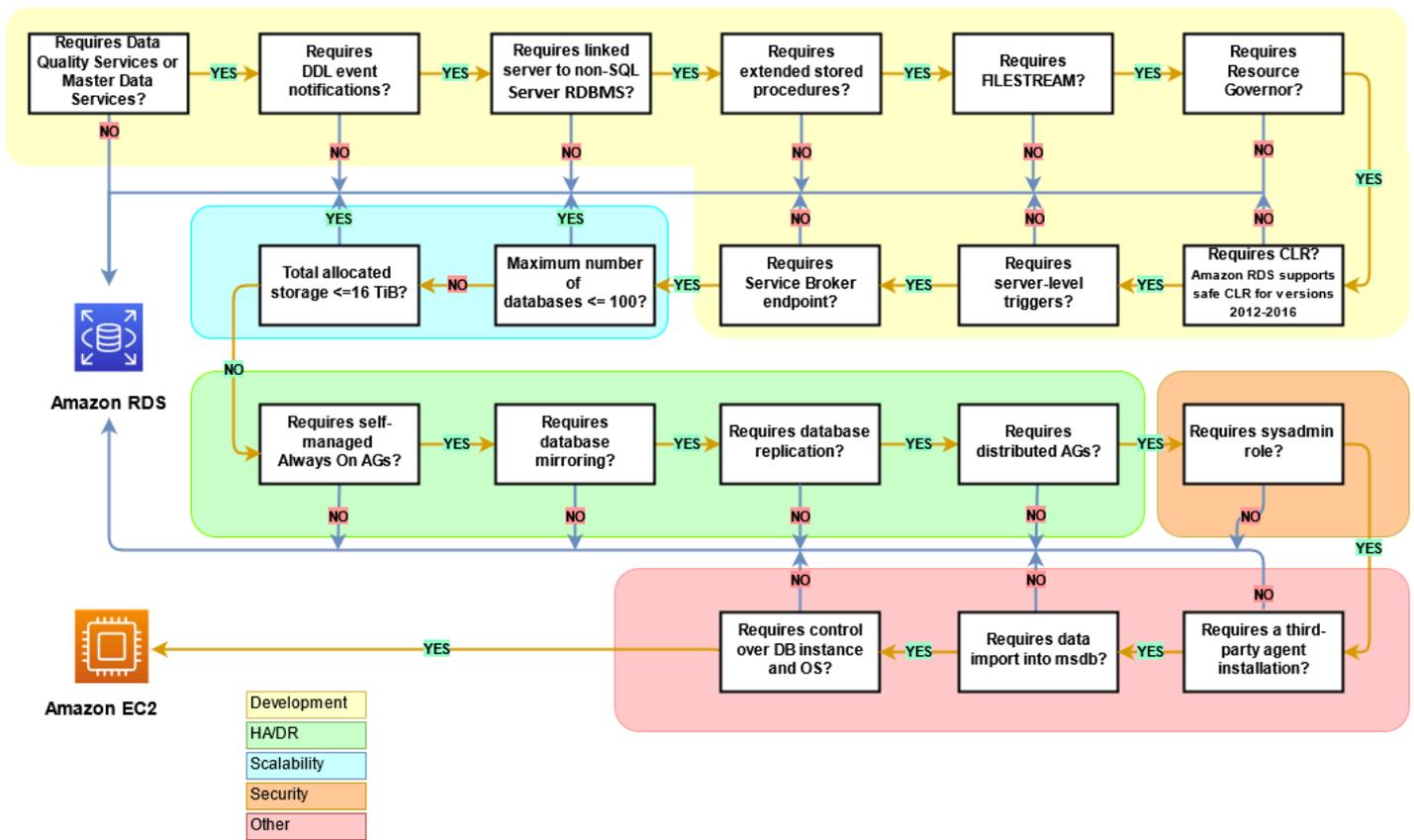
Entwicklungsfunktion	Amazon RDS	Amazon RDS Custom	Amazon EC2	Hinweise
Importieren von Daten in die msdb-Datenbank		N 	J 	Wenn diese Funktion für Ihren Workload von entscheidender Bedeutung ist, sollten Sie Amazon RDS Custom oder Amazon EC2 wählen.
Installationsmethoden	N/A	N/A	Amazon Machine Image (AMI) oder manuelle Installation	
SQL-Server-Editionen	Enterprise, Standard, Web, Express	Enterprise, Standard, Entwickler	Enterprise, Standard, Web, Entwickler, Express	
SQL Server-Versionen	2014, 2016, 2017, 2019, 2022	2019, 2022	2014, 2016, 2017, 2019, 2022	

Ausführliche Informationen zu diesen Funktionen finden Sie hier:

- [Microsoft-Produkte auf AWS](#)
- [Active-Directory-Referenzarchitektur: Implementieren von Active-Directory-Domain-Services in AWS](#)
- [Remote Desktop Gateway aktiviert AWS](#) (AWS Schnellstart)

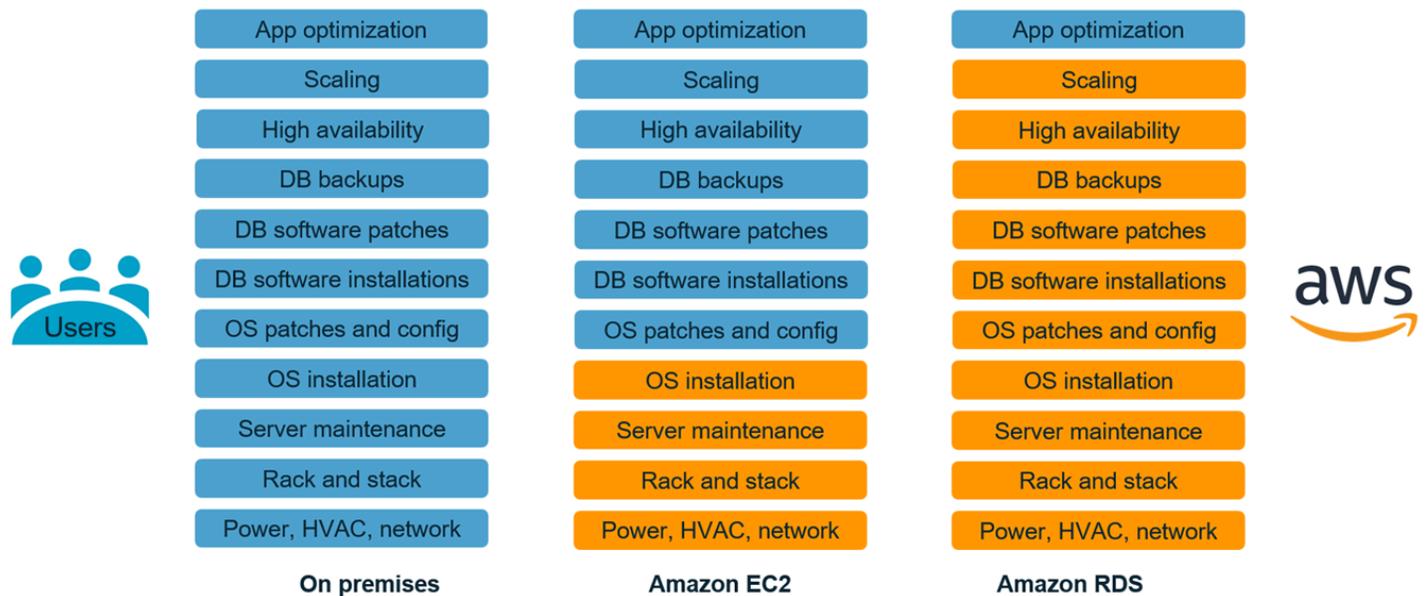
- [Sichern der Microsoft-Plattform auf AWS](#)
- [SQL Server mit AlwaysOn-Replikation auf AWS](#) (AWS Schnellstart)
- [AWS Directory Service](#)
- [AWSEC2-SQLServerDBRestore](#) (AWS Systems Manager Automation-Runbook, das SQL Server-Datenbank-Backups, die in Amazon S3 gespeichert sind, auf SQL Server 2017 wiederherstellt, der auf einer EC2-Linux-Instance ausgeführt wird)

Das folgende Diagramm hilft bei der Visualisierung der Informationen in der vorherigen Tabelle, um Sie bei Ihrem Entscheidungsprozess zu unterstützen.



## Gemeinsame Verantwortlichkeit

Das folgende Diagramm zeigt die Aufteilung der Verantwortlichkeiten zwischen AWS und dem Benutzer bei der Verwaltung von SQL Server-Funktionen und -Operationen.



Mit - AWS Services müssen Sie sich keine Gedanken über Verwaltungsaufgaben wie Serverbereitstellung, Patching, Einrichtung, Konfiguration, Backups oder Wiederherstellung machen. überwacht AWS kontinuierlich Ihre Cluster, um Ihre Workloads mit selbstverarbeitendem Speicher und automatisierter Skalierung betriebsbereit zu halten. Sie konzentrieren sich auf wichtige Aufgaben zur Anwendungsentwicklung wie Schemadesign, Abfrageausführung und Optimierung, während sich in Ihrem Namen um operative Aufgaben AWS kümmert.

Sie müssen niemals zu viel oder zu wenig Infrastruktur bereitstellen, um dem Anwendungswachstum, intermittierenden Spitzen und Leistungsanforderungen gerecht zu werden, oder es müssen feste Kapitalkosten anfallen, einschließlich Softwarelizenzierung und -support, Hardwareaktualisierung und Ressourcen zur Wartung der Hardware. AWS verwaltet diese, sodass Sie Zeit damit verbringen können, Innovationen zu entwickeln und neue Anwendungen zu erstellen, ohne die Infrastruktur zu verwalten.

Weitere Informationen finden Sie unter [Modell der geteilten Verantwortung](#) auf der - AWS Website.

# Migrationsmuster der SQL Server-Datenbank

Verwenden Sie die folgenden Links, um die AWS Präskriptive Guidance Pattern für die Migration von SQL Server-Datenbanken auf AWS:

- [Rehost-Muster \(von SQL Server zu Amazon EC2\)](#)
- [Replattformmuster \(von SQL Server zu Amazon RDS for SQL Server\)](#)
- [Pattern neu erstellen \(von SQL Server zu Open Source und AWS Cloud-native Datenbanken\)](#)

Wenn Sie nach Mustern suchen, die die Verwendung eines bestimmten Tools abdecken, geben Sie den Werkzeugnamen in das Suchfeld ein oder wählen Sie ihn aus einem Filter aus. Sie können beispielsweise die Datei verwenden [diese Abfrage](#) um alle SQL Server-Migrationsmuster anzuzeigen, die AWS DMS aus.

## Partner

Die Datenbankmigration kann ein herausforderndes Projekt sein, das Fachwissen und Tools erfordert. Durch eine Partnerschaft können Sie Ihre Migration und die Zeit bis zu Ergebnissen beschleunigen. [AWS Database Migration ServicePartner](#) verfügen über das erforderliche Fachwissen, um Kunden bei der einfachen und sicheren Migration in die Cloud zu unterstützen. Diese Partner verfügen über das Fachwissen sowohl für homogene Migrationen wie SQL Server zu SQL Server als auch für heterogene Migrationen zwischen verschiedenen Datenbankplattformen wie SQL Server zu Amazon Aurora oder Amazon RDS for MySQL.

Je nach Ihren Anforderungen und Präferenzen können Sie den Partner beauftragen, die gesamte Migration abzuwickeln oder nur bei einigen Aspekten der Migration zu helfen. Darüber hinaus können Sie Tools und Lösungen verwenden, die von Partnern des AWS Partner Network (APN) bereitgestellt werden, um Sie bei der Migration zu unterstützen. Einen vollständigen Katalog von Migrationstools und -lösungen finden Sie unter [AWSPartnertools und -lösungen](#).

# Weitere Ressourcen

## Blog-Posts

- [Regionsübergreifende Disaster Recovery von Amazon RDS for SQL Server](#)
- [Datenbankmigration — Was müssen Sie wissen, bevor Sie beginnen?](#)
- [Bereitstellen von Always-On-Verfügbarkeitsgruppen zwischen Amazon EC2 Windows- und Amazon Linux 2-Instances](#)
- [So erstellen Sie eine hybride Microsoft SQL Server-Lösung mithilfe von verteilten Verfügbarkeitsgruppen](#)
- [Migrieren zu Amazon RDS for SQL Server mithilfe der Transaktionsreplikation](#)
- [Einführung in die fortlaufende Replikation von Amazon RDS for SQL Server mit AWS Database Migration Service](#)
- [Erfahren Sie warum AWS ist die beste Cloud zum Ausführen von Microsoft Windows Server- und SQL Server-Workloads](#)
- [Migrieren Sie Ihre lokalen SQL Server Windows-Workloads zu Amazon EC2 Linux](#)
- [Migrieren einer SQL Server-Datenbank zu einer MySQL-kompatiblen Datenbank-Engine](#)
- [Migrieren Sie Ihre lokalen SQL Server Windows-Workloads zu Amazon EC2 Linux](#)
- [Vereinfachen Sie Ihre Microsoft SQL Server Hochverfügbarkeitsbereitstellungen mit FSx for Windows File Server](#)
- [Speichern von SQL Server-Backups in Amazon S3 mithilfe von Storage Gateway](#)

## AWS-Dokumentation

- [Amazon Aurora](#)
- [Amazon EC2](#)
- [Amazon RDS](#)
- [Amazon RDS Custom](#)
- [AWS DMS](#)
- [AWS SCT](#)
- [SQL Server-Lizenzierung](#)

# Danksagungen

Der Autor dankt den folgenden Fachleuten für ihren Beitrag zu diesem Leitfaden:

- Marcelo Fernandes, Senior Consultant für AWS Migrationen — Abschnitt „[Wahl zwischen Amazon EC2 und Amazon RDS](#)“
- Tarun Chawla, Berater für Datenbankmigrationen — Abschnitt „[Wahl zwischen Amazon EC2 und Amazon RDS](#)“
- Alex Zuo, Senior Technical Product Manager, SQL Server auf Amazon EC2 — Abschnitt [AWS Migration Hub Orchestrator](#)

# Anhang: Fragebogen zur SQL Server-Datenbank-Migration

Verwenden Sie den Fragebogen in diesem Abschnitt als Ausgangspunkt, um Informationen für die Bewertungs- und Planungsphasen Ihres Migrationsprojekts zu sammeln. Sie können diesen Fragebogen im Microsoft Excel-Format herunterladen und damit Ihre Informationen aufzeichnen.

[Fragebogen](#)[herunterladen](#)

## Allgemeine Informationen

1. Wie lautet der Name Ihrer SQL Server-Instance?
2. Wie lautet die Version Ihres SQL Servers?
3. Was ist die Edition Ihrer SQL Server-Datenbank: Standard, Entwickler oder Enterprise?
4. Was ist der Datenbanktyp (OLTP, DW, Reporting, Stapelverarbeitung)?
5. Wie viele Datenbanken haben Sie auf der SQL Server-Instanz?
6. Wie groß ist Ihre Datenbank?
7. Was ist die Sortierung der Datenbank?
8. Wie lautet die Zeitzone der Datenbank?
9. Was sind die durchschnittlichen und maximalen E/A-Transaktionen pro Sekunde (TPS)?
10. Was ist der IOPS (im Durchschnitt und Maximum) für diese Datenbank für Lese-/Schreibvorgänge?
11. Wie viele Transaktionsprotokolle generieren Sie pro Stunde (mit durchschnittlicher und maximaler Größe)?
12. Hat die Datenbank verknüpfte Server, die auf andere Datenbanken verweisen?
13. Was sind die SLA-Anforderungen für Ihre Datenbank?
14. Was sind die RTO- und RPO-Anforderungen für Ihre Datenbank?
15. Wie viel Datenbankausfallzeiten können Sie für Migrationszwecke zulassen?
16. Haben Sie Compliance-, Regulierungs- oder Audit-Anforderungen?
17. Mit welchem Tool überwachen Sie Ihre SQL Server-Datenbanken?

## Infrastruktur

1. Wie lautet der Hostname der Datenbank?
2. Welches Betriebssystem wird für diese Datenbank verwendet?
3. Wie viele CPU-Kerne hat der Server?
4. Wie groß ist die Speichergröße auf dem Server?
5. Befindet sich die Datenbank auf einem virtualisierten Computer oder einem physischen Server?
6. Verwenden Sie lokalen Speicher?
7. Verwenden Sie Storage Type Network Attached Storage (NAS) - oder Storage Area Network (SAN) -Speichertypen?
8. Haben Sie einen Cluster oder einzelne Instanzen?

## Datenbank-Backups

1. Wie sichern Sie Ihre Datenbank? Wie oft?
2. Wie hoch ist Ihre Aufbewahrungsfrist für Transaktionsprotokolle und -sicherungen?
3. Wo lagern Sie Ihr Backup?

## Funktionen der Datenbank

1. Verwenden Sie das automatische Tuning für Ihre SQL Server-Instanz?
2. Verwenden Sie parallel-indizierte Operationen?
3. Verwenden Sie Parallelitätsfunktionen für partitionierte Tabellen?
4. Verwenden Sie Tabellen- und Indexpartitionierung?

## Datenbanksicherheit

1. Verwenden Sie dynamische Datenmaskierung?
2. Verwenden Sie Sicherheitsfunktionen wie Transparent Database Encryption (TDE)?
3. Verwenden Sie Server- oder Datenbankaudits?
4. Verwenden Sie erweiterte Kompression?

# Hohe Verfügbarkeit der Datenbank und Notfallwiederherstellung

1. Was sind Ihre Anforderungen für hohe Verfügbarkeit?
2. Verwenden Sie Transaktions-Replikation?
3. Verwenden Sie Peer-to-Peer-Transaktions-Replikation?
4. Welche Art von Hochverfügbarkeitslösungen (z. B. Failover-Clustering, Always On Availability Groups, Datenbankspiegelung) verwenden Sie für Ihre SQL Server-Umgebung?
5. Wo sind Ihre Primär- und Standby-Datenbankregionen?
6. Was verwenden Sie als Disaster Recovery-Lösung (z. B. Protokollversand, Always On Availability Groups, eine SAN-basierte virtualisierte Umgebung)?
7. Verwenden Sie einen DNS-Alias (Domain Name System) für Datenbankkonnektivität?

# Dokumentverlauf

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

Änderung	Beschreibung	Datum
<a href="#">Abschnitt aktualisiert</a>	Dem Abschnitt <a href="#">Always On Failover</a> Cluster Instances wurden Informationen zu Amazon EBS Multi-Attach mit persistenten Reservierungen und anderen Optionen für AWS die Bereitstellung von FCIs hinzugefügt.	1. April 2024
<a href="#">Aktualisierte Informationen</a>	Die <a href="#">Entscheidungsmatrix</a> für SQL Server 2022 wurde aktualisiert.	18. März 2024
<a href="#">Aktualisierte Informationen</a>	Die <a href="#">Entscheidungsmatrix</a> wurde aktualisiert, um widerzuspiegeln, dass Amazon RDS Custom jetzt TDE unterstützt.	16. November 2023
<a href="#">Abschnitt hinzugefügt</a>	Informationen über hinzugefügt <a href="#">AWS Migration Hub Orchestrator</a> .	29. Juni 2023
<a href="#">Abschnitt entfernt</a>	Informationen zur CloudEndure Migration, die eingestellt wird, wurden entfernt. <a href="#">AWS Application Migration Service</a> ist der primäre Migrationsdienst, der für lift-	23. September 2022

---

	and-shift Migrationen zum empfohlen wird. AWS Cloud	
<a href="#">Abschnitt aktualisiert</a>	Der <a href="#">Entscheidungsmatrix</a> wurden weitere Informationen hinzugefügt.	03. August 2022
<a href="#">Abschnitte wurden hinzugefügt und aktualisiert</a>	Es wurden Informationen zur <a href="#">Migration von SQL Server-Datenbanken zu Amazon RDS Custom</a> und zur <a href="#">Verwendung von Babelfish</a> zur Migration von SQL Server-Datenbanken nach Aurora PostgreSQL hinzugefügt. Die <a href="#">Entscheidungsmatrix</a> wurde mit den neuesten Informationen aktualisiert.	29. Juli 2022
<a href="#">Korrigierter Text</a>	Die CLR-Informationen in der <a href="#">Entscheidungsmatrix</a> wurden korrigiert.	21. Juni 2022
<a href="#">Abschnitt aktualisiert</a>	Der Abschnitt CloudEndure Migration wurde mit den neuesten Informationen zur Produktverfügbarkeit aktualisiert.	10. Mai 2022
<a href="#">Abschnitt wurde entfernt</a>	Informationen zu Amazon RDS für VMware, die nicht mehr unterstützt werden, wurden entfernt.	19. April 2022
<a href="#">Abschnitt hinzugefügt</a>	Es wurden Informationen zum <a href="#">AWS Launch Wizard für SQL Server</a> hinzugefügt.	15. Juli 2021

[Entscheidungsmatrix  
hinzugefügt](#)

Im Abschnitt [Auswahl  
zwischen Amazon EC2 und  
Amazon RDS](#) wurde ein  
side-by-side Vergleich der  
SQL Server-Unterstützung  
bereitgestellt.

28. Juni 2021

[Die Informationen zum  
Protokollversand wurden  
aktualisiert](#)

Im Abschnitt [Protokollversand  
wurde](#) klargestellt, dass für  
den Protokollversand auf  
Amazon RDS for SQL Server  
benutzerdefinierte Skripts  
erforderlich sind.

25. März 2021

[Erste Veröffentlichung](#)

—

9. Oktober 2020

# 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 cloudbasierter Features 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-Repositorys 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 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:

- **Entwicklungsumgebung** – Eine Instance einer laufenden Anwendung, die nur dem Kernteam zur Verfügung steht, das für die Wartung der Anwendung verantwortlich ist. Entwicklungsumgebungen werden verwendet, um Änderungen zu testen, bevor sie in höhere Umgebungen übertragen werden. Diese Art von Umgebung wird manchmal als Testumgebung bezeichnet.
- **Niedrigere Umgebungen** – Alle Entwicklungsumgebungen für eine Anwendung, z. B. solche, die für erste Builds und Tests verwendet wurden.

- Produktionsumgebung – Eine Instance einer laufenden Anwendung, auf die Endbenutzer zugreifen können. In einer CI/CD-Pipeline ist die Produktionsumgebung die letzte Bereitstellungsumgebung.
- Höhere Umgebungen – Alle Umgebungen, auf die auch andere Benutzer als das Kernentwicklungsteam zugreifen können. Dies kann eine Produktionsumgebung, Vorproduktionsumgebungen und Umgebungen für Benutzerakzeptanztests umfassen.

## Epics

In der agilen Methodik sind dies funktionale Kategorien, die Ihnen helfen, Ihre Arbeit zu organisieren und zu priorisieren. Epics bieten eine allgemeine Beschreibung der Anforderungen und Implementierungsaufgaben. Zu den Sicherheitsthemen AWS von CAF gehören beispielsweise Identitäts- und Zugriffsmanagement, Detektivkontrollen, Infrastruktursicherheit, Datenschutz und Reaktion auf Vorfälle. Weitere Informationen zu Epics in der AWS - Migrationsstrategie finden Sie im [Leitfaden zur Programm-Implementierung](#).

## ERP

Siehe [Enterprise Resource Planning](#).

## Explorative Datenanalyse (EDA)

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

## F

### Faktentabelle

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

### schnell scheitern

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

## Grenze zur Fehlerisolierung

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

## Feature-Zweig

Siehe [Zweig](#).

## Features

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

## Bedeutung der Feature

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

## Featuretransformation

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

## 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

### laC

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.

### IIoT

Siehe [Industrielles Internet der Dinge](#).

### unveränderliche Infrastruktur

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

## Eingehende (ingress) VPC

In einer Architektur AWS mit mehreren Konten ist dies eine VPC, die Netzwerkverbindungen von außerhalb einer Anwendung akzeptiert, überprüft und weiterleitet. Die [AWS -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-Servicemanagement)

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

## BIS

Siehe [IT-Informationsbibliothek](#).

## ITSM

Siehe [IT-Servicemanagement](#).

## L

### Labelbasierte Zugangskontrolle (LBAC)

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

### Landing Zone

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

### 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 Netzwerk 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-Entitäts in der IAM-Dokumentation.

## persönlich identifizierbare Informationen (PII)

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

## Personenbezogene Daten

Siehe [persönlich identifizierbare Informationen](#).

## Playbook

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

## PLC

Siehe [programmierbare Logiksteuerung](#).

## PLM

Siehe [Produktlebenszyklusmanagement](#).

## policy

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

## Polyglotte Beharrlichkeit

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

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

### Portfoliobewertung

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

### predicate

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

### Prädikat Pushdown

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

### Präventive Kontrolle

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

### Prinzipal

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

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

### Service-Level-Indikator (SLI)

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

### Service-Level-Ziel (SLO)

Eine Zielkennzahl, die den Zustand eines Dienstes darstellt, gemessen anhand eines [Service-Level-Indikators](#).

### Modell der geteilten Verantwortung

Ein Modell, das die Verantwortung beschreibt, mit der Sie gemeinsam AWS für Cloud-Sicherheit und Compliance verantwortlich sind. AWS ist für die Sicherheit der Cloud verantwortlich, 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 unbekannt 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.