

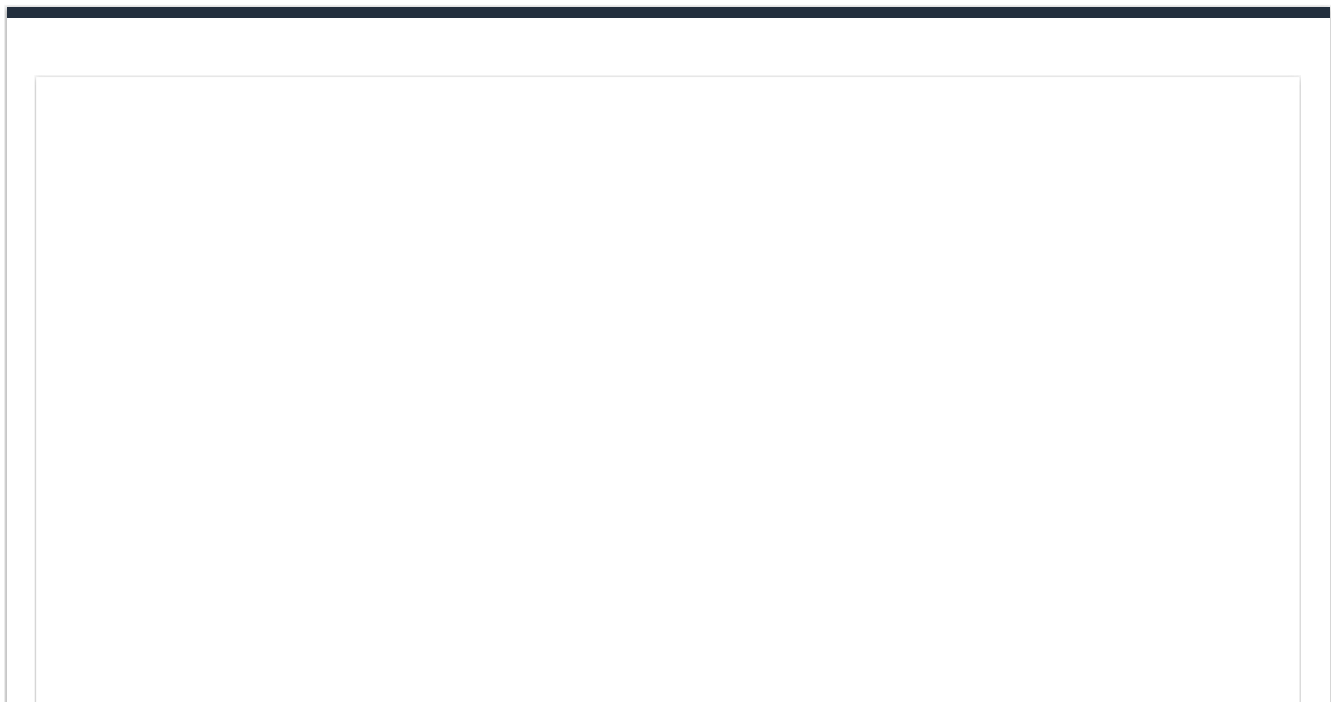
[Kostenfreier Einstieg](#)[Kontakt](#)

Was ist ereignisgesteuerte Architektur?

Entkoppelte Systeme, die als Reaktion auf Ereignisse laufen

Eine ereignisgesteuerte Architektur ist in modernen Anwendungen, die mit Microservices entwickelt werden, üblich. Sie verwendet Ereignisse zum Auslösen und Kommunizieren zwischen entkoppelten Services. Ein Ereignis ist eine Zustandsänderung oder eine Aktualisierung, so wie ein Artikel, das auf einer E-Commerce-Website in einen Einkaufswagen gelegt wird. Ereignisse können entweder den Status enthalten (den gekauften Artikel, seinen Preis und eine Lieferadresse) oder Identifikatoren sein (eine Benachrichtigung, dass eine Bestellung versandt wurde).

Ereignisgesteuerte Architekturen haben drei Schlüsselkomponenten: Ereignisproduzenten, Ereignisrouter und Ereignisverbraucher. Ein Hersteller veröffentlicht ein Ereignis an den Router, der die Ereignisse filtert und an die Verbraucher weiterleitet. Hersteller-Services und Verbraucher-Services sind voneinander entkoppelt, wodurch sie unabhängig voneinander skaliert, aktualisiert und bereitgestellt werden können.



[Kostenfreier Einstieg](#)[Kontakt](#)

weitere Informationen finden Sie in unserer Reihe AWS Online Tech Talks [zur Verwendung von Amazon EventBridge zum Aufbau entkoppelter, ereignisgesteuerter Architekturen](#). Lernen Sie die Grundlagen von ereignisgesteuerten Architekturen und beginnen Sie mit Amazon EventBridge, einschließlich der Erstellung eines Event Bus, dem Einrichten von SaaS-Ereignisquellen und mehr.

Vorteile von ereignisgesteuerter Architektur

Skalieren und fehlschlagen unabhängig voneinander

Durch die Entkopplung Ihrer Services nehmen diese nur den Ereignis-Router wahr, aber nicht einander. Das bedeutet, dass Ihre Services interoperabel sind, aber wenn ein Service ausfällt, laufen die anderen weiter. Der Ereignis-Router fungiert als elastischer Puffer, der Lastspitzen auffangen wird.

Mit Agilität entwickeln

Sie brauchen keinen benutzerdefinierten Code mehr zu schreiben, um Ereignisse abzufragen, zu filtern und weiterzuleiten; der Ereignis-Router filtert und leitet Ereignisse automatisch an die Verbraucher weiter. Durch den Router entfällt auch die aufwändige Koordination zwischen Produzenten- und Konsumenten-Services, was den Entwicklungsprozess beschleunigt.

Prüfung mit Leichtigkeit

Ein Ereignis-Router dient als zentraler Ort für die Prüfung Ihrer Anwendung und die Festlegung von Richtlinien. Diese Richtlinien können einschränken, wer einen Router veröffentlichen und abonnieren darf, und steuern, welche Benutzer und Ressourcen die Berechtigung zum Zugriff auf Ihre Daten haben. Darüber hinaus können Sie Ihre Ereignisse sowohl während der Übertragung als auch im Ruhezustand verschlüsseln.

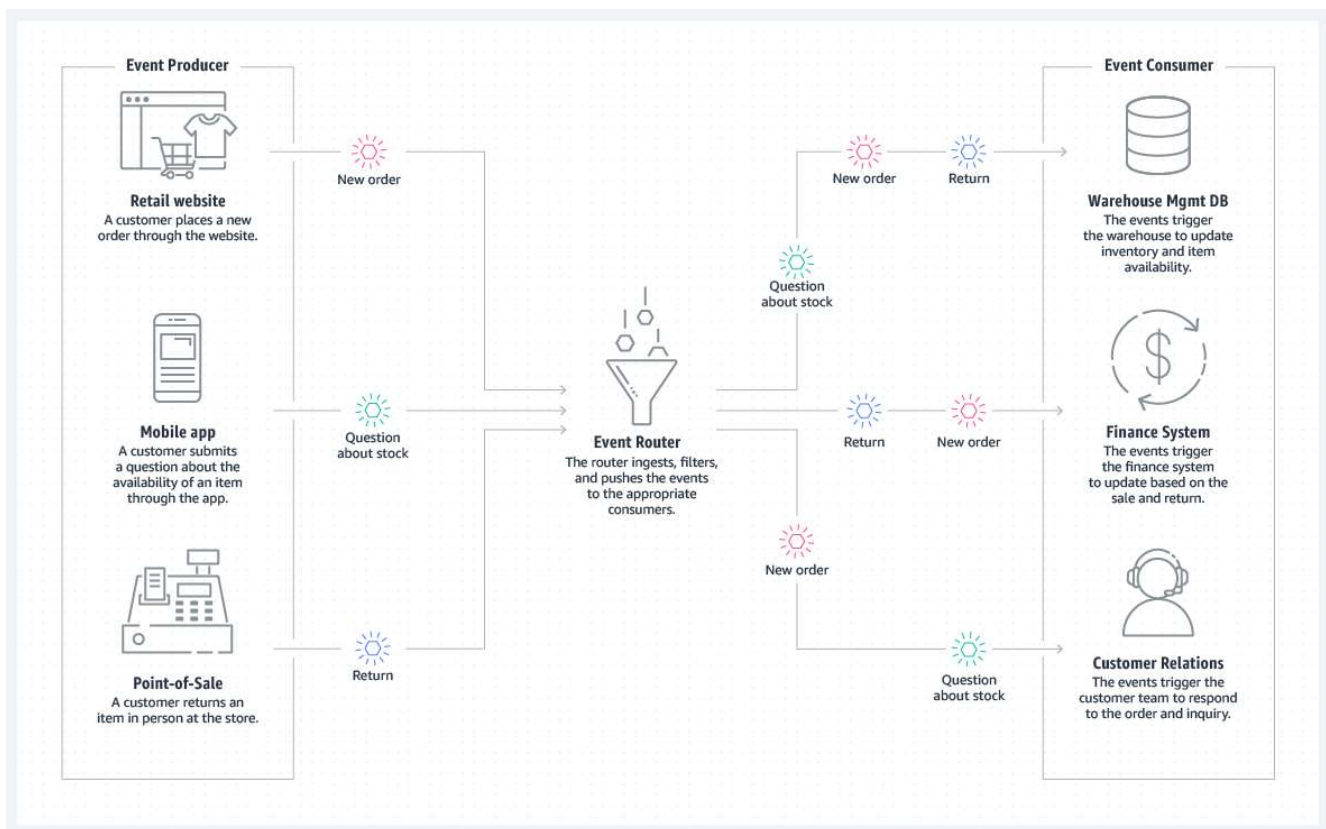
Kosten senken

Ereignisgesteuerte Architekturen sind Push-basiert, das heißt, alles geschieht nach Bedarf, wenn das Ereignis im Router auftritt. Auf diesem Weg zahlen Sie nicht für ständige Abfragen, um ein Ereignis zu

[Kostenfreier Einstieg](#)[Kontakt](#)

So funktioniert es: Beispielarchitektur

Hier ist ein Beispiel für eine ereignisgesteuerte Architektur für eine E-Commerce-Website. Diese Architektur gestattet es der Website, auf Änderungen aus einer Vielzahl von Quellen in Zeiten hoher Nachfrage zu reagieren, ohne die Anwendung zum Absturz zu bringen oder die Ressourcen zu überlasten.



Wann sollte diese Architektur verwendet werden?

Konten- und regionenübergreifende Datenreplikation

Sie können eine ereignisgesteuerte Architektur verwenden, um Systeme zwischen Teams zu koordinieren, die in unterschiedlichen Regionen und auf verschiedenen Konten arbeiten und diese

[Kostenfreier Einstieg](#)[Kontakt](#)

Statt Ihre Ressourcen kontinuierlich zu überprüfen, können Sie eine ereignisgesteuerte Architektur verwenden, um alle Anomalien, Änderungen und Aktualisierungen zu überwachen und entsprechende Warnmeldungen zu erhalten. Diese Ressourcen können Speicher-Buckets, Datenbanktabellen, Serverless-Funktionen, Rechenknoten und mehr umfassen.

Fanout und Parallelverarbeitung

Wenn Sie viele Systeme haben, die auf ein Ereignis reagieren müssen, können Sie eine ereignisgesteuerte Architektur verwenden, um das Ereignis aufzufächern, ohne dafür benutzerdefinierten Code schreiben zu müssen, der an jeden Verbraucher weitergeleitet wird. Der Router leitet das Ereignis an die Systeme weiter, wobei jedes dieser Systeme das Ereignis parallel zu einem anderen Zweck verarbeiten kann.

Integration heterogener Systeme

Wenn Sie Systeme auf verschiedenen Stacks laufen haben, können Sie eine ereignisgesteuerte Architektur verwenden, um Informationen zwischen ihnen ohne Kopplung auszutauschen. Der Ereignis-Router stellt eine indirekte Verbindung und Interoperabilität zwischen den Systemen her, so dass sie Nachrichten und Daten austauschen können, ohne dabei voneinander abhängig zu sein.

Sollten Sie eine ereignisgesteuerte Architektur verwenden?

Ereignisgesteuerte Architekturen sind ideal, um die Agilität zu verbessern und schnell zu handeln. Sie sind häufig in modernen Anwendungen zu finden, die Microservices verwenden, oder in allen Anwendungen, die über entkoppelte Komponenten verfügen. Wenn Sie eine ereignisgesteuerte Architektur einführen, müssen Sie eventuell die Art und Weise, wie Sie Ihr Anwendungsdesign betrachten, neu überdenken. Um sich für den Erfolg zu rüsten, sollten Sie Folgendes bedenken:

- Die Beständigkeit Ihrer Ereignisquelle. Ihre Ereignisquelle sollte zuverlässig sein und die Lieferung garantieren, wenn Sie jedes einzelne Ereignis bearbeiten müssen.
- Ihre Anforderungen an die Leistungskontrolle. Ihre Anwendung sollte in der Lage sein, mit der asynchronen Beschaffenheit von Ereignis-Routern umzugehen.
- Ihre Ereignisablaufverfolgung. Die durch eine ereignisgesteuerte Architektur eingeführte Indirektheit ermöglicht eine dynamische Verfolgung durch Überwachungsservices, aber keine statische Verfolgung durch Codeanalyse.

[Kostenfreier Einstieg](#)[Kontakt](#)

Womit Sie beginnen sollten

Es gibt zwei Hauptarten von Routern, die in ereignisgesteuerten Architekturen verwendet werden: Ereignis-Busse und Ereignisthemen. Wir bei AWS bieten [Amazon EventBridge](#) zum Aufbau von Ereignis-Bussen und [Amazon Simple Notification Service \(SNS\)](#) zum Aufbau von Ereignisthemen an.



Die Nutzung von **Amazon EventBridge** empfiehlt sich, wenn Sie eine Anwendung erstellen möchten, die auf Ereignisse von SaaS-Anwendungen, AWS-Services und individuellen Anwendungen reagiert. EventBridge nutzt ein vordefiniertes Schema für Ereignisse und ermöglicht es Ihnen, Regeln zu erstellen, die auf den gesamten Ereigniskörper angewendet werden, um zu filtern, bevor sie an die Verbraucher weitergeleitet werden.

[Erstellen Sie Ihren ersten Event -Bus mit diesem Tutorial »](#)

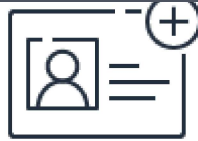
[Vertiefen Sie Ihre Kenntnisse von EventBridge mit diesem Webinar »](#)



Amazon SNS wird empfohlen, wenn Sie eine Anwendung entwickeln möchten, die auf Ereignisse mit hohem Durchsatz und geringer Latenz reagiert, die von anderen Anwendungen, Microservices oder AWS-Services veröffentlicht werden, oder für Anwendungen, die einen sehr hohen Fanout benötigen (Tausende oder Millionen von Endpunkten). SNS-Themen sind unabhängig von dem Ereignisschema, das übermittelt wird.

[Fanout-Ereignisbenachrichtigungen mit diesem Tutorial senden »](#)

[Lesen Sie diesen Blog zum Aufbau ereignisgesteuerter Architekturen »](#)

[Kostenfreier Einstieg](#)[Kontakt](#)

Für ein kostenloses Konto registrieren

Sie erhalten sofort Zugriff auf das kostenlose AWS-Kontingent.

[Registrieren »](#)

ERSTE SCHRITTE MIT AWS

Erfahren Sie, wie Sie in wenigen Minuten mit der AWS-Nutzung beginnen können



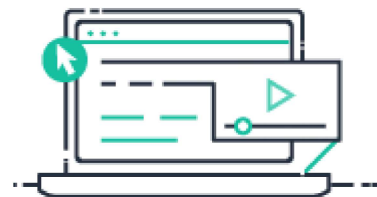
KOSTENLOSES KONTINGENT FÜR AWS

Verschaffen Sie sich 12 Monate kostenlos praktische Erfahrung mit AWS



KOSTENLOSES AWS TRAINING

Greifen Sie auf mehr als 500 kostenlose digitale Kurse für verschiedene Rollen, Kompetenzstufen und Fachgebiete zu, um Ihre AWS-Cloud-Kenntnisse zu erweitern.

[Bei der Konsole anmelden](#)

Ressourcen für AWS

[Erste Schritte](#)[Schulung und Zertifizierung](#)[Mehr über AWS erfahren](#)

[Kostenfreier Einstieg](#)[Kontakt](#)[Was ist ein Container?](#)[AWS-Partner](#)[Was ist ein Data Lake?](#)[AWS Cloud-Sicherheit](#)[Neuerungen](#)[Blogs](#)[Pressemitteilungen](#)

Entwickler in AWS Hilfe

[Entwicklerzentrum](#)[Kontakt](#)[SDKs und Tools](#)[Erhalten Sie Hilfe von Experten](#)[.NET auf AWS](#)[Support-Ticket aufgeben](#)[Python in AWS](#)[AWS re:Post](#)[Java in AWS](#)[Wissenscenter](#)[PHP in AWS](#)[AWS Support – Überblick](#)[JavaScript in AWS](#)[Rechtliche Dokumente](#)[Stellenangebote bei AWS](#)[Erstellen Sie ein AWS-Konto](#)

Amazon.com setzt als Arbeitgeber auf Gleichberechtigung: *Minderheiten/Frauen/Menschen mit Behinderungen/Veteranen/Geschlechtsidentität/sexuelle Orientierung/Alter.*

Sprache

[عربي](#)[Bahasa Indonesia](#)[Deutsch](#)[English](#)[Español](#)[Français](#)[Italiano](#)



Kostenfreier Einstieg

Kontakt

日本語 |
한국어 |
中文 (简体) |
中文 (繁體)

Datenschutz

|

Allgemeine Geschäftsbedingungen

|

Cookie-Einstellungen

|

© 2023, Amazon Web Services, Inc. bzw. Tochtergesellschaften des Unternehmens. Alle Rechte vorbehalten.