



Diagrammi di architettura

Architettura serverless per applicazioni globali



Architettura serverless per applicazioni globali: Diagrammi di architettura

Copyright © 2026 Amazon Web Services, Inc. and/or its affiliates. All rights reserved.

I marchi e l'immagine commerciale di Amazon non possono essere utilizzati in relazione a prodotti o servizi che non siano di Amazon, in una qualsiasi modalità che possa causare confusione tra i clienti o in una qualsiasi modalità che denigri o discrediti Amazon. Tutti gli altri marchi non di proprietà di Amazon sono di proprietà dei rispettivi proprietari, che possono o meno essere affiliati, collegati o sponsorizzati da Amazon.

Table of Contents

Applicazioni globali i

Architettura serverless per applicazioni globali 1

Approfondimenti 2

Storia del diagramma 2

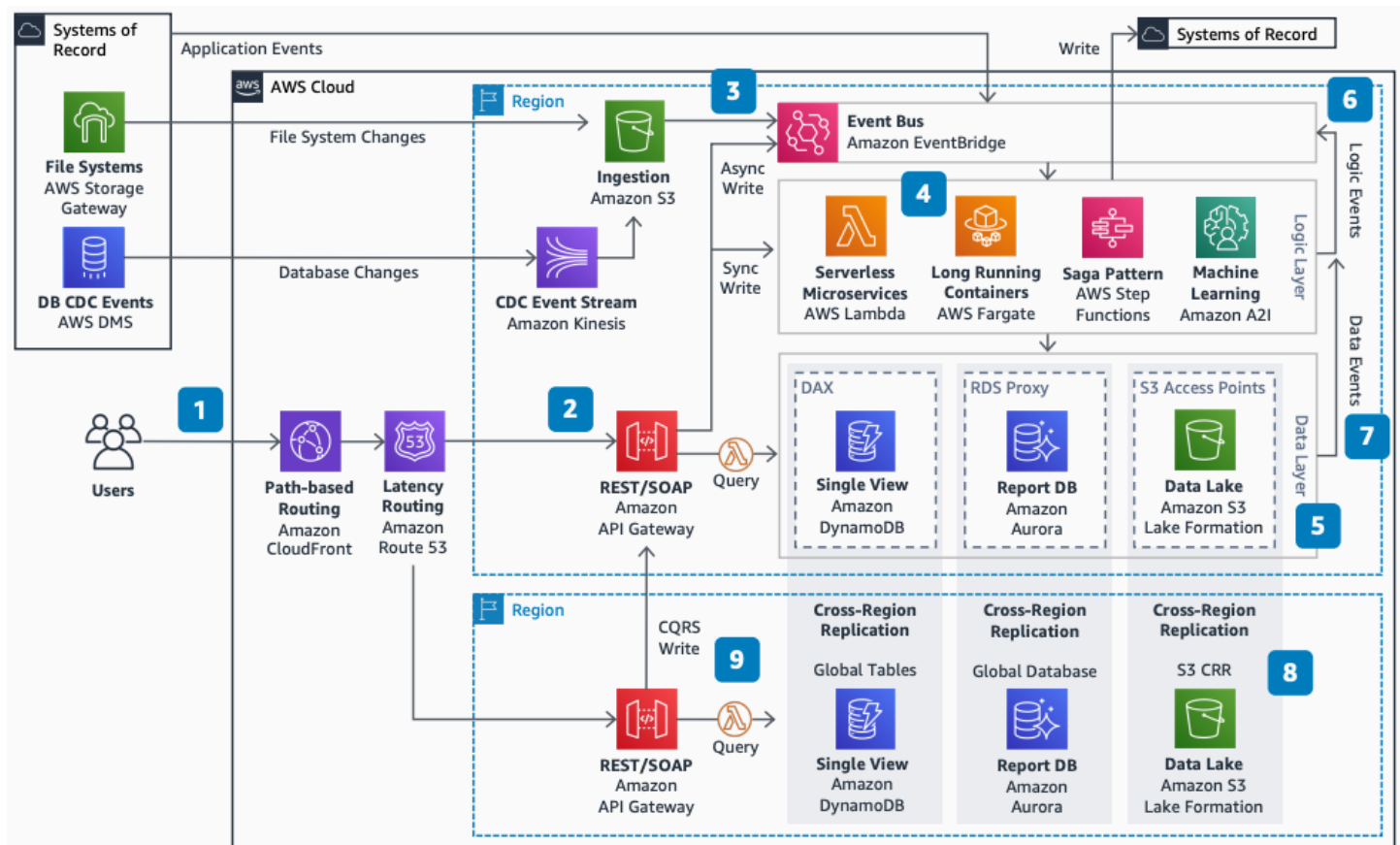
..... iv

Architettura serverless per applicazioni globali

Data di pubblicazione: 15 ottobre 2020 () [Storia del diagramma](#)

Questa architettura mostra come migliorare l'esperienza dei clienti sui servizi globali implementandoli in più AWS regioni. Puoi applicare modelli architetturici basati sugli eventi come event sourcing, saga orchestration e CQRS per ridurre la latenza e aumentare le prestazioni.

Architettura serverless per applicazioni globali



I passaggi seguenti descrivono l'architettura:

1. Indirizza il traffico dalle sedi periferiche in base al percorso della richiesta utilizzando [Amazon CloudFront](#), consentendo la migrazione graduale delle operazioni API legacy a CNAME singolo. Quindi instrada le richieste verso la regione con la latenza minima utilizzando Route 53. <https://docs.aws.amazon.com/Route53/latest/DeveloperGuide/Welcome.html>

2. Fornisci a ciascuna regione un'API di ingresso utilizzando [Amazon API Gateway](#) che implementa il pattern CQRS. Per le richieste di interrogazione, leggi dal livello dei dati. Per i comandi di scrittura sincrona, richiama il livello logico.
3. Elabora i comandi asincroni aggiungendoli al bus degli eventi utilizzando Amazon. [EventBridge](#). Genera eventi aggiuntivi da sistemi esterni.
4. Elabora la logica transazionale con [AWS Lambda](#) funzioni ed esegui attività di lunga durata con contenitori su [AWS Fargate](#). Usa il modello Saga per orchestrare transazioni distribuite per una maggiore coerenza. [AWS Step Functions](#)
5. Archivia i dati per i modelli di accesso utilizzando [Amazon DynamoDB](#) per gli archivi chiave-valore, Amazon Aurora per le query relazionali e Amazon Simple Storage Service per l'analisi dei data lake e. <https://docs.aws.amazon.com/AmazonS3/latest/userguide/Welcome.html> AI/ML
6. Genera eventi logici dal livello logico per eseguire fasi del flusso di lavoro basate sugli eventi seguendo il modello di trasformazione.
7. Genera eventi di dati dopo le modifiche al modello di dati canonico sul livello dati, riducendo la ridondanza nel livello logico.
8. Propaga le modifiche tra le regioni con la replica attiva e attiva dei dati utilizzando DynamoDB Global Tables, Amazon S3 Replication e Amazon Aurora Global Database. Cross-Region
9. Per scenari di lettura impegnativi in altre regioni, invia richieste di scrittura alle regioni principali. Per scritture globali rapide, replica la logica degli eventi e dei comandi su tutte le regioni.

Approfondimenti

Per ulteriori informazioni, consulta le seguenti risorse:

- [AWS Icone dell'architettura](#)
- [AWS Centro di architettura](#)
- [AWS Well-Architected](#)

Storia del diagramma

Per ricevere notifiche sugli aggiornamenti di questo diagramma di architettura di riferimento, iscriviti al feed RSS.

Modifica

Descrizione

Data

[Pubblicazione iniziale](#)

Diagramma dell'architettura di riferimento pubblicato per la prima volta.

15 ottobre 2020

 Note

Per iscriverti agli aggiornamenti RSS, devi avere un plugin RSS abilitato per il browser che stai utilizzando.

Le traduzioni sono generate tramite traduzione automatica. In caso di conflitto tra il contenuto di una traduzione e la versione originale in Inglese, quest'ultima prevarrà.