



Diagrammi di architettura

Esponi il tuo data lake come API GraphQL



Esponi il tuo data lake come API GraphQL: 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

API GraphQL di Data Lake i

Esponi il tuo data lake come API GraphQL 1

Approfondimenti 2

Storia del diagramma 2

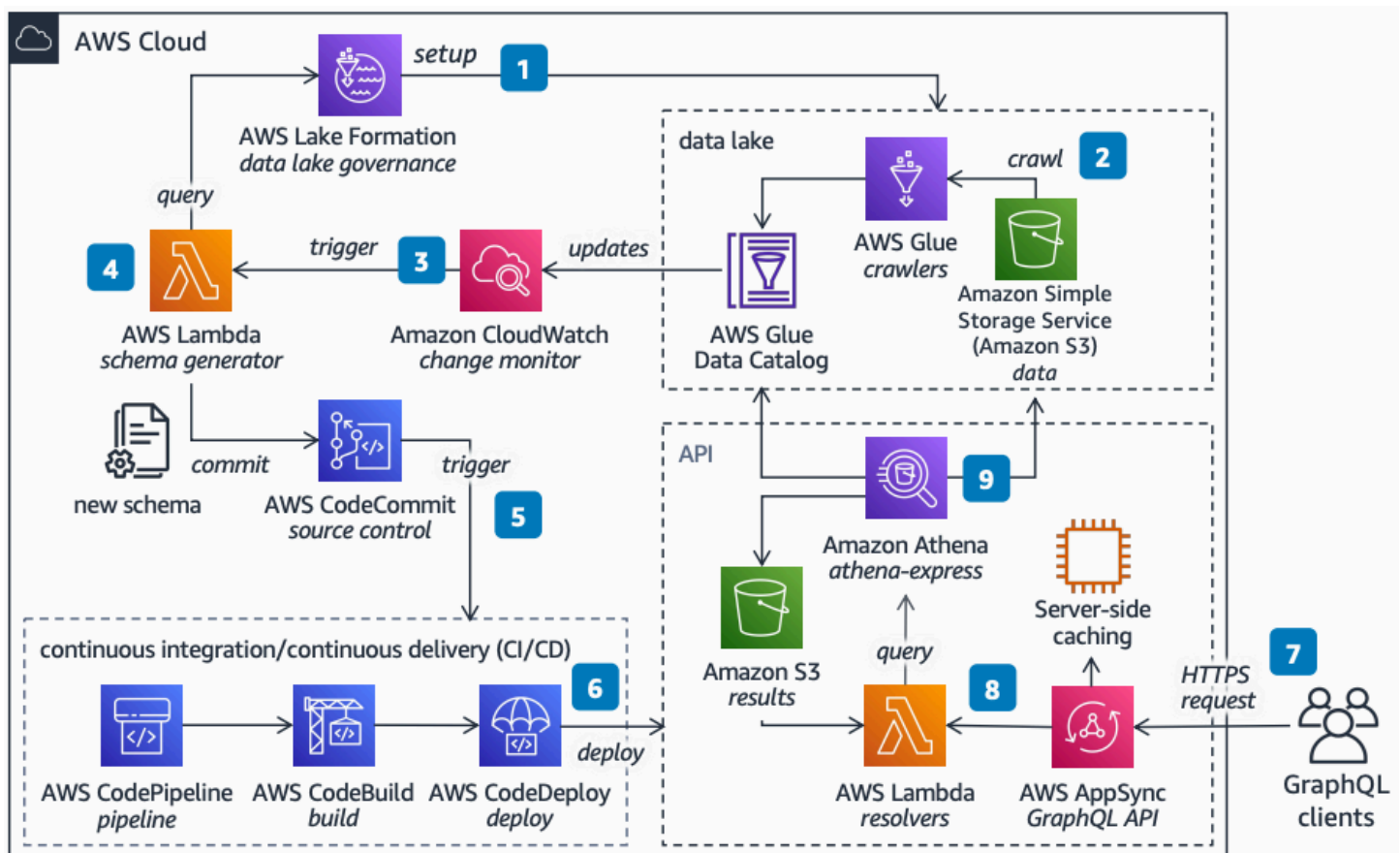
..... iv

Esponi il tuo data lake come API GraphQL

Data di pubblicazione: 25 aprile 2022 () [Storia del diagramma](#)

Questa architettura mostra come creare un data lake utilizzando [Lake Formation](#) e condividerlo come API autodocumentata GraphQL. Puoi creare automaticamente l'API con una CI/CD pipeline reattiva e consentire ai client esterni autorizzati di interrogare il tuo data lake utilizzando GraphQL la sintassi su HTTPS.

Esponi il tuo data lake come API GraphQL



I passaggi seguenti descrivono l'architettura:

1. Configura un data lake utilizzando Lake Formation e [Amazon Simple Storage Service](#).
2. Esegui la scansione dei dati nei bucket Amazon S3 con [AWS Glue](#) per creare un catalogo di dati.

3. Monitora eventuali modifiche nel catalogo dati di AWS Glue con [Amazon CloudWatch](#) e attiva una [AWS Lambda](#) funzione su ogni modifica del catalogo per aggiornare lo schema dell'API esterna.
4. Nella funzione Lambda, interroga il catalogo di dati e le impostazioni di sicurezza più recenti, genera automaticamente il codice sorgente per creare e distribuire una nuova versione dello schema e dei resolver dell'API esterna e invia il codice ad AWS. CodeCommit
5. Configura l'avvio automatico CodeCommit di una nuova CI/CD pipeline su AWS dopo ogni commit di codice. CodePipeline
6. CodePipeline Utilizzalo per creare una nuova versione dell'API con AWS CodeBuild utilizzando il codice sorgente appena generato e utilizza AWS CodeDeploy per distribuire sia la nuova AppSync [GraphQL API](#) Amazon che i nuovi risolutori di query Lambda.
7. Consenti GraphQL ai tuoi clienti di richiamare in modo sicuro l'API esterna tramite HTTPS con il meccanismo di autenticazione scelto. Attiva la memorizzazione nella cache AppSync lato server.
8. Usa i risolutori di query AppSync Lambda per tradurre la query in GraphQL query SQL supportate da Athena. <https://docs.aws.amazon.com/athena/latest/ug/what-is.html>
9. Esegui i job Athena in modo sincrono con la libreria athena-express, temporizzando e limitando opportunamente il set di risultati.

Approfondimenti

Per ulteriori informazioni, consultate 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.

25 aprile 2022

 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.