



아키텍처 다이어그램

Data Lake를 GraphQL API로 노출



Data Lake를 GraphQL API로 노출: 아키텍처 다이어그램

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

Amazon의 상표 및 트레이드 드레스는 Amazon 외 제품 또는 서비스와 함께, Amazon 브랜드 이미지를 떨어뜨리거나 고객에게 혼동을 일으킬 수 있는 방식으로 사용할 수 없습니다. Amazon이 소유하지 않은 기타 모든 상표는 Amazon과 제휴 관계이거나 관련이 있거나 후원 관계와 관계없이 해당 소유자의 자산입니다.

Table of Contents

Data Lake GraphQL API i

Data Lake를 GraphQL API로 노출 1

참조 자료 2

다이어그램 기록 2

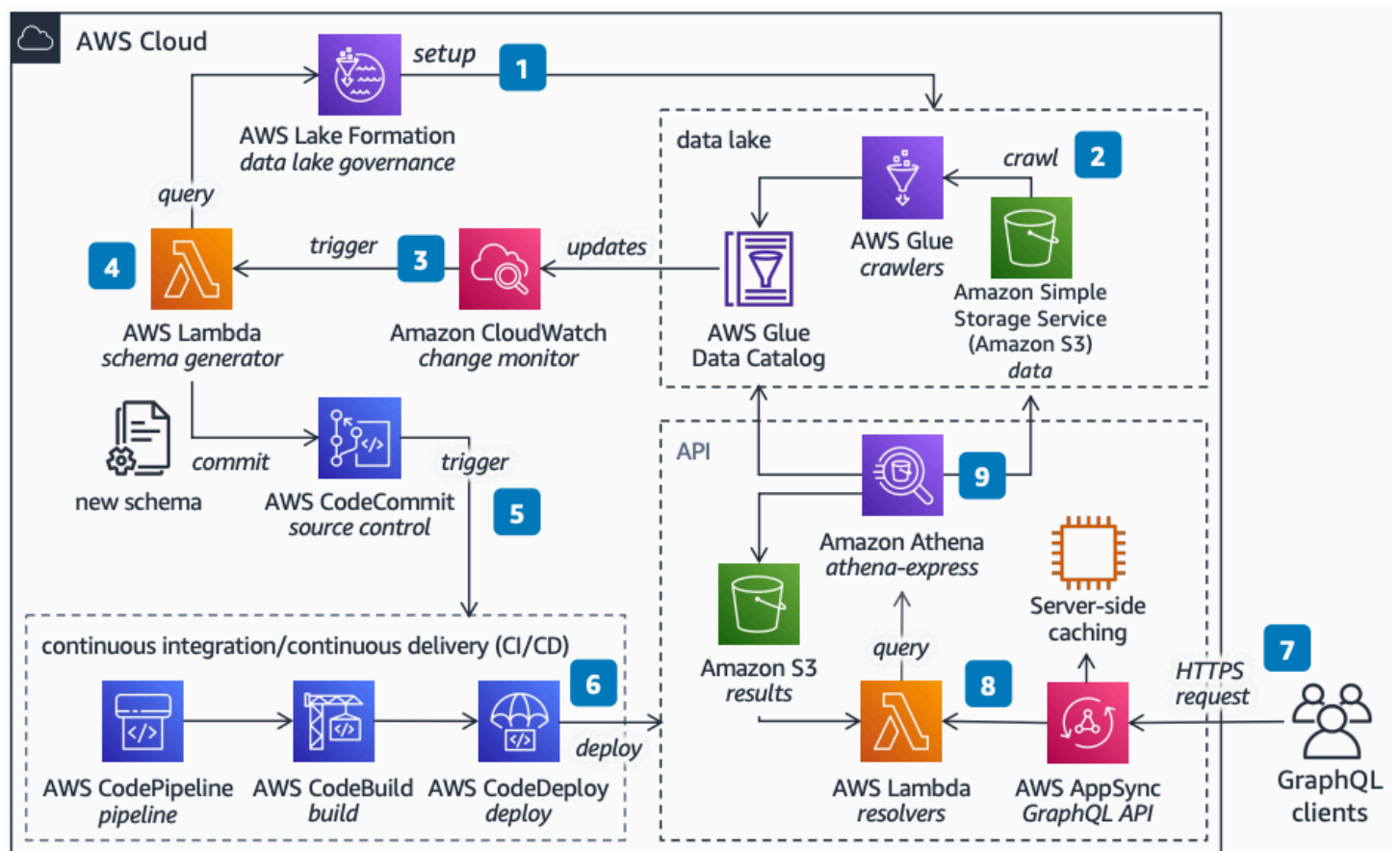
..... iv

Data Lake를 GraphQL API로 노출

게시일: 2022년 4월 25일([다이어그램 기록](#))

이 아키텍처는 [Lake Formation](#)을 사용하여 데이터 레이크를 빌드하고 이를 자체 문서화된 GraphQL API로 공유하는 방법을 보여줍니다. 사후 대응형 CI/CD 파이프라인을 사용하여 API를 자동으로 빌드하고 권한이 있는 외부 클라이언트가 HTTPS를 통한 구문을 사용하여 데이터 GraphQL 레이크를 쿼리하도록 허용할 수 있습니다.

Data Lake를 GraphQL API로 노출



다음 단계에서는 아키텍처를 설명합니다.

1. Lake Formation 및 [Amazon Simple Storage Service](#)를 사용하여 데이터 레이크를 설정합니다.
2. [AWS Glue](#)를 사용하여 Amazon S3 버킷의 데이터를 크롤링하여 데이터 카탈로그를 빌드합니다.
3. [Amazon CloudWatch](#)를 사용하여 AWS Glue 데이터 카탈로그의 변경 사항을 모니터링하고 모든 카탈로그 변경에 대해 [AWS Lambda](#) 함수를 트리거하여 외부 API의 스키마를 업데이트합니다.

4. Lambda 함수에서 최신 데이터 카탈로그 및 보안 설정을 쿼리하고, 소스 코드를 자동으로 생성하여 외부 API 스키마 및 해석기의 새 버전을 빌드 및 배포하고, 코드를 AWS CodeCommit에 커밋합니다.
5. 코드 커밋 후 AWS CodePipeline에서 새 CI/CD 파이프라인을 자동으로 시작하도록 CodeCommit을 설정합니다.
6. CodePipeline을 사용하여 새로 생성된 소스 코드를 사용하여 AWS CodeBuild로 새 버전의 API를 빌드하고 AWS CodeDeploy를 사용하여 새 [Amazon AppSync](#) GraphQL API와 새 Lambda 쿼리 해석기를 모두 배포합니다.
7. GraphQL 클라이언트가 선택한 인증 메커니즘으로 HTTPS를 통해 외부 API를 안전하게 호출하도록 허용합니다. AppSync 서버 측 캐싱을 활성화합니다.
8. AppSync Lambda 쿼리 해석기를 사용하여 GraphQL 쿼리를 [Athena](#)에서 지원하는 SQL 쿼리로 변환합니다.
9. Athena Express 라이브러리와 동기적으로 Athena 작업을 실행하여 적절한 시간 초과 및 결과 세트 제한.

참조 자료

자세한 내용은 다음 리소스를 참조하세요.

- [AWS 아키텍처 아이콘](#)
- [AWS 아키텍처 센터](#)
- [AWS Well-Architected](#)

다이어그램 기록

이 참조 아키텍처 다이어그램의 업데이트에 대한 알림을 받으려면 RSS 피드를 구독하세요.

변경 사항	설명	날짜
최초 게시	참조 아키텍처 다이어그램이 처음 게시되었습니다.	2022년 4월 25일

 Note

RSS 업데이트를 구독하려면 사용 중인 브라우저에 대해 RSS 플러그인이 활성화되어 있어야 합니다.

기계 번역으로 제공되는 번역입니다. 제공된 번역과 원본 영어의 내용이 상충하는 경우에는 영어 버전이 우선합니다.