



Architektur-Diagramme

Multi-Region GraphQL API mit CloudFront



Multi-Region GraphQL API mit CloudFront: Architektur-Diagramme

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

Die Handelsmarken und die 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 Handelsmarken, die nicht Eigentum von Amazon sind, gehören den jeweiligen Besitzern, die möglicherweise zu Amazon gehören oder nicht, mit Amazon verbunden sind oder von Amazon gesponsert werden.

Table of Contents

Multi-Region GraphQL i

Multi-Region GraphQL-API mit CloudFront 1

Weitere Informationen 2

Verlauf des Diagramms 2

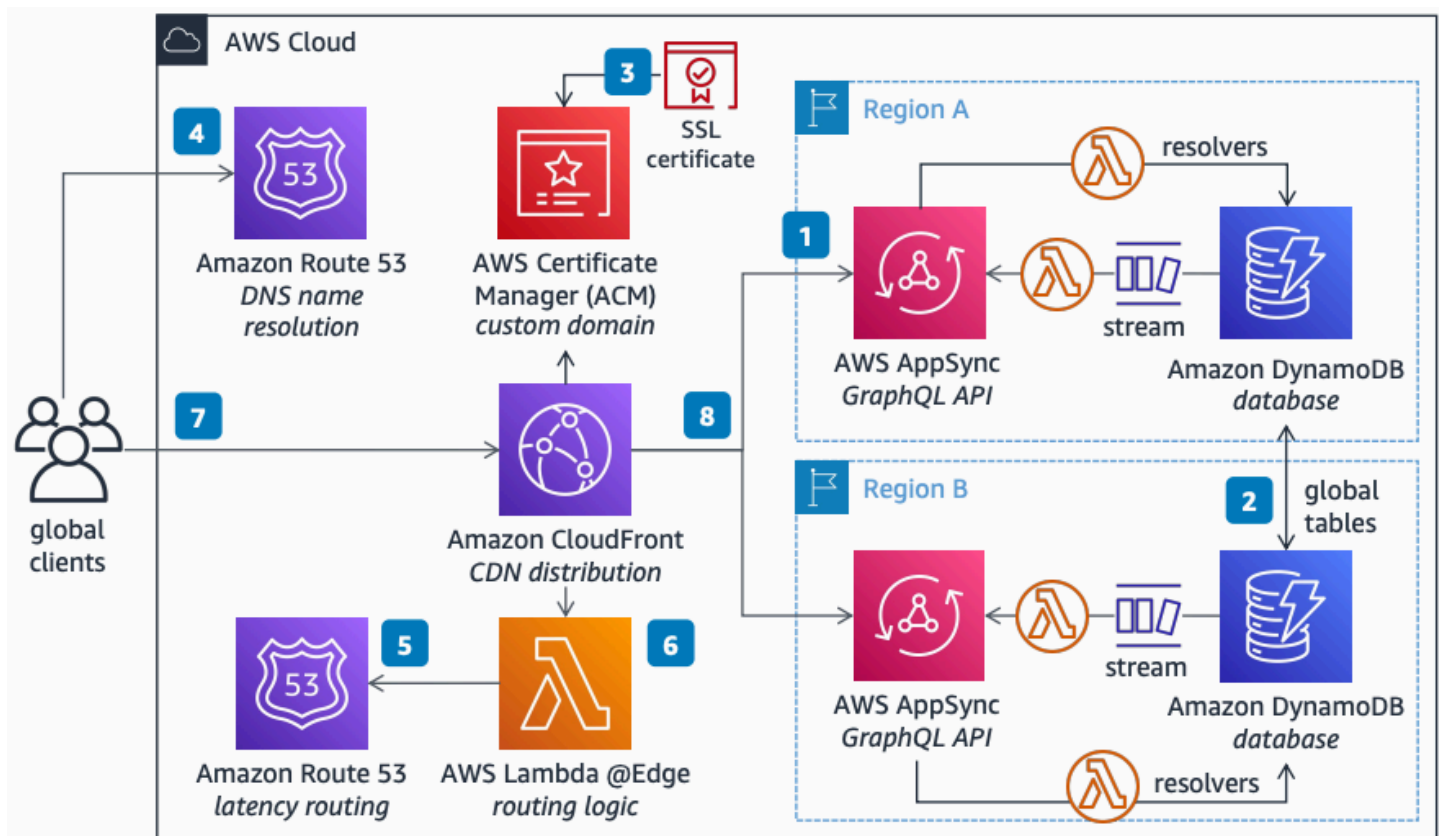
..... iv

Multi-Region GraphQL API mit CloudFront

Veröffentlichungsdatum: 25. April 2022 () [Geschichte des Diagramms](#)

Diese Architektur zeigt, wie Sie die Latenz für Endbenutzer reduzieren und gleichzeitig die Verfügbarkeit Ihrer Anwendung erhöhen können. Sie können GraphQL API-Endpunkte in mehreren AWS Regionen mit active/active Echtzeit-Datensynchronisierung bereitstellen, die von globalen [Amazon](#) DynamoDB-Tabellen unterstützt wird.

Multi-Region GraphQL-API mit CloudFront



Die folgenden Schritte beschreiben die Architektur:

1. Stellen Sie mithilfe von [Amazon eine GraphQL API in zwei oder mehr AWS Regionen bereit und bearbeiten Sie dann die AppSync Befehle und Abfragen mithilfe von AWS Lambda Resolvern \[AppSync\]\(#\)](#), die mit einer DynamoDB-Datenbank verbunden sind.
2. Um Kunden über Datenänderungen in allen Regionen zu informieren, aktivieren Sie globale DynamoDB-Tabellen, um die Daten regionsübergreifend synchron zu halten. Verarbeiten Sie

DynamoDB-Datenströme mit einem Lambda-Handler, der speziell erstellte Schemaabonnements auslöst. GraphQL

3. Um benutzerdefinierte Domains zu unterstützen, laden Sie das SSL-Zertifikat der Domain in den AWS Certificate Manager (ACM) hoch und hängen Sie es an eine Amazon-Distribution an. [CloudFront](#)
4. Verweisen Sie auf Ihren Domainnamen, CloudFront indem Sie Amazon Route 53 als Ihren DNS-Dienst zur Namensauflösung verwenden.
5. Richten Sie eine Routing-Regel auf Route 53 ein, um Ihre globalen Kunden in die Region mit der niedrigsten Latenz zu ihrem Standort weiterzuleiten.
6. Um Kunden nahtlos an AppSync Endpunkten in einer beliebigen Region zu authentifizieren, verwenden Sie Lambda @Edge, um Route 53 nach der besten Region abzufragen, an die die Anfrage weitergeleitet werden kann, und um die Autorisierung zu normalisieren, indem Sie die Besonderheiten der einzelnen Regionen abstrahieren. AppSync
7. Kunden auf der ganzen Welt können dann über einen einzigen Endpunkt, der an Edge-Standorten verfügbar ist, eine Verbindung zu Ihrer GraphQL API herstellen.
8. CloudFront leitet Kundenanfragen nahtlos an die API in der Region mit der niedrigsten Latenz zum Standort des Kunden weiter.

Weitere Informationen

Weitere Informationen finden Sie in den folgenden Ressourcen:

- [AWS Architektur-Ikonen](#)
- [AWS Zentrum für Architektur](#)
- [AWS Well-Architected](#)

Geschichte des Diagramms

Abonnieren Sie den RSS-Feed, um über Aktualisierungen dieses Referenzarchitekturdiagramms informiert zu werden.

Änderung

Beschreibung

Datum

Erste Veröffentlichung

Das Referenzarchitekturdiagramm wurde zuerst veröffentlicht.

25. April 2022

Note

Um RSS-Updates zu abonnieren, muss ein RSS-Plugin für den von Ihnen verwendeten Browser aktiviert sein.

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.