



Architektur-Diagramme

Automatisierte Archivierung für Amazon Redshift



Automatisierte Archivierung für Amazon Redshift: 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

Automatisierte Archivierung i

Automatisierte Archivierung für Amazon Redshift 1

Weitere Informationen 2

Verlauf des Diagramms 2

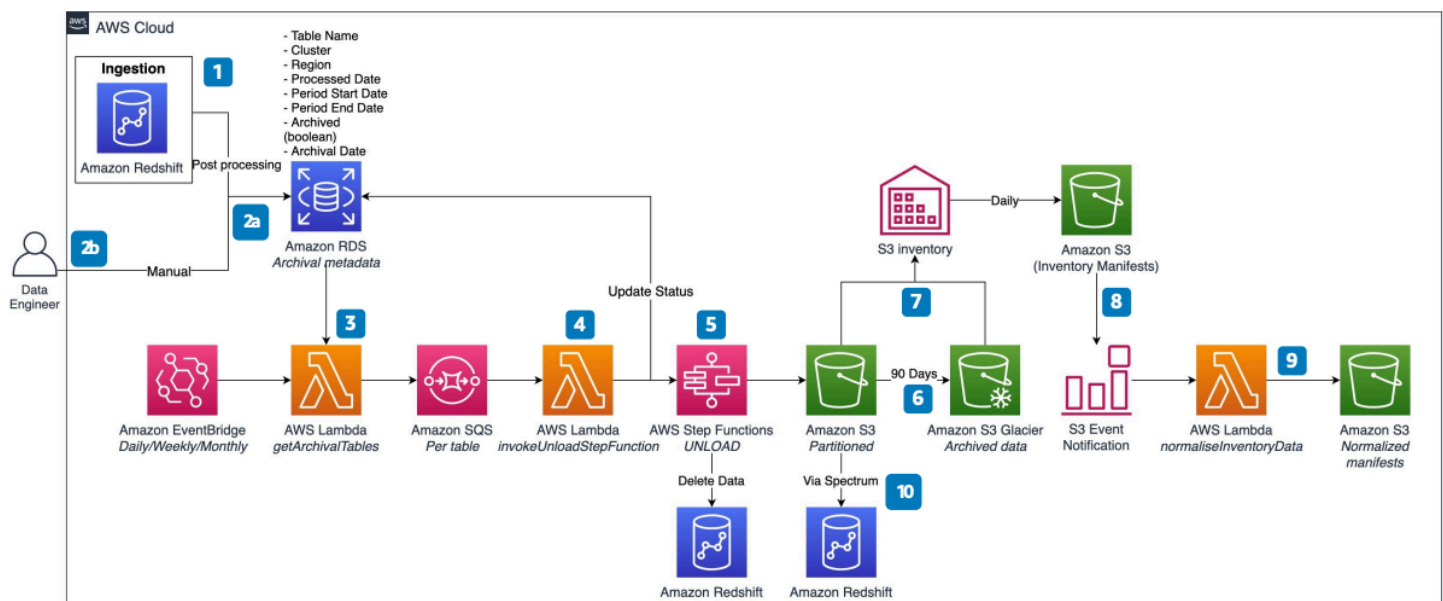
..... iv

Automatisierte Archivierung für Amazon Redshift

Veröffentlichungsdatum: 12. Juli 2021 () [Geschichte des Diagramms](#)

Diese Architektur zeigt, wie der periodische Datenarchivierungsprozess für eine [Amazon Redshift](#)-Datenbank automatisiert werden kann. Die kalten Daten bleiben sofort verfügbar, und Sie können sie mithilfe von Amazon Redshift Spectrum mit vorhandenen Datensätzen im Amazon Redshift-Cluster verbinden.

Automatisierte Archivierung für Amazon Redshift



Die folgenden Schritte beschreiben die Architektur:

1. Nehmen Sie Daten mit unterschiedlichen Frequenzen in den Amazon Redshift-Cluster auf.
2. Erstellen Sie nach jedem Ladevorgang eine Warteschlange mit Metadaten zu Tabellen, die in Amazon Redshift-Tabellen in Amazon RDS gefüllt sind. Dateningenieure können die Archivierungswarteschlange auch manuell erstellen.
3. Verwenden Sie [Amazon EventBridge](#), um eine [AWS Lambda](#) Funktion regelmäßig auszulösen. Die Funktion liest die Warteschlange aus der RDS-Tabelle und erstellt eine [Amazon Simple Queue](#) Service-Nachricht für jeden Zeitraum, der zur Archivierung fällig ist.
4. Eine Proxy-Lambda-Funktion entfernt die Amazon SQS-Nachrichten aus der Warteschlange und ruft sie für jede Nachricht auf. [AWS Step Functions](#)

5. Step Functions entlädt die Daten aus dem Amazon Redshift-Cluster in einen [Amazon Simple Storage](#) Service-Bucket für die angegebene Tabelle und den angegebenen Zeitraum.
6. Die Amazon S3 Lifecycle-Konfiguration verschiebt Daten nach 90 Tagen in Buckets von der S3 Standard-Speicherklasse in die S3 Glacier-Speicherklasse.
7. Das Amazon S3-Inventarisierungstool generiert täglich Manifestdateien aus dem Amazon S3-Bucket, der für kalte Daten vorgesehen ist, und speichert sie in einem Amazon S3-Bucket für Manifestdateien.
8. Jedes Mal, wenn eine Inventar-Manifestdatei in einem Amazon S3-Manifest-Bucket erstellt wird, wird eine Lambda-Funktion durch eine Amazon S3-Ereignisbenachrichtigung ausgelöst.
9. Eine Lambda-Funktion normalisiert die Manifestdatei, sodass sie im Falle einer Wiederherstellung einfach verwendet werden kann.
10. Fragen Sie mithilfe von Amazon Redshift Spectrum die im Amazon S3-Bucket gespeicherten Daten nach kalten Daten ab.

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.	12. Juli 2021

 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.