



Verwenden von materialisierten Ansichten in Amazon Redshift

AWS Prescriptive Guidance



AWS Prescriptive Guidance: Verwenden von materialisierten Ansichten in Amazon Redshift

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

Einführung 1

Übersicht 1

Gezielte Geschäftsergebnisse 1

Materialisierte Ansichten verstehen 2

Ansichten 2

Materialisierte Ansichten 2

Vergleich der Ansichtstypen 3

Erstellen von materialisierten Ansichten 5

Abfragen materialisierter Ansichten 7

Aktualisieren materialisierter Ansichten 9

Automatisches Umschreiben von Abfragen 11

So funktioniert das Umschreiben von Abfragen 11

Vorteile der materialisierten Ansicht 13

Ressourcen 14

Dokumentverlauf 15

..... xvi

Verwenden von materialisierten Ansichten in Amazon Redshift

Ethan Stark, Kelly Ragan und Srinivasan Krishnasamy, Amazon Web Services (AWS)

Dezember 2022 ([Dokumentverlauf](#))

Übersicht

Bei Data-Warehouse-Anwendungen müssen Sie häufig komplexe Abfragen an großen Tabellen durchführen. Die Bearbeitung dieser Anfragen kann zeitaufwändig und teuer sein. Wenn Sie beispielsweise eine Abfrage in Amazon Redshift ausführen, analysiert, validiert, plant, optimiert und führt der Leader-Knoten Ihre Abfrage aus. Dieser Prozess kann Ihre Opportunitätskosten und tatsächlichen Kosten in die Höhe treiben, da er eine beträchtliche Menge an Realzeit, CPU-Zeit und Arbeitsspeicher beansprucht.

In diesem Handbuch erfahren Sie, wie Sie materialisierte Ansichten in Amazon Redshift verwenden, um Abfragen zu beschleunigen, insbesondere vorhersehbare und häufig wiederholte Abfragen. Materialisierte Ansichten reduzieren die Abfragezeit, indem sie eine vorberechnete Ergebnismenge speichern, sodass Sie nicht direkt auf die zugrunde liegenden Basistabellen zugreifen müssen. Verkürzte Abfragezeiten können wiederum die Kosten für die Abfrageverarbeitung senken, sodass Sie Ihre Anwendung kostengünstig skalieren können. Dieses Handbuch richtet sich an Dateningenieure, Datenarchitekten und Datenanalysten.

Gezielte Geschäftsergebnisse

Sie können diesen Leitfaden verwenden, um die folgenden Geschäftsergebnisse zu erzielen:

- Senkung der Kosten für die Verarbeitung von Amazon-Redshift-Abfragen
- Effiziente und kostengünstige Erweiterung Ihrer Anwendung
- Benutzern Zeit für wichtigere Aufgaben geben

Materialisierte Ansichten verstehen

Ansichten

Eine Ansicht ist eine virtuelle Tabelle, die auf der Ergebnismenge einer SQL-SELECT-Abfrage basiert. Die virtuelle Tabelle enthält die aus dem Abfrageausdruck abgerufenen Daten, das Ergebnis wird jedoch nicht auf der Festplatte gespeichert. Wenn Sie Ansichten verwenden, erhalten Sie immer die meisten up-to-date Daten, da die Abfrage bei jeder Ausführung der Abfrage die Daten aus den Originaltabellen abrufen. Sie können eine Ansicht aus einer oder mehreren Basistabellen oder -ansichten erstellen. Sie können eine Ansicht auch auf dieselbe Weise abfragen wie die ursprünglichen Basistabellen.

Die folgende Beispielabfrage zeigt, wie man eine Ansicht erstellt:

```
CREATE VIEW tickets_view
AS
  select e.eventname,
         sum(s.price) as total_sales
  from sales s
  join event e
    on e.eventid = s.eventid
  group by e.eventname;
```

Die folgende Beispielabfrage zeigt, wie man eine Ansicht abfragt:

```
select eventname,
       total_sales
from ticket_view
where eventname = 'Gotterburg';
```

Materialisierte Ansichten

Eine materialisierte Ansicht ist ein Datenbankobjekt, das die Ergebnisse einer Abfrage enthält. Eine materialisierte Ansicht kann beispielsweise jede der folgenden sein:

- Eine lokale Kopie von Daten, die sich ausserhalb befinden
- Eine Teilmenge der Zeilen oder Spalten einer Tabelle oder eines Join-Ergebnisses

- Eine Zusammenfassung, die eine Aggregatfunktion verwendet

Vergleich der Ansichtstypen

In der folgenden Tabelle sind die Unterschiede zwischen einer Ansicht und einer materialisierten Ansicht zusammengefasst.

Key (Schlüssel)	Anzeigen	Materialisierte Ansicht
Definition	Eine virtuelle Tabelle, die keine Daten speichert, sondern stattdessen eine definierte SQL-Abfrage ausführt, um Daten aus einer oder mehreren Tabellen in einer Datenbank abzurufen	Eine virtuelle Tabelle, die durch eine bearbeitbare SQL-Abfrage definiert ist, deren Ergebnis jedoch auf der Festplatte gespeichert wird
Speicher	Das Ergebnis des Abfrageausdrucks wird nicht auf der Festplatte gespeichert – nur der Abfrageausdruck wird auf der Festplatte gespeichert	Der Abfrageausdruck und das Ergebnis des Abfrageausdrucks werden beide auf der Festplatte gespeichert
Ausführen	Die Abfrage, die die Ansicht definiert, wird jedes Mal ausgeführt, wenn in einer Abfrage auf die Ansicht verwiesen wird	Das Ergebnis der Abfrage wird auf der Festplatte gespeichert, und der Abfrageausdruck wird nicht jedes Mal ausgeführt, wenn ein Benutzer versucht, die Daten aus einer materialisierten Ansicht abzurufen
Aktualität der Daten	Stellt immer den neuesten aktualisierten Wert aus den Basistabellen bereit	Stellt nicht den zuletzt aktualisierten Wert bereit, wenn dieser Wert in der Datenbank geändert wird
Kosten	Keine Speicherkosten	Hat Speicherkosten

Design	Um eine Standardansicht zu erstellen, müssen Sie: · Zugriff auf die zugrunde liegenden Tabellen haben · Eine Standard-SELECT-Anweisung verwenden	Um eine materialisierte Ansicht zu erstellen, müssen Sie: · Zugriff auf die zugrunde liegenden Tabellen haben · Eine Standard-SELECT-Anweisung verwenden Optional können Sie Folgendes angeben: · Ob die materialisierte Ansicht in automatischen und manuellen Cluster-Snapshots enthalten ist, die in Amazon Simple Storage Service (Amazon S3) gespeichert werden · Wie die Daten in der materialisierten Ansicht verteilt und sortiert werden · Ob die materialisierte Ansicht automatisch mit den neuesten Änderungen aus den Basistabellen aktualisiert werden soll
Verwendung	Wenn Daten nur selten abgerufen oder aktualisiert werden	Wenn Daten häufig abgerufen oder aktualisiert werden

Erstellen von materialisierten Ansichten

Sie können eine materialisierte Ansicht abhängig von einem oder mehreren der folgenden Elemente erstellen:

- Amazon-Redshift-Tabellen
- Externe Tabellen, die Sie erstellen können, indem Sie [Amazon Redshift Spectrum](#) oder eine [verbundene Abfrage](#) verwenden
- Andere materialisierte Ansichten

Wenn Sie eine materialisierte Ansicht erstellen, führt Amazon Redshift die vom Benutzer angegebene SQL-Anweisung aus, um die Daten aus der Basistabelle oder den Basistabellen zu sammeln. Anschließend speichert Amazon Redshift den Ergebnissatz.

Die folgende Abbildung bietet einen Überblick über eine materialisierte Ansicht namens `mv_total_orders`. Diese Ansicht wird durch eine SQL-Abfrage definiert, die zwei Basistabellen verwendet: Kunde und Auftrag.

Customer

cust_id	first_name
1	Akua
2	Ana
3	Jane
4	Mary
5	Paulo
6	Richard
7	Saanvi
8	Wei

Order

order_id	customer_id	amount
1	5	100
2	3	200
3	2	500
4	1	300
5	2	200
6	5	300
7	6	200



```
CREATE MATERIALIZED VIEW mv_total_orders
AS
  SELECT c.cust_id,
         c.first_name,
         sum(o.amount) as total_amount
  FROM orders o
  JOIN customer c
    ON c.cust_id = o.customer_id
  GROUP BY c.cust_id,
           c.first_name;
```

mv_total_orders

cust_id	first_name	total_amount
1	Akua	300
2	Ana	700
3	Jane	200
5	Paulo	400
6	Richard	200

Abfragen materialisierter Ansichten

Wenn Sie eine materialisierte Ansicht abfragen, greifen Sie direkt auf die vorberechneten Daten in der materialisierten Ansicht zu. Sie können eine materialisierte Ansicht in jeder SQL-Abfrage verwenden, indem Sie auf den Namen der materialisierten Ansicht als Datenquelle verweisen, wie bei einer Tabelle oder Standardansicht.

Betrachten Sie dazu die Beispielillustration für die materialisierte Ansicht `mv_total_orders` vom Abschnitt [Erstellen von materialisierten Ansichten](#) dieses Handbuchs. Wenn Sie eine Abfrage `mv_total_orders` erstellen möchten für (was eine Liste von Kunden mit Bestellungen im Gesamtwert von mehr als 500 USD zurückgibt), dann könnten Sie die folgende Standardabfrage ausführen:

```
statement. SELECT c.cust_id,
                c.first_name,
                sum(o.amount) as total_amount
FROM orders o
JOIN customer c
  ON c.cust_id = o.customer_id
GROUP BY c.cust_id,
         c.first_name
HAVING sum(o.amount) > 500;
```

Die vorherige Abfrage ist jedoch nicht im Hinblick auf Geschwindigkeit optimiert. Wir empfehlen, dass Sie stattdessen die folgende Abfrage ausführen:

```
SELECT cust_id,
       first_name,
       total_amount
FROM mv_total_orders
WHERE total_amount > 500;
```

Die empfohlene Abfrage ist viel schneller, da die Abfrageergebnisse vorberechnet sind und es nicht erforderlich ist, auf die zugrundeliegenden Tabellen (Kunde und Auftrag) zuzugreifen. Amazon Redshift kann die Ergebnisse direkt aus zurückgeben `mv_total_orders`.

 Wichtig

Wenn eine Abfrage auf eine materialisierte Ansicht zugreift, sieht die Abfrage nur die in der materialisierten Ansicht gespeicherten Daten zum Zeitpunkt der letzten Aktualisierung. Daher erfasst die Abfrage möglicherweise nicht alle aktuellen Änderungen aus den korrespondierenden Basistabellen der materialisierten Ansicht.

Aktualisieren materialisierter Ansichten

Eine materialisierte Ansicht enthält eine Momentaufnahme des Abfrageergebnisses. Materialisierte Ansichten werden nicht regelmäßig aktualisiert, es sei denn, Sie konfigurieren Amazon Redshift für regelmäßige Aktualisierungen. Um die Daten in einer materialisierten Ansicht manuell zu aktualisieren, können Sie jederzeit die Anweisung `REFRESH MATERIALIZED VIEW` verwenden. Der Befehl identifiziert Änderungen, die in der oder den Basistabellen stattgefunden haben, und wendet diese Änderungen dann auf die materialisierte Ansicht an.

Es gibt zwei Möglichkeiten, eine materialisierte Ansicht zu aktualisieren: eine manuelle Aktualisierung und eine automatische Aktualisierung (genannt [Auto-Aktualisierung](#)). Die folgende Beispielabfrage zeigt, wie eine materialisierte Ansicht manuell aktualisiert wird:

```
REFRESH MATERIALIZED VIEW mv_total_orders;
```

Um eine materialisierte Ansicht automatisch zu aktualisieren, fügen Sie der Anweisung `CREATE MATERIALIZED VIEW` die Klausel `AUTO REFRESH YES` hinzu, wie das folgende Beispiel zeigt:

```
CREATE MATERIALIZED VIEW mv_total_orders
AUTO REFRESH YES -- Add this clause to auto refresh the MV
AS
SELECT c.cust_id,
       c.first_name,
       sum(o.amount) as total_amount
FROM orders o
JOIN customer c
  ON c.cust_id = o.customer_id
GROUP BY c.cust_id,
         c.first_name;
```

Amazon Redshift aktualisiert materialisierte Ansichten so schnell wie möglich, nachdem sich eine Basistabelle geändert hat. Um die Auswirkungen aktiver Workloads in Ihrem Cluster bei der Verarbeitung der Aktualisierung zu minimieren, berücksichtigt Amazon Redshift die folgenden Faktoren:

- Aktuelle Systemauslastung
- Die für eine Aktualisierung erforderlichen Ressourcen
- Verfügbare Cluster-Ressourcen

- Wie oft die materialisierten Ansichten verwendet werden

Amazon Redshift priorisiert Ihre Workloads gegenüber der automatischen Aktualisierung und kann die automatische Aktualisierung stoppen, um die Leistung des Benutzer-Workloads aufrechtzuerhalten. Bedenken Sie, dass dieser Ansatz die Aktualisierung einiger materialisierter Ansichten verzögern kann. Den Aktualisierungsstatus können Sie unter Ansicht [SVL_MV_REFRESH_STATUS](#) prüfen. In dieser Ansicht werden Abfragen aufgezeichnet, die vom Benutzer initiiert oder automatisch aktualisiert wurden.

Automatisches Umschreiben von Abfragen

Amazon Redshift kann automatisch SQL-Abfragen neu schreiben, die nicht explizit auf bestehende materialisierte Ansichten verweisen, die zur Leistungsverbesserung verwendet werden. Die direkte Abfrage einer materialisierten Ansicht reduziert die Verarbeitungszeit der Abfrage, aber die Abfrageanweisung muss geändert werden. Wenn Sie materialisierte Ansichten verwenden, um Abfragen zu beschleunigen, müssen Sie sich überlegen, wie materialisierte Ansichten systematisch und automatisch zur Beantwortung von Abfragen verwendet werden können. Sie können transparentes Umschreiben verwenden, um materialisierte Ansichten auf die gleiche Weise wie einen Index hinzuzufügen oder zu löschen, ohne dass sich dies auf bestehende SQL-Anweisungen auswirkt. Amazon Redshift verwendet nur up-to-date materialisierte Ansichten, wenn Abfragen automatisch neu geschrieben werden.

So funktioniert das Umschreiben von Abfragen

Erwägen Sie die folgende SQL-Abfrage. Die Abfrage verknüpft zwei Tabellen: Kunde (mehr als 10 Millionen Zeilen) und Bestellung (mehr als 10 Milliarden Zeilen):

```
SELECT c.cust_id,
       c.first_name,
       sum(o.amount) as total_amount
FROM orders o
JOIN customer c
  ON c.cust_id = o.customer_id
GROUP BY c.cust_id,
         c.first_name
HAVING sum(o.amount) > 500;
```

Diese Abfrage verknüpft zwei große Tabellen und wendet eine Summenaggregation auf die `o.amount`-Spalte an und filtert dann die Ergebnisse so, dass nur die Kunden angezeigt werden, die mehr als 500 USD bestellt haben. Diese Abfrage könnte eine Menge Ressourcen verbrauchen.

Betrachten Sie dazu die materialisierte Ansicht `mv_total_orders` vom Abschnitt [Erstellen von materialisierten Ansichten](#) dieses Leitfadens. Nachdem diese Ansicht erstellt wurde, schreibt Amazon Redshift die vorherige Abfrage automatisch wie folgt um:

```
SELECT cust_id,
       first_name,
```

```
total_amount
FROM mv_total_orders
WHERE total_amount > 500;
```

Diese neue Abfrage wird viel schneller ausgeführt, da sie eine materialisierte Ansicht verwendet, anstatt zwei große Tabellen zu verbinden.

Das Umschreiben von Abfragen hat aus den folgenden Gründen zu einer breiteren Akzeptanz materialisierter Ansichten geführt:

1. Materialisierte Ansichten unterstützen das transparente Umschreiben von Abfragen. Beim Umschreiben der Abfrage wird eine in Form von Tabellen ausgedrückte SQL-Anweisung in eine Anweisung umgewandelt, die auf eine oder mehrere materialisierte Ansichten zugreift. Diese Transformation ist für den Endbenutzer transparent und erfordert kein Eingreifen und keinen Verweis auf die materialisierte Ansicht in der SQL-Anweisung.
2. Materialisierte Ansichten vereinfachen die Anwendung von zwischengespeicherten allgemeinen Ergebnismengen und unterstützen abfrageübergreifende Optimierungen wie das Precomputing.
3. In Data Warehouses und Datenintegrationsszenarien können materialisierte Ansichten fremde Tabellenergebnisse materialisieren, um die Unterschiede zwischen mehreren Datenquellen zu verbergen, lokale Replikate zu implementieren oder Lese-/Schreibzugriffe durchzuführen.
4. Das Feature zum Umschreiben von Abfragen in Kombination mit der automatischen Erstellung materialisierter Ansichten beschleunigt die Autonomie der Datenbank.

Vorteile der materialisierten Ansicht

Die Verwendung materialisierter Ansichten bietet mehrere Vorteile:

- **Weniger Updates** – Eine Standardansicht wird nicht physisch umgesetzt; die Abfrage, mit der die Ansicht definiert wird, wird jedes Mal ausgeführt, wenn die Ansicht in einer Abfrage referenziert wird. Im Gegensatz dazu wird eine materialisierte Ansicht wie bei regulären Ansichten aufgrund eines Abfrageausdrucks vorab berechnet und auf der Festplatte gespeichert (ähnlich einem Objekt). Im Gegensatz zu Standardansichten werden materialisierte Ansichten nicht bei jeder Verwendung aktualisiert.
- **Schnellere Reaktionszeiten** – Eine materialisierte Ansicht reagiert schneller als eine Ansicht. Das liegt daran, dass die materialisierte Ansicht im Voraus berechnet wird und somit keine Zeit damit verschwendet wird, die Abfrage oder die Verknüpfungen in der Abfrage zu lösen, die die materialisierte Ansicht erstellen.
- **Gespeicherte SQL-Anweisung** – Sie können anstelle einer materialisierten Ansicht eine Rollup- oder Aggregationstabelle verwenden. Rollup-Tabellen werden im Voraus berechnet und auf der Festplatte gespeichert (ähnlich wie materialisierte Ansichten), aber sie speichern ihre SQL-Anweisungen nicht in der Datenbank. Materialisierte Ansichten speichern ihre SQL-Anweisungen.
- **Einfach zu aktualisieren** – Materialisierte Ansichten lassen sich leicht aktualisieren. Führen Sie einfach den `REFRESH MATERIALIZED VIEW`-Befehl aus.
- **Automatisches Umschreiben von Abfragen** – Der Abfrageoptimierer kann Ihre SQL-Anweisung neu schreiben, um Daten aus einer vorhandenen materialisierten Ansicht abzurufen, auch wenn die materialisierte Ansicht nicht explizit in Ihrer SQL-Anweisung verwendet wird.

Ressourcen

Referenzen

- [Erstellen von materialisierten Ansichten in Amazon Redshift](#)
- [Abfragen einer materialisierten Ansicht](#)
- [Automatisches Umschreiben von Abfragen zur Verwendung materialisierter Ansichten](#)
- [Aktualisieren einer materialisierten Ansicht](#)

Tools

- [Amazon Redshift Query Editor v2.0.](#)

Dokumentverlauf

In der folgenden Tabelle werden wichtige Änderungen in diesem Leitfaden beschrieben. Um Benachrichtigungen über zukünftige Aktualisierungen zu erhalten, können Sie einen [RSS-Feed](#) abonnieren.

Änderung	Beschreibung	Datum
Erste Veröffentlichung	—	15. Dezember 2022

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.