



Utilisation de vues matérialisées dans Amazon Redshift

AWS Conseils prescriptifs



AWS Conseils prescriptifs: Utilisation de vues matérialisées dans Amazon Redshift

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

Les marques et la présentation commerciale d'Amazon ne peuvent être utilisées en relation avec un produit ou un service qui n'est pas d'Amazon, d'une manière susceptible de créer une confusion parmi les clients, ou d'une manière qui dénigre ou discrédite Amazon. Toutes les autres marques commerciales qui ne sont pas la propriété d'Amazon appartiennent à leurs propriétaires respectifs, qui peuvent ou non être affiliés ou connectés à Amazon, ou sponsorisés par Amazon.

Table of Contents

Introduction	1
Présentation	1
Résultats commerciaux ciblés	1
Compréhension des vues matérialisées	2
Vues	2
Vues matérialisées	2
Comparaison des types de vue	3
Création de vues matérialisées	5
Interrogation de vues matérialisées	7
Actualisation de vues matérialisées	9
Réécriture automatique des requêtes	11
Fonctionnement de la réécriture de requêtes	11
Avantages des vues matérialisées	13
Ressources	14
Historique du document	15
.....	xvi

Utilisation de vues matérialisées dans Amazon Redshift

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

Décembre 2022 ([historique du document](#))

Présentation

Les applications d'entrepôt des données vous demandent souvent d'effectuer des requêtes complexes sur de grandes tables. Le traitement de ces requêtes peut être long et coûteux. Par exemple, lorsque vous exécutez une requête dans Amazon Redshift, le nœud principal analyse, valide, planifie, optimise et exécute votre requête. Ce processus peut augmenter vos coûts d'opportunité et vos coûts réels en consommant une quantité importante de temps d'horloge murale, de temps de processeur et de mémoire.

Ce guide explique comment utiliser les vues matérialisées dans Amazon Redshift pour accélérer les requêtes, en particulier les requêtes prévisibles et fréquemment répétées. Les vues matérialisées réduisent le temps de requête en stockant un jeu de résultats préalablement calculé, de sorte que vous n'avez pas à accéder directement aux tables de base sous-jacentes. La réduction des temps de requête peut à son tour réduire le coût du traitement des requêtes, ce qui vous permet de mettre votre application à l'échelle de manière rentable. Ce guide est destiné aux ingénieurs de données, aux architectes de données et aux analystes de données.

Résultats commerciaux ciblés

Vous pouvez utiliser ce guide pour obtenir les résultats métier suivants :

- Réduire le coût de traitement des requêtes Amazon Redshift
- Développer votre application de manière efficace et rentable
- Décharger les utilisateurs pour qu'ils puissent consacrer du temps à des tâches à plus forte valeur ajoutée

Compréhension des vues matérialisées

Vues

Une vue est une table virtuelle basée sur l'ensemble de résultats d'une requête SELECT SQL. La table virtuelle contient les données extraites de l'expression de requête, mais le résultat n'est pas stocké sur le disque. Lorsque vous utilisez des vues, vous obtenez toujours le plus de up-to-date données, car la requête extrait les données des tables d'origine chaque fois que vous exécutez la requête. Vous pouvez créer une vue à partir d'une ou de plusieurs tables ou vues de base. Vous pouvez également interroger une vue de la même manière que vous interrogez les tables de base d'origine.

L'exemple de requête suivant montre la création d'une vue :

```
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;
```

L'exemple de requête suivant montre l'interrogation d'une vue :

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

Vues matérialisées

Une vue matérialisée est un objet de base de données qui contient les résultats d'une requête. Par exemple, une vue matérialisée peut se présenter comme suit :

- Une copie locale de données se trouvant à distance
- Un sous-ensemble des lignes ou des colonnes d'une table ou d'un résultat de jointure
- Un résumé utilisant une fonction d'agrégation

Comparaison des types de vue

Le tableau suivant récapitule les différences entre une vue et une vue matérialisée.

Clé	Afficher	Vue matérialisée
Définition	Table virtuelle qui ne stocke aucune donnée, mais qui exécute une requête SQL définie pour obtenir des données d'une ou de plusieurs tables d'une base de données.	Table virtuelle définie par une requête SQL modifiable, mais le résultat de la requête est stocké sur le disque.
Stockage	Le résultat de l'expression de requête n'est pas stocké sur le disque : seule l'expression de requête est stockée sur le disque.	L'expression de requête et son résultat sont tous deux stockés sur le disque.
Exécuter	La requête qui définit la vue est exécutée chaque fois qu'elle est référencée dans une requête.	Le résultat de la requête est stocké sur le disque et l'expression de requête ne s'exécute pas chaque fois qu'un utilisateur essaie de récupérer les données d'une vue matérialisée.
Récence des données	Fournit toujours la dernière valeur mise à jour à partir des tables de base.	Ne fournit pas la dernière valeur mise à jour si cette valeur est modifiée dans la base de données.
Coût	Aucun coût de stockage	Présence d'un coût de stockage
Conception	Pour créer une vue standard, vous devez effectuer les actions suivantes :	Pour créer une vue matérialisée, vous devez effectuer les actions suivantes :

- Avoir accès aux tables sous-jacentes

- Utiliser une instruction SELECT standard

- Avoir accès aux tables sous-jacentes

- Utiliser une instruction SELECT standard

Le cas échéant, vous pouvez spécifier les options suivantes :

- Si la vue matérialisée est incluse dans des instantanés de cluster manuels et automatisés, qui sont stockés dans Amazon Simple Storage Service (Amazon S3).

- La façon dont les données de la vue matérialisée sont distribuées et triées

- Si la vue matérialisée doit être automatiquement actualisée avec les dernières modifications de ses tables de base ou non.

Utilisation

Lorsque les données sont rarement consultées ou mises à jour

Lorsque les données sont fréquemment consultées ou mises à jour

Création de vues matérialisées

Vous pouvez créer une vue matérialisée basée sur un ou plusieurs des éléments suivants :

- Tables Amazon Redshift
- Tables externes que vous pouvez créer en utilisant [Amazon Redshift Spectrum](#) ou une [requête fédérée](#)
- Toute autre vue matérialisée

Lorsque vous créez une vue matérialisée, Amazon Redshift exécute l'instruction SQL définie par l'utilisateur pour collecter les données de la ou des tables de base. Amazon Redshift enregistre ensuite l'ensemble de résultats.

L'illustration suivante fournit une vue d'ensemble d'une vue matérialisée appelée `mv_total_orders`. Cette vue est définie par une requête SQL qui utilise deux tables de base : `client` et `commande`.

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

Interrogation de vues matérialisées

Lorsque vous interrogez une vue matérialisée, vous accédez directement aux données précalculées dans la vue matérialisée. Vous pouvez utiliser une vue matérialisée dans une requête SQL en référençant le nom de la vue matérialisée comme source de données, comme dans une table ou une vue standard.

Par exemple, considérez l'illustration de l'exemple de vue matérialisée `mv_total_orders` de la section [Création de vues matérialisées](#) de ce guide. Si vous souhaitez créer une requête pour `mv_total_orders` (qui renvoie une liste de clients dont les commandes totalisent plus de 500 USD), vous pouvez alors exécuter la requête standard suivante :

```
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;
```

Cependant, la requête précédente n'est pas optimisée en termes de rapidité. Nous vous recommandons plutôt d'exécuter la requête suivante :

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

La requête recommandée est beaucoup plus rapide, dans la mesure où les résultats de la requête sont précalculés et qu'il n'est pas nécessaire d'accéder aux tables sous-jacentes (client et commande). Amazon Redshift peut renvoyer les résultats directement depuis `mv_total_orders`.

Important

Lorsqu'une requête accède à une vue matérialisée, elle ne voit que les données stockées dans la vue matérialisée actualisée la plus récemment. Par conséquent, la requête ne voit

pas toujours toutes les dernières modifications des tables de base correspondantes de la vue matérialisée.

Actualisation de vues matérialisées

Une vue matérialisée contient un instantané du résultat de la requête. Les vues matérialisées ne sont pas mises à jour régulièrement, sauf si vous configurez Amazon Redshift pour effectuer des mises à jour périodiques. Pour actualiser et mettre à jour manuellement les données de la vue matérialisée, vous pouvez utiliser l'instruction `REFRESH MATERIALIZED VIEW` à tout moment. Cette commande identifie les modifications qui ont lieu dans les tables de base et applique ces modifications à la vue matérialisée.

Il existe deux méthodes pour actualiser une vue matérialisée : une actualisation manuelle et une actualisation automatique (appelée [actualisation automatique](#)). L'exemple de requête suivant montre comment actualiser manuellement une vue matérialisée :

```
REFRESH MATERIALIZED VIEW mv_total_orders;
```

Pour actualiser automatiquement une vue matérialisée, ajoutez la clause `AUTO REFRESH YES` à l'instruction `CREATE MATERIALIZED VIEW` comme le montre l'exemple suivant :

```
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 actualise automatiquement les vues matérialisées dès que possible après la modification des tables de base. Pour réduire l'impact des charges de travail actives dans votre cluster lors du traitement de l'actualisation, Amazon Redshift prend en compte les facteurs suivants :

- Charge actuelle du système
- Les ressources requises pour une actualisation
- Ressources de cluster disponibles
- Fréquence d'utilisation des vues matérialisées

Amazon Redshift privilégie vos charges de travail par rapport à l'actualisation automatique et peut arrêter cette dernière pour préserver les performances de l'application des utilisateurs. N'oubliez pas que cette approche peut retarder l'actualisation de certaines vues matérialisées. Pour connaître l'état de l'actualisation, vous pouvez consulter la vue [SVL_MV_REFRESH_STATUS](#). Cette vue enregistre les requêtes initiées par l'utilisateur ou actualisées automatiquement.

Réécriture automatique des requêtes

Amazon Redshift peut réécrire automatiquement les requêtes SQL qui ne font pas explicitement référence à des vues matérialisées existantes utilisées pour améliorer les performances.

L'interrogation directe d'une vue matérialisée réduit le temps de traitement de la requête, mais l'instruction de la requête doit être modifiée. Si vous utilisez des vues matérialisées pour accélérer les requêtes, il est important de réfléchir à la manière d'utiliser systématiquement et automatiquement des vues matérialisées pour répondre aux requêtes. Vous pouvez utiliser la réécriture transparente pour ajouter ou supprimer des vues matérialisées de la même manière qu'un index, sans affecter les instructions SQL existantes. Amazon Redshift utilise uniquement des vues up-to-date matérialisées lorsqu'il réécrit automatiquement les requêtes.

Fonctionnement de la réécriture de requêtes

Considérez la requête SQL suivante. La requête joint deux tableaux : client (plus de 10 millions de lignes) et commande (plus de 10 milliards de lignes) :

```
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;
```

Cette requête joint deux grands tableaux, applique une agrégation de somme sur la colonne `o.amount`, puis filtre les résultats pour afficher uniquement les clients dont la commande dépasse 500 USD. Cette requête peut consommer beaucoup de ressources.

À titre d'exemple, considérez la vue matérialisée `mv_total_orders` de la section [Création de vues matérialisées](#) de ce guide. Une fois cette vue créée, Amazon Redshift réécrit automatiquement la requête précédente comme suit :

```
SELECT cust_id,
       first_name,
       total_amount
```

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

Cette nouvelle requête s'exécute beaucoup plus rapidement, car elle utilise une vue matérialisée au lieu de joindre deux grands tableaux.

La réécriture des requêtes a conduit à une adoption plus large des vues matérialisées pour les raisons suivantes :

1. Les vues matérialisées permettent la réécriture transparente des requêtes. La réécriture des requêtes transforme une instruction SQL exprimée en termes de tableaux en une instruction accédant à une ou plusieurs vues matérialisées. Cette transformation est transparente pour l'utilisateur final, ne nécessite aucune intervention et aucune référence à la vue matérialisée dans l'instruction SQL.
2. Les vues matérialisées simplifient l'application des ensembles de résultats communs mis en cache et prennent en charge les optimisations des requêtes croisées telles que le calcul préalable.
3. Dans les entrepôts des données et les scénarios d'intégration de données, les vues matérialisées peuvent matérialiser les résultats des tableaux étrangers afin de masquer les différences entre plusieurs sources de données, en implémentant des répliques locaux ou la répartition lecture/écriture.
4. La fonctionnalité de réécriture des requêtes associée à la création automatique de vues matérialisées accélère l'autonomie de la base de données.

Avantages des vues matérialisées

L'utilisation de vues matérialisées présente plusieurs avantages :

- **Moins de mises à jour** : une vue standard n'est pas matérialisée physiquement, ce qui signifie que la requête qui définit une vue standard s'exécute chaque fois que la vue est référencée dans une requête. En revanche, une vue matérialisée est préalablement calculée et stockée sur disque (comme sur un objet) à la suite d'une expression de requête telle que les vues normales. Contrairement aux vues standard, les vues matérialisées ne sont pas mises à jour chaque fois qu'elles sont utilisées.
- **Temps de réponse plus rapides** : une vue matérialisée répond plus rapidement qu'une vue. Cela s'explique par le fait que la vue matérialisée est préalablement calculée et qu'il ne faut donc pas perdre de temps à résoudre la requête ou à joindre la requête qui crée la vue matérialisée.
- **Instruction SQL stockée** : vous pouvez utiliser une table cumulative ou d'agrégation au lieu d'une vue matérialisée. Les tables cumulatives sont préalablement calculées et stockées sur disque (comme dans les vues matérialisées), mais elles ne stockent pas leurs instructions SQL dans la base de données. Les vues matérialisées stockent leurs instructions SQL.
- **Facilité d'actualisation** : les vues matérialisées sont faciles à actualiser. Exécutez simplement la commande `REFRESH MATERIALIZED VIEW`.
- **Réécriture automatique des requêtes** : l'optimiseur de requêtes peut réécrire votre instruction SQL pour récupérer les données d'une vue matérialisée existante, même si la vue matérialisée n'est pas explicitement utilisée dans votre instruction SQL.

Ressources

Références

- [Création de vues matérialisées dans Amazon Redshift](#)
- [Interrogation d'une vue matérialisée](#)
- [Réécriture automatique des requêtes pour utiliser des vues matérialisées](#)
- [Actualisation d'une vue matérialisée](#)

Outils

- [Éditeur de requêtes Amazon Redshift v2.0.](#)

Historique du document

Le tableau suivant décrit les modifications importantes apportées à ce guide. Pour être averti des mises à jour à venir, abonnez-vous à un [fil RSS](#).

Modification	Description	Date
Publication initiale	—	15 décembre 2022

Les traductions sont fournies par des outils de traduction automatique. En cas de conflit entre le contenu d'une traduction et celui de la version originale en anglais, la version anglaise prévaudra.