

 Google Discussions

## Exam Professional Data Engineer All Questions

View all questions & answers for the Professional Data Engineer exam

[Go to Exam](#)

### EXAM PROFESSIONAL DATA ENGINEER TOPIC 1 QUESTION 184 DISCUSSION

Actual exam question from Google's Professional Data Engineer

Question #: 184

Topic #: 1

[\[All Professional Data Engineer Questions\]](#)

You are building a report-only data warehouse where the data is streamed into BigQuery via the streaming API. Following Google's best practices, you have both a staging and a production table for the data. How should you design your data loading to ensure that there is only one master dataset without affecting performance on either the ingestion or reporting pieces?




- A. Have a staging table that is an append-only model, and then update the production table every three hours with the changes written to staging.
- B. Have a staging table that is an append-only model, and then update the production table every ninety minutes with the changes written to staging.
- C. Have a staging table that moves the staged data over to the production table and deletes the contents of the staging table every three hours.
- D. Have a staging table that moves the staged data over to the production table and deletes the contents of the staging table every thirty minutes.

[Show Suggested Answer](#)

by  AWSandeep at Sept. 2, 2022, 10:36 p.m.

### Comments

Type your comment...

  **NicolasN** Highly Voted  2 years, 6 months ago

**Selected Answer: C**

[C]

I found the correct answer based on a real case, where Google's Solutions Architect team decided to move an internal process to use BigQuery.

The related doc is here: <https://cloud.google.com/blog/products/data-analytics/moving-a-publishing-workflow-to-bigquery-for-new-data-insights>

   upvoted 20 times

  **NicolasN** 2 years, 6 months ago

The interesting excerpts:

"Following common extract, transform, load (ETL) best practices, we used a staging table and a separate production table so that we could load data into the staging table without impacting users of the data. The design we created based on ETL best practices called for first deleting all the records from the staging table, loading the staging table, and then replacing the production table with the contents."

"When using the streaming API, the BigQuery streaming buffer remains active for about 30 to 60 minutes or more after use, which means that you can't delete or change data during that time. Since we used the streaming API, we scheduled the load every three hours to balance getting data into BigQuery quickly and being able to subsequently delete the data from the staging table during the load process."

   upvoted 18 times

  **squishy\_fishy** 1 year, 6 months ago

I second this. At my work, I run into this exact steaming buffer thing, it will not let me delete the data until after 60 minutes.

   upvoted 2 times

  **AzureDP900** 2 years, 4 months ago

Agreed C is right



   upvoted 1 times

  **nwk** Highly Voted  2 years, 8 months ago

Vote B - "Some recently streamed rows might not be available for table copy typically for a few minutes. In rare cases, this can take up to 90 minutes"

<https://cloud.google.com/bigquery/docs/streaming-data-into-bigquery#dataavailability>

   upvoted 11 times

  **jkhong** 2 years, 4 months ago


Aren't there other aspects of data pipelining that we should be aware of? other than merely referring to the number of 'recommended' minutes stated in docs. B doesn't address how the appended data is subsequently deleted, since the table is append-only, the size will constantly grow, and so the user may unnecessarily incur more storage costs.

   upvoted 1 times

  **YorelNation** 2 years, 8 months ago

They don't seem concerned too much by data accuracy in the question

   upvoted 1 times

  **devaid** 2 years, 7 months ago



A and B are discarded because the UPDATE statement, is not performance efficient. Neither appending more and more values to the staging table. It's better cleaning the staging table, and merging with the master dataset.

   upvoted 4 times

  **MaxNRG** 1 year, 4 months ago

You can use BigQuery's features like MERGE to efficiently update the production table with only the new or changed data from the staging table, reducing processing time and costs.

   upvoted 1 times

  **SamuelTsch** Most Recent  6 months, 1 week ago

**Selected Answer: A**

deleting data from my point of view is not a good practice to build datawarehouse solutions. So, C and D are excluded. according to the official documentation, the updating/merging process could last till 90 minutes. 3 hours could be enough.

   upvoted 5 times

  **TVH\_Data\_Engineer** 11 months, 2 weeks ago

**Selected Answer: A**

An append-only staging table ensures that all incoming data is captured without risk of data loss or overwrites, which is crucial for maintaining data integrity in a streaming ingestion scenario.

Three Hour Update Interval:

Three-hour update interval.

Updating the production table every three hours strikes a good balance between minimizing the latency of data availability for reporting and reducing the frequency of potentially resource-intensive update operations. This interval is frequent enough to keep the production table relatively up-to-date for reporting purposes while ensuring that the performance of both ingestion and reporting processes is not significantly impacted. Frequent updates (like every ninety minutes or every thirty minutes) could introduce unnecessary overhead and contention, especially if the dataset is large or if there are complex transformations involved.

   upvoted 5 times

  **MaxNRG** 1 year, 4 months ago

**Selected Answer: A**

Not C nor D. Moving and deleting:

Deleting data from the staging table every 3 or 30 minutes could lead to data loss if the production table update fails, and it also requires more frequent and potentially resource-intensive operations.

Options C and D cause rebuilding of the staging table, which slows down ingestion, and may lose data if errors occur during recreation.

A or B

   upvoted 3 times

  **MaxNRG** 1 year, 4 months ago

When designing a report-only data warehouse in BigQuery, where data is streamed in and you have both staging and production tables, the key is to balance the frequency of updates with the performance needs of both the ingestion and reporting processes. Let's evaluate each option:

   upvoted 1 times

  **MaxNRG** 1 year, 4 months ago

A. Staging table as append-only, updating production every three hours: This approach allows for a consistent flow of data into the staging table without interruptions. Updating the production table every three hours strikes a balance between having reasonably fresh data and not overloading the system with too frequent updates. However, this may not be suitable if your reporting requirements demand more up-to-date data.

B. Staging table as append-only, updating production every ninety minutes: This is similar to option A but with a more frequent update cycle. This could be more appropriate if your reporting needs require more current data. However, more frequent updates can impact performance, especially during the update windows.

   upvoted 1 times

  **MaxNRG** 1 year, 4 months ago

C. Staging table moves data to production and clears staging every three hours: Moving data from staging to production and then clearing the staging table ensures that there is only one master dataset. However, this method might lead to more significant interruptions in data availability, both during the move and the clearing process. This might not be ideal if continuous access to the latest data is required.

D. Staging table moves data to production and clears staging every thirty minutes: This option provides the most up-to-date data in the production table but could significantly impact performance. Such frequent data transfers and deletions might lead to more overhead and could interrupt both the ingestion and reporting processes.

   upvoted 1 times

  **MaxNRG** 1 year, 4 months ago

Considering these options, A (Staging table as append-only, updating production every three hours) seems to be the most balanced approach. It provides a good compromise between having up-to-date data in the production environment and maintaining system performance. However, the exact frequency should be fine-tuned based on the specific performance characteristics of your system and the timeliness requirements of your reports.

It's also important to implement efficient mechanisms for transferring data from staging to production to minimize the impact on system performance. Techniques like partitioning and clustering in BigQuery can be used to optimize query performance and manage large datasets more effectively.

   upvoted 2 times

  **Aman47** 1 year, 4 months ago

Neither. In the current scenario, DataStream (a new google resource) captures the CDC data and uses Dataflow to Replicate the changes to big query.

   upvoted 1 times

  **hauhau** 2 years, 5 months ago

**Selected Answer: B**



Vote B

C : the doc says streaming data can be used up to 60 minutes not 3 hours

C : the doc says streaming data can be used up to 90 minutes not 3 hours

B : correct , insert staging table first with append  
and use merge from staging into production table

   upvoted 2 times

  **hauhau** 2 years, 5 months ago



B just say "update", not specifically mention DML. update can be merge

   upvoted 2 times

  **MaxNRG** 1 year, 4 months ago

You can use BigQuery's features like MERGE to efficiently update the production table with only the new or changed data from the staging table, reducing processing time and costs.

   upvoted 1 times

  **zellick** 2 years, 5 months ago

**Selected Answer: C**

C is the answer.

<https://cloud.google.com/blog/products/data-analytics/moving-a-publishing-workflow-to-bigquery-for-new-data-insights>  
Following common extract, transform, load (ETL) best practices, we used a staging table and a separate production table so that we could load data into the staging table without impacting users of the data. The design we created based on ETL best practices called for first deleting all the records from the staging table, loading the staging table, and then replacing the production table with the contents.

When using the streaming API, the BigQuery streaming buffer remains active for about 30 to 60 minutes or more after use, which means that you can't delete or change data during that time. Since we used the streaming API, we scheduled the load every three hours to balance getting data into BigQuery quickly and being able to subsequently delete the data from the staging table during the load process.

   upvoted 7 times

  **Atnafu** 2 years, 5 months ago

C

Following common extract, transform, load (ETL) best practices, we used a staging table and a separate production table so that we could load data into the staging table without impacting users of the data. The design we created based on ETL best practices called for first deleting all the records from the staging table, loading the staging table, and then replacing the production table with the contents.

When using the streaming API, the BigQuery streaming buffer remains active for about 30 to 60 minutes or more after use, which means that you can't delete or change data during that time. Since we used the streaming API, we scheduled the load every three hours to balance getting data into BigQuery quickly and being able to subsequently delete the data from the staging table during the load process.

Building a script with BigQuery on the back end

<https://cloud.google.com/blog/products/data-analytics/moving-a-publishing-workflow-to-bigquery-for-new-data-insights>

   upvoted 3 times

  **John\_Pongthorn** 2 years, 7 months ago

**Selected Answer: C**

D : read more on Streaming inserts and timestamp-aware queries as the following link

it is the same as this question exactly, but it is quite similar.

<https://cloud.google.com/blog/products/bigquery/performing-large-scale-mutations-in-bigquery>

read carefully in the content below.

When using timestamps to keep track of updated and deleted records, it's a good idea to periodically delete stale entries. To illustrate, the following pair of DML statements can be used to remove older versions as well as deleted records.

You'll notice that the above DELETE statements don't attempt to remove records that are newer than 3 hours. This is because data in BigQuery's streaming buffer is not immediately available for UPDATE, DELETE, or MERGE operations, as described in DML Limitations. These queries assume that the actual values for RecordTime roughly match the actual ingestion time.

   upvoted 4 times

  **John\_Pongthorn** 2 years, 7 months ago

[https://cloud.google.com/architecture/database-replication-to-bigquery-using-change-data-capture#prune\\_merged\\_data](https://cloud.google.com/architecture/database-replication-to-bigquery-using-change-data-capture#prune_merged_data)


<https://cloud.google.com/bigquery/docs/reference/standard-sql/data-manipulation-language#limitations>

   upvoted 1 times

  **John\_Pongthorn** 2 years, 7 months ago

Either C or D But When will we delete stale data on staging table ? Every xxx????


[https://cloud.google.com/architecture/database-replication-to-bigquery-using-change-data-capture#prune\\_merged\\_data](https://cloud.google.com/architecture/database-replication-to-bigquery-using-change-data-capture#prune_merged_data)

   unvoted 1 times

 **Oleksandr0501** 1 year, 12 months ago

gpt: "Overall, deleting the staging table every 30 minutes is a better choice than every 3 hours because it reduces the risk of data inconsistencies and performance issues."


   upvoted 1 times

 **TNT87** 2 years, 8 months ago

**Selected Answer: D**

D. Have a staging table that moves the staged data over to the production table and deletes the contents of the staging table every thirty minutes.

   upvoted 2 times

 **TNT87** 2 years, 8 months ago

Ans D

D. Have a staging table that moves the staged data over to the production table and deletes the contents of the staging table every thirty minutes.

   upvoted 1 times

 **AWSandeep** 2 years, 8 months ago

**Selected Answer: D**

D. Have a staging table that moves the staged data over to the production table and deletes the contents of the staging table every thirty minutes.

   upvoted 2 times



## Platform

> Home

> All Exams

> Examtopics PRO

> Training Courses

