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 31 DISCUSSION

Actual exam question from Google's Professional Data Engineer

Question #: 31

Topic #: 1

[All Professional Data Engineer Questions]

You have Google Cloud Dataflow streaming pipeline running with a Google Cloud Pub/Sub subscription as the source. You need to make an update to the code that will make the new Cloud Dataflow pipeline incompatible with the current version. You do not want to lose any data when making this update. What should you do?

A. Update the current pipeline and use the drain flag.

B. Update the current pipeline and provide the transform mapping JSON object.

C. Create a new pipeline that has the same Cloud Pub/Sub subscription and cancel the old pipeline.

D. Create a new pipeline that has a new Cloud Pub/Sub subscription and cancel the old pipeline.

Show Suggested Answer

by [deleted] at *March 20, 2020, 2:13 p.m.*

Comments

Type your comment...

Submit

VishalB Highly Voted 4 years, 9 months ago

Correct Option : A

Explanation:-This option is correct as the key requirement is not to lose

Explanation:- This option is correct as the key requirement is not to lose
the data, the Dataflow pipeline can be stopped using the Drain option.
Drain options would cause Dataflow to stop any new processing, but would
also allow the existing processing to complete

upvoted 80 times

BigQuery 3 years, 5 months ago

To all the New Guys Here. Please don't get confused with all the people's fight over here. Just google the question and you will get the correct ans in many website. Still I recommend to refer this website for question. for this Particular problem ans is A. Reason is here --> https://cloud.google.com/dataflow/docs/guides/updating-a-pipeline#python
have time to read the full page when to use Update using Json mapping and when to use Drain. (you will have question following for Drain option though).
Thumb rule is this,
# If any major change to windowing transformation (like completely changing window fn from fixed to sliding) in Beam/Dataflow/you want to stop pipeline but want inflight data --> use Drain option.
# For all other use cases and Minor changing to windowing fn (like just changing window time of sliding window) --> Use Update with Json mapping.

In this case it is Code change to new version. so, Update with Json mapping. Simple as that.

All the Best Guys.

upvoted 33 times

BigQuery 3 years, 5 months ago

SORRY I MEANT TO SAY ANS IS 'B'. In this case it is Code change to new version. so, Update with Json mapping.

upvoted 6 times

anji007 3 years, 1 month ago

Its clearly mentioned in the question that pipeline in compatible, if it is so you can not update with JSON mapping. Only way is to stop the pipeline with Drain and replace it with a new one. So the closest answer is A only.

upvoted 7 times

maxdataengineer 2 years, 6 months ago

JSON Mapping is a way to solve compatibility issues when updating

upvoted 1 times

sergio6 3 years, 8 months ago

C and D are incorrect because canceling the old pipeline can cause data loss
https://cloud.google.com/dataflow/docs/guides/stopping-a-pipeline
A is incorrect because updating pipeline does not include any drain flag
https://cloud.google.com/dataflow/docs/guides/updating-a-pipeline

upvoted 7 times

Tanzu 3 years, 3 months ago

drain is in the guide ...stopping-a-pipeline. Just ...updating-a-pipeline is not enough to evaluate this question.

that's why drainnin is not a flag in a pipeline update. it is a process about how to stop a pipeline w/o data loss !

data in dataflow is in 3 stages. ingestion data, buffered data and in-flight data which is processing by old pipeline.

upvoted 2 times

sergio6 3 years, 6 months ago

B is correct: Update the current pipeline and provide the transform mapping JSON object.
Dataflow always performs a compatibility check between the old and new job and without the mapping (necessary as old and new are incompatible) it would give an error and the old job would continue to be executed
https://cloud.google.com/dataflow/docs/guides/updating-a-pipeline#Mapping
https://cloud.google.com/dataflow/docs/guides/updating-a-pipeline#CCheck

upvoted 5 times

Tanzu 3 years, 3 months ago

new pipeline is incompatible means, compatibility check will fail. so you wil not be able to update as new pipeline.

that's why B cannot be valid answer here in this context.

upvoted 2 times

maxdataengineer 2 years, 6 months ago

B is a way to solve compatibility issues

upvoted 2 times

VishalB 4 years, 9 months ago

Option C & D are incorrect as Cancel Option will lead to loose the data
Option B is very Close, since the new Code make pipeline incompatible by providing transform mapping JSON file you can handle this

upvoted 3 times

sergio6 3 years, 6 months ago

A is incorrect because updating pipeline does not include any drain flag

upvoted 2 times

Tanzu 3 years, 3 months ago

two steps.. 1st drain the job w/ sdk or console. then, update the pipeline. cause it is OK to update a job while in draining

upvoted 2 times

maxdataengineer 2 years, 6 months ago

Yes but the compatibility problem will still be there, stopping the pipeline does not solve that

upvoted 1 times

Tanzu 3 years, 3 months ago

There are 5 update scenarios in a job in update-a-pipeline context.
a- changing transform name (requires mapping) , adding a new step (no need for mapping)
b- windowing or triggering (only for minor changes, otherwise don't do that)
c- coders (don't do that
d- schema (adding or required to nullable is possible) other scenarios not possible
e- stateful operations

none of them are relevant, here. cause there is no specific detail, secondly incompatible w. new pipeline.

and mostly if in compatible only a has a solve. but not for all cases.
so, drain == no data loss (ingesting, buffered and in-flight data) is the only scenario.

upvoted 6 times

maxdataengineer 2 years, 6 months ago

As you said, Drain stops the pipeline but it does not solve the compatibility issue. The pipeline will not be able to be updated which is the core problem of the question.

upvoted 3 times

assU2 2 years, 5 months ago

You do not want to lose any data when making this update - is the core problem. You are doing it ANYWAY.

upvoted 2 times

[Removed] Highly Voted 5 years, 1 month ago

Correct B - https://cloud.google.com/dataflow/docs/guides/updating-a-pipeline#preventing_compatibility_breaks

upvoted 24 times

[Removed] 5 years, 1 month ago

Changing the pipeline graph without providing a mapping. When you update a job, the Dataflow service attempts to match the transforms in your prior job to the transforms in the replacement job in order to transfer intermediate state data for each step. If you've renamed or removed any steps, you'll need to provide a transform mapping so that Dataflow can match state data accordingly.

upvoted 4 times

arnabbis4u 5 years ago

The job can be incompatible for reasons other than transformation changes. Since it is clearly mentioned that the change job is incompatible, I think we have to create a new job and D should be correct.

upvoted 9 times

sergio6 3 years, 6 months ago

Canceling the job will cause data loss, against the requirement

upvoted 3 times

abhaya2608 Most Recent 1 month, 1 week ago

Selected Answer: B

Please refer the google doc link below,
https://cloud.google.com/dataflow/docs/guides/stopping-a-pipeline#drain

upvoted 1 times

Parandhaman_Margan 1 month, 3 weeks ago

Selected Answer: D

Create a new pipeline that has a new Cloud Pub/Sub subscription and cancel the old pipeline

upvoted 1 times

dans_puts 4 months, 3 weeks ago

Selected Answer: D

D. Create a new pipeline that has a new Cloud Pub/Sub subscription and cancel the old pipeline:

By creating a new subscription, the new pipeline will consume messages independently of the old pipeline. This ensures no data is lost as messages published to Pub/Sub are delivered to all subscriptions.
Once the new pipeline is verified to be running as expected, the old pipeline can be safely canceled.

upvoted 1 times

Smakyel79 5 months, 2 weeks ago

Why option D is better for this case - In this scenario: the pipeline is incompatible with the old one; running the two pipelines concurrently ensures no data loss and allows for easier debugging of the new pipeline; a new subscription ensures the old pipeline can finish processing its messages while the new pipeline starts fresh

upvoted 1 times

Smakyel79 5 months, 2 weeks ago

Draining a pipeline stops it in an orderly manner but does not address incompatibility issues. Once the pipeline is drained, no more data is processed, and the new pipeline starts fresh. This can lead to data loss if there are messages in Pub/Sub that the drained pipeline didn't process.

upvoted 1 times

philli1011 1 year, 3 months ago

Option: C
using draining will stop the subscription totally while allowing the existing data to complete processing. While the pipeline is stopped, will lose streaming data. The best option is to create a new pipeline that is connected to the same subscription, then we can apply drain to the old pipeline and end it. That way we will capture all the streaming data.

upvoted 2 times

MaxNRG 1 year, 4 months ago

Selected Answer: A

Drain flag: This flag allows the pipeline to finish processing all existing data in the Pub/Sub subscription before shutting down. This ensures no data is lost during the update.
Current pipeline: Updating the current pipeline minimizes disruption and avoids setting up entirely new infrastructure.
Incompatible changes: Even with incompatible changes, the drain flag ensures existing data is processed correctly.

upvoted 1 times

> **MaxNRG** 1 year, 4 months ago
>
> While other options might work in some cases, they have drawbacks:
>
> B. Transform mapping JSON: This option is mainly for schema changes and doesn't guarantee data completion before shutdown.
> C. New pipeline, same subscription: This risks duplicate processing of data if both pipelines run concurrently.
> D. New pipeline, new subscription: This loses the current pipeline's state and potentially data, making it impractical for incompatible changes.
> Therefore, the most reliable and data-safe approach is to update the current pipeline with the drain flag for seamless transition and data integrity.
>
> Remember, always test updates in a staging environment before deploying to production.
>
> upvoted 1 times

TVH_Data_Engineer 1 year, 4 months ago

Selected Answer: C

Same Cloud Pub/Sub Subscription: By using the same Cloud Pub/Sub subscription for the new pipeline, you ensure that no messages are lost during the transition. Pub/Sub manages message delivery, ensuring that unacknowledged messages (those that haven't been processed by your old pipeline) will be available for the new pipeline to process.

Creating a New Pipeline: Since the update makes the new pipeline incompatible with the current version, it's necessary to create a new pipeline. Attempting to update the current pipeline in place (options A and B) would not be feasible due to compatibility issues.

Cancel the Old Pipeline: Once the new pipeline is up and running and processing messages, you can safely cancel the old pipeline. This ensures a smooth transition with no data loss.

upvoted 2 times

JOKKUNO 1 year, 5 months ago

In order to make an update to a Google Cloud Dataflow streaming pipeline without losing any data, the recommended approach is:

A. Update the current pipeline and use the drain flag.

Explanation:

The drain flag is designed to allow the current pipeline to finish processing any remaining data before shutting down. This helps ensure that no data is lost during the update process.
By updating the current pipeline and using the drain flag, you allow the pipeline to complete its current processing before the update takes effect, minimizing the risk of data loss.
This approach is a safe way to transition from the old version to the new version without interrupting data processing.

upvoted 1 times

axantroff 1 year, 5 months ago

I would vote for A because of the structure of the exam, but there are other options worth considering as well

upvoted 1 times

RT_G 1 year, 5 months ago

Selected Answer: C

My answer is C. Chatted with ChatGPT and narrowed down on this option. Let me know your thoughts on this perspective.
Option C - By using the existing subscription, you can ensure that the data flow remains uninterrupted, and there is no loss of data during the transition from the old pipeline to the new one.

Creating a new pipeline that uses the same Cloud Pub/Sub subscription allows for a seamless transition without any interruptions to the data flow. This approach ensures that the new pipeline can continue to consume data from the same subscription as the old pipeline, thereby maintaining data continuity throughout the update process.

upvoted 1 times

rocky48 1 year, 5 months ago

Selected Answer: A

Correct Option : A
Explanation:-This option is correct as the key requirement is not to lose
the data, the Dataflow pipeline can be stopped using the Drain option.

upvoted 1 times

mk_choudhary 1 year, 6 months ago

It should be B
Drain will stop the existing job only and it does not suffice the updated schema.
In order to bring updated schema into effect, updated JSON mapping need to be applied.

upvoted 1 times

Simhamed2015 1 year, 6 months ago

The two cores of this question are: 1- Don't lose data ← Drain, is perfect for this because you process all buffer data and stop reviving messages; normally this message is alive for 7 days of retry, so when you start a new job you will receive all without lose any data. 2- Incompatible new code ← mapping solve some incompatibilities like change name of a ParDO but no a version issue. So launch a new job with the new code, it's the only option.
So, option is A.

upvoted 1 times

imran79 1 year, 7 months ago

The best choice is D. Create a new pipeline that has a new Cloud Pub/Sub subscription and cancel the old pipeline.

upvoted 1 times

Load full discussion...