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 139 DISCUSSION

Actual exam question from Google's Professional Data Engineer
Question #: 139
Topic #: 1
[All Professional Data Engineer Questions]

You are building a new data pipeline to share data between two different types of applications: jobs generators and job runners. Your solution must scale to accommodate increases in usage and must accommodate the addition of new applications without negatively affecting the performance of existing ones. What should you do?

A. Create an API using App Engine to receive and send messages to the applications

B. Use a Cloud Pub/Sub topic to publish jobs, and use subscriptions to execute them

C. Create a table on Cloud SQL, and insert and delete rows with the job information

D. Create a table on Cloud Spanner, and insert and delete rows with the job information

Show Suggested Answer

by ducc at *Sept. 3, 2022, 6:40 a.m.*

Comments

Type your comment...

Submit

jkhong Highly Voted 1 year, 11 months ago

Selected Answer: B

Job generators (they would be the publishers).
Job runners = subscribers

Question mentions that it must scale (of which push subscription has automatic scaling) and can accommodate additional new applications (this can be solved by having multiple subscriptions, with each relating to a unique application) to a central topic

upvoted 9 times

AzureDP900 1 year, 10 months ago

Yes it is
B. Use a Cloud Pub/Sub topic to publish jobs, and use subscriptions to execute them

upvoted 4 times

srivastavas08 Most Recent 8 months, 4 weeks ago

A. App Engine API: While scalable, it introduces a central point of failure and might not be as performant as Pub/Sub for high-volume data.
C. Cloud SQL: Not designed for real-time data sharing and continuous updates, leading to potential bottlenecks and performance issues.
D. Cloud Spanner: Offers strong consistency and global distribution, but its pricing model might be less suitable for high-volume, cost-sensitive workloads compared to Pub/Sub.

upvoted 1 times

juliorevk 1 year, 1 month ago

Selected Answer: B

B to decouple jobs being generated and run. Pub/Sub also scales seamlessly

upvoted 1 times

barnac1es 1 year, 1 month ago

Selected Answer: B

B. Use a Cloud Pub/Sub topic to publish jobs, and use subscriptions to execute them.

Scalability: Cloud Pub/Sub is a highly scalable messaging service that can handle a significant volume of messages and subscribers. It can easily accommodate increases in usage as your data pipeline scales.

Decoupling: Using Pub/Sub decouples the job generators from the job runners, which is a good architectural choice for flexibility and scalability. Job generators publish messages to a topic, and job runners subscribe to that topic to execute jobs when they are available.

Adding New Applications: With Cloud Pub/Sub, adding new applications (new publishers or subscribers) is straightforward. You can simply create new publishers to send jobs or new subscribers to consume jobs without impacting existing components.

upvoted 2 times

musumusu 1 year, 8 months ago

key words here: job generators (pushlish message on pub/sub) and job runners(subscribe message for further analysis). You may add as much as pushlishing job and subscribing job to same topic. So Answer B.
Using API , app engine is also good approach but its more complex than pub/sub.

upvoted 2 times

zellck 1 year, 11 months ago

Selected Answer: B

B is the answer.

upvoted 2 times

Atnafu 1 year, 11 months ago

A
Since it's application i will go with

upvoted 1 times

arpitagrawal 2 years, 2 months ago

Selected Answer: B

use pubsub

upvoted 3 times

YorelNation 2 years, 2 months ago

Selected Answer: B

I would tend to think B , one of the use of pub/sub is decoupling app

upvoted 2 times

ducc 2 years, 2 months ago

Selected Answer: B

I choose B

upvoted 2 times