

[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 37 DISCUSSION

Actual exam question from Google's Professional Data Engineer

Question #: 37

Topic #: 1

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

Flowlogistic Case Study -

Company Overview -

Flowlogistic is a leading logistics and supply chain provider. They help businesses throughout the world manage their resources and transport them to their final destination. The company has grown rapidly, expanding their offerings to include rail, truck, aircraft, and oceanic shipping.

Company Background -

The company started as a regional trucking company, and then expanded into other logistics market. Because they have not updated their infrastructure, managing and tracking orders and shipments has become a bottleneck. To improve operations, Flowlogistic developed proprietary technology for tracking shipments in real time at the parcel level. However, they are unable to deploy it because their technology stack, based on Apache Kafka, cannot support the processing volume. In addition, Flowlogistic wants to further analyze their orders and shipments to determine how best to deploy their resources.

Solution Concept -

Flowlogistic wants to implement two concepts using the cloud:

- ⇒ Use their proprietary technology in a real-time inventory-tracking system that indicates the location of their loads
- ⇒ Perform analytics on all their orders and shipment logs, which contain both structured and unstructured data, to determine how best to deploy resources, which markets to expand into. They also want to use predictive analytics to learn earlier when a shipment will be delayed.

Existing Technical Environment -

Flowlogistic architecture resides in a single data center:

⇒ Databases

8 physical servers in 2 clusters

- SQL Server `` user data, inventory, static data

3 physical servers

- Cassandra `` metadata, tracking messages

10 Kafka servers `` tracking message aggregation and batch insert

⇒ Application servers `` customer front end, middleware for order/customs

60 virtual machines across 20 physical servers

- Tomcat `` Java services

- Nginx `` static content

- Batch servers

⇒ Storage appliances

- iSCSI for virtual machine (VM) hosts

- Fibre Channel storage area network (FC SAN) `` SQL server storage

- Network-attached storage (NAS) image storage, logs, backups

⇒ 10 Apache Hadoop /Spark servers

- Core Data Lake

- Data analysis workloads

⇒ 20 miscellaneous servers

- Jenkins, monitoring, bastion hosts,

Business Requirements -

⇒ Build a reliable and reproducible environment with scaled panty of production.

⇒ Aggregate data in a centralized Data Lake for analysis

⇒ Use historical data to perform predictive analytics on future shipments

⇒ Accurately track every shipment worldwide using proprietary technology

⇒ Improve business agility and speed of innovation through rapid provisioning of new resources

⇒ Analyze and optimize architecture for performance in the cloud

⇒ Migrate fully to the cloud if all other requirements are met

Technical Requirements -

⇒ Handle both streaming and batch data

⇒ Migrate existing Hadoop workloads

⇒ Ensure architecture is scalable and elastic to meet the changing demands of the company.

⇒ Use managed services whenever possible

⇒ Encrypt data flight and at rest

⇒ Connect a VPN between the production data center and cloud environment

SEO Statement -

We have grown so quickly that our inability to upgrade our infrastructure is really hampering further growth and efficiency. We are efficient at moving shipments around the world, but we are inefficient at moving data around.

We need to organize our information so we can more easily understand where our customers are and what they are shipping.

CTO Statement -

IT has never been a priority for us, so as our data has grown, we have not invested enough in our technology. I have a good staff to manage IT, but they are so busy managing our infrastructure that I cannot get them to do the things that really matter, such as organizing our data, building the analytics, and figuring out how to implement the CFO' s tracking technology.

CFO Statement -

Part of our competitive advantage is that we penalize ourselves for late shipments and deliveries. Knowing where out shipments

are at all times has a direct correlation to our bottom line and profitability. Additionally, I don't want to commit capital to building out a server environment.

Flowlogistic is rolling out their real-time inventory tracking system. The tracking devices will all send package-tracking messages, which will now go to a single

Google Cloud Pub/Sub topic instead of the Apache Kafka cluster. A subscriber application will then process the messages for real-time reporting and store them in

Google BigQuery for historical analysis. You want to ensure the package data can be analyzed over time.

Which approach should you take?

- A. Attach the timestamp on each message in the Cloud Pub/Sub subscriber application as they are received.
- B. Attach the timestamp and Package ID on the outbound message from each publisher device as they are sent to Cloud Pub/Sub.
- C. Use the NOW () function in BigQuery to record the event's time.
- D. Use the automatically generated timestamp from Cloud Pub/Sub to order the data.

Show Suggested Answer

by [deleted] at March 20, 2020, 4:42 p.m.

Comments

Type your comment...


Submit

  **Manue** Highly Voted 3 years ago

"However, they are unable to deploy it because their technology stack, based on Apache Kafka, cannot support the processing volume."

Sure man, Kafka is not performing, let's use PubSub instead hahaha...

   upvoted 37 times

  **sfsdeniso** 1 year, 5 months ago

google send via pub sub web indexes
twice a day a whole internet is being sent via pub sub

   upvoted 2 times

  **ralf_cc** 2 years, 10 months ago

lol this is a vendor exam...

   upvoted 8 times

  **[Removed]** Highly Voted 4 years, 1 month ago

Answer: B

   upvoted 23 times

  **rtcpost** Most Recent 6 months, 2 weeks ago

Selected Answer: B

B. Attach the timestamp and Package ID on the outbound message from each publisher device as they are sent to Cloud Pub/Sub.

Here's why this approach is the most suitable:

By attaching a timestamp and Package ID at the point of origin (publisher device), you ensure that each message has a clear and consistent timestamp associated with it from the moment it is generated. This provides a reliable and accurate record of when each package-tracking message was created, which is crucial for analyzing the data over time.

This approach allows you to maintain the chronological order of events as they occurred at the source, which is important for real-time reporting and historical analysis.

Option A suggests attaching the timestamp in the Cloud Pub/Sub subscriber application. While this can work, it introduces a potential delay and the risk of timestamps not being accurate if there are issues with message processing.

Option C, using the NOW() function in BigQuery, records the time when the data is ingested into BigQuery, which may not reflect the actual time of the event.

👍 ↩ 🚩 upvoted 6 times

🗄 👤 **JJJJim** 1 year, 2 months ago

Selected Answer: B

Answer is B, attach the timestamp and ID is necessary to analyze data easily.

👍 ↩ 🚩 upvoted 1 times

🗄 👤 **nidmed** 2 years ago

Selected Answer: B

Answer: B

👍 ↩ 🚩 upvoted 4 times

🗄 👤 **Arkon88** 2 years, 2 months ago

Selected Answer: B

we need package ID + Timestamp so B

👍 ↩ 🚩 upvoted 1 times

🗄 👤 **davidqianwen** 2 years, 3 months ago

Selected Answer: B

Answer: B

👍 ↩ 🚩 upvoted 1 times

🗄 👤 **exnaniantwort** 2 years, 3 months ago

Selected Answer: B

agree with humza

👍 ↩ 🚩 upvoted 1 times

🗄 👤 **sraakesh95** 2 years, 3 months ago

Selected Answer: D

<https://cloud.google.com/pubsub/docs/reference/rest/v1/PubsubMessage>

👍 ↩ 🚩 upvoted 4 times

🗄 👤 **[Removed]** 2 years, 6 months ago

D is enough.. we have publish timestamp which is enough for this requirement

👍 ↩ 🚩 upvoted 1 times

🗄 👤 **Tanzu** 2 years, 3 months ago

there are 2 requirements

1- is about ordering due to historical data analysis

2- what it means to write a single topic and its impact.. why some sentence added here.

1st is primary, 2nd is secondary req. in this context.

So,

- in pub/sub, processTime is filled by server, not publisher. but that does not guarantee the ordering due to latency, pub/sub handling, sensors or any other reasons..

- you need to populate orderingKey field too, so that subscribers can get in ordered.

👍 ↩ 🚩 upvoted 3 times

🗄 👤 **Pinko1497** 2 years ago

Also, since this is a International company, adding timestamp on message receiving would help catch local time.

👍 ↩ 🚩 upvoted 1 times

🗄 👤 **Chelseajcole** 2 years, 6 months ago

It is about processing time and event time.. Answer is B.

👍 ↩ 🚩 upvoted 1 times

🗄 👤 **Tanzu** 2 years, 3 months ago

not just timing, but also package-id .. cause they are sending 1 topic in gcp instead of to many in kafka. that means there must be added some additional critical data too.

👍 ↩ 🚩 upvoted 1 times

🗄 👤 **anji007** 2 years, 6 months ago

Ans: B


A: Adding timestamp as they received is not a better option, messages may not arrive in order at the receiver/ subscriber, could be due to connectivity or network.

B: Timestamp should be added here.

C: Doesn't make sense at all.

D: Ordering should be based on the order how messages are generated at the publisher but not as per order they reach the pub/sub.

   upvoted 4 times

  **humza** 2 years, 9 months ago

Answer: B

A. There is no indication that the application can do this. Moreover, due to networking problems, it is possible that Pub/Sub doesn't receive messages in order. This will analysis difficult.

B. This makes sure that you have access to publishing timestamp which provides you with the correct ordering of messages.

C. If timestamps are already messed up, BigQuery will get wrong results anyways.

D. The timestamp we are interested in is when the data was produced by the publisher, not when it was received by Pub/Sub.

   upvoted 7 times


  **sumanshu** 2 years, 10 months ago

Vote for B

   upvoted 2 times

  **funtoosh** 3 years, 2 months ago


Better if the publisher attached the package ID and Timestamp as packages can come in an Asynchronous fashion.

   upvoted 3 times

  **naga** 3 years, 2 months ago

Correct B

   upvoted 3 times

  **Radhika7983** 3 years, 5 months ago

The answer is B.

JSON representation

```
{
  "data": string,
  "attributes": {
    string: string,
    ...
  },
  "messageId": string,
  "publishTime": string,
  "orderingKey": string
}
```

In the attribute, we can have package id and timestamp.

   upvoted 6 times

[Load full discussion...](#)



Platform

> Home

> Examtopics PRO

> All Exams

> Training Courses

