**If we design aws kinesis streams data to send to Kinesis Firehose and to S3, what is the potential issue, if we need to restart the process or data delete**

you design your AWS Kinesis Streams data to send to Kinesis Firehose and to S3, there are a few potential issues that you should be aware of if you need to restart the process or delete data.

Restarting the process

If you need to restart the process of sending data from Kinesis Streams to Kinesis Firehose and to S3, you will need to make sure that you have a way to identify and recover any data that was not processed during the previous run. This can be done by using a checkpointing mechanism or by using a durable message queue.

Deleting data

If you need to delete data from Kinesis Streams, you will need to make sure that you delete it from both Kinesis Streams and Kinesis Firehose. This is because Kinesis Firehose can continue to read data from Kinesis Streams even after you have deleted it from Kinesis Streams.

To avoid these issues, you should carefully design your data streaming architecture and make sure that you have a plan for restarting the process or deleting data.

Here are some additional tips to help you avoid these issues:

* Use a checkpointing mechanism to ensure that data is not lost if the process needs to be restarted.
* Use a durable message queue to store data that is waiting to be processed by Kinesis Firehose.
* Delete data from both Kinesis Streams and Kinesis Firehose to ensure that it is not retained in either service.
* Monitor your data streaming application to identify and address any potential issues.

By following these tips, you can help to ensure that your data streaming application is reliable and that you can recover from any unexpected events.

## **give me different possible best design for data streaming architecture using aws**

Sure, here are some different possible best designs for data streaming architecture using AWS:

Design 1

This design uses Amazon Kinesis Data Streams to ingest data from a variety of sources, such as IoT devices, web applications, and cloud-based services. The data is then processed by Amazon Kinesis Data Analytics, which can be used to perform real-time analytics on the data. The results of the analytics can be stored in Amazon S3 or Amazon Redshift for further analysis or visualization.

Design 2

This design uses Amazon Kinesis Data Firehose to ingest data from a variety of sources and then load it into Amazon S3 or Amazon Redshift. The data can then be processed by Amazon Redshift Spectrum, which allows you to query the data without having to load it into an Amazon Redshift cluster.

Design 3

This design uses Amazon Managed Streaming for Apache Kafka (Amazon MSK) to ingest data from a variety of sources and then store it in Amazon S3. The data can then be processed by Amazon EMR, which can be used to run Apache Spark or Apache Hadoop jobs on the data.

Design 4

This design uses Amazon Kinesis Data Streams to ingest data from a variety of sources and then send it to Amazon Lambda. Amazon Lambda can be used to process the data in real time or to store it for later processing.

These are just a few possible best designs for data streaming architecture using AWS. The best design for your specific needs will depend on the volume and type of data you are collecting, the processing requirements for the data, and your budget.

Here are some additional factors to consider when designing your data streaming architecture:

* The volume of data you are collecting.
* The latency requirements for your application.
* The processing requirements for your application.
* Your budget.

By considering these factors, you can design a data streaming architecture that meets your specific needs.

## **compare the cost among those suggested options for 5 GB data, dont worry about latency**

| Option | Cost |
| --- | --- |
| Amazon Kinesis Data Streams | $0.0125 per GB |
| Amazon Kinesis Data Firehose | $0.008 per GB |
| Amazon Managed Streaming for Apache Kafka (Amazon MSK) | $0.0075 per GB |
| Amazon EMR | $0.005 per GB |
| Amazon Lambda | $0.001 per 100,000 requests |

As you can see, Amazon Kinesis Data Streams is the most expensive option, followed by Amazon MSK. Amazon Kinesis Data Firehose and Amazon EMR are both less expensive than Amazon Kinesis Data Streams, but more expensive than Amazon MSK. Amazon Lambda is the least expensive option, but it is also the least scalable.

The best option for you will depend on your specific needs and budget. If you need to process a large volume of data in real time, Amazon Kinesis Data Streams may be the best option. If you need to process a smaller volume of data or if you can process the data in batches, Amazon Kinesis Data Firehose or Amazon EMR may be a better option. If you need to process a very small volume of data or if you only need to process the data occasionally, Amazon Lambda may be the best option.

Here are some additional factors to consider when choosing a data streaming architecture:

* The volume of data you are collecting.
* The processing requirements for your application.
* Your budget.

By considering these factors, you can choose a data streaming architecture that meets your specific needs.

## **for any reason kinesis stream is broken, what would happen to firehose**

If the Kinesis stream is broken, Kinesis Firehose will stop receiving data from the stream. This will prevent Kinesis Firehose from loading the data into Amazon S3 or Amazon Redshift.

If the Kinesis stream is broken for a short period of time, Kinesis Firehose will automatically retry loading the data when the stream becomes available again. However, if the Kinesis stream is broken for an extended period of time, you may need to manually restart Kinesis Firehose to load the data.

To avoid this issue, you can configure Kinesis Firehose to use a durable message queue, such as Amazon SQS. This will allow Kinesis Firehose to continue loading data from the stream even if the stream is broken.

Here are some additional tips to help you avoid this issue:

* Use a durable message queue to store data that is waiting to be processed by Kinesis Firehose.
* Monitor your Kinesis stream to identify and address any potential issues.

By following these tips, you can help to ensure that your data streaming application is reliable and that you can recover from any unexpected events.