**Azure Event Hubs and stream analytics Manual**

Batch processing

There is a business need for reporting, however, the reports can refresh every morning or once a day, or twice a day. For this scenario, batch processing is preferred as the data pipelines run once or twice a day.

Real-time processing /stream processing

There is a business need for the reports to generate regularly, for example every 5 minutes. You can take real-time data and then run the pipelines to generate a report. This is where the real-time processing/stream analytics is important.

Azure Event Hubs

This is a data streaming resource, and it can receive and process millions of events per second. You can stream your log data. So lets say that you have an application and it is hosted on a virtual machine, you can send your telemetric or log data to event hubs.

A diagram of a process flow

Description automatically generated

Go to event hubs in resources and create a new event. The throughput units are just the capacity at which you can stream events. The lower the number, the fewer the data events that can be ingested for streaming. So the number 1 means 1MB per seconds or 1000 events per second. If you exceed this, you will get an exception error. The click create.

A screenshot of a computer

Description automatically generated

So next go to the overview page. This is a way to monitor what is happened in your namespace.

A computer screen shot of a computer screen

Description automatically generated

Next if we go to the event hub tab on the left, we can create the specific event hub that will be responsible for ingesting the data that we will be streaming from the python application.

A screenshot of a computer

Description automatically generated

After clicking add event hub, you add the name of the event hub. You can partition the data and store it. The data is partitioned and is processed in parallel for optimised compute. In the scenario above, we will keep 2 partitions for parallel processing, but the more you have, the more oprtimisation.

A screenshot of a computer

Description automatically generated

Capture allows you to capture the streaming data in adls. When event data is streamed, you cannot see it when event hubs is capturing it. You can only see it in stream analytics. So, if you store it in adls first, then you can add that step. However, this capture feature is not allowed with a basic subscription.

A screenshot of a computer

Description automatically generated

So after this is created, it is important to note that we can have multiple event hubs in one event hub namespace. So in the same way above, I can create more event hubs to capture streaming data. Event hubs has an endpoint which can be used in the python application so that the application and event hubs can make a connection. Under shared access policy in event hubs, you will see a shared access policy. If you don’t, then create one.

A screenshot of a computer

Description automatically generated

A screenshot of a computer

Description automatically generated

Lets now create stream analytics resource to recognise the data that is being streamed into the event hub, and to recognise the data and analyse it. Click on create:

A screenshot of a computer

Description automatically generated

A screenshot of a computer

Description automatically generated

The more the streaming units, the quicker the streaming data will be analysed. Go to the created resources, then inputs, and then Event Hub.

A screenshot of a computer

Description automatically generated

Consumer groups are multiple groups that could be using the data that is being ingested. You can specify the specific consumer groups that this user needs to use. Select the information as below, and then click save.

A screenshot of a computer

Description automatically generated

A screenshot of a computer

Description automatically generated

Once this is done, go to the resource and look at the query tab:

A screenshot of a computer

Description automatically generated

You can query incoming data in here by writing SQL and then perhaps put it in the output as seen above.

Python programme

Create a python realtime programme which can read stock information.