# DP 200 - Implementing a Data Platform Solution

# Lab 6 - Performing Real-Time Analytics with Stream Analytics

**Estimated Time**: 60 minutes

**Pre-requisites**: It is assumed that the case study for this lab has already been read. It is assumed that the content and lab for module 1: Azure for the Data Engineer has also been completed

**Lab files**: The files for this lab are located in the Allfiles\Labfiles\Starter\DP-200.6 folder.

## Lab overview

The students will be able to describe what data streams are and how event processing works and choose an appropriate data stream ingestion technology for the AdventureWorks case study. They will provision the chosen ingestion technology and integrate this with Stream Analytics to create a solution that works with streaming data.

## Lab objectives

After completing this lab, you will be able to:

1. Explain data streams and event processing
2. Data Ingestion with Event Hubs
3. Processing Data with Stream Analytics Jobs

## Scenario

As part of the digital transformation project, you have been tasked by the CIO to help the marketing departments become more productive with aspects of their work. Over the last few years the marketing department has been using Twitter to amplify marketing message around the bicycle products that are sold.

Whilst the department can provide reach numbers post campaign, they are unable to understand who is interacting with their campaigns in real-time, as the volumes are difficult to track manually. As a result, they would like to implement a system that can track in real-time who is responding to their campaign.

At the end of this lad, you will have:

1. Explain data streams and event processing
2. Data Ingestion with Event Hubs
3. Processing Data with Stream Analytics Jobs

**IMPORTANT**: As you go through this lab, make a note of any issue(s) that you have encountered in any provisioning or configuration tasks and log it in the table in the document located at \Labfiles\DP-200-Issues-Doc.docx. Document the Lab number, note the technology, Describe the issue, and what was the resolution. Save this document as you will refer back to it in a later module.

## Exercise 1: Explain data streams and event processing

Estimated Time: 15 minutes

Group exercise

The main task for this exercise are as follows:

1. From the case study and the scenario, identify the data stream ingestion technology for AdventureWorks, and the high-level tasks that you will conduct as a data engineer to complete the social media analysis requirements.
2. The instructor will discuss the findings with the group.

### Task 1: Identify the data requirements and structures of AdventureWorks.

1. From the lab virtual machine, start **Microsoft Word**, and open up the file **DP-200-Lab06-Ex01.docx** from the **Allfiles\Labfiles\Starter\DP-200.6** folder.
2. As a group, spend **10 minutes** discussing and listing the data requirements and data structure that your group has identified within the case study document.

### Task 2: Discuss the findings with the Instructor

1. The instructor will stop the group to discuss the findings.

**Result**: After you completed this exercise, you have created a Microsoft Word document that shows a table of data streaming ingestion and the high-level tasks that you will conduct as a data engineer to complete the social media analysis requirements .

## Exercise 2: Data Ingestion with Event Hubs.

Estimated Time: 15 minutes

Individual exercise

The main tasks for this exercise are as follows:

1. Create and configure an Event Hub Namespace.
2. Create and configure an Event Hub.
3. Configure Event Hub security.

### Task 1: Create and configure an Event Hub Namespace.

1. In the Azure portal, select **+ Create a resource**, type **Event Hubs**, and then select **Event Hubs**  from the resulting search. Then select **Create**.
2. Create an Event Hub Namespace with the following options, then click **Create**:
   * **Name**: xx-socialtwitter-eh, where xx are your initials
   * **Pricing Tier**: Standard
   * **Subscription**: Your subscription
   * **Resource group**: awrgstudxx
   * **Location**: select the location closest to you
   * Leave other options to their default settings

**Note**: The creation of the Event Hub Namespace takes approximately 1 minute.

### Task 2: Create and configure an Event Hub

1. In the Azure portal, in the blade, click **Resource groups**, and then click **awrgstudxx**, and then click on **awdlsstudxx**, where **xx** are your initials
2. Click on **xx-socialtwitter-eh**, where **xx** are your initials.
3. In the **xx-socialtwitter-eh** screen, click on **+ Event Hubs**.
4. Provide the name **socialtwitter-eh**, and then select **Create**.

**Note**: You will receive a message stating that the Event Hub is created after about 10 seconds

### Task 3: Configure Event Hub security

1. In the Azure portal, in the **xx-socialtwitter-eh**, where **xx** are your initials. Scroll to the bottom of the window, and click on **socialtwitter-eh** event hub.
2. To grant access to the event hub, click **Shared access policies**.
3. Under the **socialtwitter-eh - Shared access policies** screen, create a policy with **Manage** permissions by selecting **+ Add**. Give the policy the name of **socialtwitter-eh-sap** , check **Manage**, and then click **Create**.
4. Click on your new policy **socialtwitter-eh-sap** after it has been created, and then select the copy button for the **CONNECTION STRING - PRIMARY KEY** and paste the CONNECTION STRING - PRIMARY KEY into Notepad, this is needed later in the exercise.
5. Close down the Event hub screens in the portal

**Result**: After you completed this exercise, you have created an Azure Event Hub within an Event Hub Namespace and set the security for the Event Hub that can be used to provide access to the service.

## Exercise 3: Processing Data with Stream Analytics Jobs

Estimated Time: 30 minutes

Individual exercise

The main tasks for this exercise are as follows:

1. Create a Twitter developer account
2. Configure the Twitter access keys
3. Configure and start the Twitter client application
4. Provision a Stream Analytics job.
5. Specify the a Stream Analytics job input.
6. Defining a Stream Analytics query.
7. Specify the a Stream Analytics job output.
8. Modify a Stream Analytics query.
9. Start the Stream Analytics job.
10. Validate streaming data is collected

### Task 1: Create a Twitter developer account.

1. Open a new tab in your browser and go to <https://developer.twitter.com/en/apps/> and login with your twitter account to set up a Twitter application.
2. In **Apps** page, to the right, click on the button **Create an app**.
3. In the "Please apply for a twitter account", click on **Apply**.
4. In the "User profile" page, confirm your twitter account is the correct account to associate with a developer account, and click on **Continue**.
5. In the “Account details” page, under "Who are you requesting access for?" click **I am requesting access for my own personal use**. Under "Account name", type **learn how to code as twitter\_name**. Finally, select **Primary country of operation** as the country where you reside. Click on **Continue**.

**Note**: please remember to observe all applicable laws and ethics for your country of operation

1. In the "Tell us about your project" page, choose an option for "What use case(s) are you interested in?". For the purpose of this course select **Student**. In the "Describe in your own words what you are building", spend 300 words to explain why this is being used. below is an example:
   1. I’m using Twitter’s APIs to learn how to embed twitter statements within a learning application. this account is being used to learn how to do that
   2. I plan to analyze Tweets to understand how to count and read streaming tweets in real time as part of a learning exercise
   3. I do not intend on interacting with other peoples tweets in the form of retweeting or liking content. I will be reading and counting tweets and looking at sentiment.
   4. Both individual and aggregate tweets will be analysed in the application
2. Finally, select the option for the question "Will your product, service, or analysis make Twitter content or derived information available to a government entity?" and then click on **Continue**.
3. Read and agree to the Terms of Service by clicking **Accept** if you are happy to do so. Then select the checkboxes as you see fit. Click on **Submit application**
4. Check your inbox for the verification email, click on the **confirm your email** within the email to complete the process. You will be directed back to twitter to a Welcome screen.

### Task 2: Configure the Twitter access keys

1. If you are not at the Twitter welcome screen, Open a new tab in your browser and go to <https://developer.twitter.com> and login with your twitter account.
2. In the "Welcome" or "Apps" page, click on the button **Create an app**.
3. In the “Create an App" page, provide the following information and click on **Create an App**:
   * **name**: xx-social-app, where xx are your initials
   * **Application description**: add a description such as "used to collect and aggregate tweets"
   * **website URL** a personal address that you can use for the application.
   * **Untick Sign in with Twitter**
   * **Tell us how this app will be used**: write text that outlines the app is a demo app to learn how to use streaming data
4. Review the Developer terms and click on **Create**. A new page will appear with the title of you application name.
5. Click on the page tab named **Keys and tokens**
6. Under Access token & access token secret, Click on the **Create** button.

**Note**: An authorized access token and secret will be generated for your account and the current application.

1. Copy the following keys to Notepad file you had previously opened:
   * Consumer Key (API Key)
   * Consumer Secret (API Secret)
   * Access Token
   * Access Token Secret

### Task 3: Configure and start the Twitter client application

1. Double click the **TwitterWPFClient.exe** application in the **Allfiles\Labfiles\Starter\DP-200.6\Twitter client\TwitterWPFClient\TwitterWPFClient** location. You can also download the [TwitterWPFClient here](https://github.com/Azure/azure-stream-analytics/tree/master/Samples/TwitterClient).
2. Enter your data from Notepad into the TwitterWPFClient.exe application.
   * **Twitter Consumer Key (API Key)**: Your twitter consumer key
   * **Twitter Consumer Secret (API Secret)**: Your twitter consumer secret
   * **Twitter Access Token**: Your twitter access token
   * **Twitter Access Secret**: Your twitter access secret
   * **Azure EventHub Name**: socialtwitter-eh.
   * **Azure Event Hub Connection string**: Endpoint=sb://xx-socialtwitter-eh.servicebus.windows.net/;SharedAccessKeyName=xx-social-eh-sap;SharedAccessKey=<Paste in the CONNECTION STRING - PRIMARY KEY from Notepad>

**Important**: IT IS IMPORTANT TO REMOVE ;EntityPath=socialtwitter-eh AT THE END OF THE STRING.

* + **Search Groups** define which keywords you want to determine sentiment for. An example could be “#Brexit”, or brexit, or you can add multiple keywords separated by a comma.

1. Click the **Play** button in the **TwitterWPFClient.exe** application, wait until you see tweets appearing in the console and leave the application running.

**Note**: The **TwitterWPFClient.exe** application capture the date, search term sentiment score, twitter name and tweet details for the search term defined.

1. Keep all applications running

### Task 4: Provision a Stream Analytics job.

1. Go back to the Azure portal, select **+ Create a resource**, type **STREAM**, and then click the **Stream Analytics Job**, and then click **CREATE**.
2. In the **New Stream Analytics job** screen, fill out the following details and then click on **Create**:
   * **Job name**: socialtwitter-asa-job.
   * **Subscription**: select your subscription
   * **Resource group**: awrgstudxx
   * **Location**: choose a location nearest to you.
   * Leave other options to their default settings

**Note**: You will receive a message stating that the Stream Analytics job is created after about 10 seconds

### Task 5: Specify the a Stream Analytics job input.

1. In the Azure portal, in the blade, click **Resource groups**, and then click **awrgstudxx**, and then click on **awdlsstudxx**, where **xx** are your initials
2. Click on **socialtwitter-asa-job**.
3. In your **socialtwitter-asa-job** Stream Analytics job window, click **Inputs**.
4. In the **Inputs** screen, click **+ Add stream input**, and then click **Event Hubs**.
5. In the Event Hub screen, type in the following values and click the **Save** button.
   * **Input alias**: Enter a name for this job input as TwitterStream.
   * **Select Event Hub from your subscriptions**: checked
   * **Subscription**: Your subscription name
   * **Event Hub Namespace**: xx-socialtwitter-eh
   * **Event Hub Name**: Use existing named socialtwitter-eh
   * **Event Hub Policy Name**: socialtwitter-eh-sap
   * Leave remaining options to their default
6. Once completed, the TwitterStream Input job will appear under the input window. Close the input widow to return to the Streaming Analytics Job Page

### Task 6: Defining a Stream Analytics query.

1. In your **socialtwitter-asa-job** window, in the **Query** screen in the middle of the window, click on \*\* Edit query\*\*
2. Replace the following query in the code editor:

SELECT

\*

INTO

[YourOutputAlias]

FROM

[YourInputAlias]

1. Replace with

SELECT

[Topic]

,[SentimentScore]

,[created\_at]

,[Author]

,[text]

FROM [TwitterStream]

1. Select Save.
2. Close the Query window to return to the Stream Analytics job page.

### Task 7: Specify the a Stream Analytics job output.

1. In your **socialtwitter-asa-job** Stream Analytics job window, click **Outputs**.
2. In the **Outputs** screen, click **+ Add**, and then click **Blob Storage**.
3. In the **Blob storage** window, type or select the following values in the pane:

* **Output alias**: Output
* **Select Event Hub from your subscriptions**: checked
* **Subscription**: Your subscription name
* **Storage account**: awsastudxx, where xx is your initials
* **Container**: Use existing and select tweets

1. Leave the rest of the entries as default values. Finally, click **Save**\*.
2. Close the output screen to return to the Stream Analytics job page

### Task 8: Defining a Stream Analytics query.

1. In your **socialtwitter-asa-job** window, in the **Query** screen in the middle of the window, click on **Edit query**
2. Replace the following query in the code editor:

SELECT

[Topic]

,[SentimentScore]

,[created\_at]

,[Author]

,[text]

FROM [TwitterStream]

1. Replace with

SELECT

[Topic]

,[SentimentScore]

,[created\_at]

,[Author]

,[text]

INTO [Outputs]

FROM [TwitterStream]

1. Select Save.
2. Close the Query window to return to the Stream Analytics job page.

### Task 9: Start the Stream Analytics job

1. In your **socialtwitter-asa-job** window, in the **Query** screen in the middle of the window, click on **Start**
2. In the **Start Job** dialog box that opens, click **Now**, and then click **Start**.

**Note**: In your **socialtwitter-asa-job** window, a message appears after a minute that the job has started, and the started field changes to the time started

**Note**: Leave this running for 2 minutes so that data can be captured.

### Task 10: Validate streaming data is collected

1. In the Azure portal, in the blade, click **Resource groups**, and then click **awrgstudxx**, and then click on **awsastudxx**, where **xx** are your initials.
2. In the Azure portal, click **Blobs** screen.
3. In the Azure portal, in the **awsastudxx - Blobs** screen, click on the **tweets** item in the list.
4. Confirm that a JSON file appears, and note the size column.
5. Refresh Microsoft Edge, and when the screen has refreshed not the size of the file

**Note**: You could download the file to query the JSON data, you could also output the data to Power BI.

**Result**: After you completed this exercise, you have configured Azure Stream Analytics to collect streaming data into an JSON file store in Azure Blob. You have done this with streaming twitter data.

## Close down

1. In the Azure portal, in the blade, click **Resource groups**, and then click **awrgstudxx**, and then click on **socialtwitter-asa-job**.
2. In the **socialtwitter-asa-job** screen, click on **Stop**. In the **Stop Streaming job** dialog box, click on **Yes**.
3. Close down the **TwitterWPFClient.exe** application.