Lab 04

Delivering a Real-Time Microsoft Power BI Dashboard with Azure Stream Analytics

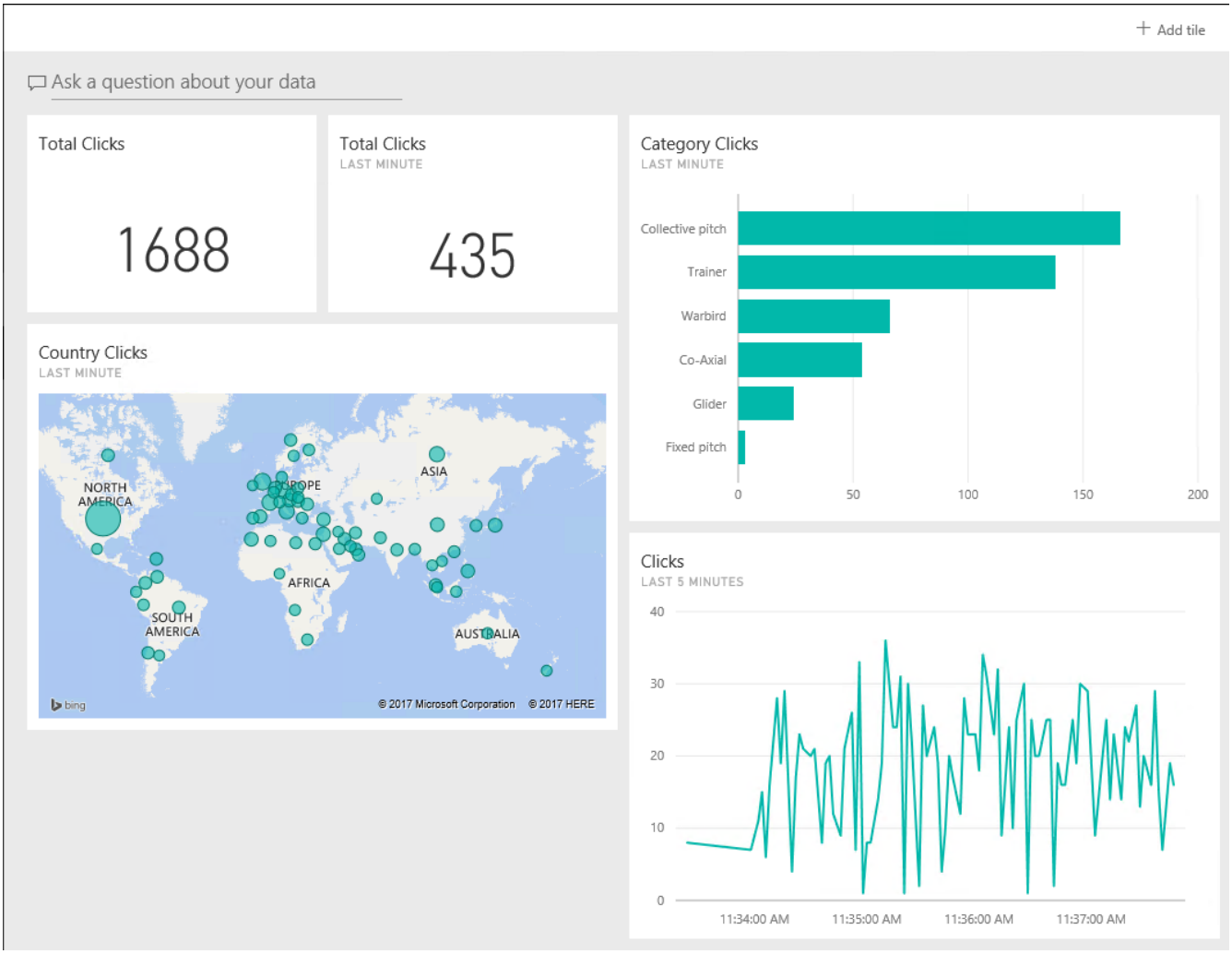
Overview

The estimated time to complete lab is 60 minutes

In this lab, you will produce a real-time Power BI dashboard by using an Azure Event Hub and Azure Stream Analytics. You will commence by signing in to Power BI, and then signing in to the Azure Portal to create an Event Hub, and then a Stream Analytics job.

With the assistance of an event generator, product clickstream events will be sent, queried and output to Power BI to be expressed in the following real-time dashboard.

In order to complete this lab, you will require an Azure subscription. There is also a requirement to sign in to Power BI.



You will learn how to:

- Create an Azure Event Hub
- Create an Azure Stream Analytics (ASA) job to output the temporal aggregation of events
- Configure a real-time Power BI dashboard based on an ASA job output

Setting Up the Lab Portal

In this exercise, you will sign in to the Virtual Machine.

Signing in to the Virtual Machine

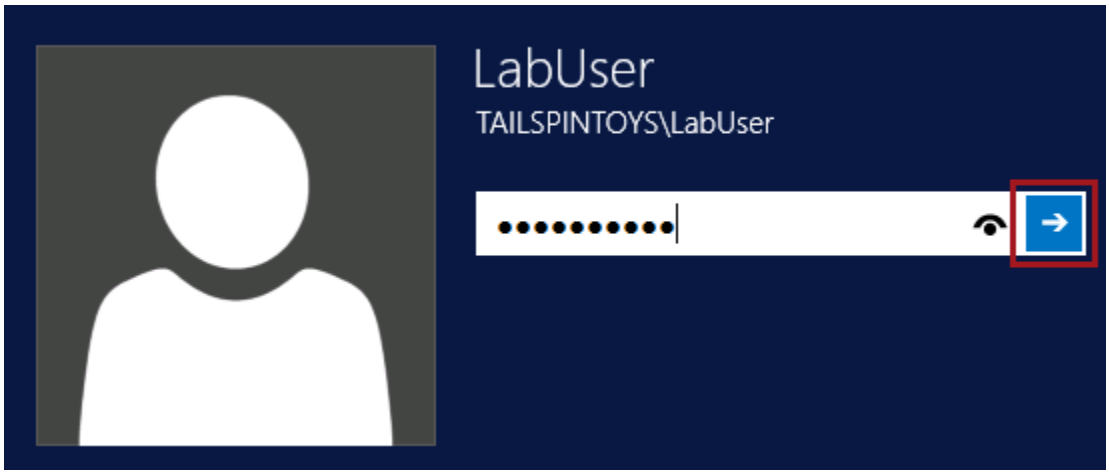
In this task, you will sign in to the virtual machine.

1. In the right pane, select the **Machines** tab.

2. Click **Ctrl+Alt+Delete**.
3. To enter the password, to the left of the password value, click the **T** command.

*T means **Type Text**.*

4. In the sign in screen, to enter the credentials, click the right-pointing arrow.



5. In the right pane, switch to the **Content** tab.

Signing In to the Power BI Service

In this exercise, you will sign in to the Power BI service.

Signing In to the Power BI Service

In this task, you will sign in to the Power BI service.

A personal trial account has been created for you, and it is recommended that you use this account to complete all Power BI labs within this virtual machine session. If you plan to complete several labs within the same virtual machine session (i.e. classroom training), you should use the same account for all labs.

1. To open Internet Explorer, on the taskbar, click the **Internet Explorer** program shortcut.

2. In Internet Explorer, navigate to http://powerbi.com.

   *Tip: You can also use the **Power BI Site** Internet Explorer favorite.*

3. Click **Sign In** (located at the top-right corner).

4. Enter the account details provided in the right-pane, **Content** tab.

   This is a personal trial account made available to you to complete the labs. It will provide at least 30 days of access to the Power BI service.

   Remember to use the T commands to type the text into the appropriate boxes.

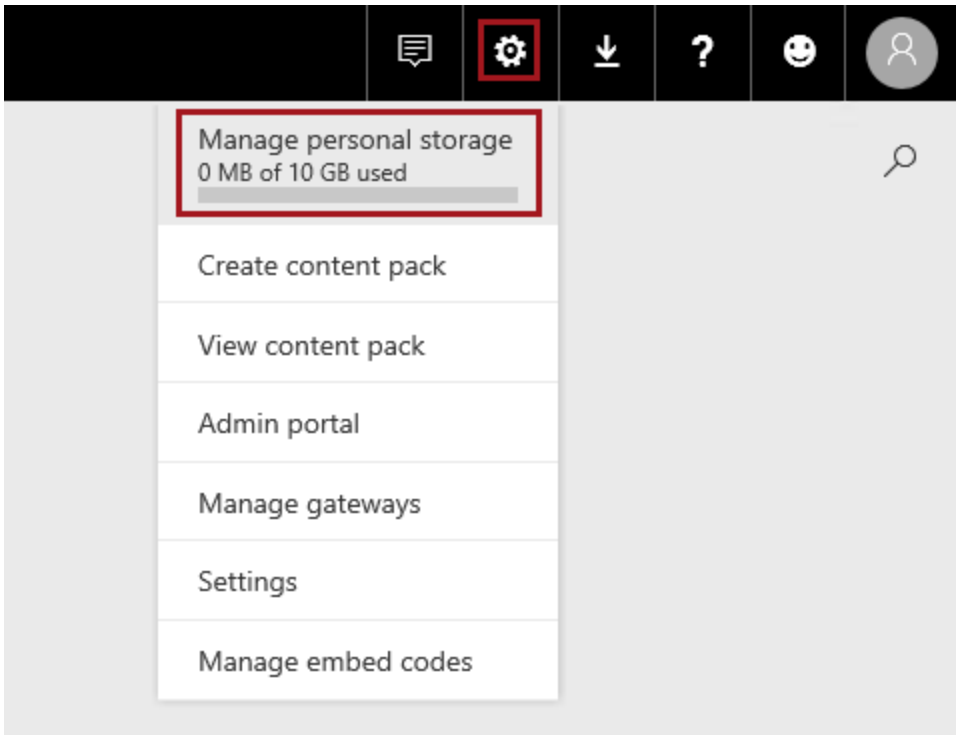
5. When prompted to update the password, reenter the provided password, and then enter and confirm a new password.

   It is important that you remember the new password, as you will be required to authenticate with the Power BI service later in the lab.

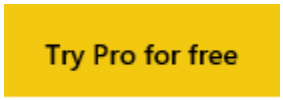
6. Complete the sign in process.

7. If prompted to stay signed in, click **Yes**.

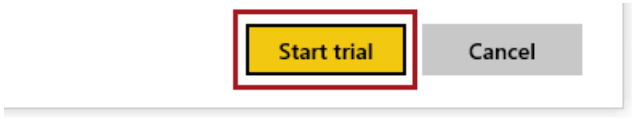
8. To upgrade the trial account to the **Power BI Pro** trial license, at the top right corner, click the **Settings** command (cog), and then select **Manage Personal Storage**.



9. At the top-right corner, click **Try Pro for Free**.



10. In the dialog window, if you agree to the terms, click **Start Trial**.



11. When the trial extension has been confirmed, click **Close**.
12. Leave the Internet Explorer window open.

Provisioning an Event Hub

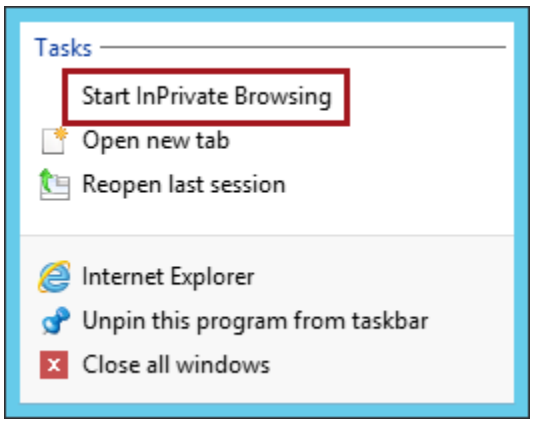
In this exercise, you will provision an Azure Event Hub.

*At the end of this lab, you must complete the **Finishing Up** exercise to delete the services you create in this exercise.*

Creating an Azure Resources Group

In this task, you will create an Azure Resource Group.

1. To open Internet Explorer, on the taskbar, right-click the **Internet Explorer** program shortcut, and then select **Start InPrivate Browsing**.



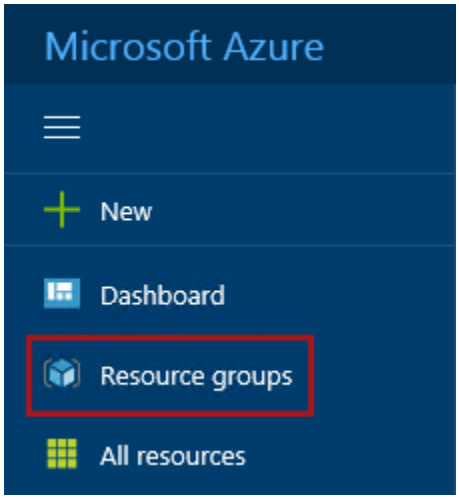
2. Navigate to https://portal.azure.com/.

   *Tip: You can also use the **Sign In to Azure Portal** Internet Explorer favorite.*

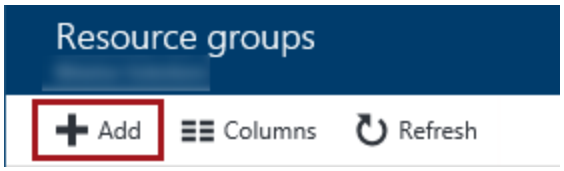
3. Sign in to the Azure Portal by using your own subscription.

   You may be able to create an Azure free account to help you complete this lab: https://azure.microsoft.com/free/.

4. In the menu (located at the left), select **Resource Groups**.



5. In the **Resource Groups** blade, click **Add**.

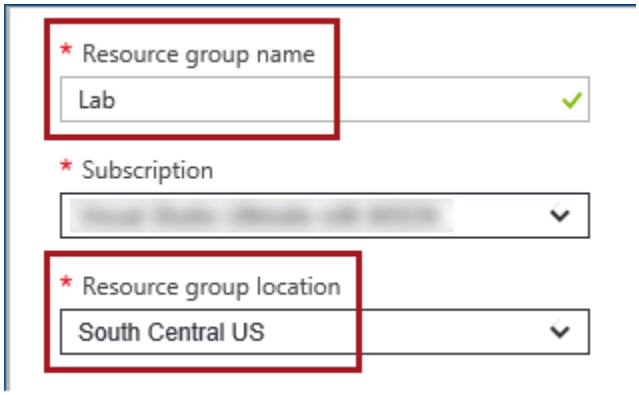


6. In the **Resource Group** blade, in the **Resource Group Name** box, enter **Lab**.

   Usually Azure resource names must be globally unique. The resource group name is an exception, and must only be unique within your Azure subscription.

7. In the **Resource Group Location** box, ensure that **South Central US** is selected.

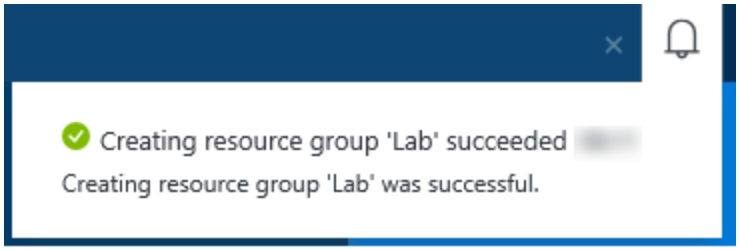
8. Verify that the configuration looks like the following.



9. Click **Create**.

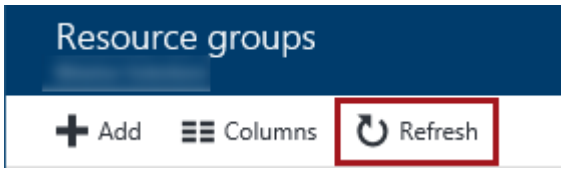
10. At the top-right corner of the portal, notice that there is a single notification.

11. Click the **Notifications** icon, and verify that the resource group was successfully deployed.



*Tip: Leave the **Notifications** pane open during this lab when creating Azure resources.*

12. In the **Resource Groups** blade, to reload the list, click **Refresh**.



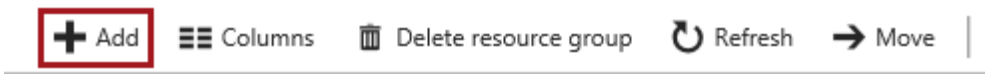
13. Verify that the **Lab** resource group is listed.

14. Select the **Lab** resource group.

Creating an Azure Event Hub Namespace

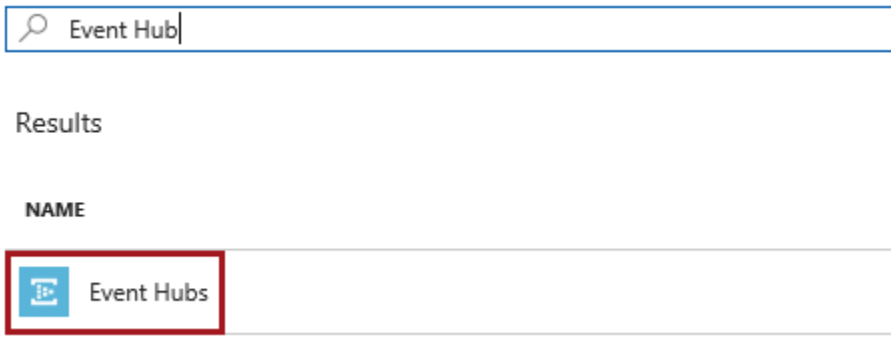
In this task, you will create an Azure Event Hub Namespace.

1. In the **Lab** resource group blade, to add a new resource, click **Add**.



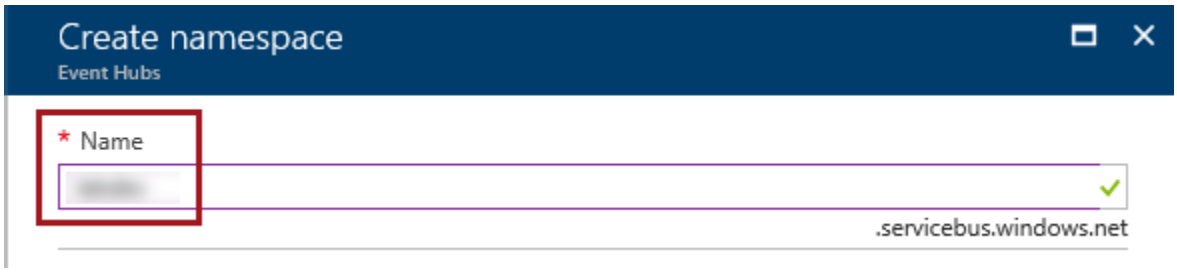
2. In the **Everything** blade, in the search box, enter **Event Hub**, and then press **Enter**.

3. Select **Event Hubs**.

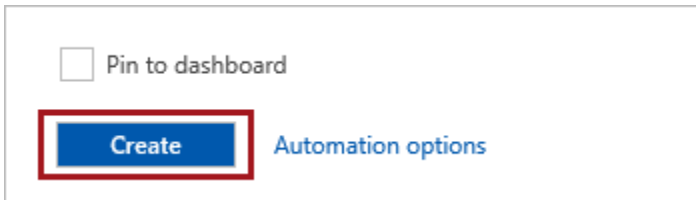


4. In the **Event Hubs** blade, click **Create**.

5. In the **Create Namespace** blade, in the **Name** box, enter a globally unique name (perhaps based on your name, e.g. **labuser0001**).



6. Click **Create**.

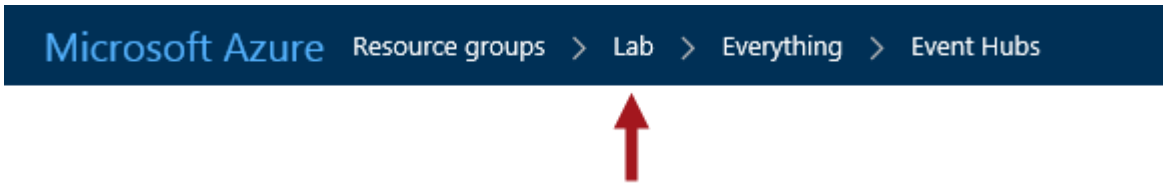


7. Wait until the Event Hub Namespace deployment has completed.

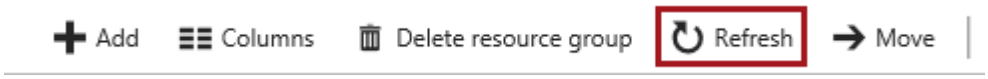
Adding an Azure Event Hub

In this task, you will add an Azure Event Hub.

1. On the bread crumb trail, click the **Lab** resource group.

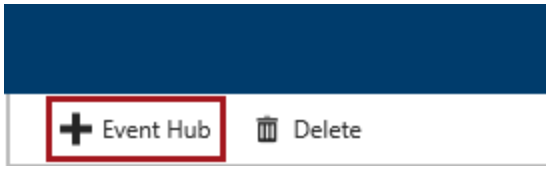


2. To refresh the **Lab** resource group blade, click **Refresh**.



3. Select the Event Hub Namespace.

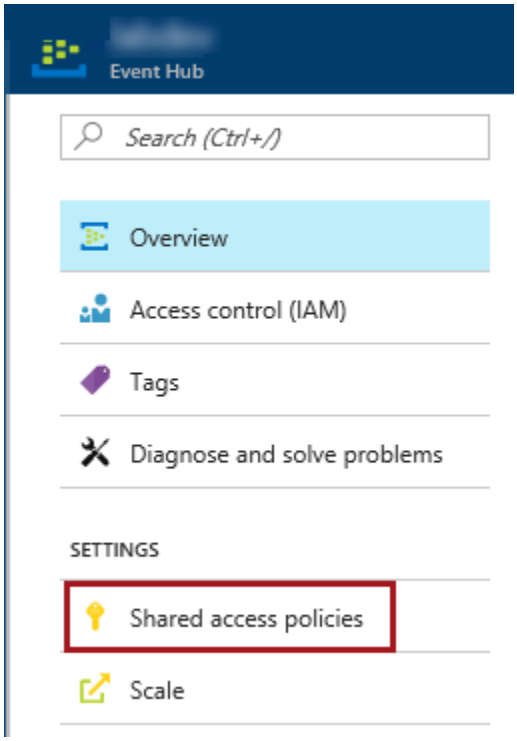
4. In the **Event Hub** blade, to add an event hub, click **Event Hub**.



5. In the **Create Event Hub** blade, in the **Name** box, enter **ProductClickstream**.
6. Click **Create**.

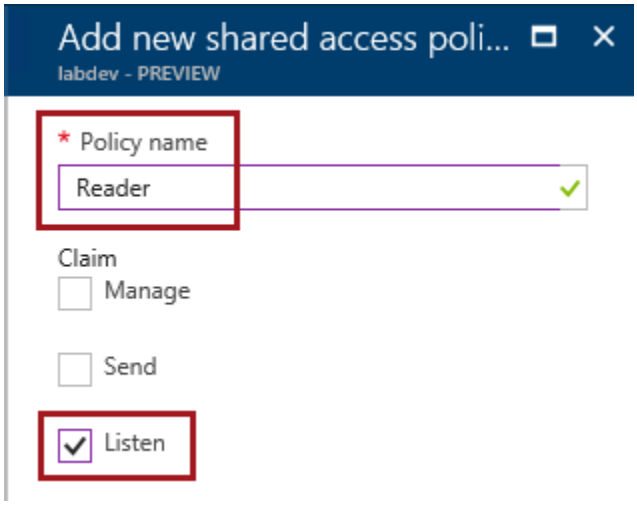
7. Wait until the Event Hub creation has completed.

8. In the left pane, select **Shared Access Policies**.



9. Click **Add**.

10. In the **Add New Shared Access Policy** blade, in the **Policy Name** box, enter **Reader**.
11. Check the **Listen** claim.

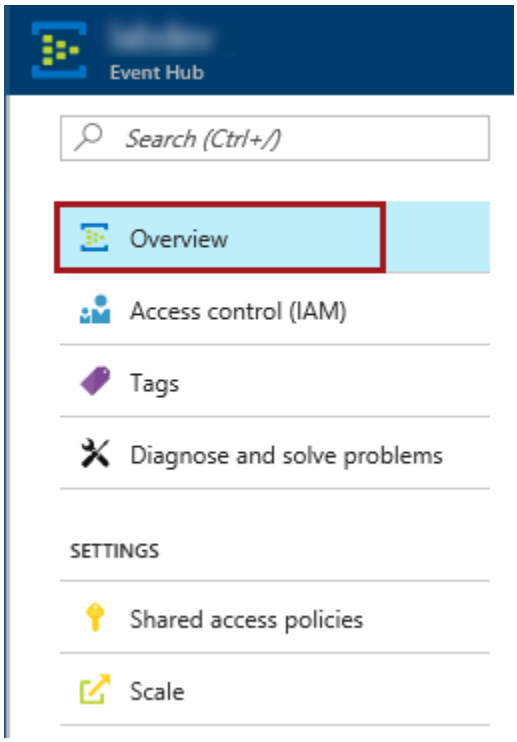


12. Click **Create**.

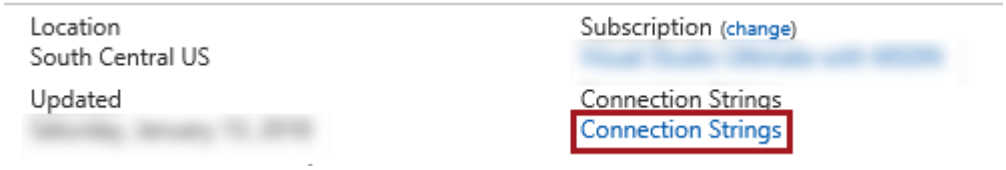
13. Wait until the policy creation has completed.

    This policy will be used by Azure Stream Analytics to retrieve events from the event hub.

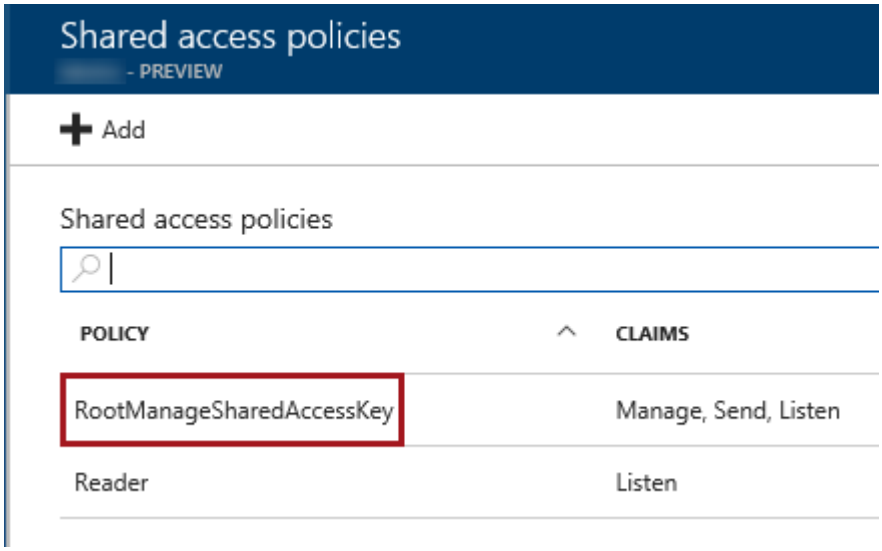
14. In the left pane, select **Overview**.



15. Click the **Connection Strings** link.

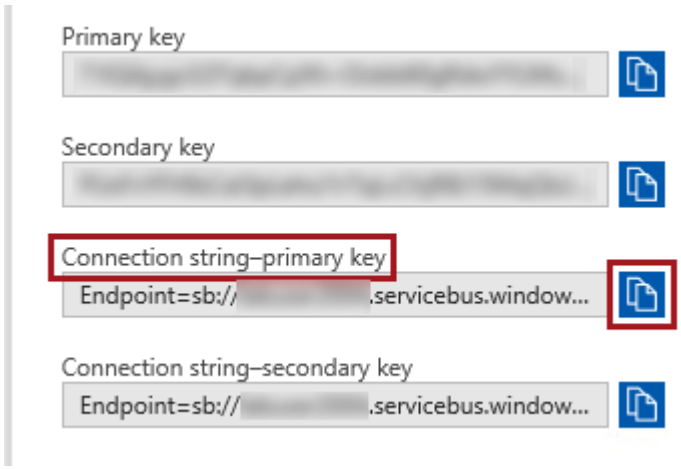


16. In the **Shared Access Policies** blade, select the **RootManageSharedAccessKey** policy.



17. Locate the **Connection String—Primary Key**.

18. To copy the connection string to the clipboard, click the copy icon located at the right of **Connection String—Primary Key**.



Preparing the Event Streamer Application

In this task, you will open the Event Streamer application and configure an event generator to cache simulation events for delivery to the productclickstream event hub.

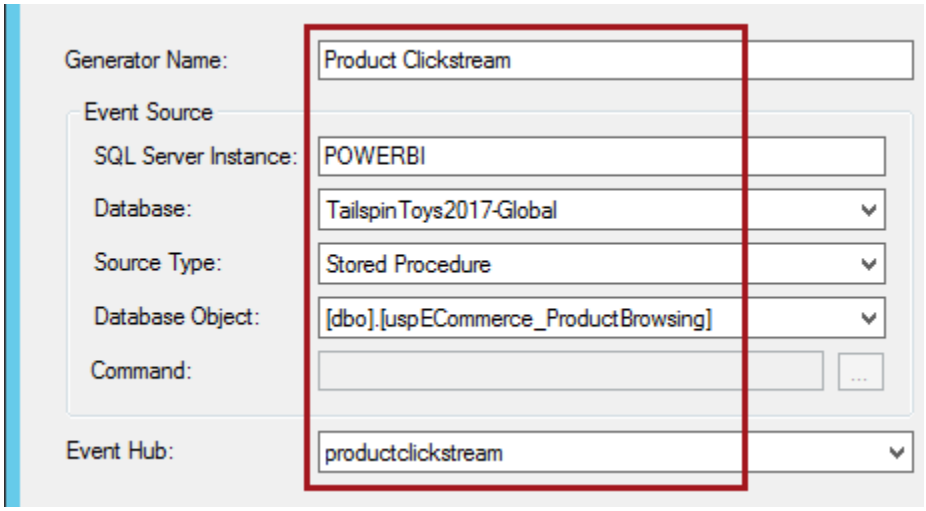
1. To open File Explorer, on the taskbar, right-click the **File Explorer** program shortcut.

2. Navigate to **D:\PowerBI\Lab04\Assets**, and then double-click the **Event Streamer** shortcut.

3. In the **Event Streamer** application, on the **Application** menu, select **Options**.

4. Use **Control-V** to paste the copied key inside the **Service Bus Connection String** box.
5. Click **OK**.
6. To create an event generator, on the **Generator** menu, select **Add Generator**.
7. In the **Create Generator** window, configure the following properties.

Property	Value
Generator Name	Product Clickstream
SQL Server Instance	POWERBI
Database	TailspinToys2017-Global
Source Type	Stored Procedure
Database Object	[dbo].[uspECommerce_ProductBrowsing]
Event Hub	productclickstream



This configuration will retrieve a result set from a SQL Server stored procedure. When events are sent to the event hub, each row will be output as a JSON string using UTF-8 encoding.

8. Click **OK**.
9. When the event generator has been created, in the confirmation message box, click **OK**.
10. In the **Event Streamer** application, to preview the simulation event data, click **Preview**.
11. In the **Data Cache Preview** window, review the data, and in particular notice that each clickstream event is described by product name, category, and the originating country of the web user.
12. Click **Close**.
13. Leave the **Event Streamer** application open.

    You will start event generation once the Azure Stream Analytics job is created and started.

Provisioning a Streaming Job

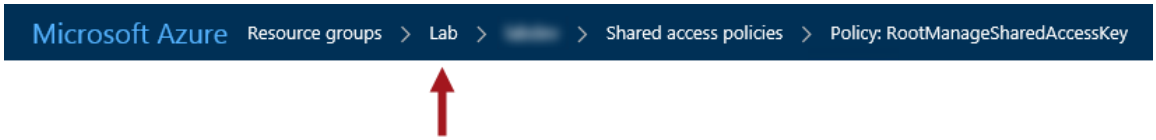
In this exercise, you will create an Azure Stream Analytics job to read from the **productclickstream** event hub, and to output a temporal aggregation of the event stream.

*At the end of this lab, you must complete the **Finishing Up** exercise to remove the services you create in this exercise.*

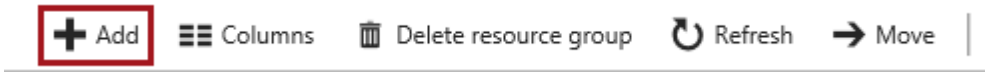
Creating an Azure Stream Analytics Job

In this task, you will create an Azure Stream Analytics job.

1. Switch to the Internet Explorer Azure Portal window.
2. On the bread crumb trail, click the **Lab** resource group.



3. In the **Lab** resource group blade, to add a new resource, click **Add**.

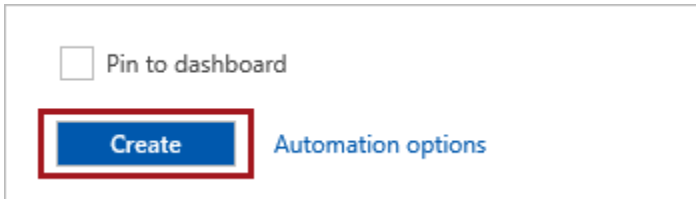


4. In the **Everything** blade, in the search box, enter **Stream**, and then press **Enter**.
5. Select **Stream Analytics Job**.

6. In the **Stream Analytics Job** blade, click **Create**.

7. In the **Job Name** box, enter **ProductClickstream**.
8. Select the existing **Lab** resource group.

9. Click **Create**.



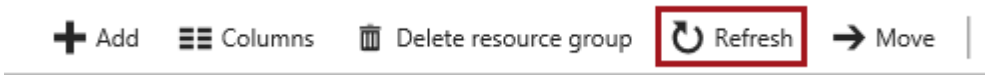
10. Wait until the Stream Analytics Job deployment has completed.

Configuring the Azure Stream Analytics Job

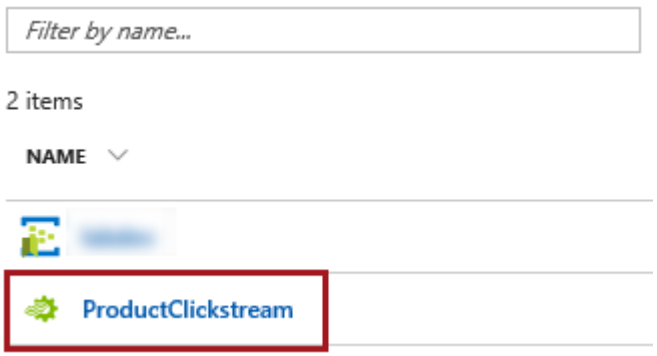
In this task, you will configure the Azure Stream Analytics Job.

1. On the bread crumb trail, click the **Lab** resource group.

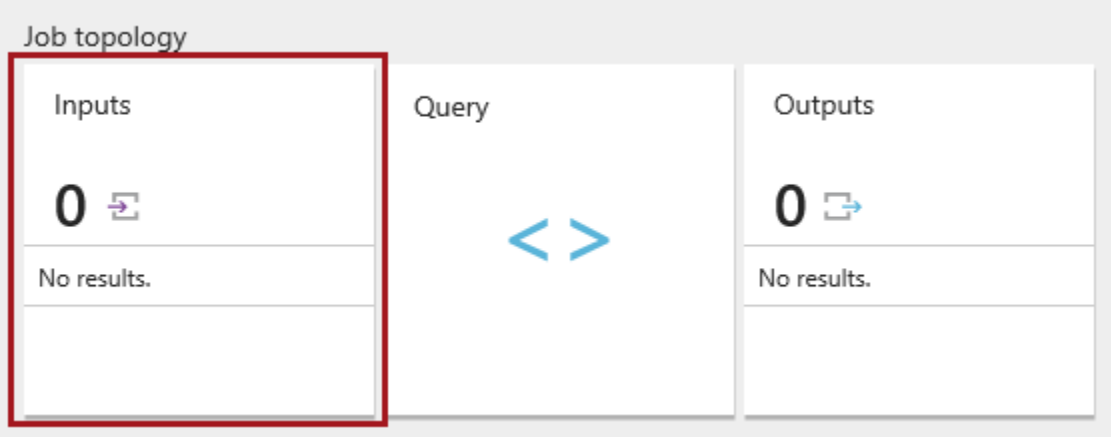
2. To refresh the **Lab** resource group blade, click **Refresh**.



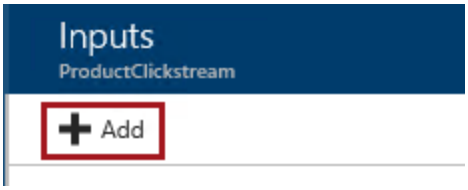
3. Select the Stream Analytics Job.



4. In the **ProductClickstream** blade, click the **Inputs** tile.



5. In the **Inputs** blade, to add an input, click **Add**.



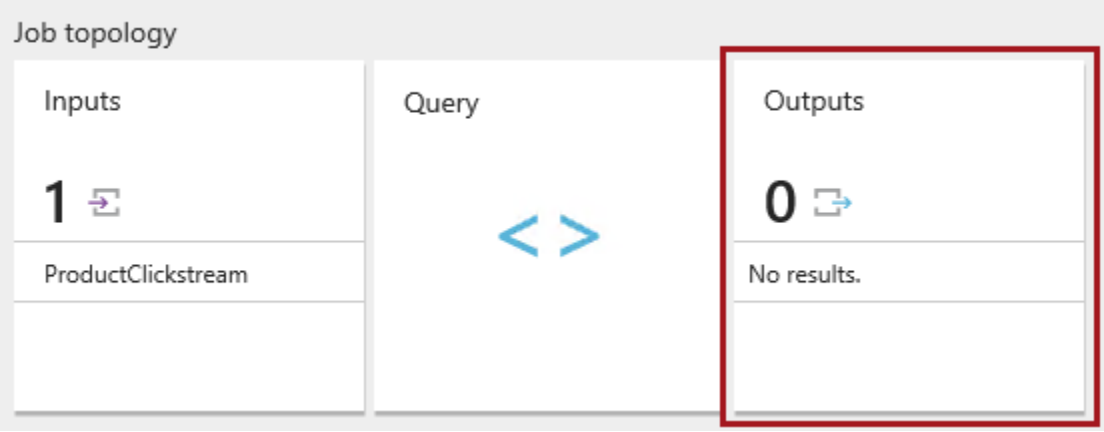
6. In the **New Input** blade, in the **Name** box, enter an available input name (like **ProductClickstream**).

   This name will be used by the job query. You will need to update the job query to use the name you submit—take note of the name for future reference.

7. In the **Service Bus Namespace** dropdown list, select the namespace you created in the previous exercise.
8. In the **Event Hub Name** dropdown list, ensure that the event hub created in the previous exercise is selected.
9. In the **Event Hub Policy Name** dropdown list, ensure that the **Reader** policy is selected.
10. Click **Create**.

11. Wait until the input has been created.
12. To close the **Inputs** blade, located at the top-right corner, click **X**.

13. In the **ProductClickstream** blade, click the **Outputs** tile.



14. In the **Outputs** blade, to add an output, click **Add**.

15. In the **New Output** blade, in the **Output Alias** box, enter an available output name (like **PowerBI**).

    This name will be used by the job query. You will need to update the job query to use the name you submit—take note of the name for future reference.

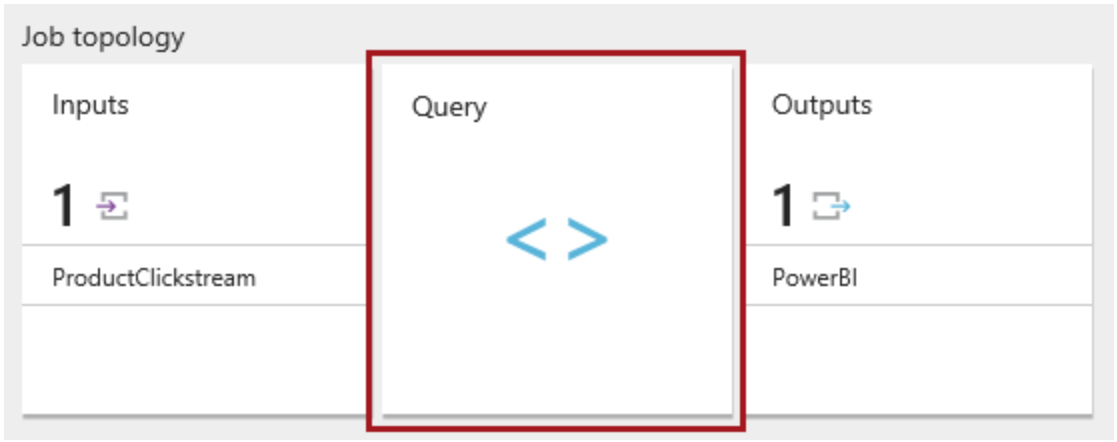
16. In the **Sink** dropdown list, select **Power BI**.
17. Click **Authorize**.

18. Complete the authorization by using the same account you used to sign in to Power BI.
19. In the **Dataset Name** box, enter **Product Clickstream**.
20. In the **Table Name** box, enter **Country Product Clicks**.
21. Click **Create**.

22. Wait until the output has been created.

23. To close the **Outputs** blade, located at the top-right corner, click **X**.

24. To configure the query, click the **Query** tile.



25. Replace all six lines of the query template with the following query.

    *For convenience, the query can be copied from the **D:\PowerBI\Lab04\Assets\Snippets.txt** file.*

Stream Analytics Query Language

```
SELECT
   System.TimeStamp AS [Time]
   ,[Country]
   ,[Category]
   ,[Product]
   ,COUNT(*) AS [Clicks]
INTO
   [PowerBI]
FROM
   [YOUR INPUT]
GROUP BY
   [Country]
   ,[Category]
   ,[Product]
   ,TUMBLINGWINDOW(SECOND, 2);
```

Stream Analytics query language is a subset of standard T-SQL syntax for doing streaming computations.

*Notice that the FROM clause describes the job input, while the INTO clause describes the job output. The remainder of the query reads like a T-SQL aggregate query, with the exception of the **TUMBLINGWINDOW** function.*

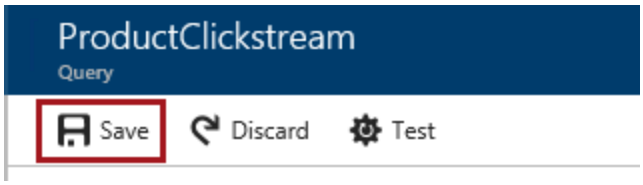
*The **TUMBLINGWINDOW** function is used to aggregate over a fixed-sized, non-overlapping and contiguous time interval, in this case a window of two seconds' duration.*

26. Verify that line 8 uses the correct output name, within square brackets.

    Tip: The name of the job inputs and outputs are listed in the left pane.

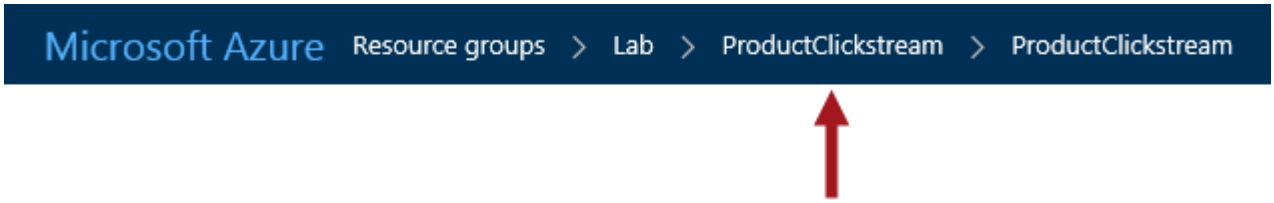
27. Replace line 10 with the name of your input, within square brackets.

28. Click **Save**.

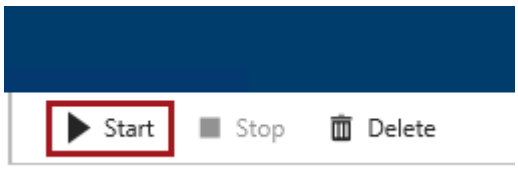


29. When prompted to confirm saving the query, click **Yes**.

30. On the bread crumb trail, click the **ProductClickstream** stream analytics job.



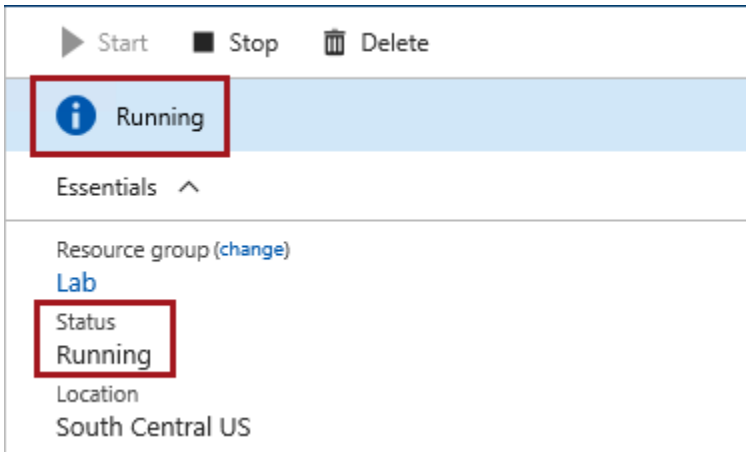
31. To start the job, click **Start**.



32. In the **Start Job** blade, click **Start**.

33. Wait until the streaming job has been started.

    The job may take some 1-2 minutes to start. The Power BI dataset will not be created until the job outputs its first query result, and this will not happen until events have been forwarded to the event hub. You will start event generation in the next exercise.



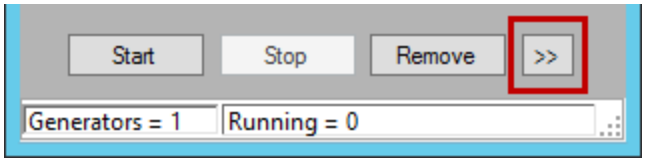
Creating a Real-Time Power BI Dashboard

In this exercise, you will create a real-time Power BI dashboard.

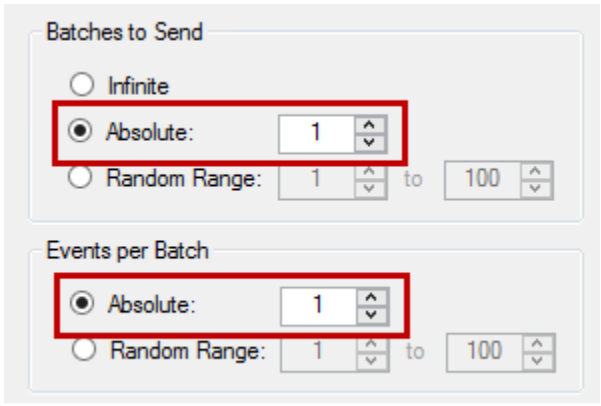
Sending a Single Event

In this task, you will configure an event generation policy to send a single event, and then start the generator.

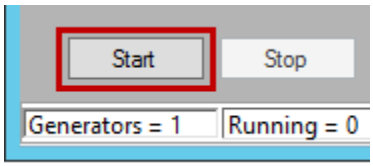
1. Switch to the **Event Streamer** application.
2. To configure the event generation policy, click the **Expand** button.



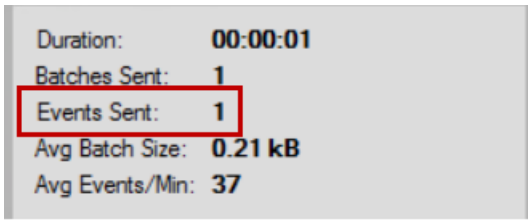
3. To send a single event, configure the following event generation policy.



4. To start the event generation, click **Start**.



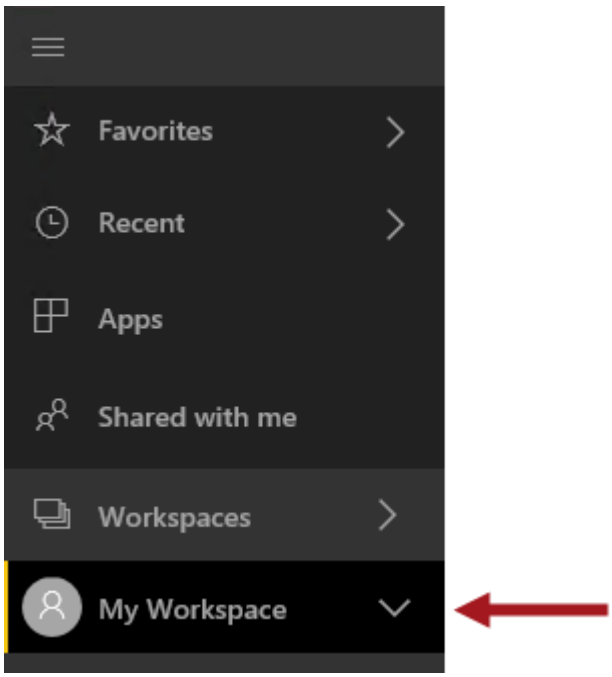
5. Verify that a single event was generated.



Reviewing the Dataset

In this task, you will verify that the dataset has been created.

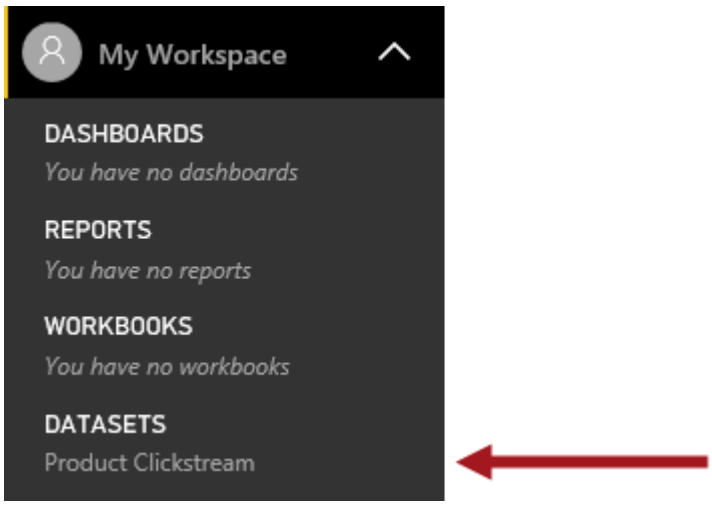
1. In the Power BI Internet Explorer window In the **Navigation Pane** (located at the left), to expand **My Workspace**, click the down-arrow.



2. In the **Datasets** group, verify that there is the **Product Clickstream** dataset.

   If the dataset does not appear, first reload (F5) the web browser.

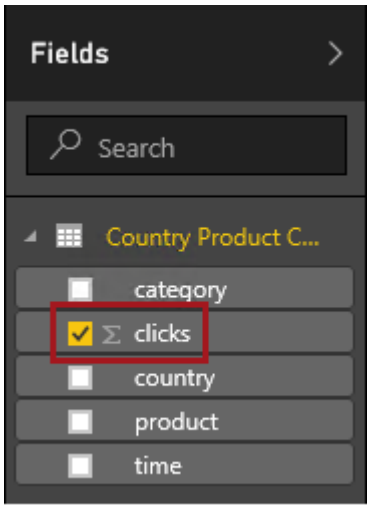
   If the dataset still does not appear, it is possible that the stream analytics job has not started, or was not properly configured.



Creating the Report

In this task, you will create a report based on the **Product Clickstream** dataset, and then pin a visualization to a dashboard.

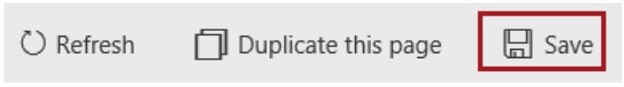
1. To create a report, in the **Navigation Pane**, click the **Product Clickstream** dataset.
2. In the **Fields** pane (located at the right), notice the **Country Product Clicks** table.
3. To add a visual, check the **clicks** field.



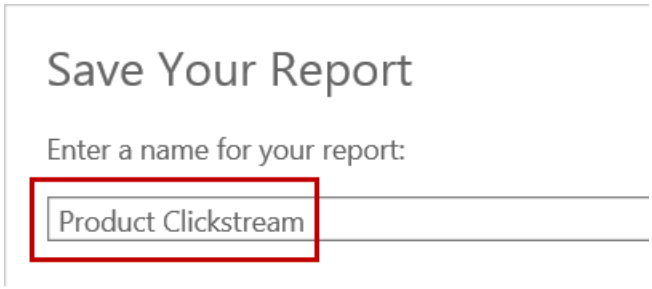
4. Notice that a bar chart visual was added to the report canvas.
5. To modify the visualization type to card, in the **Visualizations** pane, select **Card**.

   Tip: Hovering over a visualization type icon will reveal a tooltip that describes the visualization type.

6. To save the report, at the right corner, click **Save**.



7. In the **Save Your Report** dialog window, in the box, enter **Product Clickstream**.

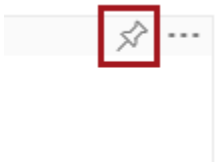


8. Click **Save**.

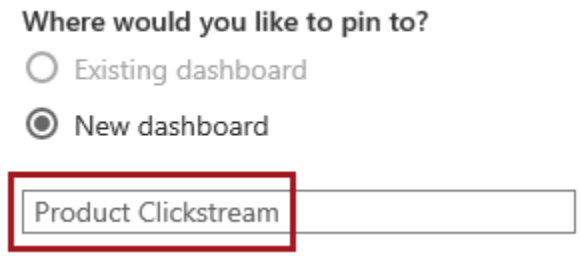
Creating the Dashboard

In this task, you will create the **Product Clickstream** dashboard.

1. To create a dashboard, hover the cursor over the card visual, and then click the **Pin Visual** icon.

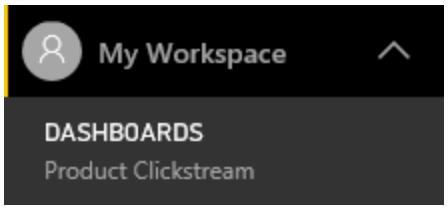


2. In the **Pin to Dashboard** dialog window, in the box, enter **Product Clickstream**.

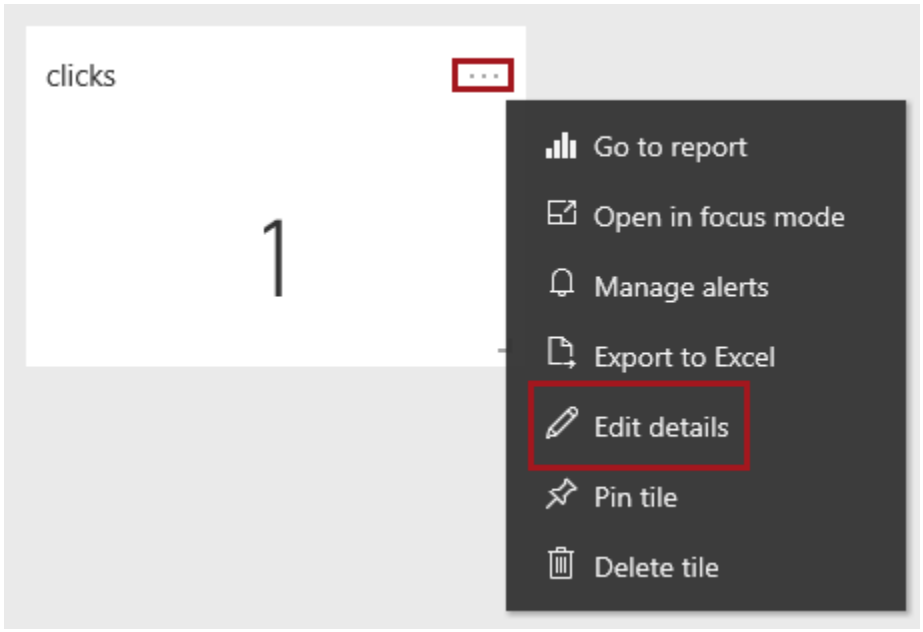


3. Click **Pin** to add a tile to a new dashboard.

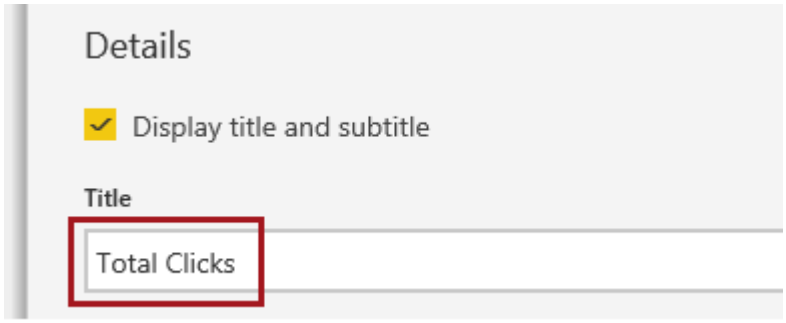
4. To navigate to the dashboard, in the **Navigation Pane**, select the **Product Clickstream** dashboard.



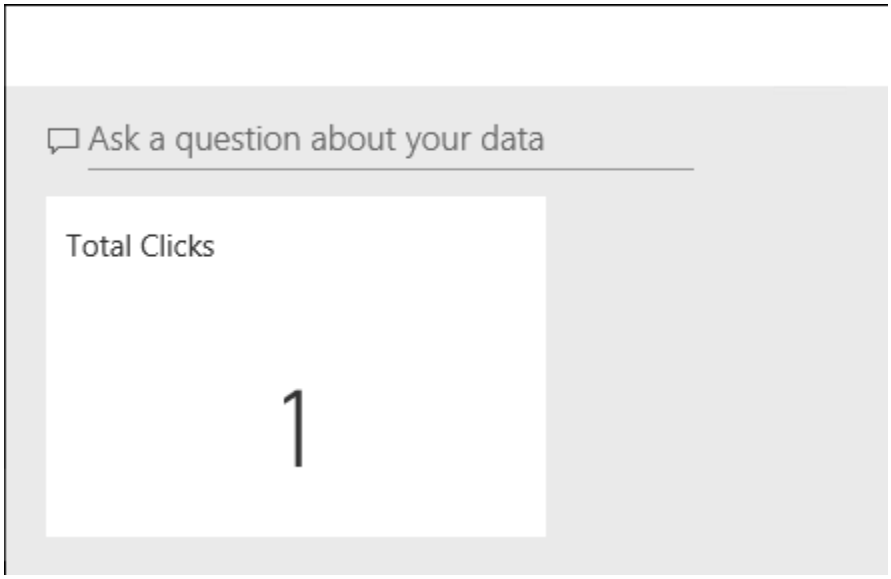
5. To modify the tile title, hover the cursor over the tile, click the ellipsis, and then select **Edit Details**.



6. In the **Tile Details** pane (located at the right), in the **Title** box, modify the text to **Total Clicks**.



7. Click **Apply**.
8. Verify that the dashboard layout looks like the following.



Creating Additional Dashboard Tiles

In this task, you will use Q&A to create additional dashboard tiles.

1. In the Q&A box, enter **Clicks for time in last 60 seconds**, and then review the response.

   *For convenience, all Q&A questions can be copied from the **D:\PowerBI\Lab04\Assets\Snippets.txt** file.*

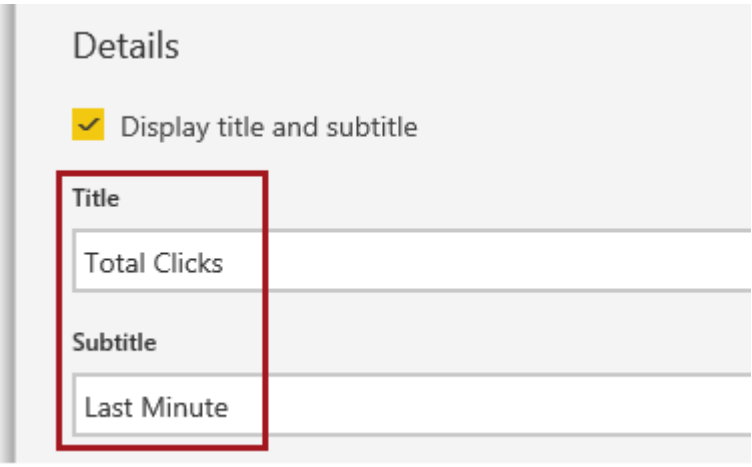
   *As the Stream Analytics job query uses a two second tumbling window, the value will be the aggregation of the last 30 query results. At this point in time, the expected response is **(Blank)**.*

2. To create a dashboard tile for this question, at the top-right corner, click the **Pin Visual**.

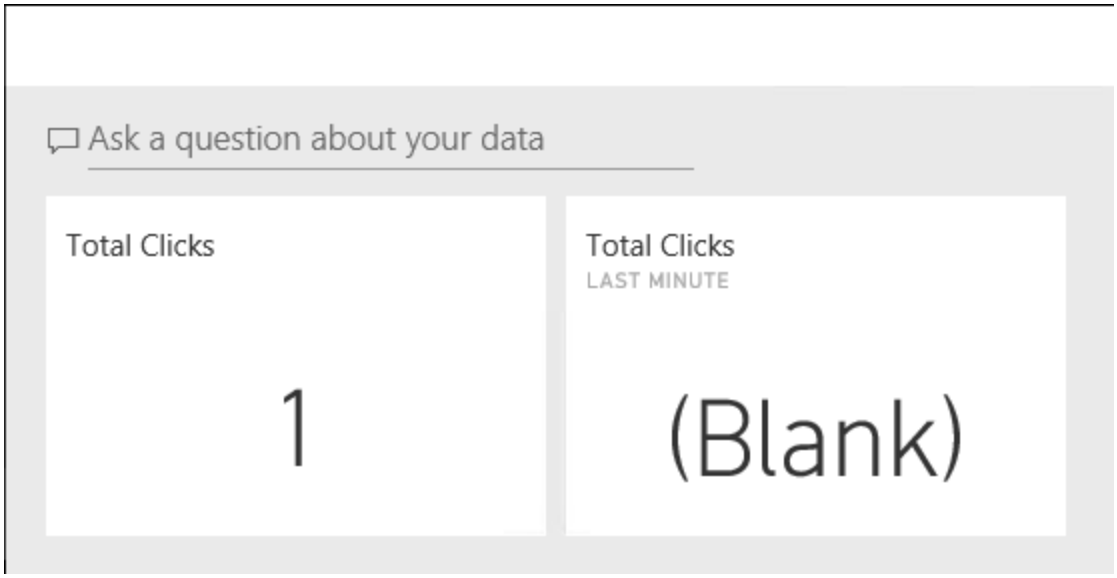
3. Pin the visualization to the existing dashboard.

4. To return to the dashboard page, click **Exit Q&A**.

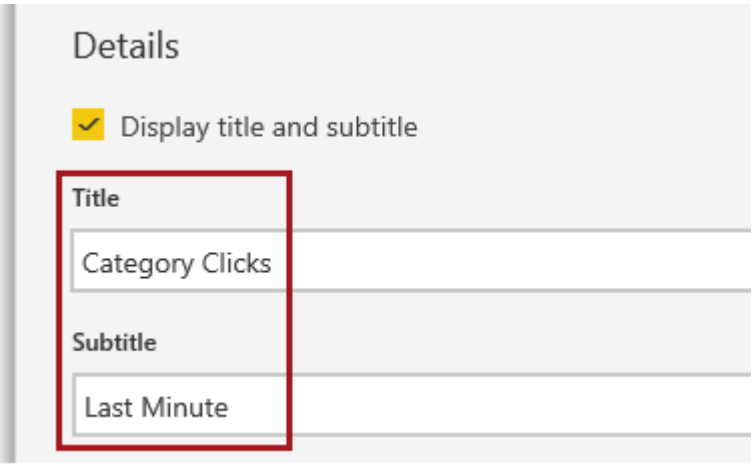
5. Modify the title of the new tile to **Total Clicks**, and the subtitle to **Last Minute**.



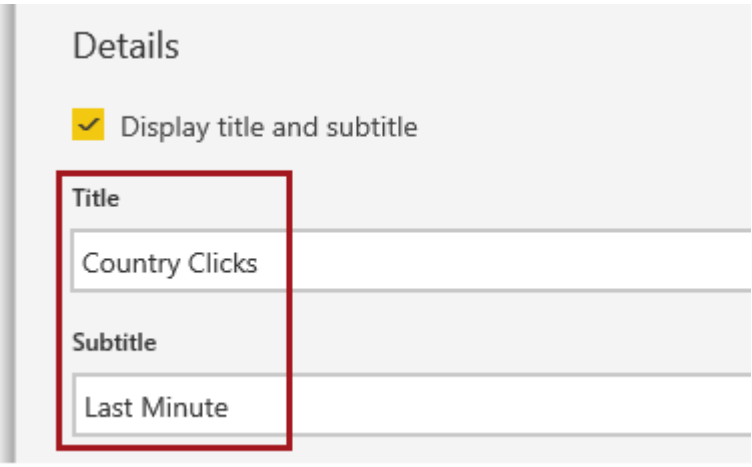
6. Verify that the dashboard layout looks like the following.



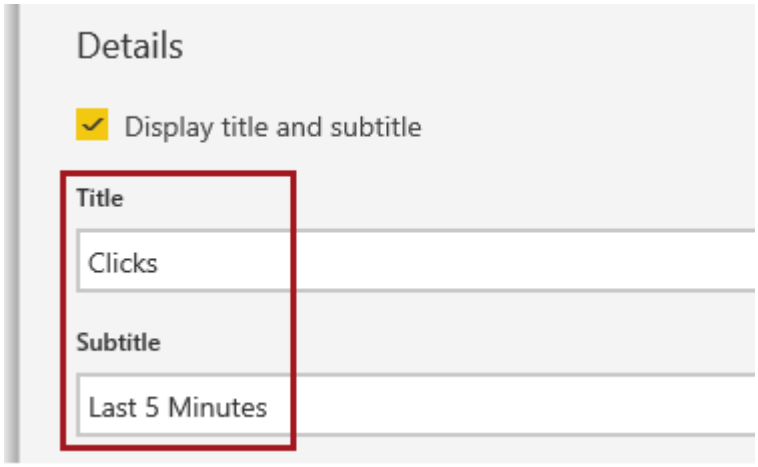
7. Create another tile by asking **Clicks for time in last 60 seconds by category sorted by clicks**.
8. Modify the tile titles as follows.



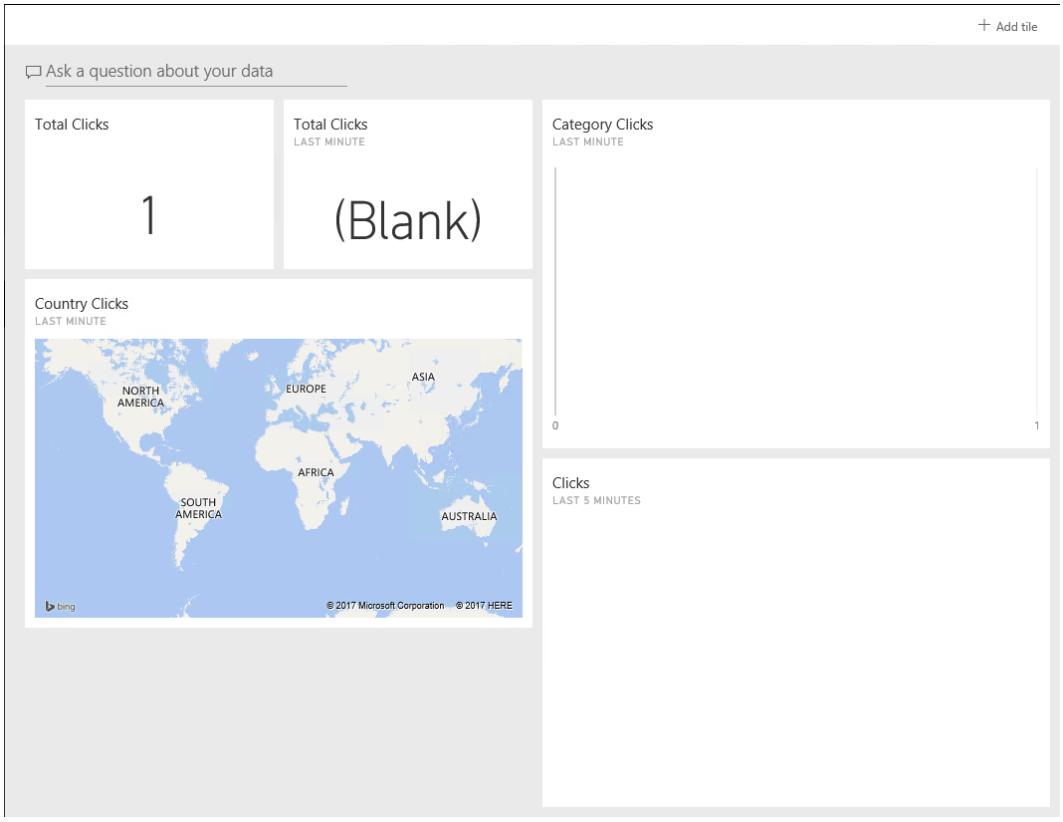
9. Create another tile by asking **Clicks for time in last 60 seconds by country as map**.
10. Modify the tile titles as follows.



11. Create another tile by asking **Clicks by time in the last 300 seconds**.
12. Modify the tile titles as follows.



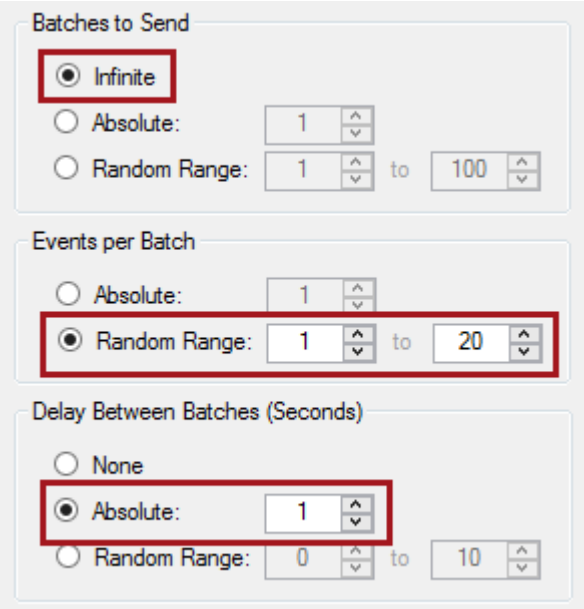
13. Verify that the dashboard layout looks like the following.



Starting the Event Generation

In this task, you will reconfigure the event generation policy, and then start the generator.

1. Switch to the Event Streamer application.
2. If necessary, position the application window so that the dashboard tiles are fully visible in the background.
3. Modify the event generator policy as follows.



   There are limits to the amount of data that can be streamed to the Power BI Service. According to the Power BI Pricing, for the Power BI (free) tier and for the Power BI Pro tier, it is one million rows/hour.

4. Click **Start**.
5. Immediately, switch back to the Power BI Internet Explorer window, and notice that the tiles update every two seconds.

   It may take several seconds for the tiles to commence updating.

Finishing Up

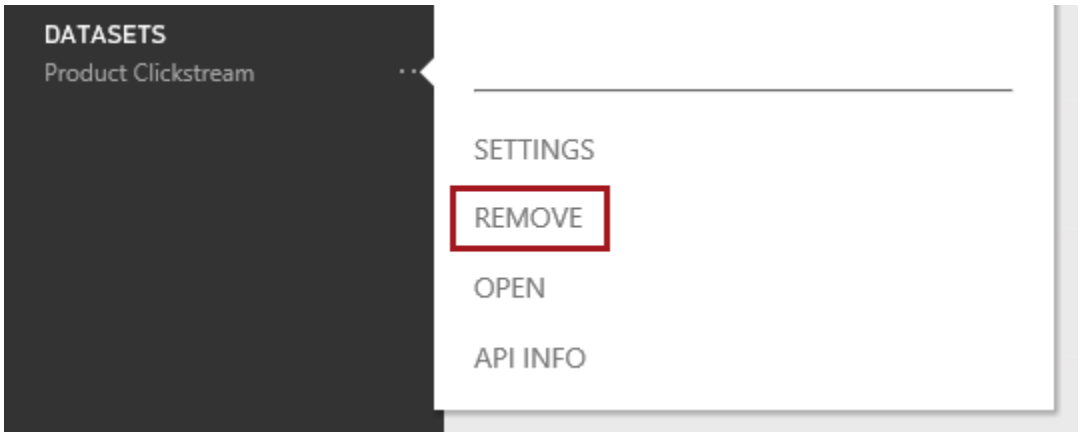
In this exercise, you will finish up by undoing the configurations made in this lab.

To limit cost for your Azure subscription, you must follow the steps in this exercise to delete the Azure Resource Group.

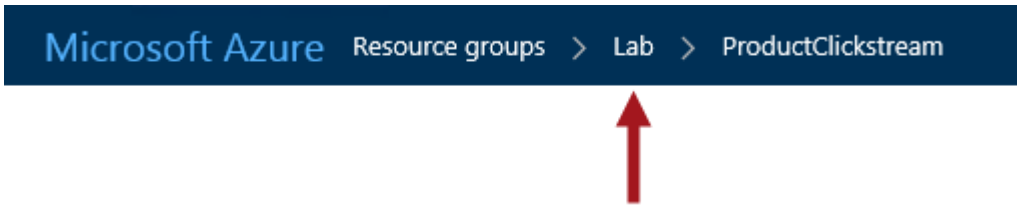
Finishing Up

In this task, you will finish up by undoing the configurations made in this lab.

1. In the Event Streamer application, on the **Application** menu, select **Exit**.
2. Switch to the Power BI Internet Explorer window.
3. To remove the dataset, in the **Navigation Pane**, right-click the **Product Clickstream** dataset, and then select **Remove**.



4. When prompted to confirm the streaming deletion, click **Delete**.
5. Notice that the **Product Clickstream** report is deleted as a consequence.
6. Use the same steps to delete also the **Product Clickstream** dashboard.
7. Switch to the Azure Portal Internet Explorer window.
8. On the bread crumb trail, click the **Lab** resource group.



9. To delete the resource group, in the command bar, click **Delete Resource Group**.

10. In the blade, in the **Type the Resource Group Name** box, enter **Lab**.
11. Click **Delete**.

12. Sign out of the Azure Portal.
13. Close the Azure Portal Internet Explorer window.

Summary

In this lab, you produced a real-time Power BI dashboard by using an Azure Event Hub and Azure Stream Analytics. You commenced by signing in to Power BI, and then signing in to the Azure Portal to create an Event Hub, and then a Stream Analytics job.

Terms of Use

By using this hands-on lab, you agree to the following terms:

The technology/functionality described in this hands-on lab is provided by Microsoft Corporation in a "sandbox" testing environment for purposes of obtaining your feedback and to provide you with a learning experience. You may only use the hands-on lab to evaluate such technology features and functionality and provide feedback to Microsoft. You may not use it for any other purpose. Without written permission, you may not modify, copy, distribute, transmit, display, perform, reproduce, publish, license, create derivative works from, transfer, or sell this hands-on lab or any portion thereof.

COPYING OR REPRODUCTION OF THE HANDS-ON LAB (OR ANY PORTION OF IT) TO ANY OTHER SERVER OR LOCATION FOR FURTHER REPRODUCTION OR REDISTRIBUTION WITHOUT WRITTEN PERMISSION IS EXPRESSLY PROHIBITED.

THIS HANDS-ON LAB PROVIDES CERTAIN SOFTWARE TECHNOLOGY/PRODUCT FEATURES AND FUNCTIONALITY, INCLUDING POTENTIAL NEW FEATURES AND CONCEPTS, IN A SIMULATED ENVIRONMENT WITHOUT COMPLEX SET-UP OR INSTALLATION FOR THE PURPOSE DESCRIBED ABOVE. THE TECHNOLOGY/CONCEPTS REPRESENTED IN THIS HANDS-ON LAB MAY NOT REPRESENT FULL FEATURE FUNCTIONALITY AND MAY NOT WORK THE WAY A FINAL VERSION MAY WORK. WE ALSO MAY NOT RELEASE A FINAL VERSION OF SUCH FEATURES OR CONCEPTS. YOUR EXPERIENCE WITH USING SUCH FEATURES AND FUNCITONALITY IN A PHYSICAL ENVIRONMENT MAY ALSO BE DIFFERENT.

FEEDBACK If you give feedback about the technology features, functionality and/or concepts described in this hands-on lab to Microsoft, you give to Microsoft, without charge, the right to use, share and commercialize your feedback in any way and for any purpose. You also give to third parties, without charge, any patent rights needed for their products, technologies and services to use or interface with any specific parts of a Microsoft software or service that includes the feedback. You will not give feedback that is subject to a license that requires Microsoft to license its software or documentation to third parties because we include your feedback in them. These rights survive this agreement.

MICROSOFT CORPORATION HEREBY DISCLAIMS ALL WARRANTIES AND CONDITIONS WITH REGARD TO THE HANDS-ON LAB, INCLUDING ALL WARRANTIES AND CONDITIONS OF MERCHANTABILITY, WHETHER EXPRESS, IMPLIED OR STATUTORY, FITNESS FOR A PARTICULAR PURPOSE, TITLE AND NON-INFRINGEMENT. MICROSOFT DOES NOT MAKE ANY ASSURANCES OR REPRESENTATIONS WITH REGARD TO THE ACCURACY OF THE RESULTS, OUTPUT THAT DERIVES FROM USE OF THE VIRTUAL LAB, OR SUITABILITY OF THE INFORMATION CONTAINED IN THE VIRTUAL LAB FOR ANY PURPOSE.

DISCLAIMER This lab contains only a portion of new features and enhancements in Microsoft Power BI. Some of the features might change in future releases of the product.

Document Version

#	Date	Author	Comments
0	24-JUL-2015	Peter Myers	Initial release
1	28-AUG-2015	Peter Myers	Migrate to new template Update lab steps to Power BI service v11.0.9167.752
2	26-SEP-2015	Peter Myers	Power BI service v11.0.9168.106
3	24-OCT-2015	Peter Myers	Power BI service v11.0.9168.431
4	30-NOV-2015	Peter Myers	Power BI service v11.0.9168.864
5	30-DEC-2015	Peter Myers	Power BI service v11.0.9169.163
6	28-JAN-2016	Peter Myers	Power BI service v11.0.127.195746
7	20-JUL-2016	Peter Myers	Updated Power BI logo Not fully tested nor ready for release
8	04-SEP-2016	Peter Myers	Updated lab title Power BI service v13.0.1605.427
9	31-DEC-2016	Peter Myers	WIP—awaiting release of new Power BI service UX
10	24-JUN-2017	Peter Myers	Power BI service v13.0.1721.165 (new UX)
11	15-JAN-2018	Peter Myers	(Self-paced lab version) Power BI service v13.0.3847.155