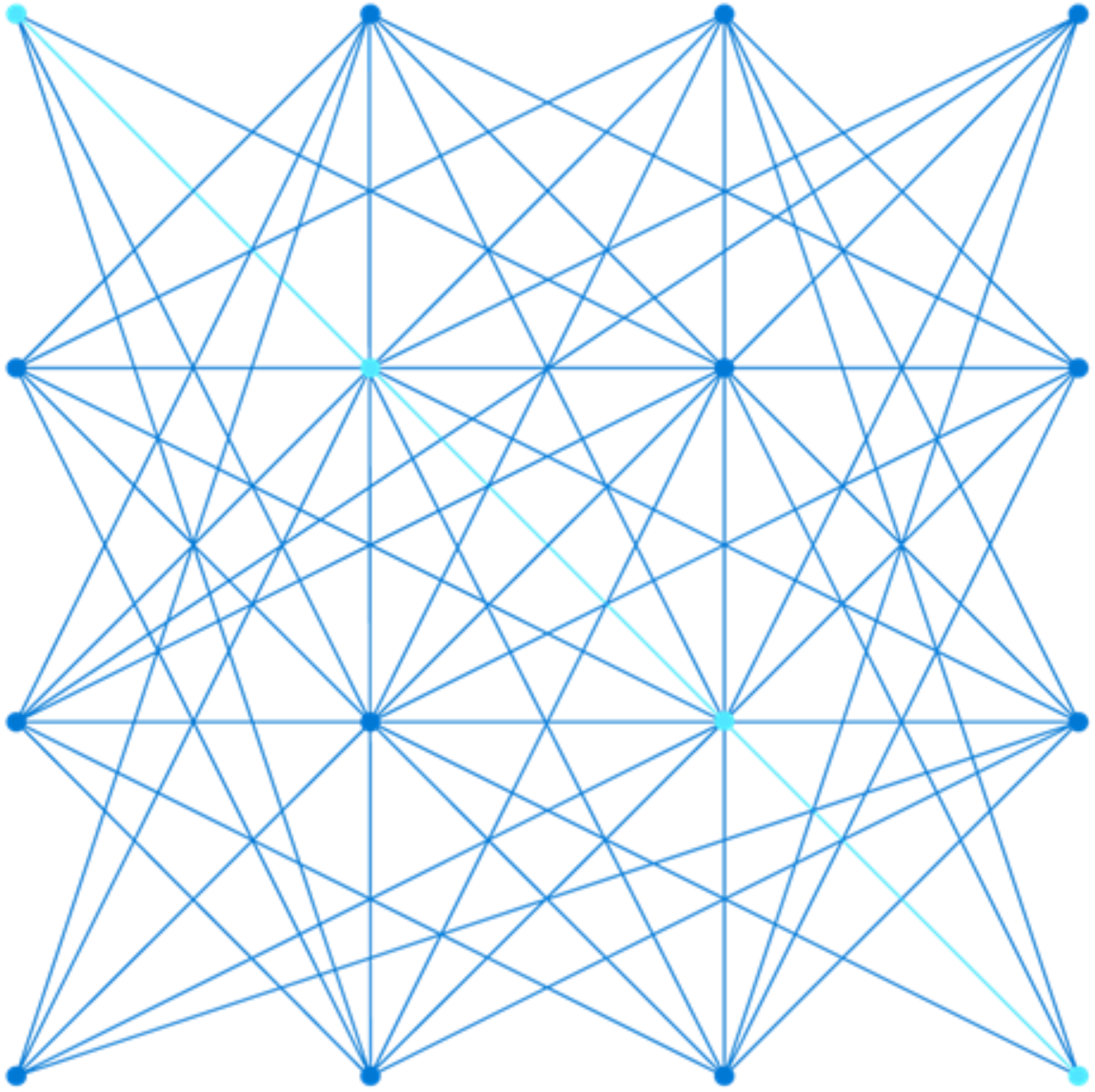
Microsoft Azure

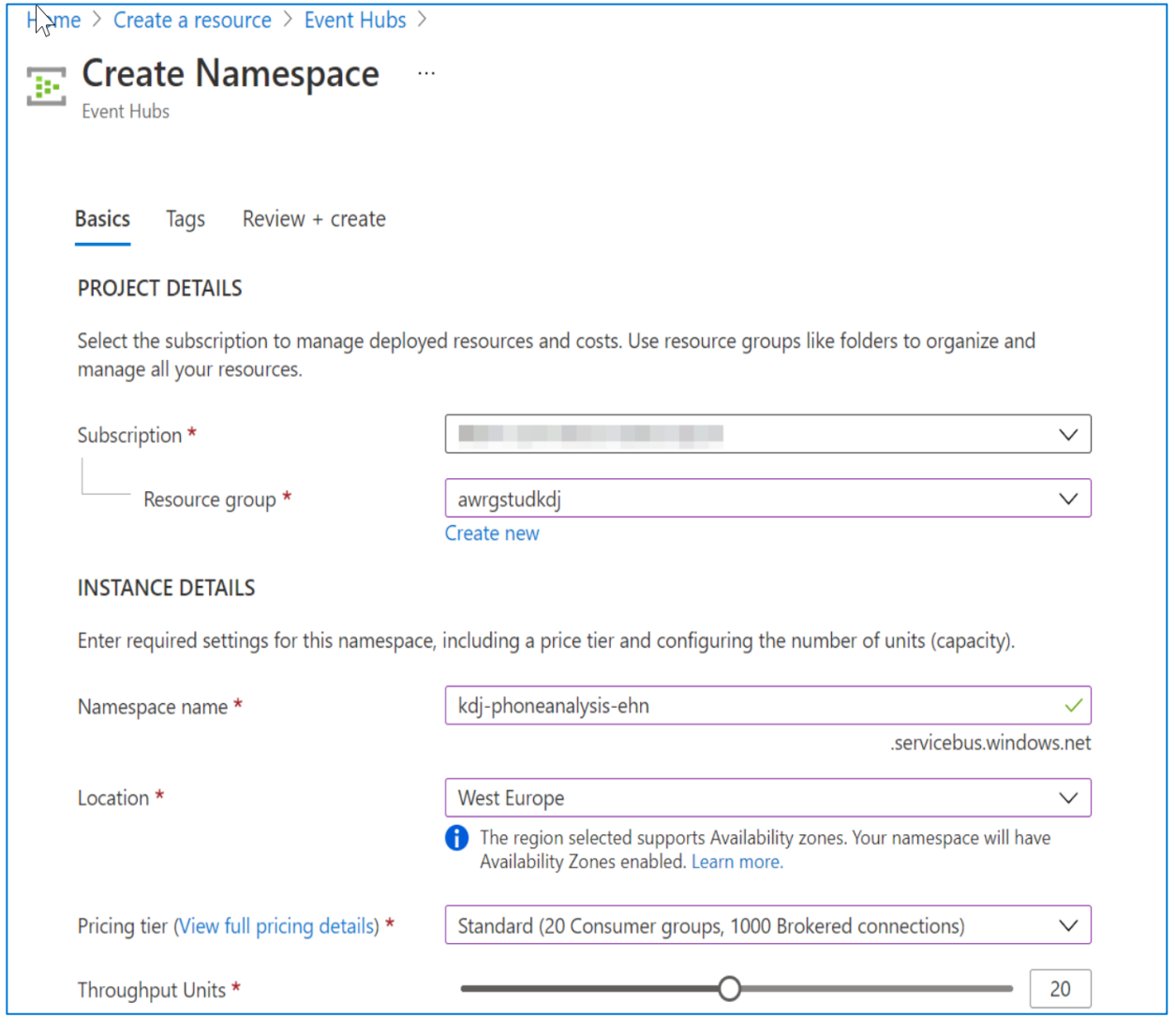
DP-203T00: Real-time Stream Processing with Stream Analytics



Lesson 01: Enable reliable messaging for Big Data applications using Azure Event Hubs

Azure Event Hubs

Azure Event Hubs is a highly scalable publish-subscribe service that can ingest millions of events per second and stream them into multiple applications



Create an Event Hub

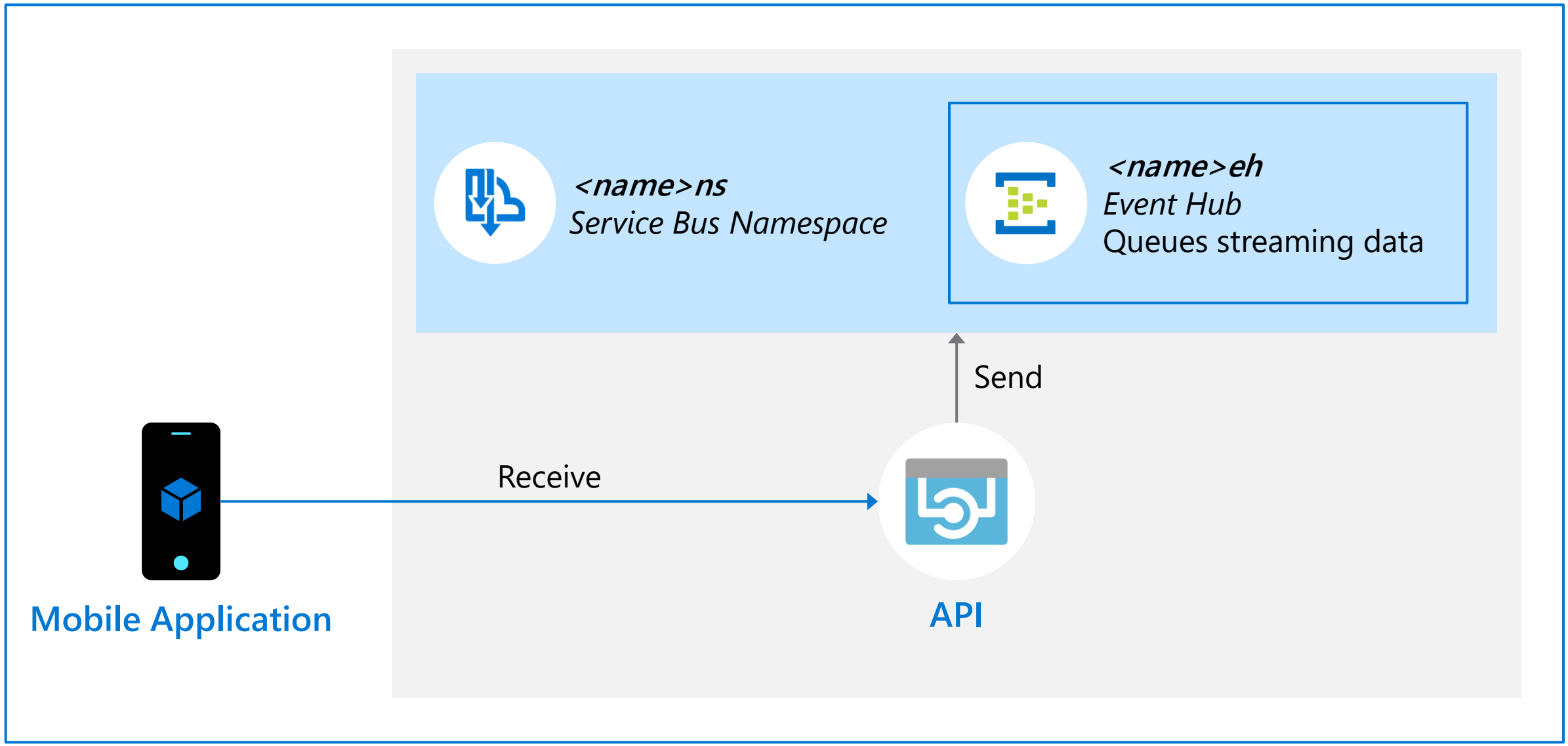
Create an event hub namespace

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. Provide a name for the event hub, and then create a resource group. Specify **xx-name-eh** and **xx-name-rg** respectively, XX- represent your initials to ensure uniqueness of the Event Hub name and Resource Group name
3. Click the checkbox to **Pin to the dashboard**, then select the **Create** button

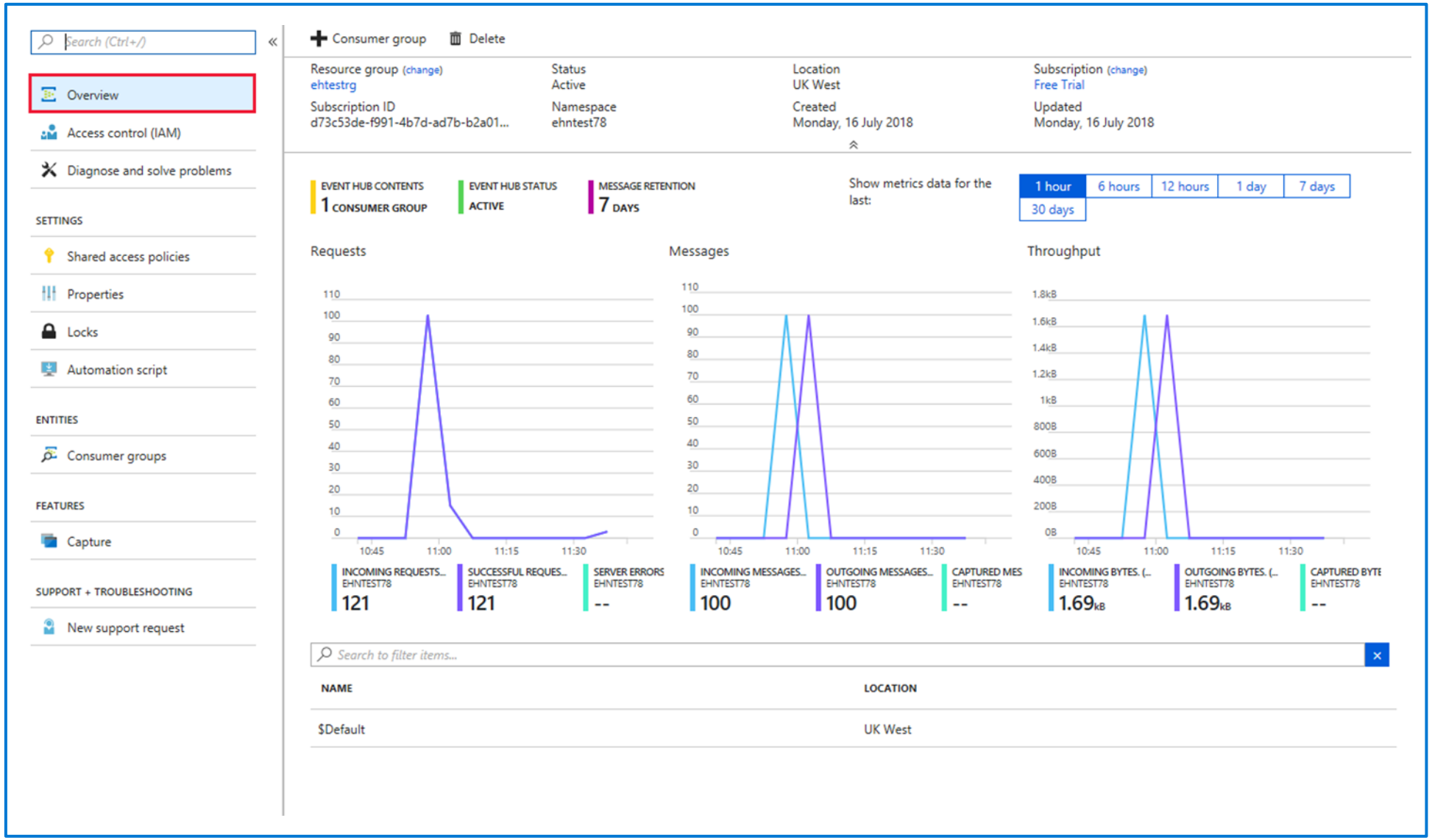
Create an event hub

1. After the deployment is complete, click the **xx-name-eh** event hub on the dashboard
2. Then, under **Entities**, select **Event Hubs**
3. To create the event hub, select the **+ Event Hub** button. Provide the name **socialstudy-eh**, and then select **Create**
4. To grant access to the event hub, we need to create a shared access policy. Select the **socialstudy-eh** event hub when it appears, and then, under **Settings**, select **Shared access policies**
5. Under **Shared access policies**, create a policy with **MANAGE** permissions by selecting **+ Add**. Give the policy the name of **xx-name-eh-sap**, check **MANAGE**, and then select **Create**
6. Select your new policy after it has been created, and then select the copy button for the **CONNECTION STRING – PRIMARY KEY** entity
7. Paste the **CONNECTION STRING – PRIMARY KEY** entity into Notepad, this is needed later in the exercise
8. Leave all windows open

Configure applications to use Event Hubs



Evaluating the performance of Event Hubs



Lesson 02: Work with data streams by using Azure Stream Analytics

What are data streams

Data streams:

In the context of analytics, data streams are event data generated by sensors or other sources that can be analyzed by another technology

Data stream processing approach:

There are two approaches. Reference data is streaming data that can be collected over time and persisted in storage as static data. In contrast, streaming data have relatively low storage requirements. And run computations in sliding windows

Data streams are used to:

Analyze data:
Continuously analyze data to detect issues and understand or respond to them

Understand systems:
Understand component or system behavior under various conditions to fuel further enhancements of said system

Trigger actions:
Trigger specific actions when certain thresholds are identified

Event processing

The process of consuming data streams, analyzing them, and deriving actionable insights out of them is called Event Processing and has three distinct components:

Event producer	Examples include sensors or processes that generate data continuously such as a heart rate monitor or a highway toll lane sensor
Event processor	An engine to consume event data streams and deriving insights from them. Depending on the problem space, event processors either process one incoming event at a time (such as a heart rate monitor) or process multiple events at a time (such as a highway toll lane sensor)
Event consumer	An application which consumes the data and takes specific action based on the insights. Examples of event consumers include alert generation, dashboards, or even sending data to another event processing engine

Processing events with Azure Stream Analytics

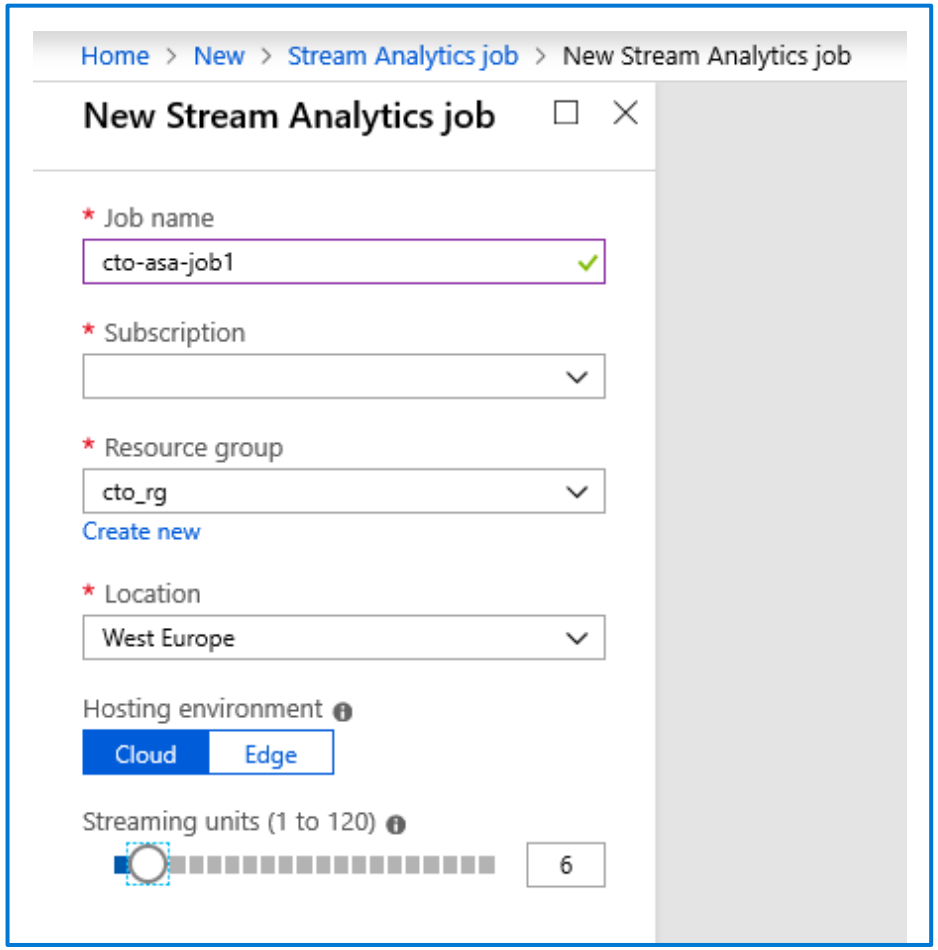
Microsoft Azure Stream Analytics is an event processing engine. It enables the consumption and analysis of high volumes of streaming data in real time

Source	Ingestion	Analytical engine	Destination
Sensors	Event Hubs	Stream Analytics Query Language	Azure Data Lake
Systems	IoT Hubs	.NET SDK	Cosmos DB
Applications	Azure Blob Store		SQL Database
			Blob Store
			Power BI

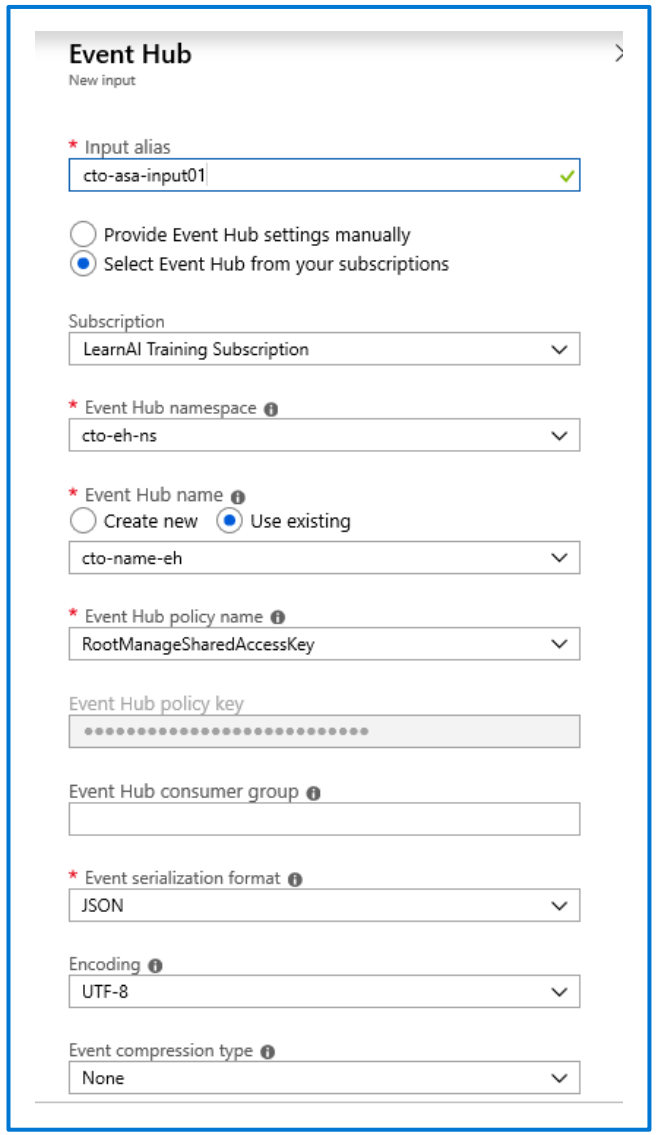
Lesson 02: Transform data by using Azure Stream Analytics

Create Stream Analytics service

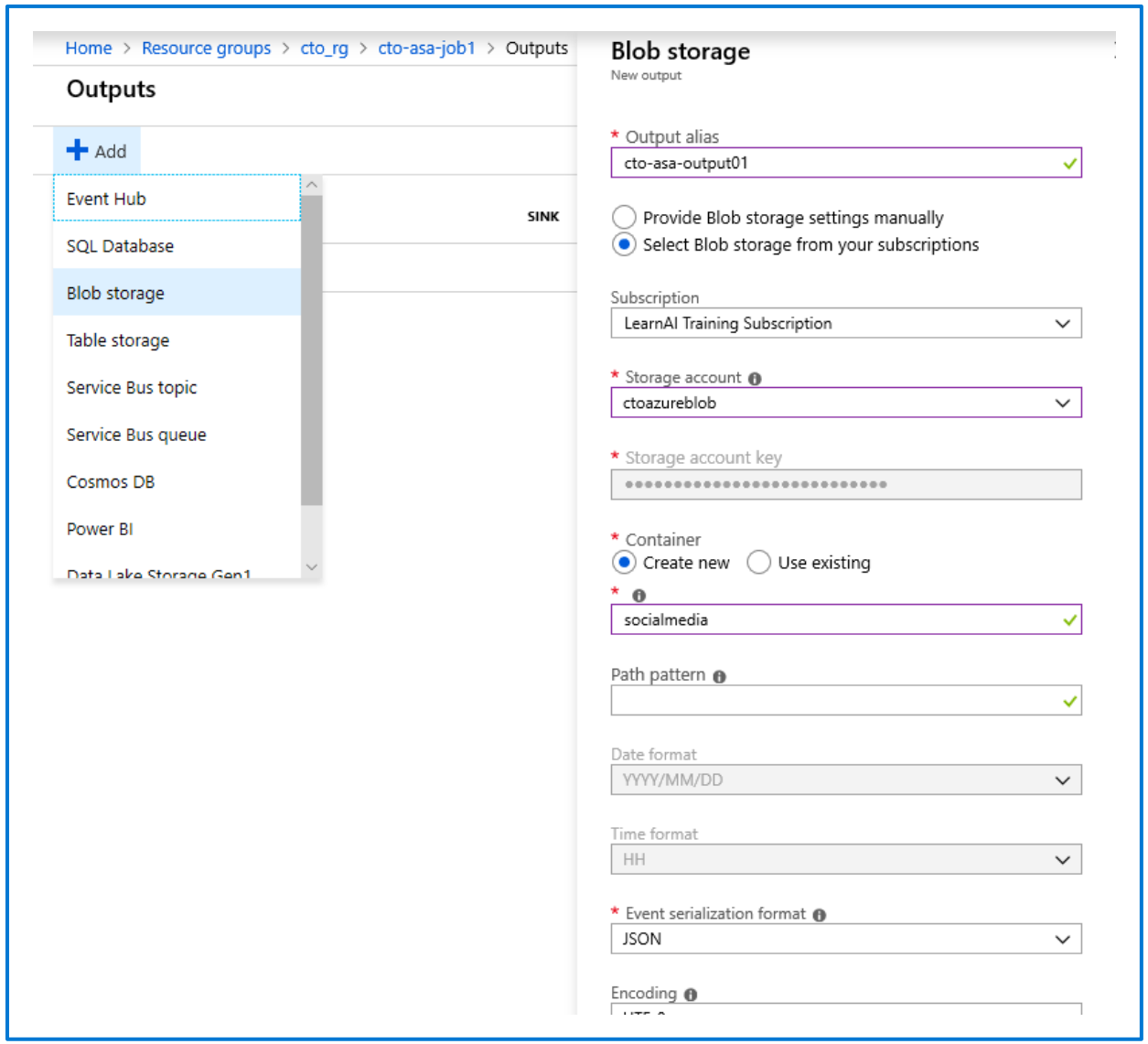
Job name	Subscription	Resource group	Location



Create a Stream Analytics Job input

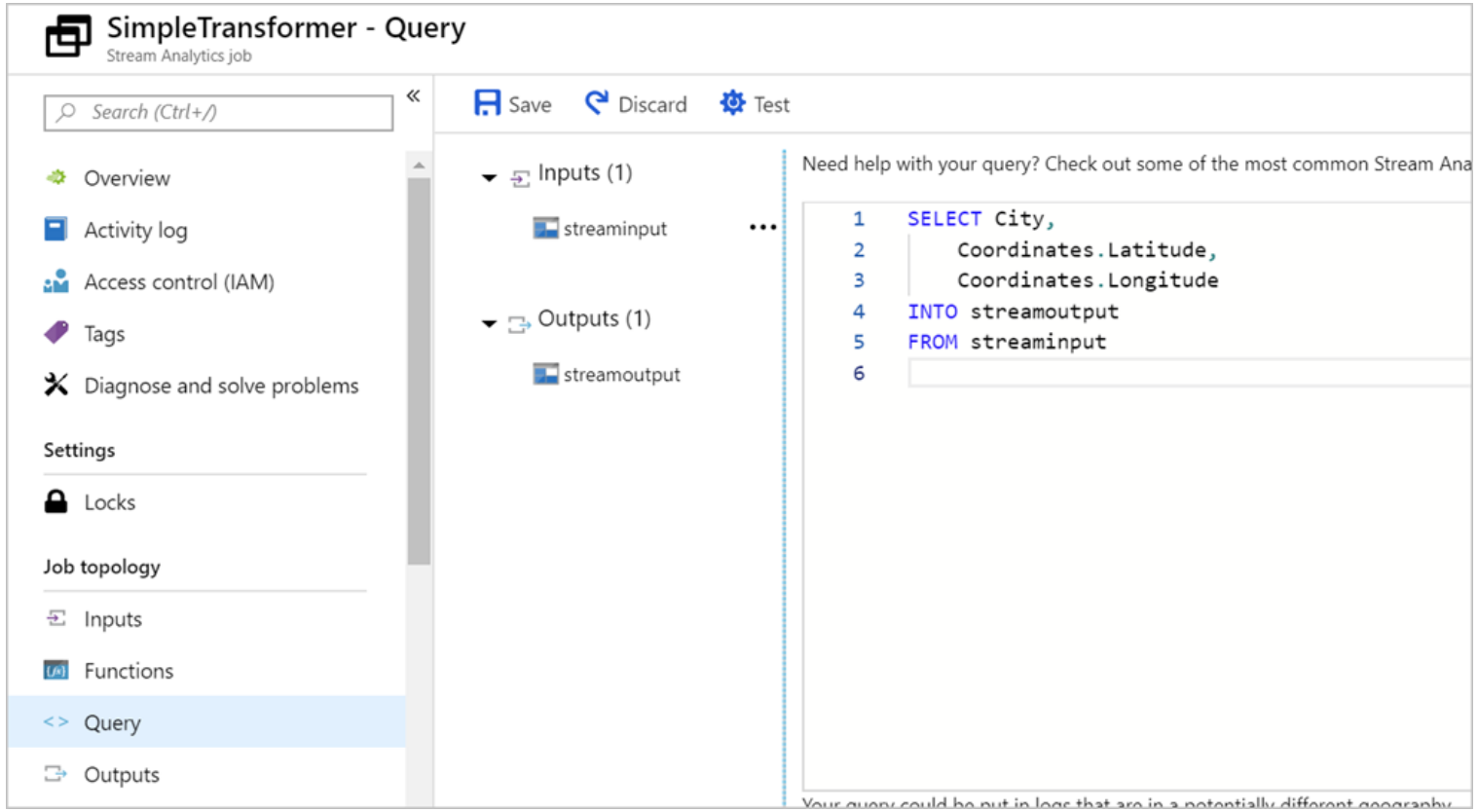


Create a Stream Analytics Job output



Transform data by using Azure Stream Analytics

- Declarative SQL like language
- Contains testing capabilities
- Performs aggregations over windows of time

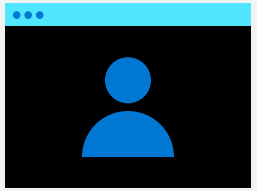


Using Windowing functions with Azure Stream Analytics

Performing operations on the data contained in temporal windows is a common pattern. Stream Analytics has native support for windowing function including

Tumbling	Hopping	Sliding	Session
Tumbling window functions are used to segment a data stream into distinct time segments and perform a function against them. For example, tell me the count of tweets per time zone every 10 seconds	Hopping window functions hop forward in time by a fixed period. For example, every 5 seconds, give me the count of tweets for the last 10 seconds	Sliding windows, output events only for points in time when the content of the window actually changes. For example, give a count of all tweets which are tweeted more than 10 times on a given topic	Session window functions group events that arrive at similar times, filtering out periods of time where there is no data. For example, give the count of tweets that occur within 5 minutes of each other

Lab: Real-time Stream Processing with Stream Analytics



Lab overview

This lab teaches you how to process streaming data with Azure Stream Analytics. You will ingest vehicle telemetry data into Event Hubs, then process that data in real time, using various windowing functions in Azure Stream Analytics. You will output the data to Azure Synapse Analytics. Finally, you will learn how to scale the Stream Analytics job to increase throughput.

Lab objectives

After completing this lab, you will be able to:

- Use Stream Analytics to process real-time data from Event Hubs
- Use Stream Analytics windowing functions to build aggregates and output to Synapse Analytics
- Scale the Azure Stream Analytics job to increase throughput through partitioning
- Repartition the stream input to optimize parallelization