BIG DATA HANDS-ON LAB

Azure Data Services:

Data Factory

Data Lake Store

SQL Data Warehouse

Databricks

Microsoft

OVERVIEW

SUMMARY

Most companies already have one or more data warehouses. However, extending and maintaining this data warehouse can be difficult. Source systems are changing faster than ever before, and end users want to make deeper analyses.

Therefore, a more flexible architecture is needed which makes it easier to add different types of data.

During this workshop you will extend the data warehouse using the Azure Data Services.

The use case during this workshop is about airdelays and preparing the data for Data Scientists on the one hand but also providing it for analysts via the Data Warehouse.

Lab 1 will guide through the data acquisition and how to create data pipelines with Azure Data Factory and load data into Azure SQL Datawarehouse

Lab 2 guides through the creation and usage of a Databricks Cluster. You will use Python and SQL to analyze and massage the data and provide it for further usage with other services.

Lab 3 will then examine the possibilities with Azure SQL Data Warehouse and provides some insights into the world of MPP- (massive parallel processing) databases. You will get data with Polybase from your Data Lake and join a dimension table that lives in the database.

Good luck and enjoy the labs!

PREREQUISITES

SQL Server Management Studio: https://docs.microsoft.com/en-us/sql/ssms/download-sql-server-management-studio-ssms

Azure Storage Explorer: https://azure.microsoft.com/en-us/features/storage-explorer/

This Link will be needed later on in the lab. (Don't click, it won't take you anywhere 😉)

https://patsqlstorage.blob.core.windows.net/?sv=2017-11-09&ss=bfqt&srt=sco&sp=rl&se=2018-11-16T20:17:43Z&st=2018-11-16T12:17:43Z&spr=https&sig=ejhXpG8RJ6KF5Tu2jxT7y%2FXT9nZfGL%2BAVAqTlA4Uzdw%3D

LAB 1: CREATE THE DATA WAREHOUSE AND LOAD A DIMENSION TABLE

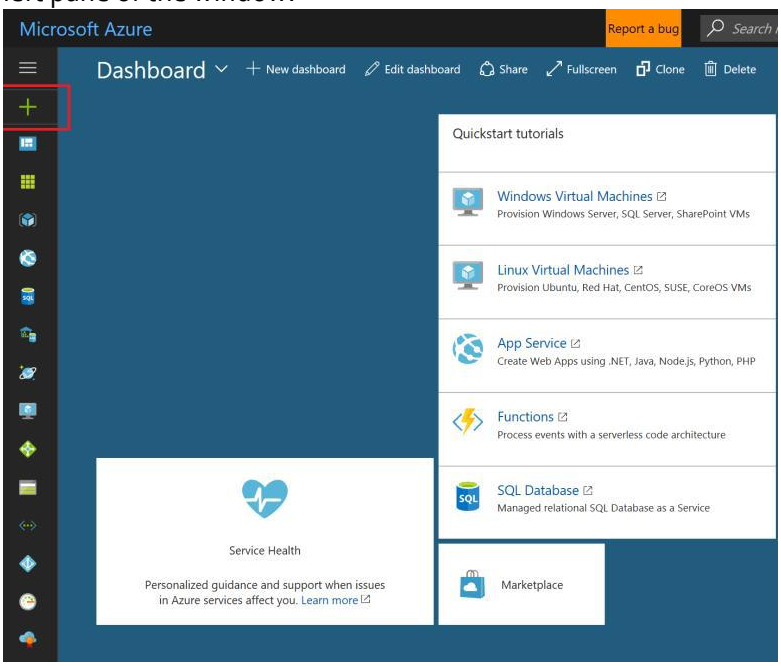
OVERVIEW

This first lab for today will walk you through the creation of a SQL Data Warehouse and the use of Azure Data Factory to fetch a file from the web, that we then will use as a Dimension table in the subsequent labs.

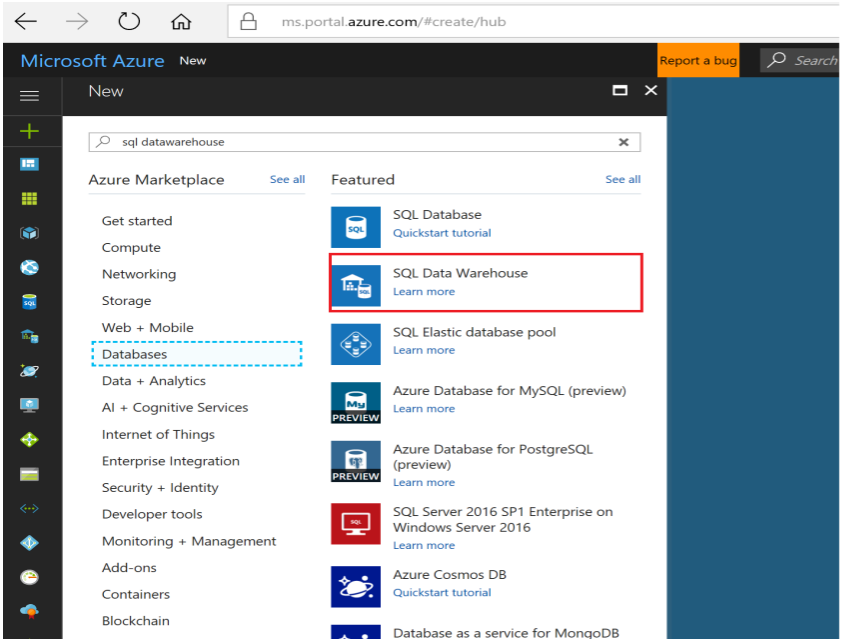
CREATE A SQL DATAWAREHOUSE

In this step we will walk through the creation of a SQL Data warehouse.

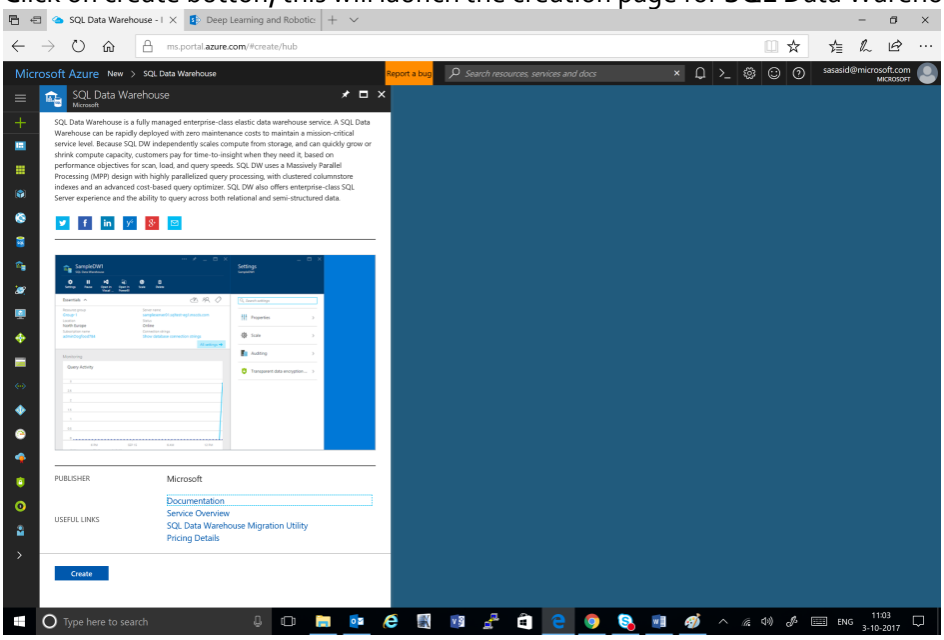
1. Open a browser and go to the Azure Portal: http://portal.azure.com. Please login with your user.
2. Next step is to create a new SQL Data warehouse. Click on the "+" sign which you see on your left pane of the window.



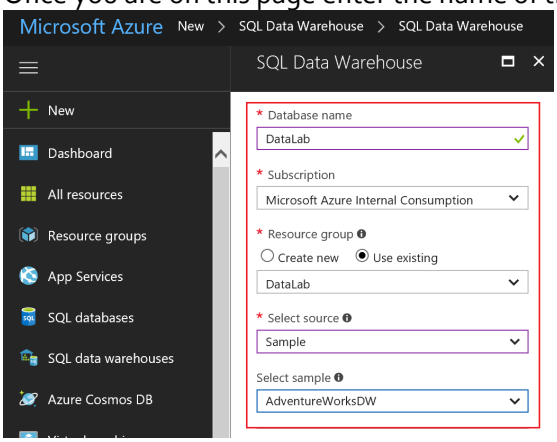
3. Select databases and then click on SQL Data Warehouse.



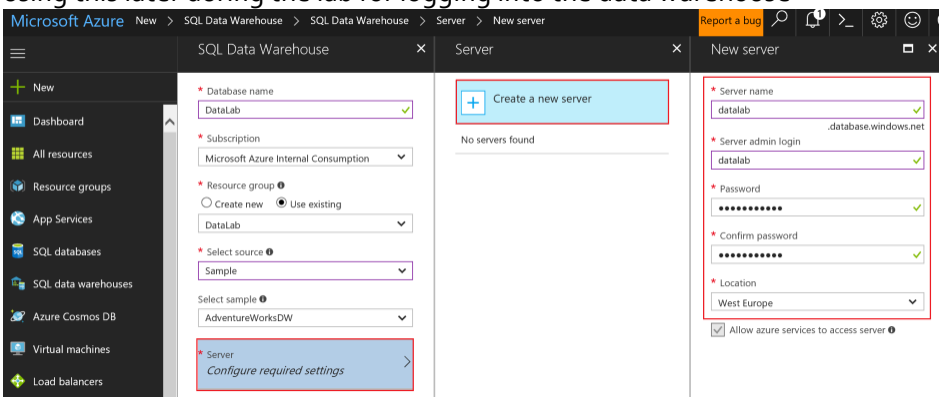
4. Click on create button, this will launch the creation page for SQL Data Warehouse



5. Once you are on this page enter the name of the SQL Data Warehouse you want to create.

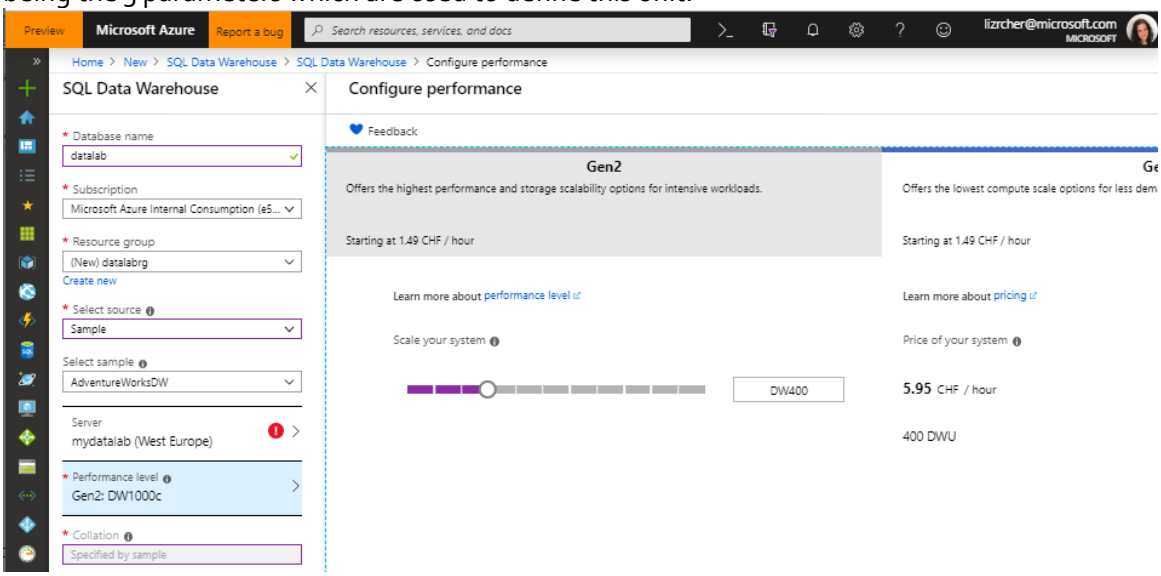


6. In the subscription tab click on the drop down and select the subscription which is listed.
7. For Resource Group click on the "create new" link under the Combo Box and provide a name like "datalabrg".
8. Select Sample in the select source option
9. In the next selection, click on the server option. This will prompt you to create a new SQL Server. Under the new Server tab enter the SQL Server name, password and location as West Europe and Click on Select. Please take a note of the user name and password since we will be using this later during the lab for logging into the data warehouse

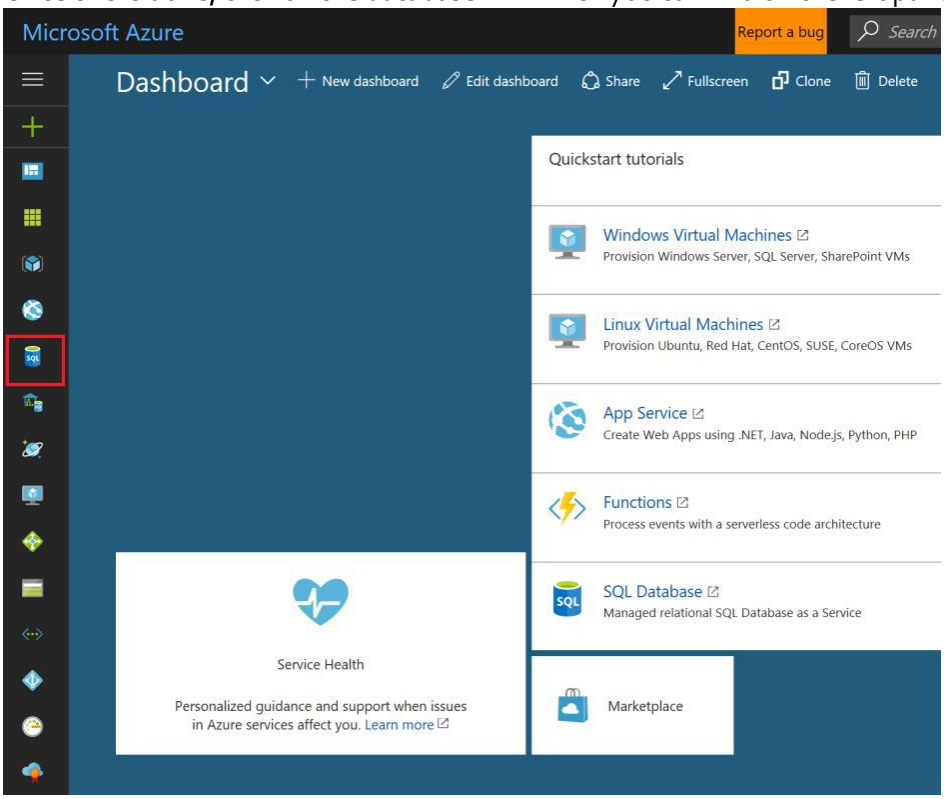


10. In the performance tier select Gen1 and the default of 400 DWU (Data Warehouse Units) for now. This is a measure of performance capacity for Data Warehouse with CPU, IO and Memory

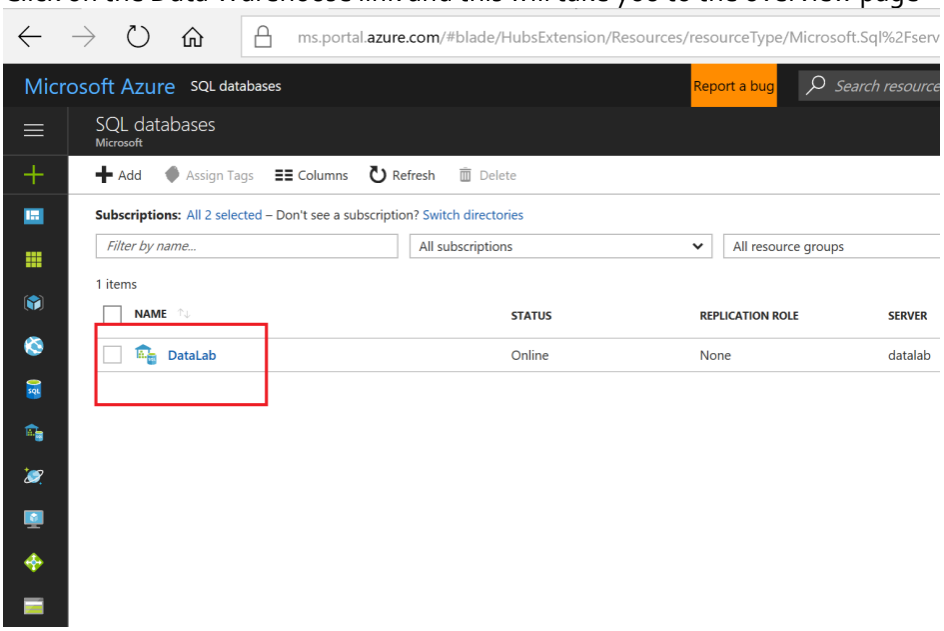
being the 3 parameters which are used to define this unit.



11. On the top right corner of your screen, you will be able to see the notification that the SQL DWH is being deployed. Take your time as this might take some minutes.
12. Once this is done, click on the database link which you can find on the left panel of the portal



13. Click on the Data Warehouse link and this will take you to the overview page

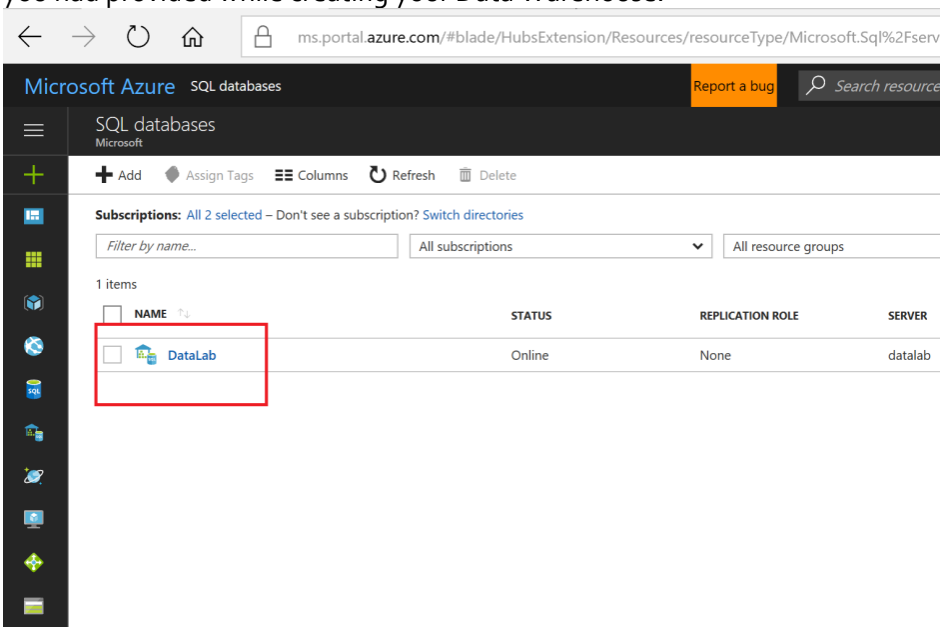


14. You have successfully created a new SQL Data Warehouse.

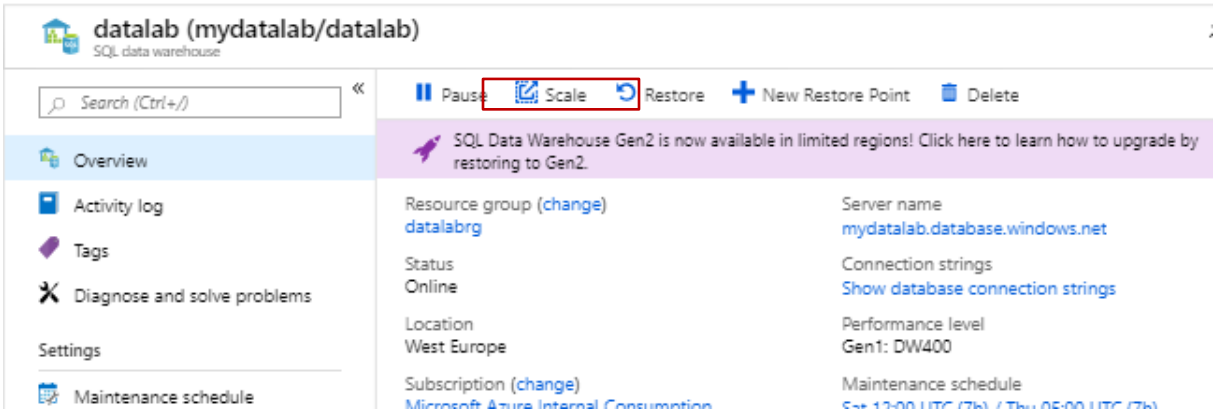
SCALABILITY OF SQL DATA WAREHOUSE IN THE AZURE PORTAL

In this part of the lab we will connect to a Data Warehouse to get some insight in customer data. In this case it is a relational database designed for Big Data (massive parallel processing technology). On Azure there are also other technologies available for analyzing data, like Hadoop, Python or R.
To start, let's explore the scalability features of SQL Data Warehouse, the Azure service is hosting the database.

1. Go to All Resources and when the new pane opens, click on your data warehouse with the name you had provided while creating your Data Warehouse.

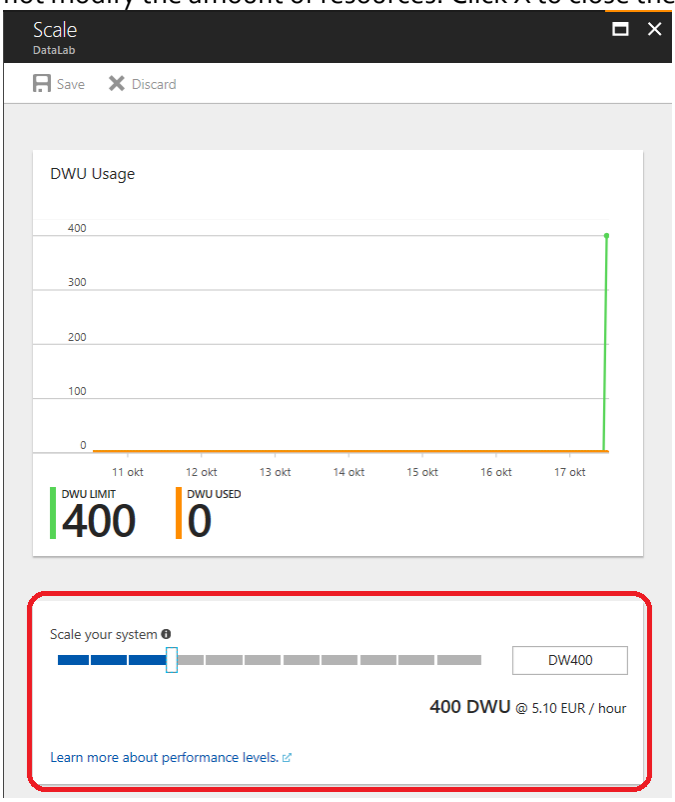


2. Some of the advantages of using Azure SQL Data Warehouse include dynamic scaling and the ability to pause your data warehouse when you are not using it. For example, if you have a period during the day where data is regularly being loaded or processed, you can scale up your data warehouse by increasing the number of DWUs. When the load process finishes, you can scale the data warehouse down by reducing DWUs. Similarly, if there is a time where you will not need compute resources at all, you can pause your data warehouse.
3. Make sure your DWH is running by checking if the status is online. Notice that there is also a button Scale. Click on the Scale button.



4. You will now see a pane where you see the actual usage of the last days. Also, you can increase or decrease the resources of the data warehouse on-demand with a slider. In this lab we'll will

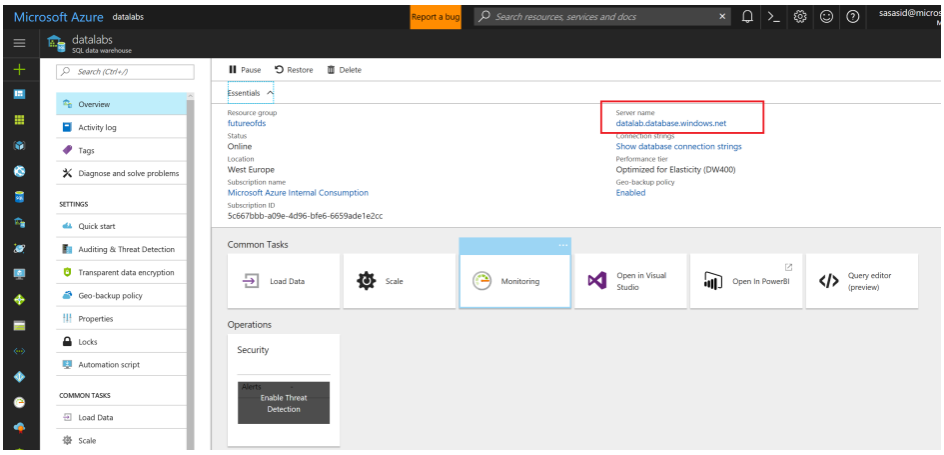
not modify the amount of resources. Click X to close the pane.



Analyze customer data with data engineering tools

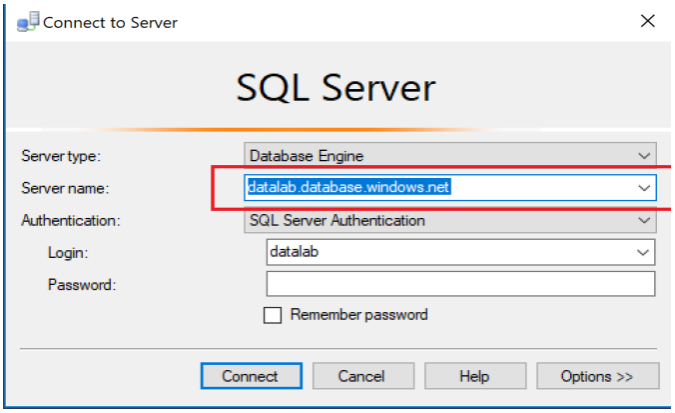
We will execute some queries to the data warehouse by connecting to the Azure SQL Data Warehouse by using SQL Server Management Studio (SSMS). SSMS is usually used by developers and administrators to access SQL Server, Azure SQL Database or Azure SQL Data Warehouse.

1. On the pane displaying the key information about the Azure SQL Data Warehouse, notice that there is a server name. We'll use this server name to connect to the data warehouse. Copy the server name.



2. Go to your desktop in Windows and open the tool SQL Server Management Studio (SSMS).

3. When opening SSMS, you are asked to connect to a server. Fill in the details as described below and click on Connect.

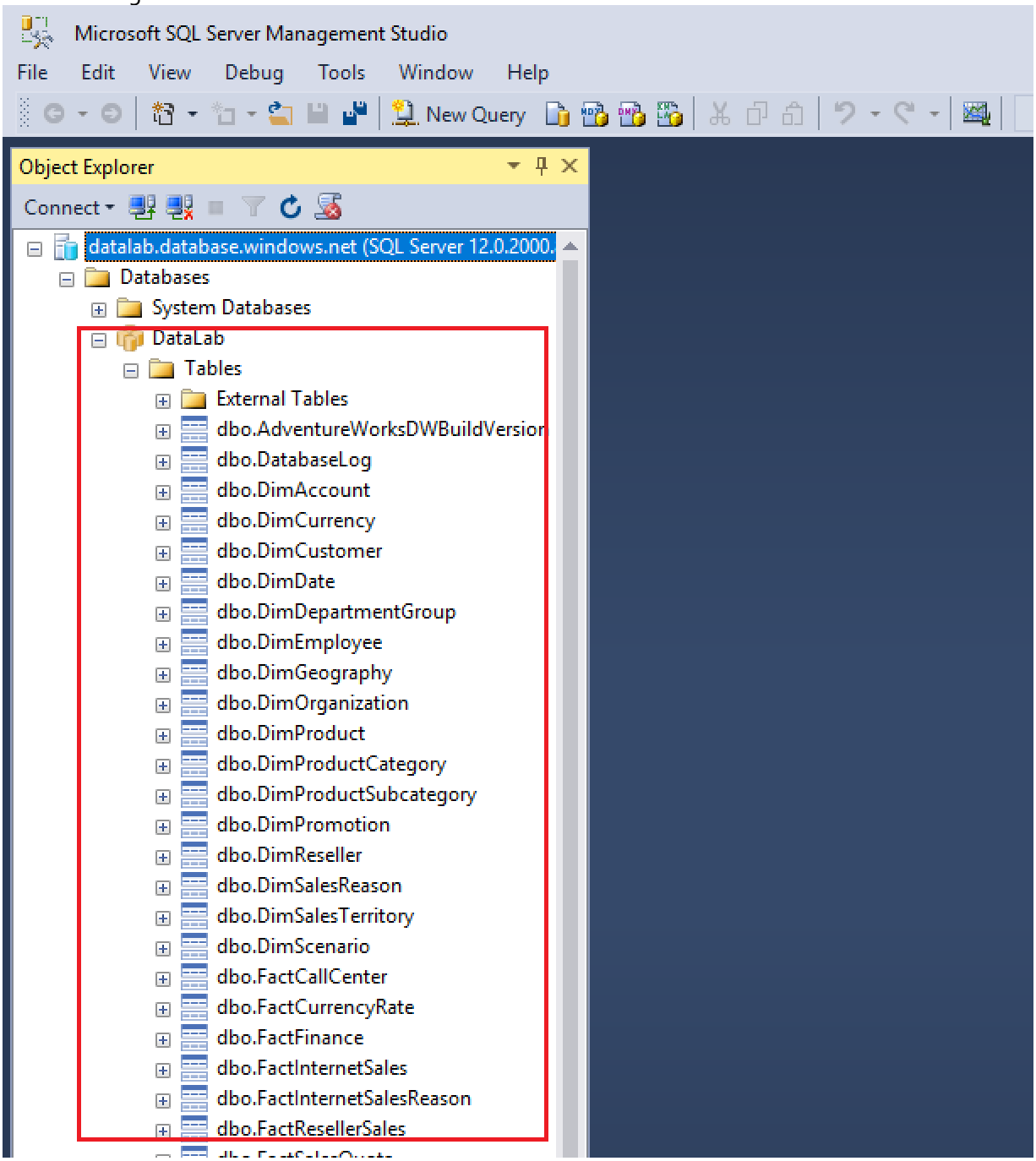


Configuration	Value
Server Type	Database Engine
Server name	[YOURSERVERNAME] (something like datalab.database.windows.net)
Authentication	SQL Server Authentication
Login	[YOURADMINUSER]
Password	[YOURADMINPASSWORD]

If prompted, please on the dialog sign into Azure and then create a new firewall rule by leaving the default of Add my client IP, then click OK. This will create a new rule that adds your IP address, and you should be able to connect to your server.

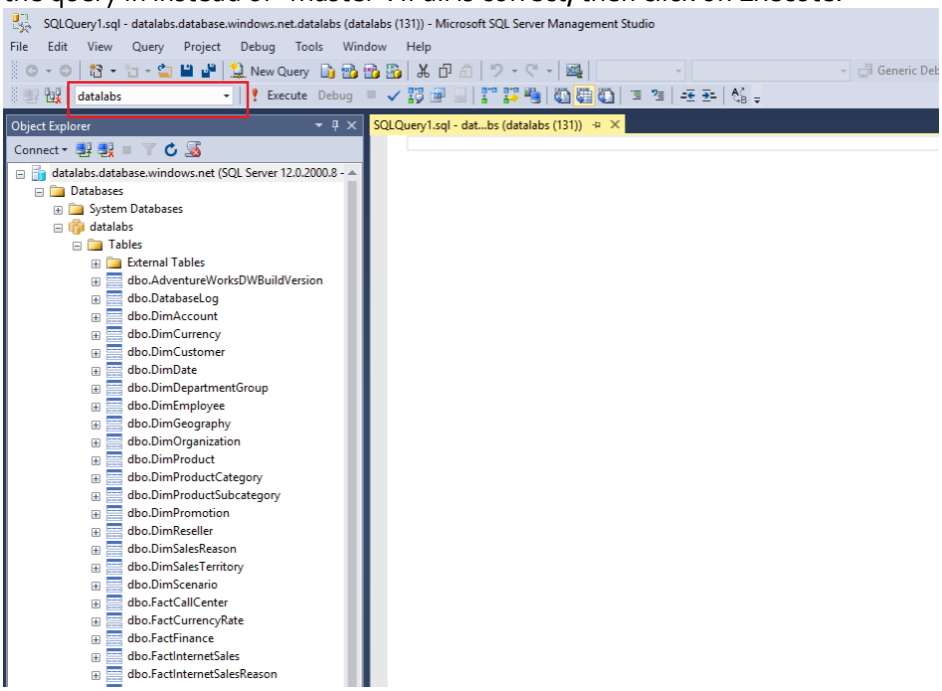
4. When connected, you see in the Object Explorer the databases on the server. When you click on the + the objects in the database will expand. Click on the + of your database and look at the

tables storing customer data that are available in the data warehouse.



5. Let's execute a query against this data warehouse to get some insight in the sales. Click on the New Query button and a new pane will open. Copy and paste the query below in the text window, and make sure that the database name you have used above is select (s. below) to run

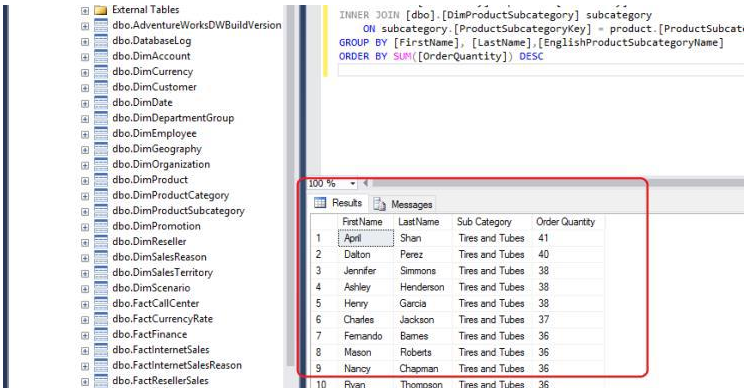
the query in instead of "master". If all is correct, then click on Execute.




```
SELECT [FirstName]
, [LastName]
, [EnglishProductSubcategoryName] [Sub Category]
, SUM([OrderQuantity]) [Order Quantity]
FROM [dbo].[FactInternetSales] sales
INNER JOIN [dbo].[DimCustomer] customer
  ON sales.[CustomerKey] = customer.[CustomerKey]
INNER JOIN [dbo].[DimProduct] AS product
  ON sales.[ProductKey] = product.[ProductKey]
INNER JOIN [dbo].[DimProductSubcategory] subcategory
  ON subcategory.[ProductSubcategoryKey] =
product.[ProductSubcategoryKey]
GROUP BY [FirstName],
[LastName],[EnglishProductSubcategoryName]

ORDER BY SUM([OrderQuantity]) DESC
```

The result of this query shows ordered quantity of products per customer. It will take a couple of seconds to run.

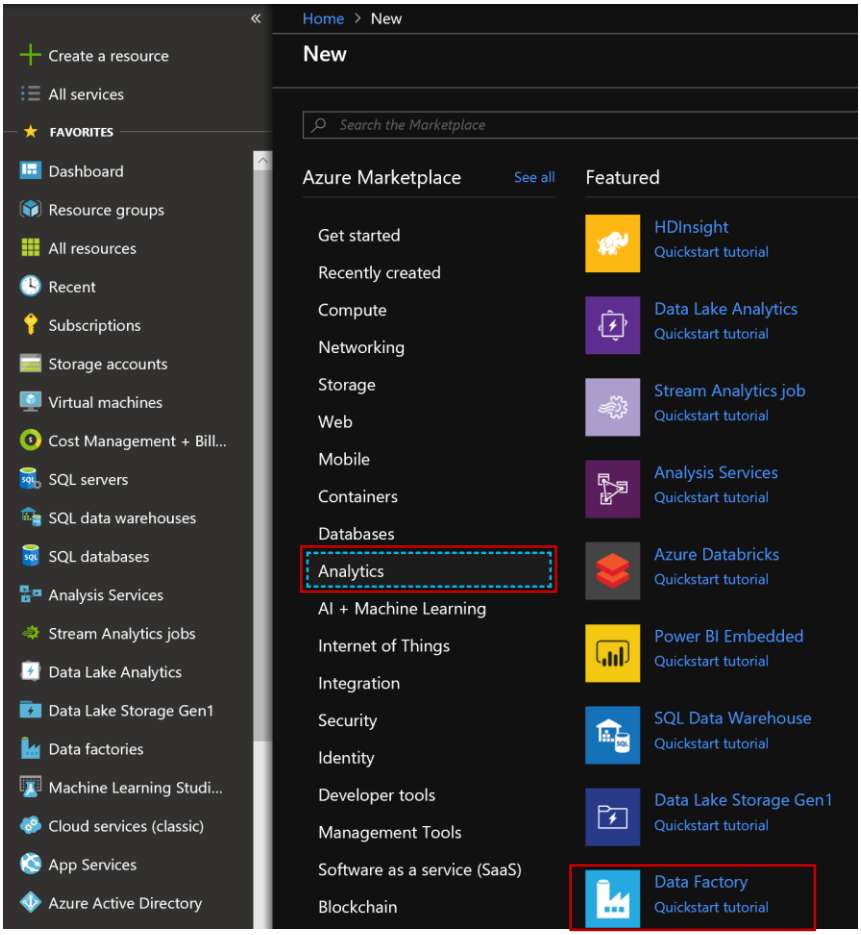


For the following steps in this lab, please create a table with this SQL-Code:

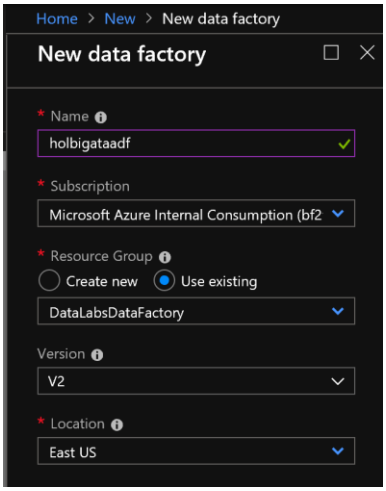
```
CREATE TABLE [dbo].[DimAirports]
(
      [AirportID] [int] NULL,
      [Airport] [nvarchar](100) NULL,
      [City] [nvarchar](100) NULL,
      [Country] [nvarchar](100) NULL,
      [IATA] [nvarchar](100) NULL,
      [ICAO] [nvarchar](100) NULL,
      [Latitude] [nvarchar](100) NULL,
      [Longitude] [nvarchar](100) NULL,
      [Altitude] [nvarchar](100) NULL,
      [Timezone] [nvarchar](100) NULL,
      [DST] [nvarchar](100) NULL,
      [Tz] [nvarchar](100) NULL,
      [AirportType] [nvarchar](100) NULL,
      [DataSource] [nvarchar](100) NULL
)
WITH
(
      DISTRIBUTION = REPLICATE,
      CLUSTERED COLUMNSTORE INDEX
)
GO
```

CREATE AN AZURE DATA FACTORY AND LOAD A FILE FROM THE INTERNET INTO OUR SQL DATA WAREHOUSE

1. Go to your Azure portal and click +, choose Analytics then choose Data Factory



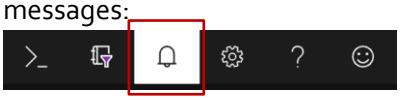
2. Enter a unique name like 'holbidataadf'
3. Choose the resource group you have created in the last sequence for your Data Lake Analytics and Store Services:



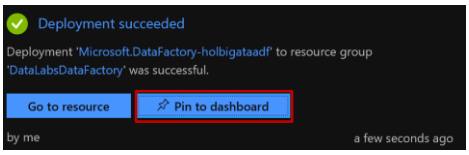
4. Choose Version: 'V2'

5. Choose the location 'West Europe' or the corresponding location, where you have created your Azure SQL Data Warehouse

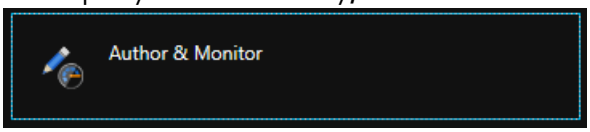
6. Hit 'create' and wait for the service to be displayed. After creation, the Bell Sign will show new messages:



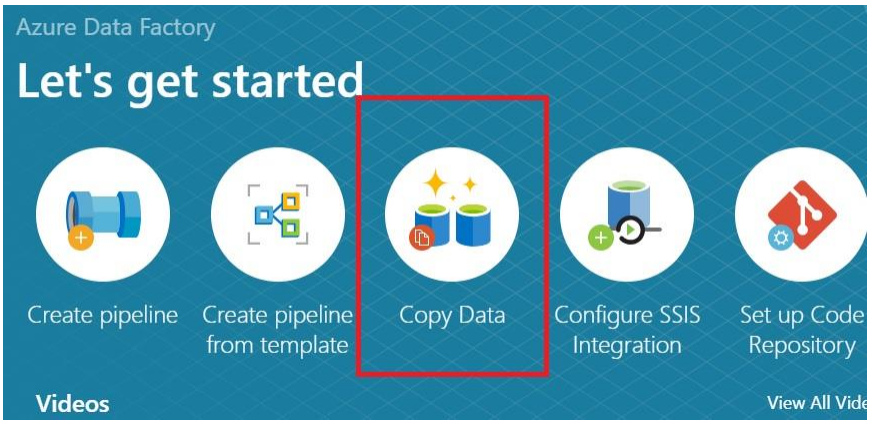
7. Click on it, choose the Deployment-Success message for your Data Factory and click 'Pin to dashboard'



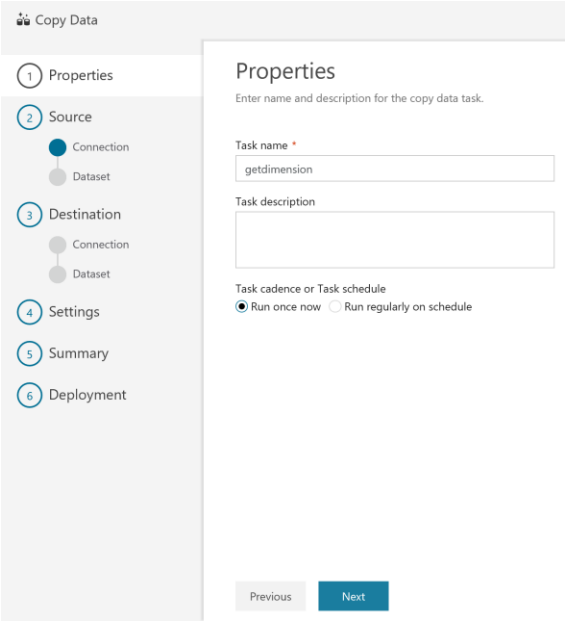
8. Now open your Data Factory, search for 'Author and Monitor'



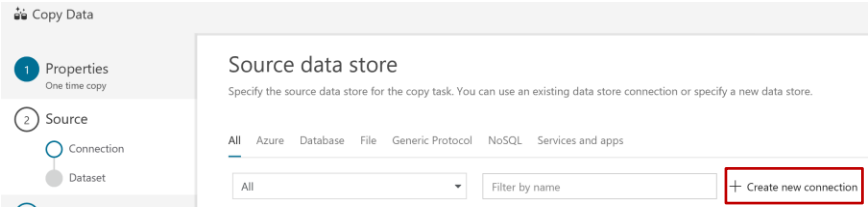
and in the new tab (takes a short amount of time) start the editor: 'Copy Data':



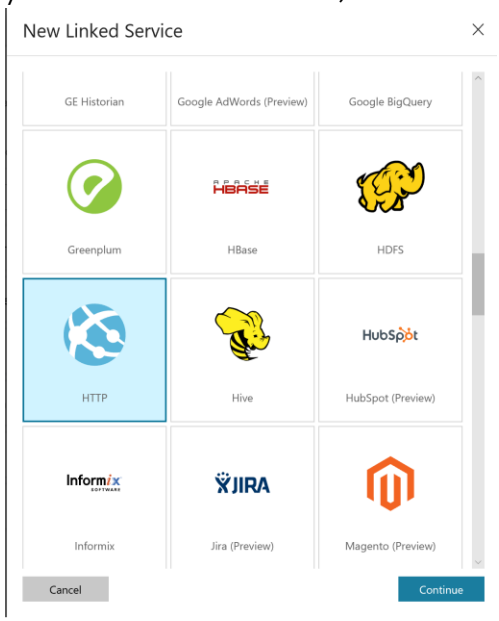
9. On the first screen name your data copy pipeline like 'holbigdatagetdimension' or just leave the default name and click 'Next'



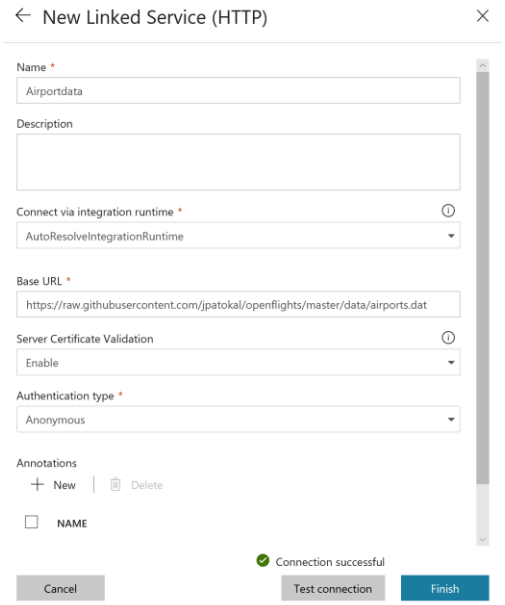
10. On the Source data store screen hit '+ Create new connection':



11. Now the 'New Linked Service' picker appears. Please choose "Http" (at the top of the window you even can search for this) and hit 'Continue'

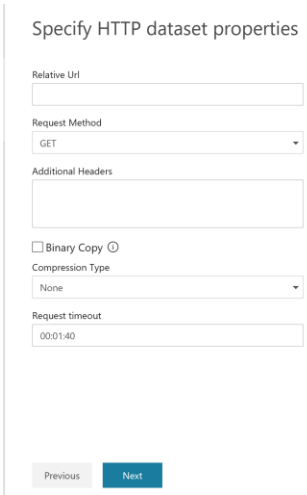


12. On the following dialog name the Linked Service and then please enter this URL in the field "Base URL": https://raw.githubusercontent.com/lzurcher/BigDataHoL/master/airports.dat
In the "Authentication type" field select "Anonymous"



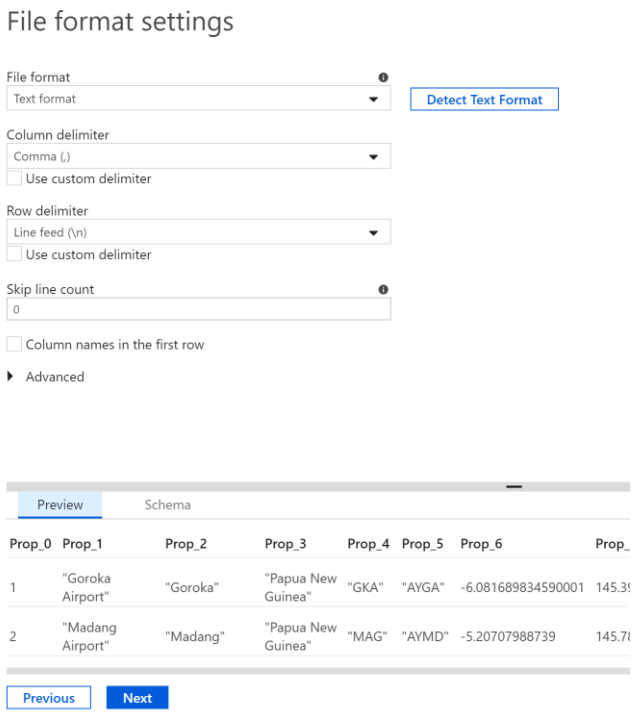
Test your connection. If it says "Connection successful" you can click "Finish" and proceed to the next step.

13. On the "Specify HTTP dataset properties" you don't need to change settings. Please just proceed with "Next"

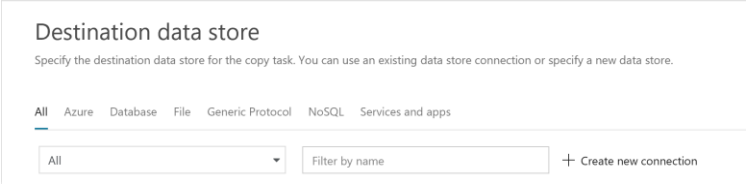


14. In the "File format settings" please choose "Text format" in the "File format" property and hit "Detect Text Format" to let the service decide what column delimiter, linebreaks, etc. are used.

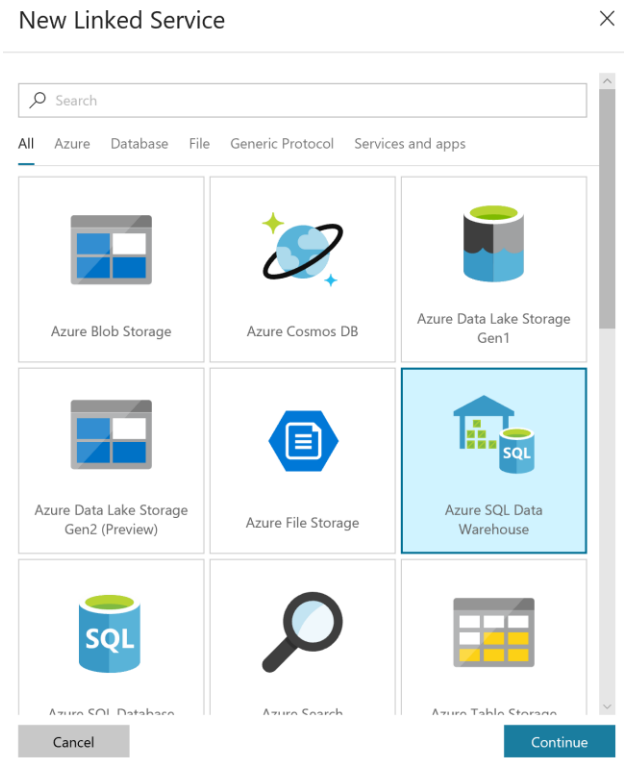
Otherwise you might enter these values yourself. Hit "Next"



15. Please click "Next" and create a new connection to your Azure SQL Data Warehouse that you have created in the steps above. ➔ "Create new connection"

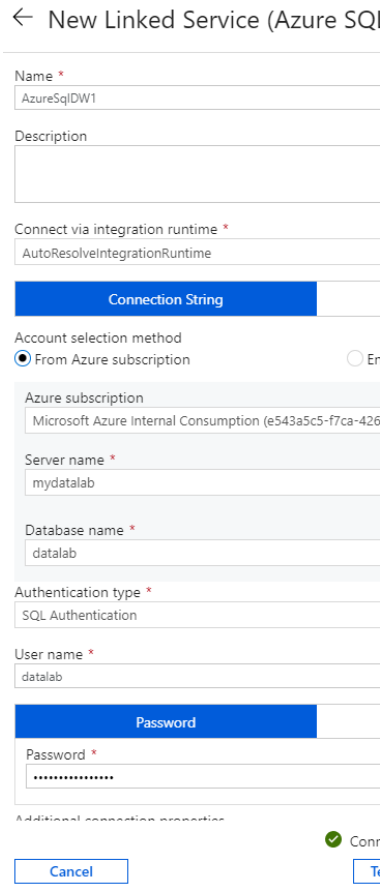


16. Please select "Azure SQL Data Warehouse" and hit "Continue"

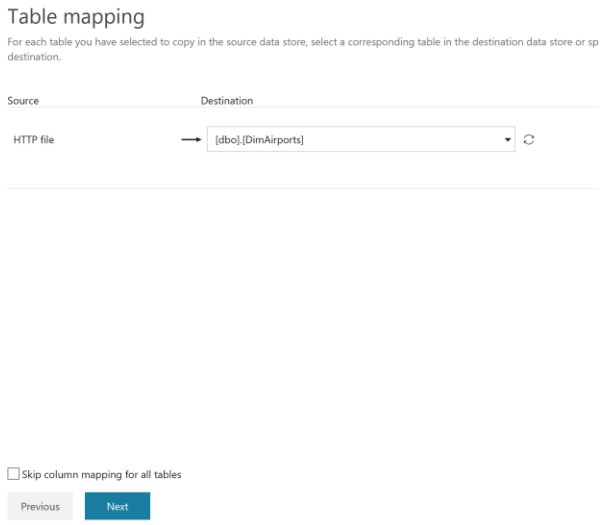


17. Name the new LinkedService after your SQL DW or in a manner that you can identify it afterwards. Use "From Azure subscription". Then select your subscription, the SQL DW Server and enter user name and password, that you have created.

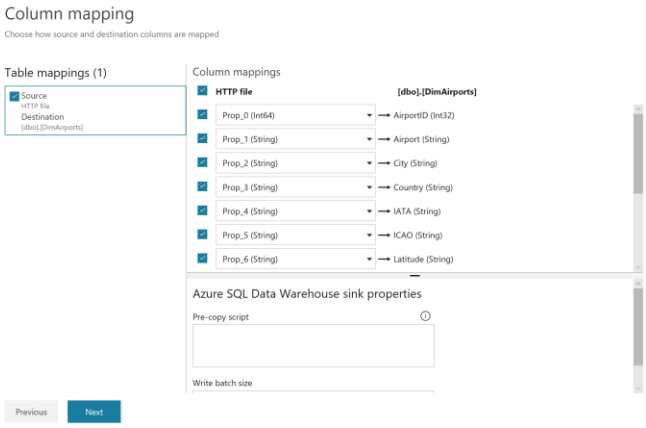
Test your connection.



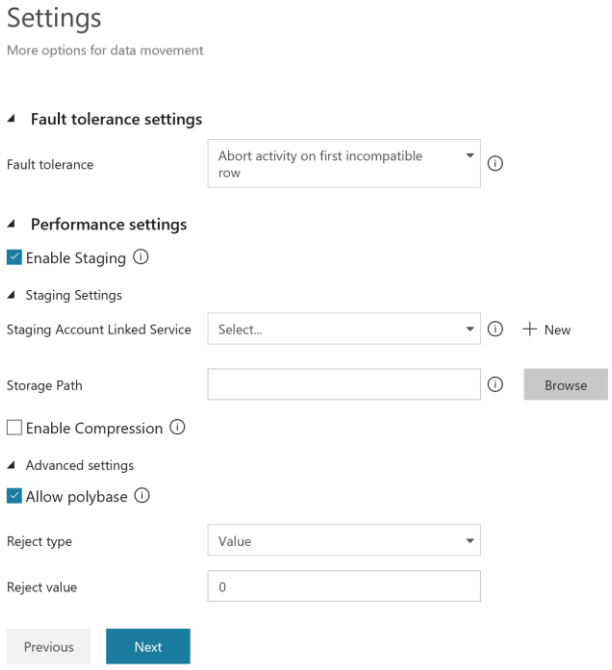
18. On the "Destination Data Store" your new Linked Service shows up and is selected. Hit "Next". You are taken to the "Table mapping" dialog.
    Select the newly created table that you have created in the former step above.



19. The "Column mapping" dialog gives you the chance to map the source columns from the file to the target columns in the database table. You can leave this as is. The columns should map in the right order in this case. Hit "Next".

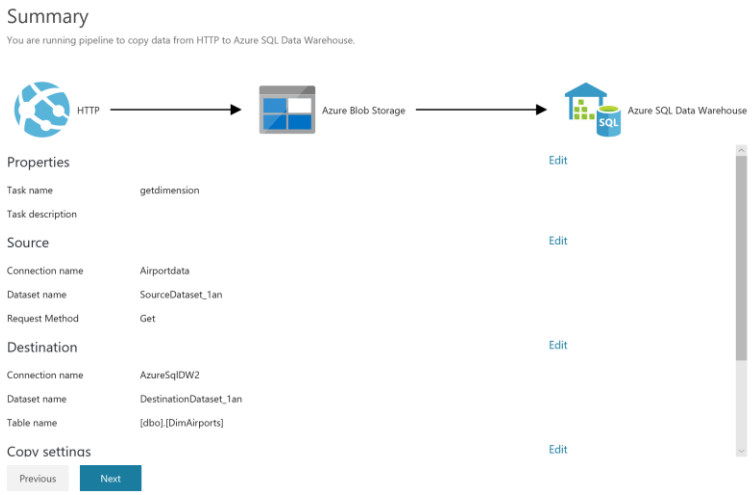


20. The "Settings" dialogue will appear and gives you the chance for final adjustments. We want to allow Azure Data Factory and Azure SQL DW to use Polybase to increase the speed when loading into the DB. So therefore please create a new Storage account, that will be used for staging during loads into the target database. Data Factory will do this job for you.
So please click " + New" or if you already have a storage account, select one in the drop down box.

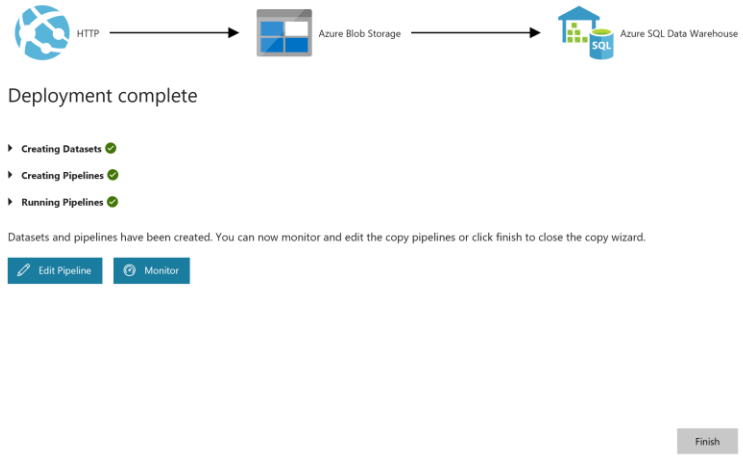


21. In the next and last step, a "Summary" about the newly created pipeline is shown. If everything is setup to your needs and requirements, you might hit "Next" and the pipeline will be deployed

and executed.



22. You might jump into the monitoring of your Data Factory to collect information about the pipeline run and check, if there where errors in the execution.



23. You could now get back to the SQL Server Management Studio and check the content of the table. (select * from dbo.DimAirports)

LAB 2: CREATING AND LOADING A DATA LAKE GEN 2

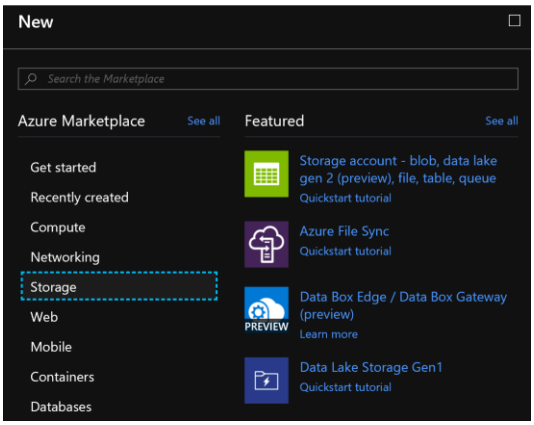
OVERVIEW

In this Lab you will create a Data Factory Pipeline, that will load data from airdelay statistics into your then newly created Azure Data Lake Gen2.
This data will then be cleansed and pre-aggregated for analysis and Machine Learning using a Databricks Spark-as-a-Service Cluster.

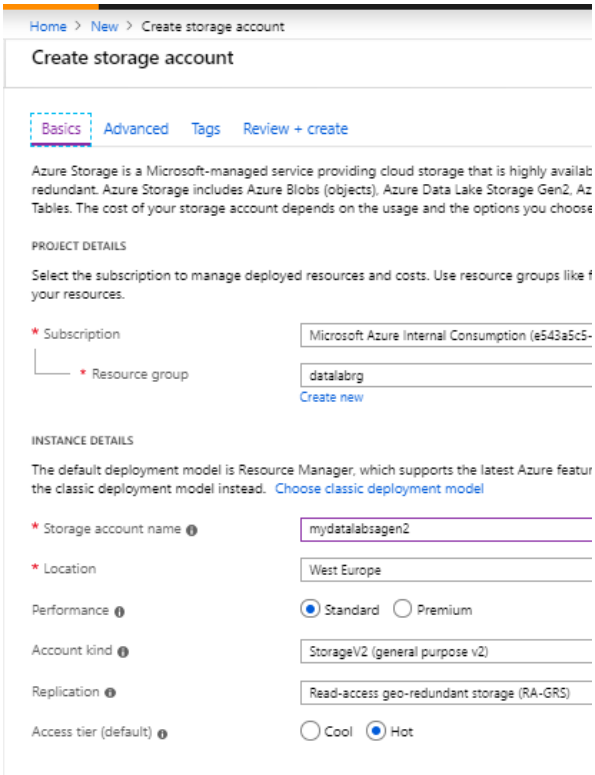
PRE-LOAD YOUR DATA TO DATA LAKE

First we will create a Storage Account, that will hold your file data.

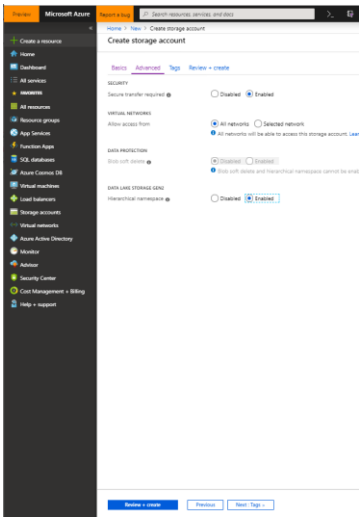
1. Please go to the portal, click '+ Create a resource' and on the Storage – Tab select 'Storage Account'



2. On the next Blade please name Storage Account, choose the resource group that you have created above and choose 'West Europe' as your selected region:

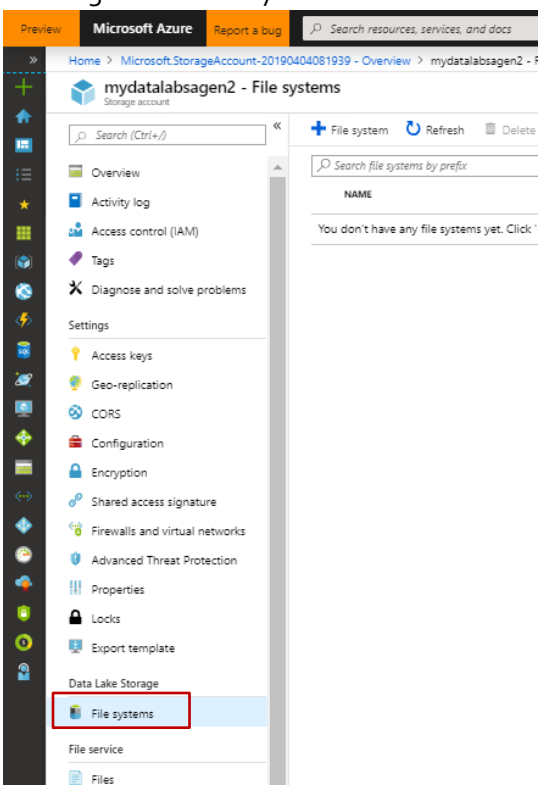


3. In the Advanced tab make sure you check Hierarchical namespace "Enabled" to enable a Data Lake Storage Gen2



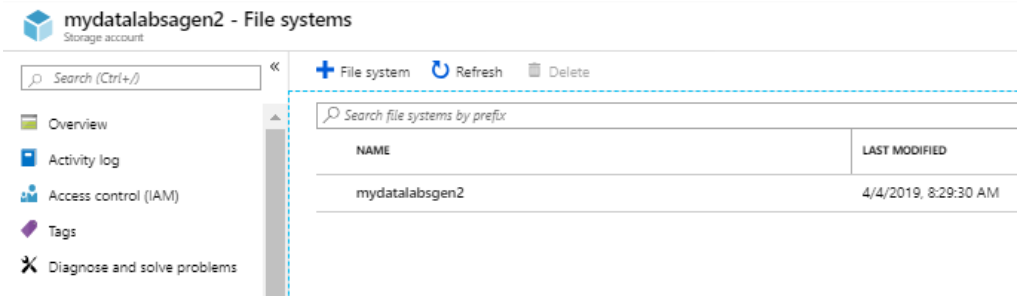
4. When you hit create it will take some minutes and the service will appear available in your resource group.

5. Please go to the newly created Data Lake Storage Gen 2 and hit 'File systems'

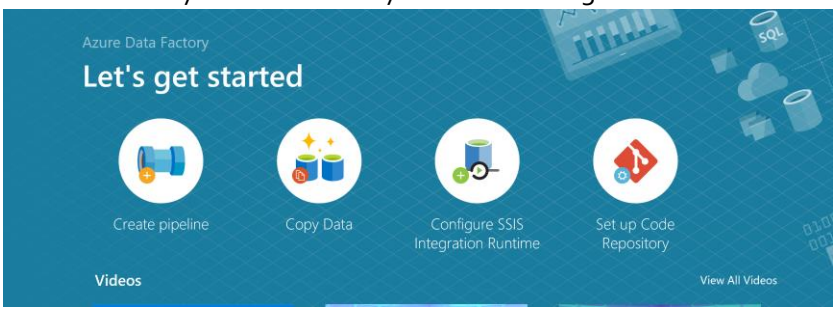


6. Add file system

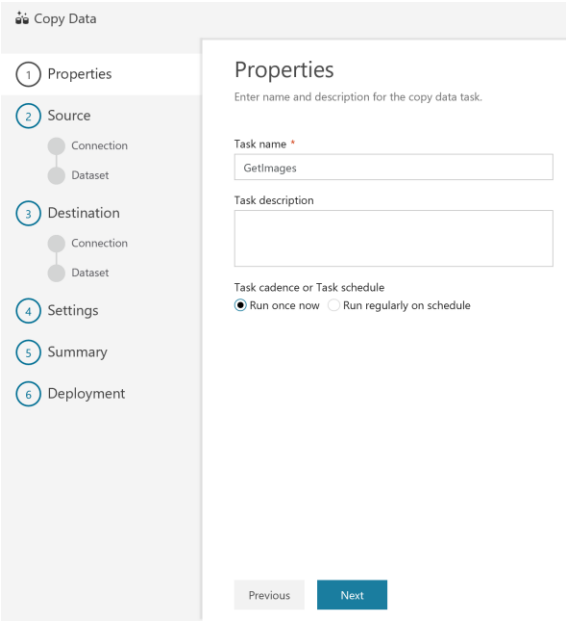
7. Create a new folder to hold your airdelay-files:



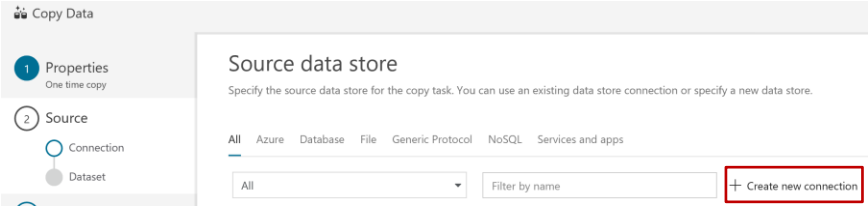
8. Now switch to your Data Factory environment again and start a new 'Copy Data' Wizard:



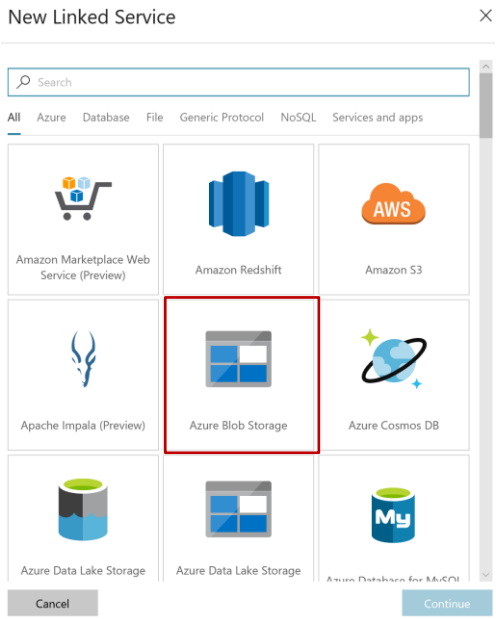
9. On the first screen name your data copy pipeline like 'holbigdataairdelay' or just leave the default name and click 'Next'



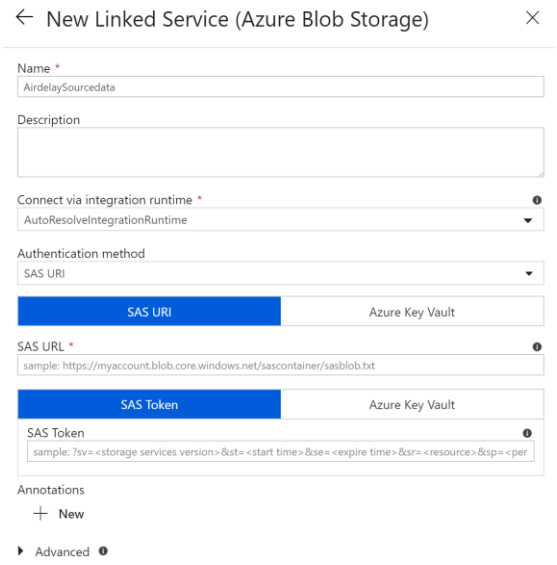
10. On the Source data store screen hit '+ Create new connection':



11. Now the 'New Linked Service' picker appears. Please choose Azure Blob Storage and hit 'Continue'



12. On the following screen select 'Use SAS URI' as 'Authentication Method' and paste the following into the SAS URI-Field:

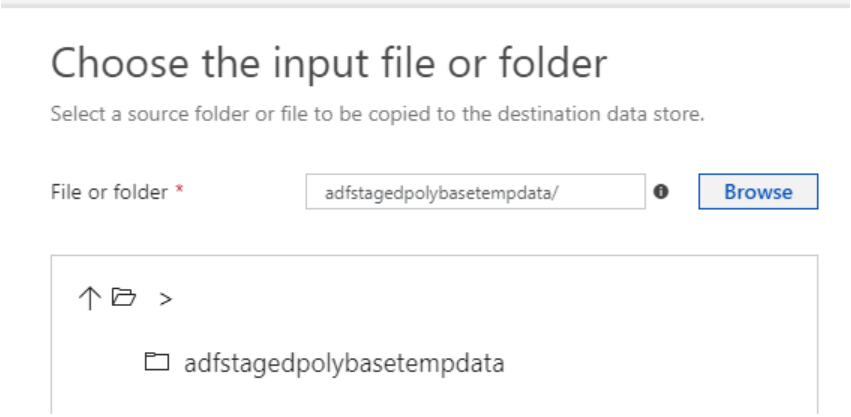


SAS URL
https://mydatalabstorage.blob.core.windows.net/

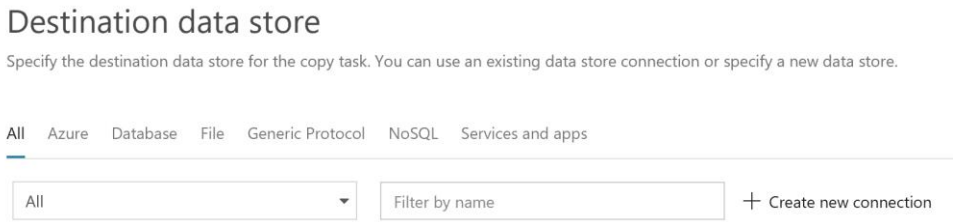
SAS Token
?sv=2018-03-28&ss=bfqt&srt=sco&sp=rwdlacup&se=2019-04-04T14:35:44Z&st=2019-04-04T06:35:44Z&spr=https&sig=FXycESAApN9PrP6fdpDXPm78l8Hx1Hp90%2BOFOW4e2hY%3D

The 'Choose input file or folder' selection is displayed. First click the checkbox 'Binary Copy', then please select 'Browse'

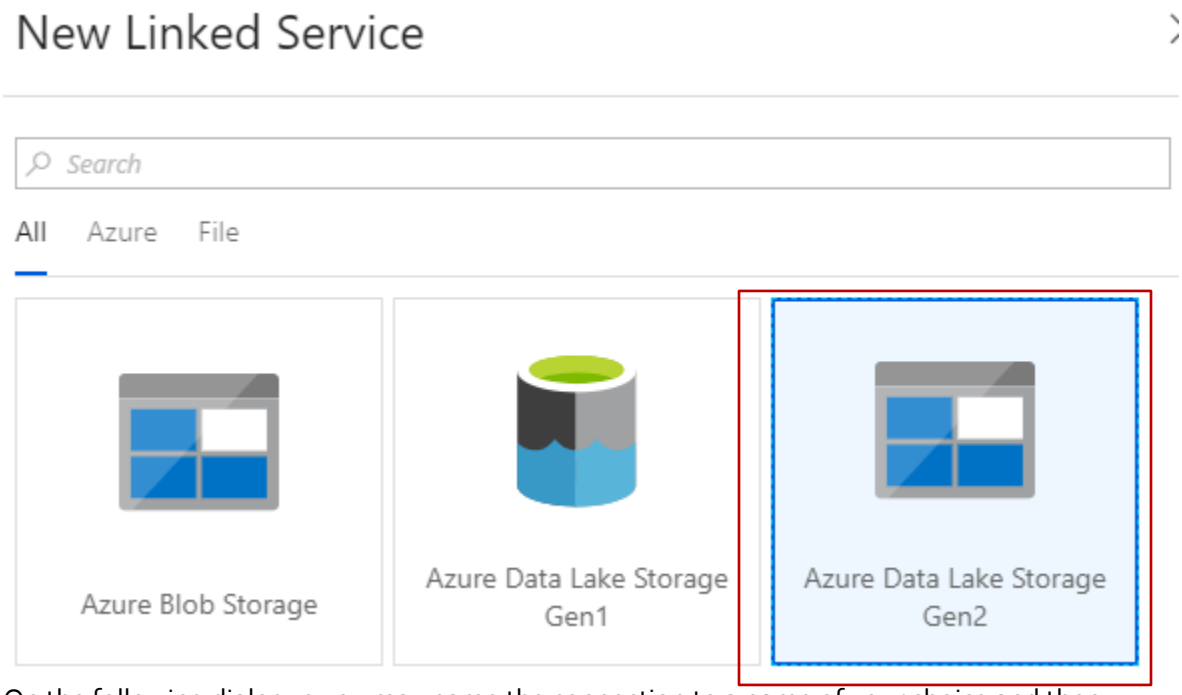
13. On the following screen choose the folder 'airdelays' (and really click on 'Choose' to make the section active. Then click 'Next'



14. On the 'Destination data store' dialogue we again create another connection:



15. Please select 'Azure Data Lake Storage Gen2' and hit 'Continue'

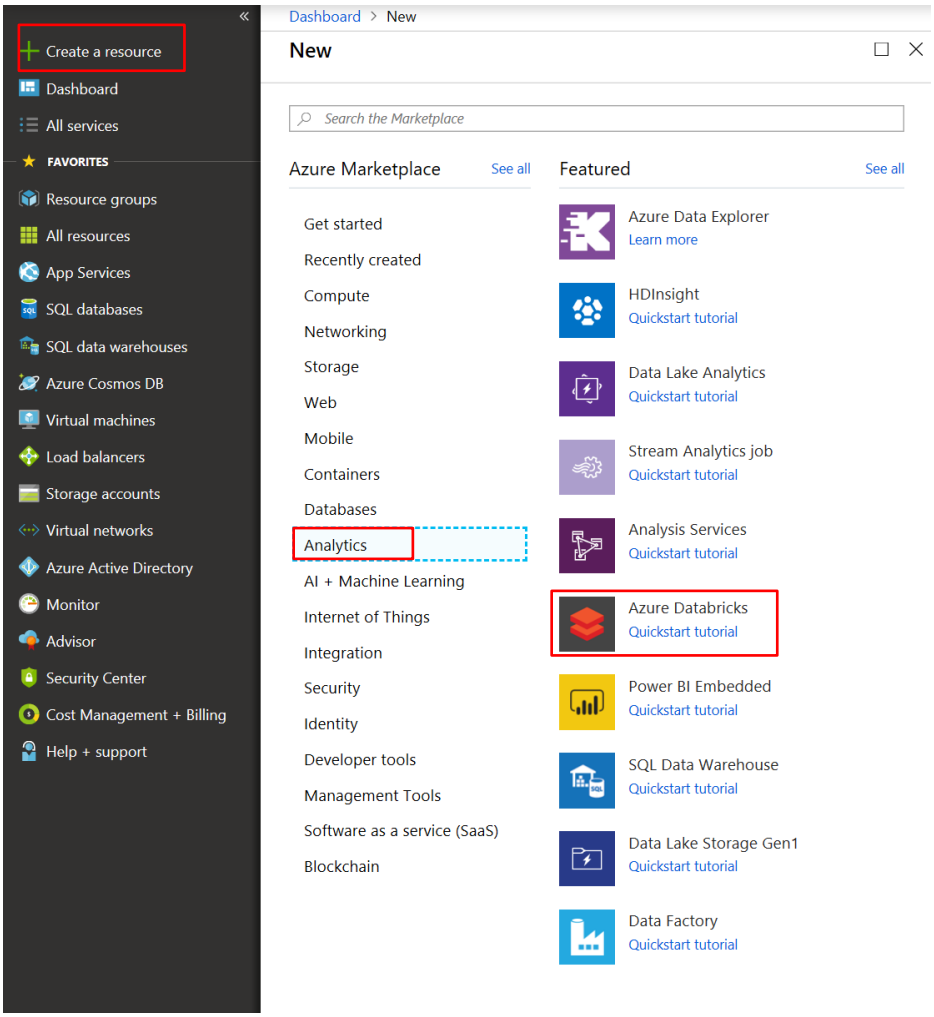


16. On the following dialogue you may name the connection to a name of your choice and then select the following entries:
    - Select your Azure Subscription from the dropdown box
    - select the Data Lake Storage Gen2 account name that you have created above
    Then hit 'Finish'

LAB 3 CONNECT WITH STORAGE EXPLORER TO SEE THE STORAGE ACCOUNTS.

Let's create a Databricks workspace to manipulate and analyze the data using Spark.

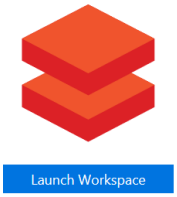
Go in the Azure Portal and create a new Azure Databricks resource.



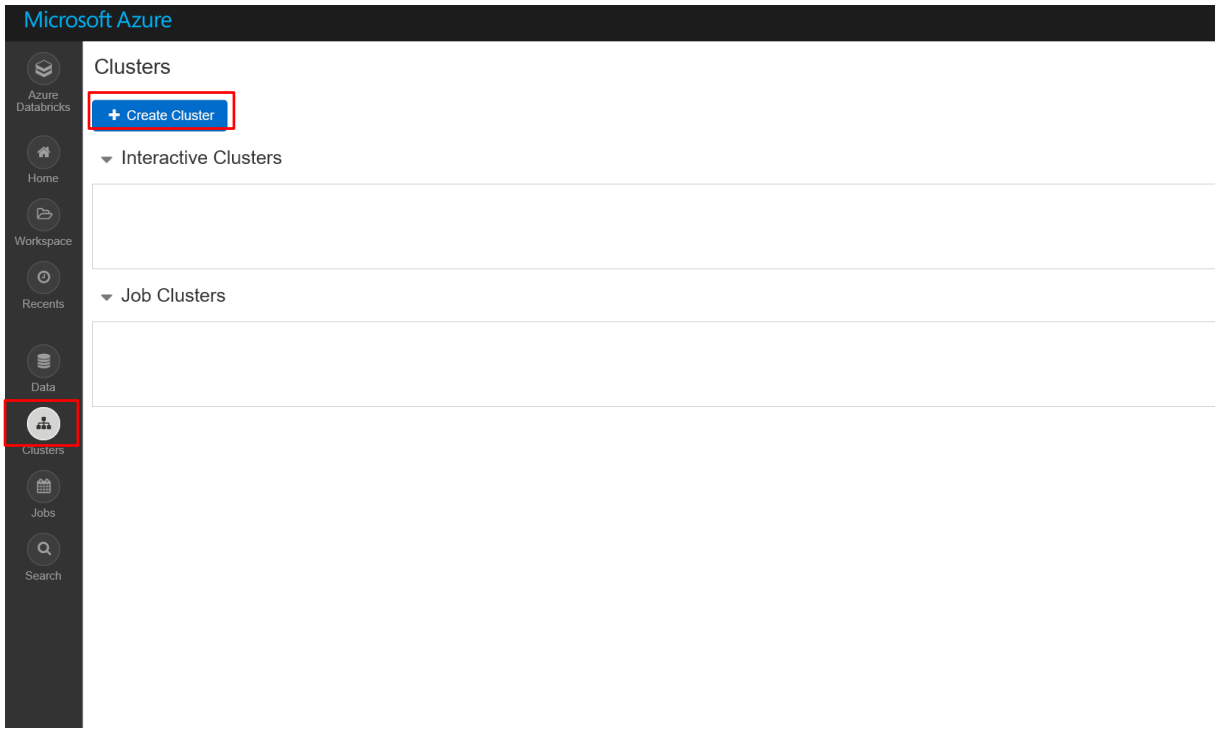
Configure it with:

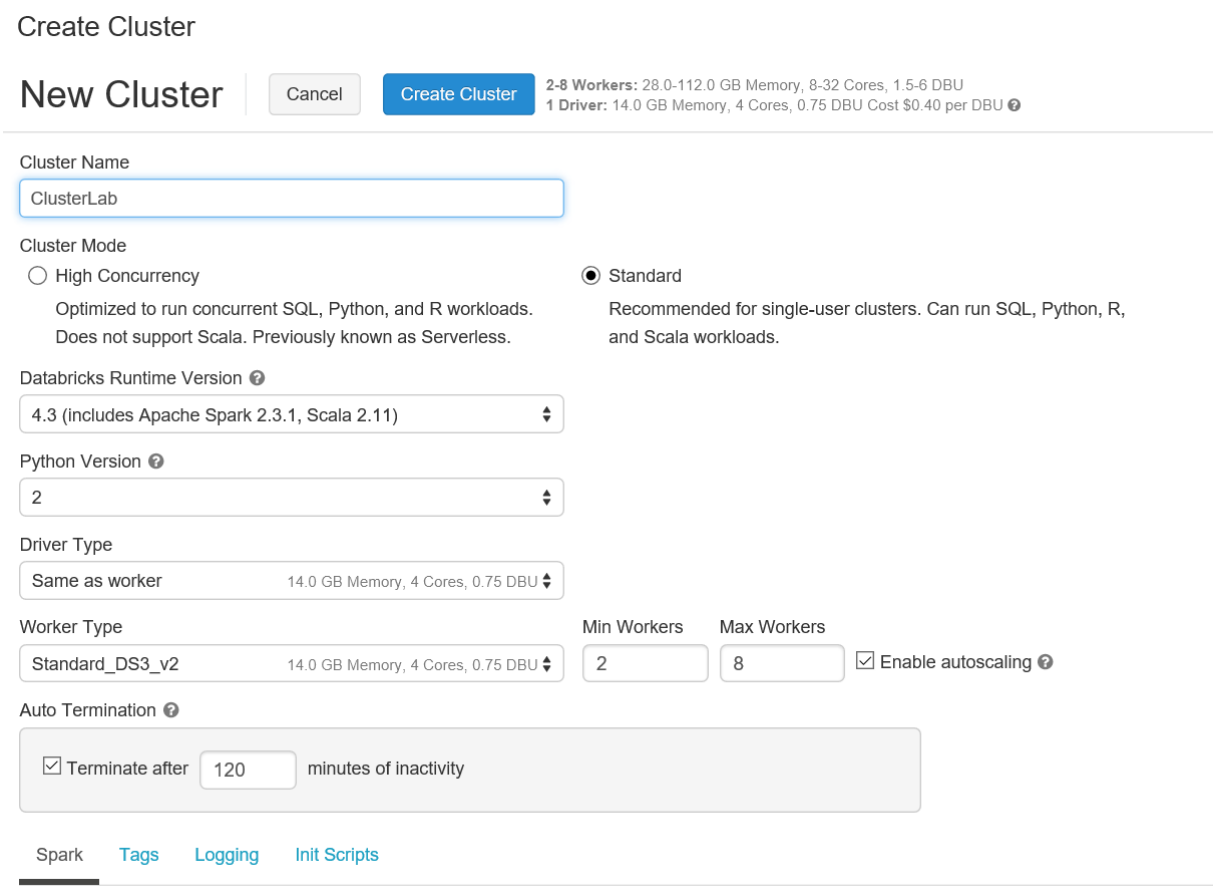
- A name
- Your subscription
- The resource group you previously created
- Location: North Europe
- Choose the Standard pricing tier

When the resource is created, click on *Launch Workspace*.


Launch Workspace

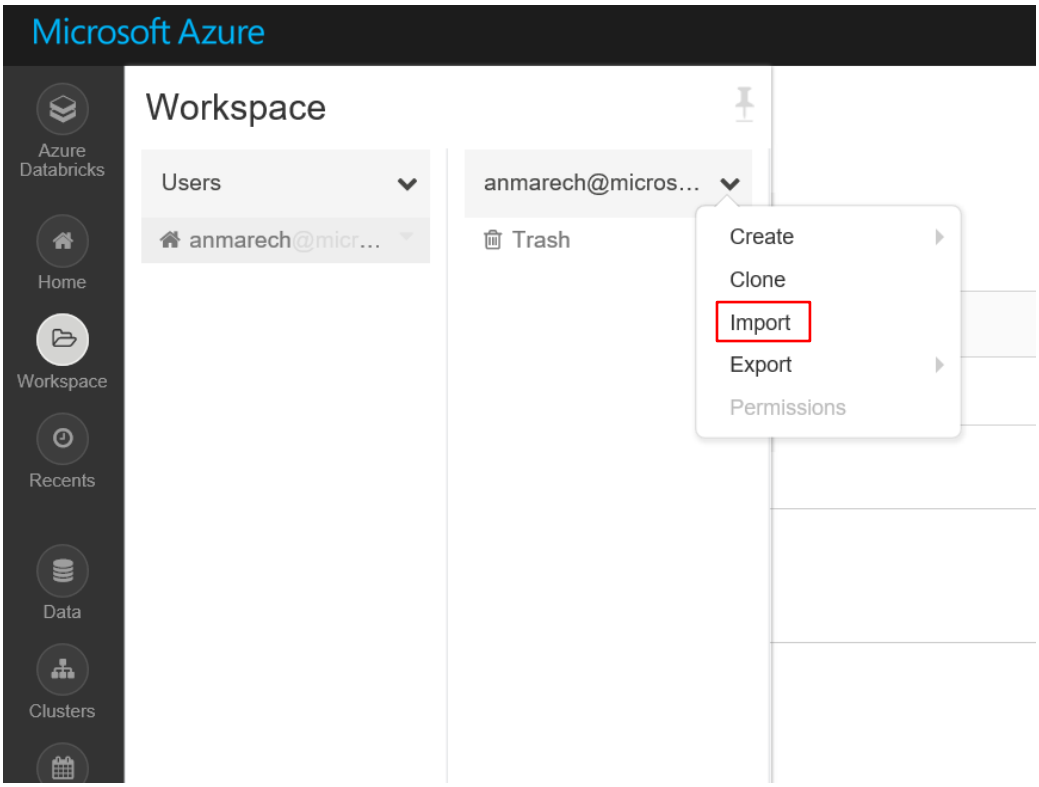
Go in the *Clusters* tab and click on *Create Cluster*.





Give a name to your cluster. You can leave the other properties as they are.

Once your cluster preparation is running, explore the Databricks environment and import the notebooks in your workspace.



For the next steps, let's run the Databricks notebook.

LAB 4

Can you create a Cosmos DB using SQL API and insert the json file from https://github.com/lzurcher/BigDataHoL ? Once inserted, copy the document and insert a new one. Add the property "my_favorate_song" to the file and give it the value of your favorite song. 😉

TERMS OF USE

DISCLAIMER

This lab contains only a portion of the features and enhancements in Microsoft Azure Data Factory, Azure SQL Data Warehouse, Azure DataBrics and Azure Data Lake Storage. Some of the features might change in future releases of the product.