

Open in app ↗



Search

Write



# Deploying the Databricks Labs Overwatch Project in Azure



Ryan Chynoweth

7 min read · Jun 2, 2022



5



2



Note #1: please be aware of the license associated with all Databricks labs projects.

Note #2: Please check out Databricks System tables. We are adding datasets and making them available to customers with practically zero effort.

## Introduction

As a solutions architect at Databricks I am part of the Field Engineering organization. Our focus is on sales activities to help our customers solve their problems and democratize data within their company. Many (if not all) of the field organization are very talented engineers with various backgrounds and expertise. This leads to our Databricks Labs project which is driven by individuals who are not dedicated engineers at Databricks, but see a problem and develop solutions to solve those problems. These

solutions are available on our [Databricks Labs Project on GitHub](#), please note that these projects do not necessarily have production support or SLAs.

Here are a handful of projects available in the Databricks Lab GitHub repositories that I would recommend checking out:

- `overwatch`
- `terraform-provider-databricks`
- `tempo`
- `dbx`
- `migrate`

In this article I would like to focus on my experience deploying the [Overwatch](#) project in Azure for a single Databricks environment. Note that this was my first time deploying Overwatch and I hope that this serves as a resource for others doing the same.

To add perspective, it took me approximately four hours. This included reading all the documentation (majority of my time), creating the resources, and deploying the Databricks job that collects the data.

## Deploying Overwatch

In Azure, the general steps to set up overwatch are as follows:

1. Create an Azure Databricks workspace.
2. Create an Azure Event Hub Namespace.
3. Create an Event Hub within the namespace.

4. Create a dedicated Azure Data Lake Gen2 storage account (ADLS). Note — this storage account can be shared across workspaces as well, but it is recommended to do it by region.
5. Create a container within the storage account.
6. Set up diagnostic logging for Azure Databricks so that the logs are streamed through the event hub in step 3.
7. Create a “default” cluster policy that all users must use to enforce cluster logs to be sent to ADLS. Note that this doesn’t have to be the only policy but all cluster policies in the workspace should enforce cluster logging for accurate reporting.
8. Create the Overwatch job using the jar or notebook.
9. Begin analyzing Databricks log data.

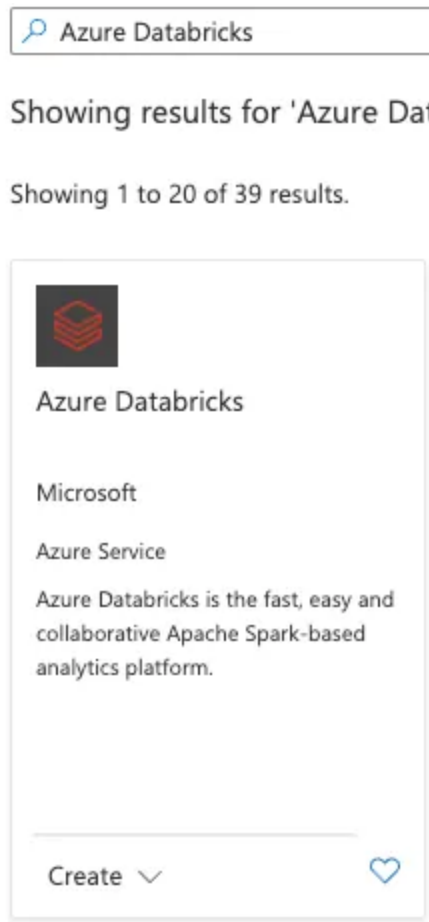
I will break these steps down into the following sections:

- Resource creation
- Resource configuration
- Data model

## Creating the Resources

### Deploy Azure Databricks

To deploy Azure Databricks you will log into the [Azure Portal](#), and click “Create a resource”. Then you will search for “Azure Databricks” which you can then click on to create. Obviously skip this step if you already have a workspace.



Configure Databricks as needed by choosing to bring your own virtual network and if you would like No Public IPs.

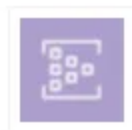
## Deploy an Azure Event Hub

Now you will need to click “Create a resource” once again. Then you will search for “Event Hubs”.

[Home](#) > [Create a resource](#) >

## Event Hubs

Microsoft



### Event Hubs

[Add to Favorites](#)

Microsoft

★ 4.2 (95 Azure ratings)

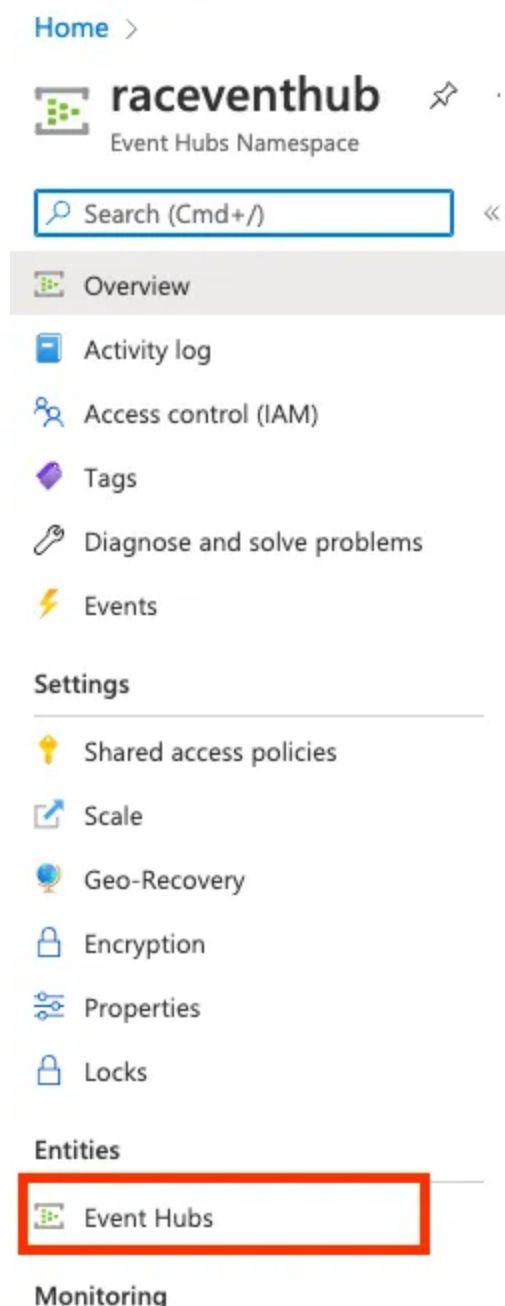
Plan

Event Hubs

[Create](#)

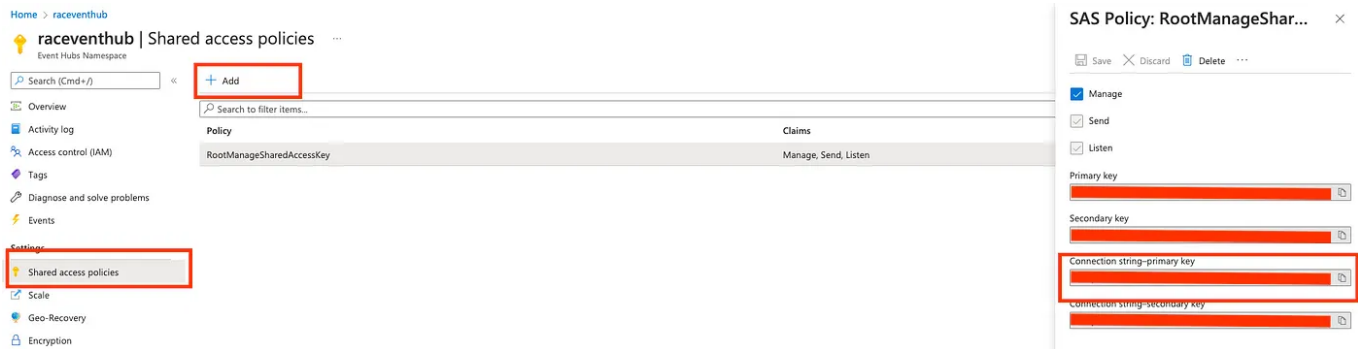
Creating an Event Hub Namespace is pretty straightforward, however, the two biggest concerns for deployment will be throughput units and pricing tier. Larger Databricks deployments (or sharing an Event Hub between workspaces) will require more throughput units. The basic pricing tier will require you to run the Overwatch job in Databricks twice a day while the standard tier allows for less often. Read more on the difference between tiers [here](#).

Next you will need to create an Event Hub within the Event Hub Namespace. To do so, navigate to “Event Hubs” in the left panel.



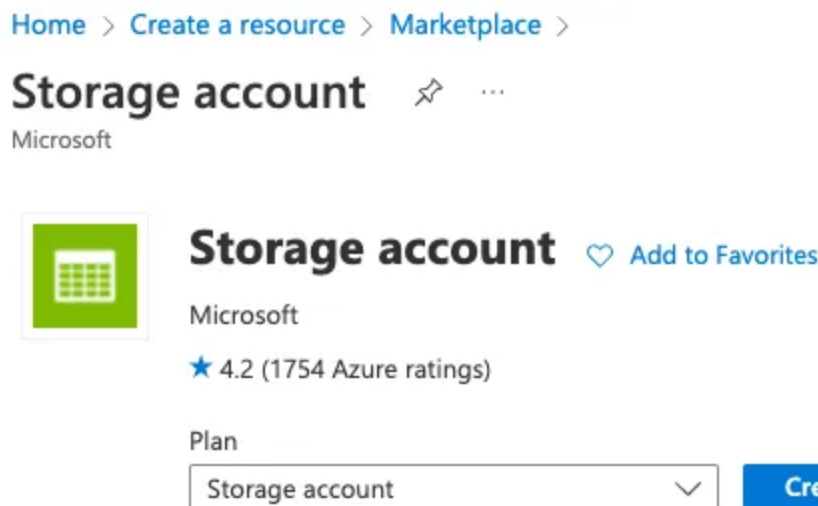
Then you can simply click the “+ Event Hub” button and create an Event Hub.

Next you will want to create a shared access policy, which is available in the left panel. Once you add the policy with “Listen” permissions, you will want to copy the “Connection string-primary key”. Please ignore the fact that my policy has “Manage” permissions in the screenshot below.



## Create an Azure Data Lake Gen2 Storage Account

Our last resource we need to deploy is an Azure Data Lake Gen2 storage account. To do so, you will need to click “Create a resource” once again, then you will search for “Storage Account”.



When creating you can select your configuration as needed, however, make sure you select “Enable hierarchical namespace” in the advanced section.

### Data Lake Storage Gen2

The Data Lake Storage Gen2 hierarchical namespace accelerates big data analytics workloads and enables file-level access control lists (ACLs). [Learn more](#)

Enable hierarchical namespace



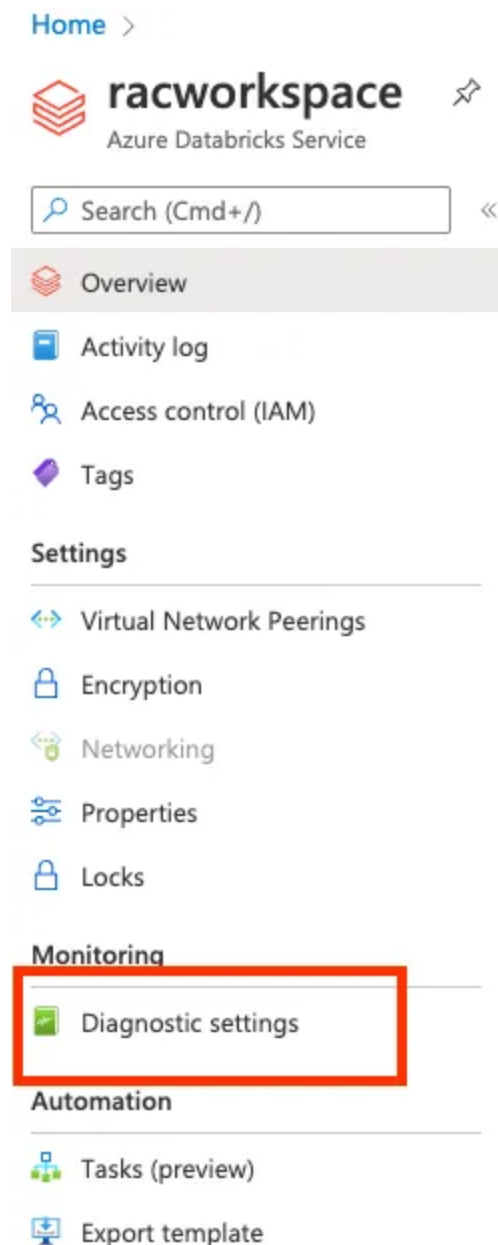
Lastly, create a container within your Azure Data Lake Gen2 storage account.

## Configuring the Resources

Now that we have deployed all the required resources, we now need to configure the resources to collect and model our data.

### Diagnostic Settings

Once deployed you will need to configure diagnostic logging on the workspace. To do so click “Diagnostic settings” in the left panel.





Then add a setting to send your logs to an Event Hub by choosing the “Stream to an event hub” destination. Below is a list of the settings I streamed to the Event Hub. I did a subset of logs but you can also choose to send “All Logs” if you wish. I selected the following as these were the datasets that I was most interested in analyzing:

- dbfs
- clusters
- accounts
- jobs
- notebook
- ssh
- workspace
- secrets
- sqlPermissions
- sql analytics
- instancePools
- databrickssql
- deltaPipelines
- repos

[Home](#) > [racworkspace](#) >

## Diagnostic setting

[Save](#)
[Discard](#)
[Delete](#)
[Feedback](#)

A diagnostic setting specifies a list of categories of platform logs and/or metrics that you want to collect from a resource, and one or more destinations that you would stream them to. Normal usage charges for the destination will occur. [Learn more about the different log categories and contents of those logs](#)

Diagnostic setting name: Overwatch

### Logs

Category groups ⓘ

☐ allLogs

Categories

☒ dbfs☒ clusters☒ accounts☒ jobs☒ notebook☒ ssh☒ workspace

### Destination details

☐ Send to Log Analytics workspace☐ Archive to a storage account☒ Stream to an event hubFor potential partner integrations, [click to learn more about event hub integration](#).

Subscription

field-eng

Event hub namespace \*

raceventhub

Event hub name (optional) ⓘ

rac\_overwatch

Event hub policy name

RootManageSharedAccessKey

Image of streaming logs to an event hub

Once you click save all your logs will be sent through the Azure Event Hub.

## Cluster Logging and Policies

The next step in the process is that you will need to send your cluster logs to ADLS Gen2. This requires configuration on all clusters that are deployed in your account. The best way to do this is through cluster policies. My recommendation would be to create a “default” cluster policy that everyone uses. It is still possible to create additional cluster policies to enforce other group requirements, but at the very least you should have a policy that contains the following. Note that you can also use an “abfss://” path which would likely be a better option for direct access to ADLS.

```
{
  "cluster_log_conf.path", "dbfs:/mnt/path/to/adlsgen2"
```

```
}
```

Please note that in the UI cluster logging would appear like the image below, the “0414-204835-wl2i2hxx” is the cluster id that is automatically added on as a suffix to the path provided:

The screenshot shows the Databricks cluster configuration page for a cluster named 'my\_cluster'. The 'Configuration' tab is selected, showing settings for autoscaling, termination, worker types, and driver type. The 'Logging' tab is also visible at the bottom. The destination path for logs is set to 'dbfs:/mnt/overwatchclusterlogs/clusterlogs/0414-204835-wl2i2hxx'.

Clusters / my\_cluster

**my\_cluster**

Configuration Notebooks (0) Libraries Event log Spark UI Driver logs Metrics All

☐ Enable autoscaling

☒ Terminate after 240 minutes of inactivity

Worker type 14 GB Memory, 4 Cores Workers 1 ☐ Spot instances

Standard\_DS3\_v2

Driver type 14 GB Memory, 4 Cores

Standard\_DS3\_v2

DBU / hour: 1.5 Standard\_DS3\_v2

Advanced options

Azure Data Lake Storage credential passthrough

☐ Enable credential passthrough for user-level data access

Spark Tags **Logging** Init Scripts JDBC/ODBC Permissions

Destination

dbfs:/mnt/overwatchclusterlogs/clusterlogs/0414-204835-wl2i2hxx

✓ Last delivered at 2022-05-25 13:57:12 PDT

Cluster

## Databricks Overwatch Job

The last configuration step in Azure Databricks is to deploy the job that transforms your log data into a data model. To do this you will need to create a job within a Databricks workspace that either executes the jar or the

notebook. Note that you will need the cluster dependencies below to run the job in Azure.

```
com.microsoft.azure:azure-eventhubs-spark_2.12:2.3.21
```

I decided to run the job as a notebook which you will need to provide the widget values below:

- ETL Storage Prefix: this is the location of your cluster logs
- ETL Database Name: this is the schema/database that is used to store the raw data and transform the log data into the ERD
- Consumer DB Name: this is the reporting schema that you will use to analyze the data
- Secret scope: the scope that contains your Databricks personal access token and your event hub key
- Secret Key (DBPAT): key value to your Databricks personal access token
- Secret Key (EH): key value to your event hub access token
- EH Topic Name: the name of your event hub topic
- Primordial Date: the date you wish to pull data from
- Max Days: the total number of days you wish to pull for the job
- A1. Scopes: the data namespaces that you wish to transform. This likely corresponds to the diagnostic logs that you set for your workspace

The screenshot shows the 'OverwatchNotebook' interface with a 'Scala' tab. The top bar includes a 'Schedule' button and a 'Share' button. Below the bar, there are nine configuration fields:

1. ETL Storage Prefix	2. ETL Database Name	3. Consumer DB Name	4. Secret Scope	5. Secret Key (DBPAT)	6. Secret Key (EH)	7. EH Topic Name	8. Primal Date	9. Max Days
/mnt/overwatchclusterlogs/	overwatch_etl	overwatch_target	overwatch	pat	ehKey	rac_overwatch	2022-04-01	60

Below these fields, there is a section for 'A1. Scopes' with a text input containing 'audit,sparkEvents,jobs,cluste'.

## Widgets for Overwatch Job

To run the job as a jar, please use the following maven coordinate.

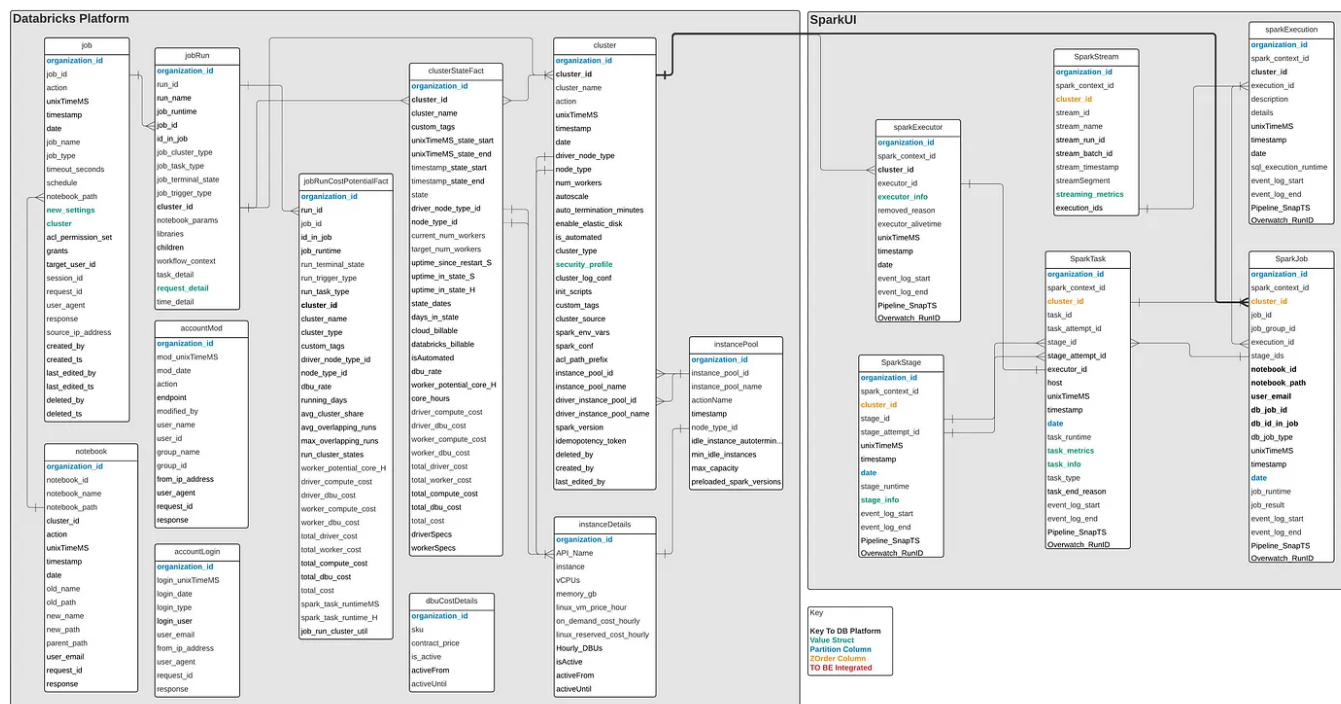
```
com.databricks.labs:overwatch_2.12:<latest>
```

In the end, you will end up with two Overwatch databases. The first is used for ETL work and the second is the “gold” tables that are transformed and should be used for reporting.

## Data Model

Overwatch is essentially a solution accelerator for analyzing your Databricks log data. All this data is available to you, but Overwatch allows you to easily analyze your data with a standard data model. The data model below is what overwatch provides but you can easily add on additional datasets as you see fit. My point of view is that the job, jobRun, clusterStateFact, and cluster tables hold the most interesting data in terms of governing Databricks assets. The SparkUI portion will be most important to understand cluster utilization and optimizations.

Overwatch, Gold 0.6.0  
David Torres | December 13, 2023



Overwatch ERD

On thing to note is that you should not use Overwatch for cost/pricing purposes. Overwatch tracks DBUs very well but you need to hard code your pricing dollar figure into the notebook/jar. Always reference your cloud bill for exact pricing!

## Conclusion

In the end, I always recommend customers to run Overwatch in their most critical workspaces in order to properly govern and analyze their Databricks usage. For more information check out the [documentation!](#)

*Disclaimer: these are my own thoughts and opinions and not a reflection of my employer*

Databricks

Databricks Labs

Databricks Overwatch

Overwatch



## Written by Ryan Chynoweth

[Edit profile](#)

312 Followers

Senior Solutions Architect Databricks — anything shared is my own thoughts and opinions

### More from Ryan Chynoweth



## Open standard for secure data sharing

Industry's first open protocol for secure data sharing, making it easy for organizations regardless of which computing platform they use.

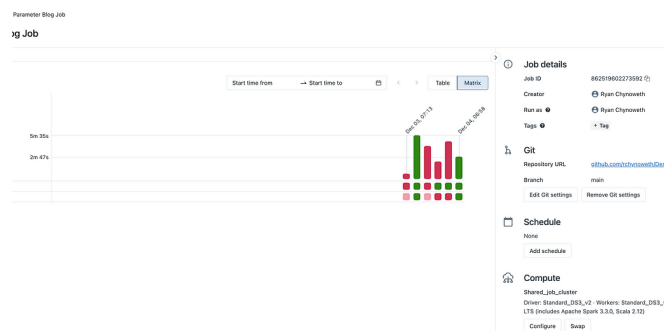


Ryan Chynoweth

## Delta Sharing: An Implementation Guide for Multi-Cloud Architecture

Introduction

8 min read · 3 days ago

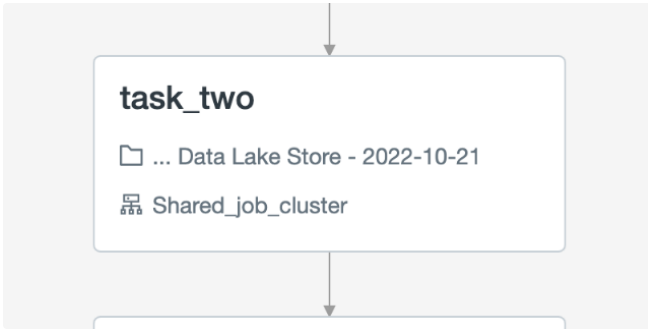


Ryan Chynoweth


## Task Parameters and Values in Databricks Workflows

Databricks provides a set of powerful and dynamic orchestration capabilities that are...

11 min read · Dec 7, 2022



 Ryan Chynoweth

 Ryan Chynoweth

## Converting Stored Procedures to Databricks

Special thanks to co-author Kyle Hale, Sr. Specialist Solutions Architect at Databricks.

14 min read · Dec 29, 2022



## Recursive CTE on Databricks

Introduction

3 min read · Apr 20, 2022



See all from Ryan Chynoweth

## Recommended from Medium





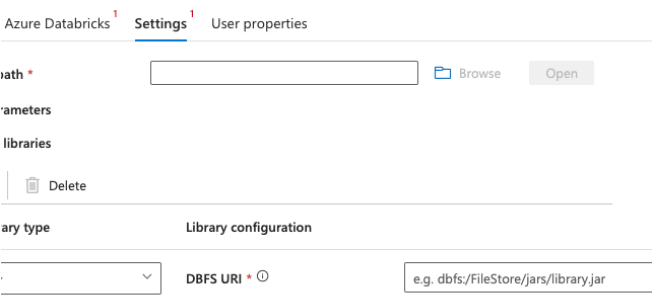
 Daan Rademaker


## Do-it-yourself, building your own Databricks Docker Container

In my previous LinkedIn article, I aimed to persuade you of the numerous advantages o...

7 min read · Oct 16, 2023

 26   



 Matt Bradley

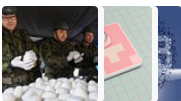
## Azure Data Factory tips for running Databricks jobs

Following on from an earlier blog around using spot instances with Azure Data...

4 min read · Nov 2, 2023

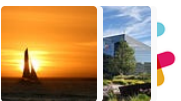
 14   

### Lists



#### Staff Picks

547 stories · 597 saves



#### Stories to Help You Level-Up at Work

19 stories · 395 saves



#### Self-Improvement 101

20 stories · 1146 saves



#### Productivity 101

20 stories · 1047 saves





Nnaemezue Obi-Eyisi

## Unveiling the Secrets: External Tables vs. External Volumes in...

While reviewing the Databricks documentation about Unity Catalog, I came...

🌟 · 7 min read · Sep 25, 2023



39



1



Manasreddy

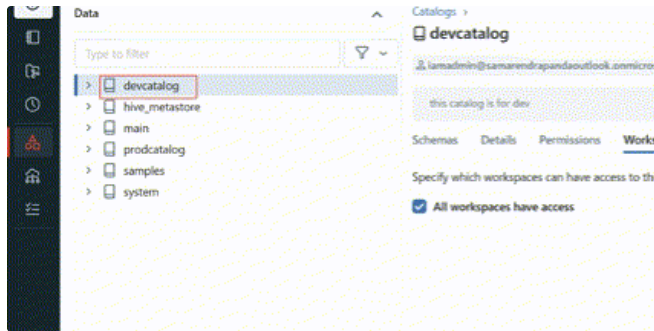
## How to pass: Databricks Data Engineer Professional Certification

Conquering the Databricks Data Engineer Professional Exam: A Definitive Guide

3 min read · Aug 24, 2023



8



Samarendra Panda

## Environment (dev, prod) creation in Azure Databricks Unity Catalog.

What is Unity Catalog?

2 min read · Jul 13, 2023



15



1



Oindrila Chakraborty

## Introduction to “Unity Catalog” in Databricks

What is “Unity Catalog”?

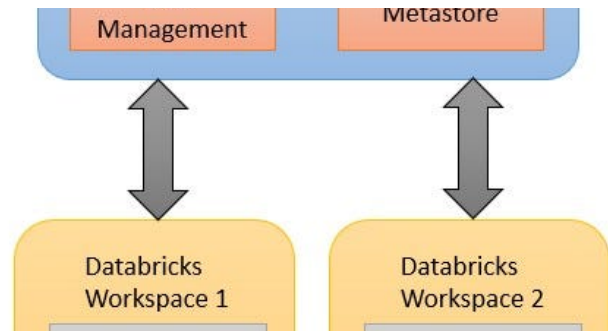
10 min read · Aug 14, 2023



18



2

[See more recommendations](#)