motifworks
An Accionlabs Company

# 15 Best Practices for Azure Databricks Implementation

Optimize your Azure Databricks platform with prescriptive and actionable best practices

## 1 Databricks Workspaces

Assign workspaces based on a related group of people working together collaboratively. It helps in streamlining access control matrix within the workspace (folders, notebooks etc.) and also across all resources that the workspace interacts with (ADLS, Azure SQL DB, Synapse etc.)

## 2 Separate workspaces by environment and deploy them into separate subscriptions

There are various limits at the workspace levels that may impact your environment. To get the maximum out the limits, it is recommended to use separate workspaces for prod, test, and dev environments
Key workspace limits are:

- The maximum number of jobs that a workspace can create in an hour is 5000
- At any time, you cannot have more than 1000 jobs simultaneously running in a workspace
- There can be a maximum of 145 notebooks attached to a cluster

## 3 Isolate Each Workspace in its own VNet

Only deploy one workspace in a Vnet. It aligns with the ADB's Workspace level isolation model. Use Vnet Peering to extend the private IP space of the workspace Vnet.

## 4 Do not store data in Default DBFS Folders

Every Workspace comes with a default DBFS, primarily designed to store libraries and other system-level configuration artifacts such as Init scripts. You should not store any production data in it because the lifecycle of default DBFS is tied to the Workspace and deleting the workspace will also delete the default DBFS and permanently remove its contents.

## 5 Use Log Analytics workspace to understand Databricks utilization

Monitoring databricks resource utilization is useful in arriving at the correct cluster and VM sizes. Each VM have a set of limits which play an important role in determining the performance of an Azure Databricks job. In order to get utilization metrics of an Azure Databricks cluster, you can stream the VM's metrics to an Azure Log Analytics Workspace by installing the Log Analytics Agent on each cluster node.

## 6 Selecting Programming Language

Databricks offers Standard and High Concurrency mode clusters. A High Concurrency cluster supports R, Python, and SQL, whereas a Standard cluster supports Scala, Java, SQL, Python, and R.

Databricks uses Scala for the background processin engine. Hence Scala performs better than Python and SQL. Therefore, for the Standard cluster, Scala is the recommended language for developing Spark jobs.

## 7 Use Key Vault for Storing Access Keys

Avoid hardcoding of sensitive information within the code. Store all the sensitive information such as storage account keys, database username, database password, etc., in a key vault. Access the key vault in Databricks through a secret scope.

## 8 Notebooks Organization

It is recommended to create separate folders for each group of users. Store the notebooks in group folders.

## 9 AutoComplete to Avoid Typographical Errors

Use the 'tab' button for auto-complete suggestions and eliminate typographical errors.

motifworks
An Accionlabs Company

## 10 Use the 'Format SQL' Option for Formatting the SQL Cells

Databricks offers a dedicated feature for formatting SQL cells. The "Format SQL code" option can be found in the "Edit" section or pressing CTRL+Shift+F

## 11 Use 'Advisor' Option

The Advisor option analyses the entire run and suggest required optimizations to increase the efficiency of the job.

## 12 Notebook Chaining

It is always a good practice to include all the repeatedly used operations such as read/write on Data Lake, SQL Database, etc., in one generic Notebook. The same Notebook can be used to set the Spark configurations, mounting ADLS path to DBFS, fetching the secrets from secret scope, etc.

For using the operations defined in the generic Notebook from other notebooks, it should be invoked using the "run" command.

## 13 Viewing the Content of a File

Use dbutils command (dbutils.fs.head("<path>")) to inspect file content and avoid loading the data into a Dataframe and then displaying the data.

## 14 Use ADF for orchestrating Databricks Notebooks

The ADF pipeline uses pipeline variables for storing the configuration details. During the Databricks notebook invocation within the ADF pipeline, the configuration details are transferred from pipeline variables to Databricks widget variables, thereby eliminating hardcoding in the Databricks notebooks.

## 15 Widget Variables

The configuration details are made accessible to the Databricks code through the widget variables. The configuration data is transferred from pipeline variable to widget variables when the notebook is invoked in the ADF pipeline.

## About Motifworks

Motifworks Inc, an Accion Labs company, is AMMP certified in Advanced Specialization for Azure Kubernetes Service, Win/SQL Server Migration, and Web Application Modernization and is recognized as one of the fastest-growing cloud solutions companies transforming businesses. As a group company of Accion Labs, combined we have a global presence in over 20 locations with over 4200 employees. We are a trusted Microsoft Partner with strong credentials including Partner of the Year Finalist (2020), Advanced Specialization, and Gold Partner in multiple competencies.

Gold Microsoft Partner Microsoft
ADVANCED SPECIALIZATION
Azure Kubernetes Service
Web Application Modernization
Windows Server and SQL Server Migration
Amongst Top Microsoft Partners in USA

Inc. 5000 America's Fastest-Growing Private Companies
BRONZE 2022 STEVIE WINNER American Business Awards
Microsoft Partner Power BI