Validation Framework as a Service – Configuration and Execution steps

Contacts

Email	karishmaa@maqsoftware.com;

CONTENTS

DESCRIPTION

Common Monitoring and Validation Framework (CMVF) is an integrated framework which provides functionalities of following tools:

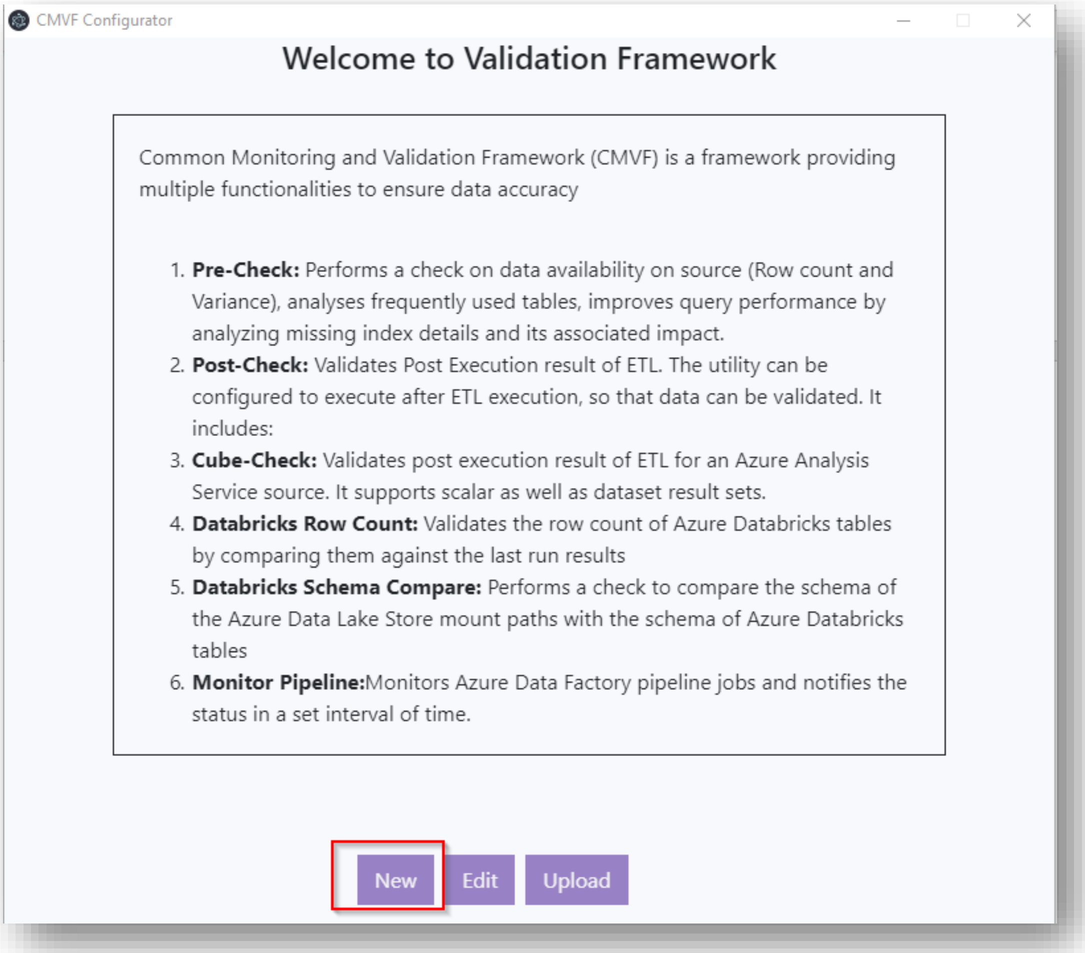
1. **Pre-Check:** Performs a check on data availability on source (row count and variance), analyses frequently used tables, improves query performance by analyzing missing index details and its associated impact.
2. **Post-Check:** Validates Post Execution result of ETL. The utility can be configured to execute after ETL execution, so that basic data can be validated
   The utility checks for the following three scenarios:
   a. **Data Existence:** To check if the record exists in table or query
   b. **KPI Variance:** To compare the source and target scalar values and return outcome as 'Pass' in case of exact match else 'Fail'
   c. **Dataset Comparison:** To compare the source and target dataset values and return outcome as Pass in case of exact match else Fail
3. **ADF Pipeline Monitoring:** Pipeline Monitoring Utility is created to monitor execution status and activity level details of Azure Data Factory (ADF) v2 Pipelines for any project. The utility can be configured to execute at a regular interval of time to give the running status of **triggered** pipelines.
   The Utility checks for the following three scenarios:
   a. **In Progress Pipeline:** To check if any pipeline is in progress at regular interval of time in the specified ADF.
   b. **Threshold Checking:** To check if any pipeline run is taking more than its normal runtime by comparing its previous runs. If it is running for more than threshold time, it will trigger an alert message.
   c. **Error Details:** To show error messages on failure of pipeline. If failed, it will send error details along with the execution details. If succeeded, it will show the execution details of all activities.

   Note: The Utility only monitors execution of triggered pipelines. Pipelines executed using 'Debug' mode, or any pipeline which is not published to master, will not be monitored.
4. **Cube-Check:** Validates post execution result of ETL for an Azure Analysis Service source. It supports scalar as well as dataset result sets.
5. **Databricks Row Count:** Validates the row count of Azure Databricks tables by comparing them against the last run results
6. **Databricks Schema Compare:** Performs a check to compare the schema of the Azure Data Lake Store mount paths with the schema of Azure Databricks tables

STEPS TO CONFIGURE:

CREATE NEW CONFIGURATION FILE

1. Once you open installed application, click on **New** button.



Welcome to Validation Framework

Common Monitoring and Validation Framework (CMVF) is a framework providing multiple functionalities to ensure data accuracy

1. **Pre-Check:** Performs a check on data availability on source (Row count and Variance), analyses frequently used tables, improves query performance by analyzing missing index details and its associated impact.
2. **Post-Check:** Validates Post Execution result of ETL. The utility can be configured to execute after ETL execution, so that data can be validated. It includes:
3. **Cube-Check:** Validates post execution result of ETL for an Azure Analysis Service source. It supports scalar as well as dataset result sets.
4. **Databricks Row Count:** Validates the row count of Azure Databricks tables by comparing them against the last run results
5. **Databricks Schema Compare:** Performs a check to compare the schema of the Azure Data Lake Store mount paths with the schema of Azure Databricks tables
6. **Monitor Pipeline:**Monitors Azure Data Factory pipeline jobs and notifies the status in a set interval of time.

CONNECTION DETAILS

- For **SQL/MSOLAP connections**

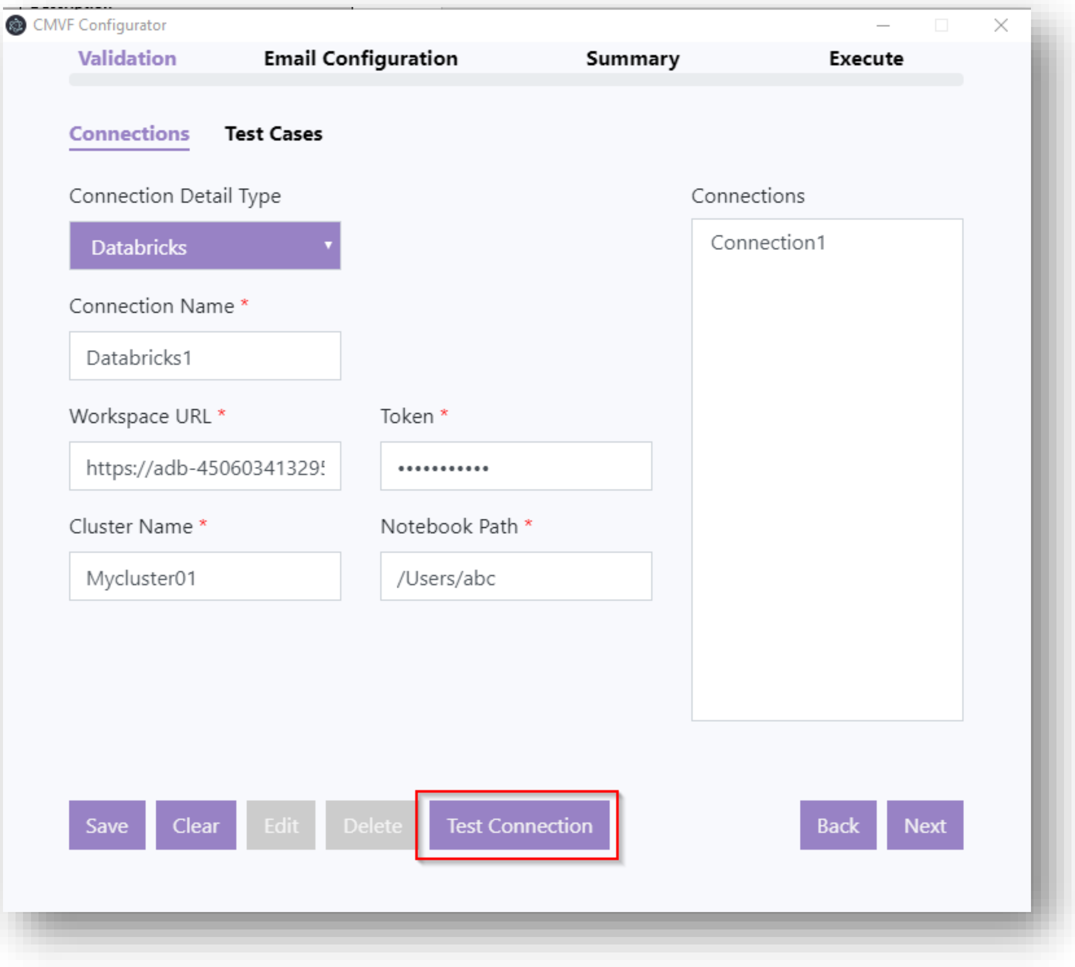
Key	Used For	Sample Value	Description
Connection Name	Acts as a unique connection identifier	Connection1	Name of connection
Connection Type	Supports SQL/MSOLAP	Select from dropdown	Required connection type
Authentication Type	Supports Windows/SQL/AAD	Select from dropdown	Required authentication type

Server Name	Establishing connection to server	Server1	Name of Server for which you are configuring the connection
Database Name	Establishing connection to database	Database1	Name of Database within the server on which queries need to run
UserName	Data Verification	SQL Auth Username	Provide username for the connection if Auth Type is SQL
Password	Data Verification	SQL Auth Password	Provide password for the connection if Auth Type is SQL
Active Directory TenantID	Cube Validation for AAD Auth type	7xxxxxxf-8xx1-4xxf-9xxb-2xxxxxxxxxx7	Enter you AAD Tenant ID
Application ID	Cube-check for AAD Auth type	6xxxxxxg-9xx1-3xxh-4xxb-2xxxxxxxxxx0	Enter you AAD Application ID (SPN)
Application Password	Cube-check for AAD Auth type		Enter you AAD Application Password

- For **Databricks connections**

Key	Used For	Sample Value	Description
Connection Name	Acts as a unique connection identifier	Databricks1	Name of connection
Workspace URL	Databricks Row Count/Databricks Schema Compare	https://adb-4506034132954994.14.azuredatabricks.net	URL of your Databricks Workspace
Token	Databricks Row Count/Databricks Schema Compare		Personal Access Token generated from Azure Databricks
Cluster Name	Databricks Row Count/Databricks Schema Compare	Mycluster01	Name of the cluster on which you want to execute it
Notebook Path	Databricks Row Count/Databricks Schema Compare	/Users/abc	Required execution notebook path of your workspace

- Once all the connection configuration details are added click on **Save** to store the connection for future reference.
- Click on **Test Connection** to ensure that connection details are correct.
- (For windows auth need to enable port)



Connection Detail Type: Databricks

Connection Name *: Databricks1

Workspace URL *: https://adb-45060341329!

Token *: •••••••••••

Cluster Name *: Mycluster01

Notebook Path *: /Users/abc

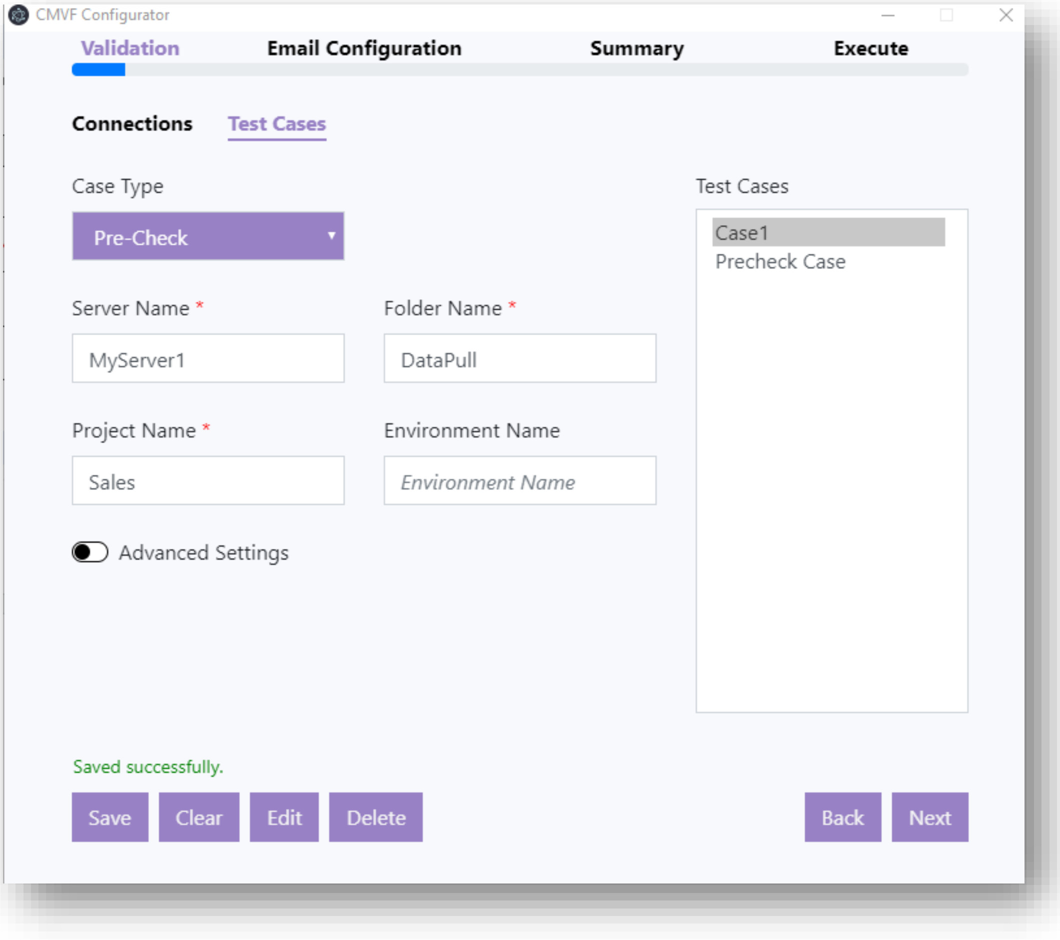
Connections: Connection1

TEST CASES

SPECIFIC TO **PRE-CHECK**

- By default, Pre-Check will extract all the queries from SSIS Project and execute it on all the connections (Source/Destination) (Currently it does not support DAX Queries)

Key	Used For	Sample Value	Description
Project Name	Pre-check	Sales	Name of SSIS Project to Run the Utility.
Folder Name	Pre-check	LearningCurvePull	Folder name within SSIS Catalog which has the project on which Utility must Run.
Server Name	Pre-check	MyServer1	Name of the server on which check is needed to be run
Environment Name	Pre-check	Dev/Prod	Name of environment on which check is needed to be run.



Case Type: Pre-Check

Server Name *: MyServer1

Folder Name *: DataPull

Project Name *: Sales

Environment Name: Environment Name

Advanced Settings

Test Cases: Case1, Precheck Case

Saved successfully.

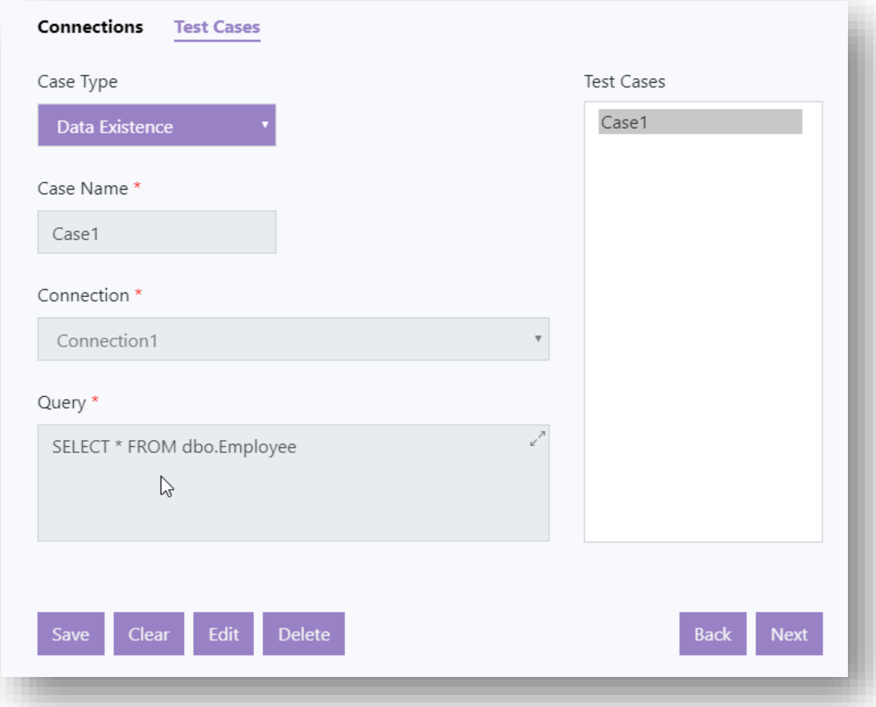
SPECIFIC TO **POST-CHECK**

Post-check includes 3 types of test cases

1. Data Existence
2. KPI Variance
3. Dataset Comparison

- For **Data Existence** test cases (Used only when you want to verify that some data exists in your target server connection provided)

Key	Used For	Sample Value	Description
Case Name	Acts as a unique test case identifier	Case1	Name of test cases
Connection	Data Verification	Select from the drop down of connections available	Configured connection name
Query	Data Verification	SELECT * FROM dbo.Employee	Enter your SQL Query



Case Type: Data Existence

Case Name *: Case1

Connection *: Connection1

Query *: SELECT * FROM dbo.Employee

Test Cases: Case1

- For **KPI Variance** test cases (Used for scalar query output validation)

Key	Used For	Sample Value	Description
Case Name	Acts as a unique test case identifier	Case1	Name of test cases
Threshold	Allowed threshold	5	Enter the threshold value allowed for the variance calculation
Source Connection	Data Verification	Select from the drop down of connections available	Configured connection name
Target Connection	Data Verification	Select from the drop down of connections available	Configured connection name
Source Query	Data Verification	SELECT COUNT(1) FROM dbo.Employee	Enter your source SQL Query
Target Query	Data Verification	SELECT COUNT(DISTINCT Employee) FROM dbo.Employee	Enter your destination SQL Query

- For **Dataset Comparison** test cases, queries resulting in same source and target schema (Used for tabular query output validation).

Key	Used For	Sample Value	Description

Case Name	Acts as a unique test case identifier	Case1	Name of test cases
Source Connection	Data Verification	Select from the drop down of connections available	Configured connection name
Target Connection	Data Verification	Select from the drop down of connections available	Configured connection name

Source Query	Data Verification	SELECT COUNT(1), Area FROM dbo.Employee
		GROUP BY Area
			Enter your source SQL Query

Target Query	Data Verification	SELECT COUNT(1), Area FROM dbo.Employee
		GROUP BY Area
			Enter your destination SQL Query

SPECIFIC TO **DATABRICKS ROWCOUNT**

- It is used to store previous and current row count for all the tables present in your Hive ADB database

Key	Used For	Sample Value	Description
Case Name	Acts as a unique test case identifier	Case1	Name of test case
Connection		Select from the available connections for ADB	Select the required ADB connection to perform row count upon.
Database Name	Rowcount	default	Name of the Hive Database
Exclusion List	Rowcount	_tmp	Contains string to exclude the names of all the tables matching with the given list of words
Positive Variance	Rowcount	5	Variance % for positive variance
Negative Variance	Rowcount	5	Variance % for negative variance

SPECIFIC TO **DATABRICKS SCHEMA COMPARE**

- It is used to compare schema for mounted table/files path from upstream with the destination Hive table.

Key	Used For	Sample Value	Description
Case Name	Acts as a unique test case identifier	Case1	Name of test case

Connection		Select from the available connections for ADB	Select the required ADB connection to perform schema compare upon.
Source Path	Schema Compare	/mnt/myblob	Give mounted path for your source
Source Table Name	Schema Compare	mytable	Name of your upstream table
Source File Type	Schema Compare	Parquet/csv/xls	Select your file type
Stream Name	Schema Compare	Platform	Name specific to project
Database Name	Schema Compare	default	Destination hive database name
Table Name	Schema Compare	mytable	Destination hive table name

SPECIFIC TO **CUBE VALIDATION**

- It is used to perform data verification for Azure Tabular models

Key	Used For	Sample Value	Description
Case Name	Acts as a unique test case identifier	Case1	Name of test cases
Result Type	Cube Validation	Select from the drop down	Select single value for query resulting in scalar output and multivalue shall be in the form of key value
Threshold	Allowed threshold	5	Enter the threshold value allowed for the variance calculation
Source Connection	Cube Validation	Select from the drop down of connections available	Configured connection name
Target Connection	Cube Validation	Select from the drop down of connections available	Configured connection name
Source Query	Cube Validation	DAX Query	Enter your source DAX Query
Target Query	Cube Validation	DAX Query	Enter your destination DAX Query

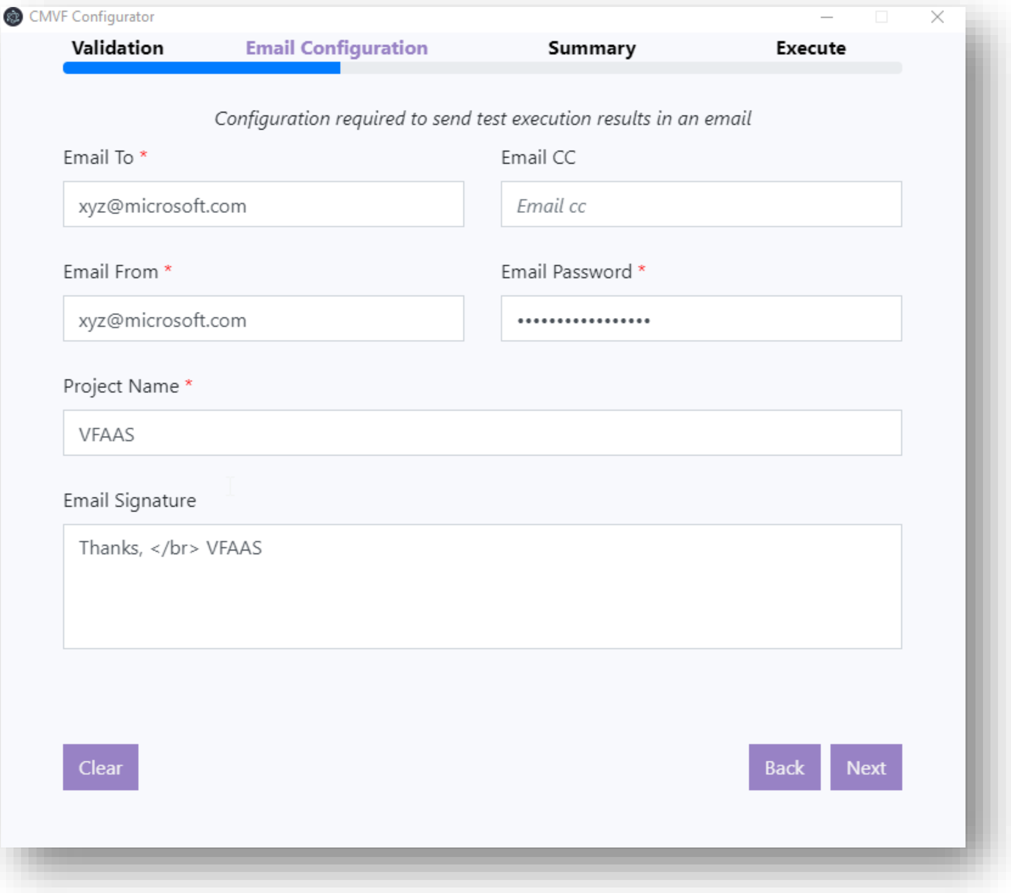
SPECIFIC TO **MONITOR PIPELINE**

- It is used to monitor pipeline run details (Currently it supports sonly triggered pipeline runs). Make sure that the entered AAD App is having access on Data Factory resource

Key	Used For	Sample Value	Description
TenantID	Monitor Pipeline	7xxxxxxf-8xx1-4xxf-9xxb-2xxxxxxxxxx7	Name of test cases
Subscription ID	Monitor Pipeline	3xxxxxb-3xx9-4xx6-bxx4-0xxxxxxxxxa	Enter the threshold value allowed for the variance calculation
Resource Group Name	Monitor Pipeline	Rg1	Resource group in which your ADF lies in.
Data Factory Name	Monitor Pipeline	Select from the drop down of connections available	Name of your data factory instance
Application ID	Monitor Pipeline	7xxxxxxf-8xx1-4xxf-9xxb-2xxxxxxxxxx7	Enter you AAD Application ID (SPN)
Application Secret	Monitor Pipeline		Enter you AAD Application Password. Used to fetch token and validate
Pipeline to Monitor	Monitor Pipeline	Master_Pipeline	Comma separated list of pipeline name
Span of Days	Monitor Pipeline	5	Number of days for which you want to fetch pipeline runs.

HOW TO EXECUTE:

1. Once test cases are configured click on **Next** (Email configuration tab) and enter the validation email details.



Configuration required to send test execution results in an email

Email To *: xyz@microsoft.com

Email CC: Email cc

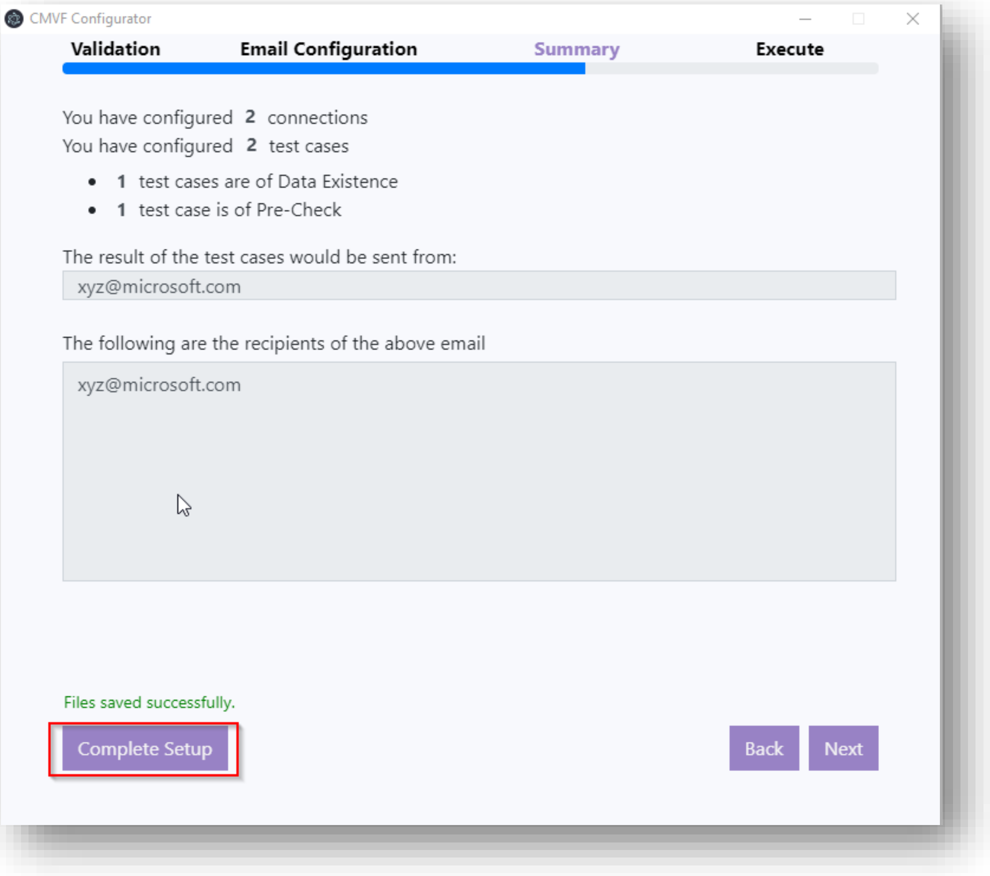
Email From *: xyz@microsoft.com

Email Password *: •••••••••••••••••

Project Name *: VFAAS

Email Signature: Thanks, </br> VFAAS

2. On the next tab verify the summary of your configured connections and click on **Complete Setup** to store the configuration file for future reference



You have configured 2 connections

You have configured 2 test cases

- 1 test cases are of Data Existence
- 1 test case is of Pre-Check

The result of the test cases would be sent from:

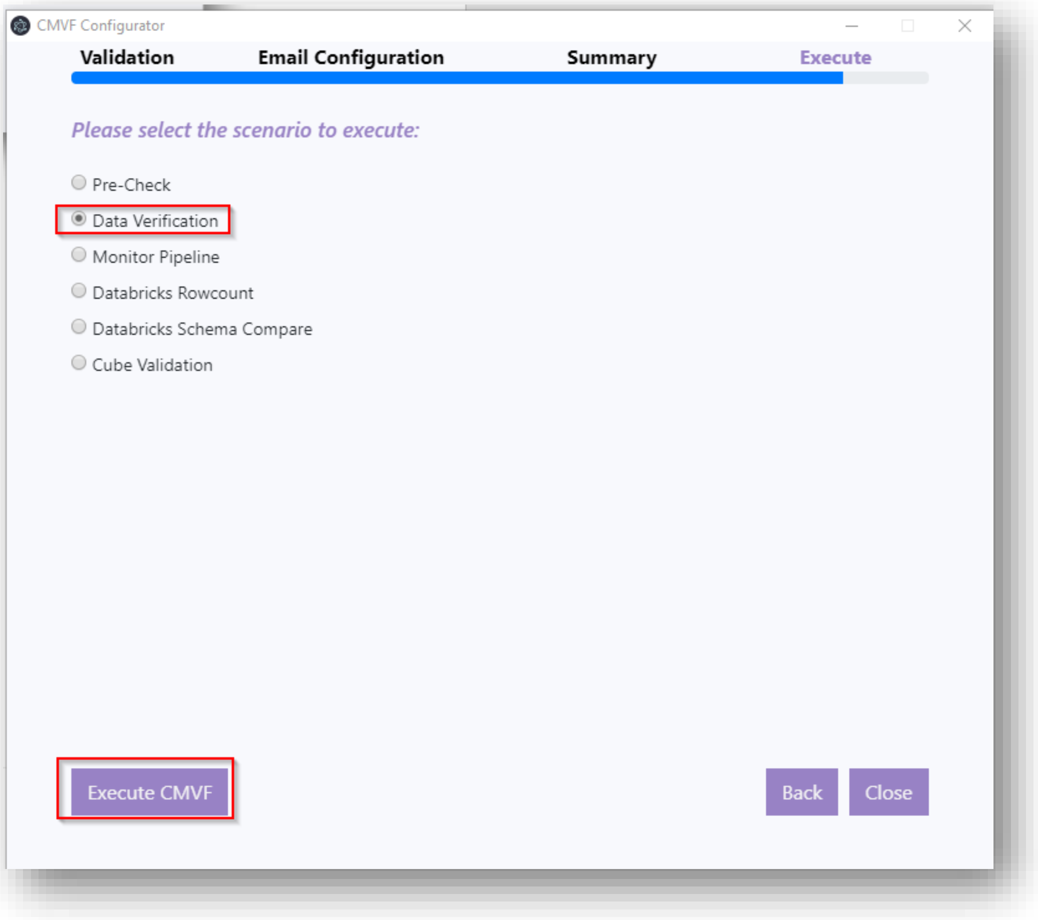
xyz@microsoft.com

The following are the recipients of the above email

xyz@microsoft.com

Files saved successfully.

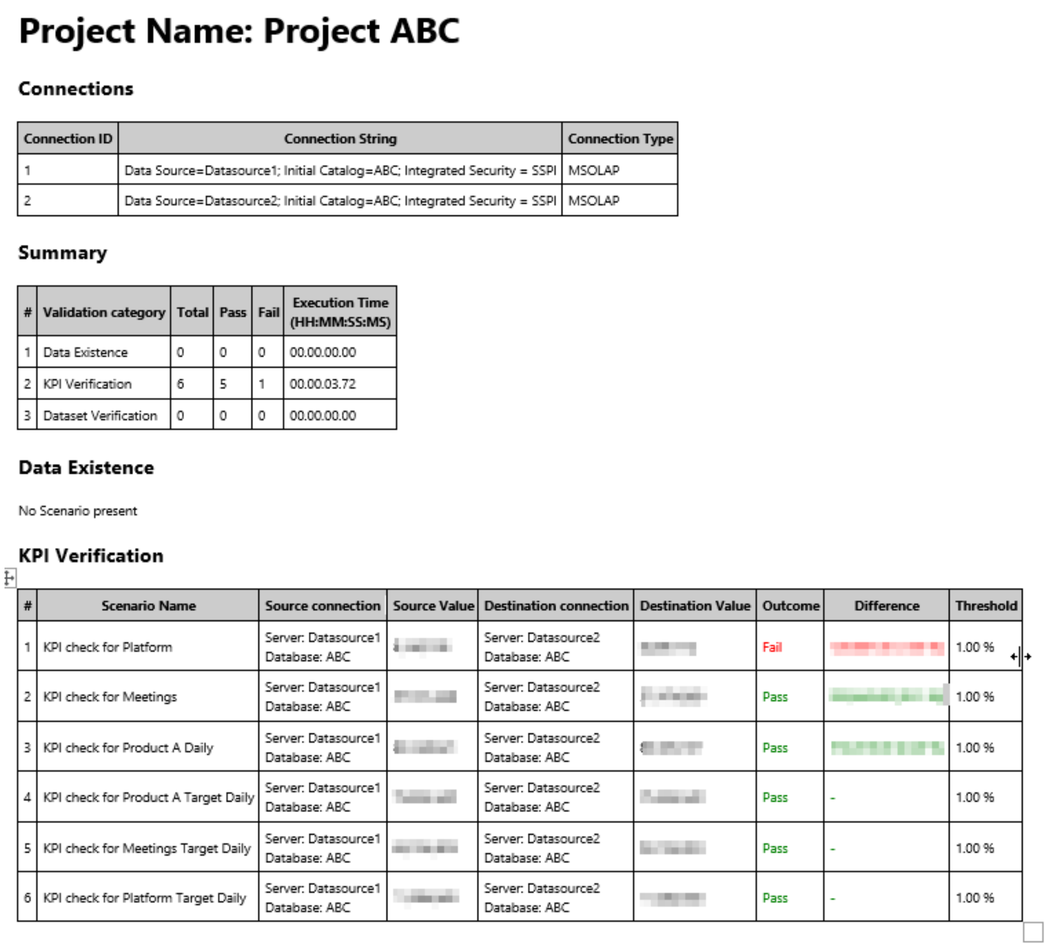
3. Select your desired test case execution scenario and click on **Execute.**



Please select the scenario to execute:

○ Pre-Check
● Data Verification
○ Monitor Pipeline
○ Databricks Rowcount
○ Databricks Schema Compare
○ Cube Validation

4. An E-mail gets generated summarizing all the details.



Project Name: Project ABC

Connections

Connection ID	Connection String	Connection Type
1	Data Source=Datasource1; Initial Catalog=ABC; Integrated Security = SSPI	MSOLAP
2	Data Source=Datasource2; Initial Catalog=ABC; Integrated Security = SSPI	MSOLAP

Summary

#	Validation category	Total	Pass	Fail	Execution Time (HH:MM:SS:MS)
1	Data Existence	0	0	0	00.00.00.00
2	KPI Verification	6	5	1	00.00.03.72
3	Dataset Verification	0	0	0	00.00.00.00

Data Existence

No Scenario present

KPI Verification

#	Scenario Name	Source connection	Source Value	Destination connection	Destination Value	Outcome	Difference	Threshold
1	KPI check for Platform	Server: Datasource1 Database: ABC	[redacted]	Server: Datasource2 Database: ABC	[redacted]	Fail	[redacted]	1.00 %
2	KPI check for Meetings	Server: Datasource1 Database: ABC	[redacted]	Server: Datasource2 Database: ABC	[redacted]	Pass	[redacted]	1.00 %
3	KPI check for Product A Daily	Server: Datasource1 Database: ABC	[redacted]	Server: Datasource2 Database: ABC	[redacted]	Pass	[redacted]	1.00 %
4	KPI check for Product A Target Daily	Server: Datasource1 Database: ABC	[redacted]	Server: Datasource2 Database: ABC	[redacted]	Pass	-	1.00 %
5	KPI check for Meetings Target Daily	Server: Datasource1 Database: ABC	[redacted]	Server: Datasource2 Database: ABC	[redacted]	Pass	-	1.00 %
6	KPI check for Platform Target Daily	Server: Datasource1 Database: ABC	[redacted]	Server: Datasource2 Database: ABC	[redacted]	Pass	-	1.00 %