|  |  |
| --- | --- |
|  | Azure Data Factory Deployment & Automation |
| Deployment & Automation from DEV to UAT Instance | |

| **Document Details** | |
| --- | --- |
| Document Author | Rupesh Shelar |
| Contributors | Agglomerates Scrum |

Table Of Contents

[1 OVERVIEW 2](#_Toc401912134)

[2 PROJECT SCOPE 2](#_Toc401912135)

[2.1 DATA FLOW 1.1 2](#_Toc401912136)

[2.1.1 AUTOMATED PIPELINE 1.1.1 2](#_Toc401912137)

[2.2 POST MIGRATION STEPS AND MONITORING 1.2 2](#_Toc401912138)

[2.2.1 SECURITY 1.1.1 2](#_Toc401912139)

[3 DEPLOYMENT PLAN 2 2](#_Toc401912140)

[3.1 TESTING 2.1 2](#_Toc401912141)

# Overview

This document outlines the automation process for the Azure Data Factory pipeline named

**ppl\_ExtractDatafromDataPlatform** for migrating data between **DEV** and **UAT** instances.

The goal is to create a pipeline that automates the extraction, transformation, and loading (ETL) of data from the **StagingDEV** and **MultivueReporting** databases into their corresponding UAT instances.

.

# Project Scope

* **Source Systems**: StagingDEV, MultivueReporting (in the DEV environment)
* **Target System**: UAT Instance (same schemas, database instance)
* **Data Movement Type**: Full or Incremental Load
* **Pipeline Name**: **ppl\_ExtractDatafromDataPlatform**
* **Automation Requirements**: Full automation of the pipeline execution using triggers and parameterization.

### Data Flow 1.1

**Source Database (BDC\_Data\_Platform)** → **Data Extraction** → **Azure Data Factory Pipeline** → **Data Transformation(Inbuilt)** → **UAT Database Instance (StagingUAT, MultivueReporting- SQL Server**

### Automated Pipeline 1.2

* **Triggering Mechanism**: Azure Data Factory triggers will be used to automate the pipeline execution.(TBD)

## Post Migration Steps and monitoring 1.2

1. **Load Data**:
   * Run ADF Pipeline to load data into UAT database tables.
2. **Error Handling**:
   * Use **Azure Monitor** for logging errors and monitoring pipeline execution.
3. **Logging and Monitoring**:
   * Logging mechanisms with Azure Monitor and custom logging for debugging purposes.
   * Email notifications for failed pipeline runs.
4. **Success/Failure Notifications**:
   * Use **Azure Logic Apps** or **Azure Function** to send email notifications on the success or failure of the pipeline run.

### Security 1.1.1

* **Managed Identity**: Use Managed Identity for secure connections to Azure SQL Databases.
* **Azure Key Vault**: Store connection strings, credentials, and sensitive data in Azure Key Vault.

# Deployment Plan 2

* **Stage 1**: Implement and test the pipeline in the **Development** environment.
* **Stage 2**: Deploy and test the pipeline in **UAT** environment.( Deployement Steps in PBI)
* **Stage 3**: Implement the pipeline for **Production** deployment (if required

## Testing 2.1

* **Unit Testing**:
  + Test each activity (e.g., Copy, Data Flow) individually in a non-production environment.
* **Integration Testing**:
  + Test end-to-end data flow to ensure that the pipeline works as expected.
* **UAT Testing**:
  + Allow stakeholders to validate data in the UAT environment before final deployment.

***End of Document***