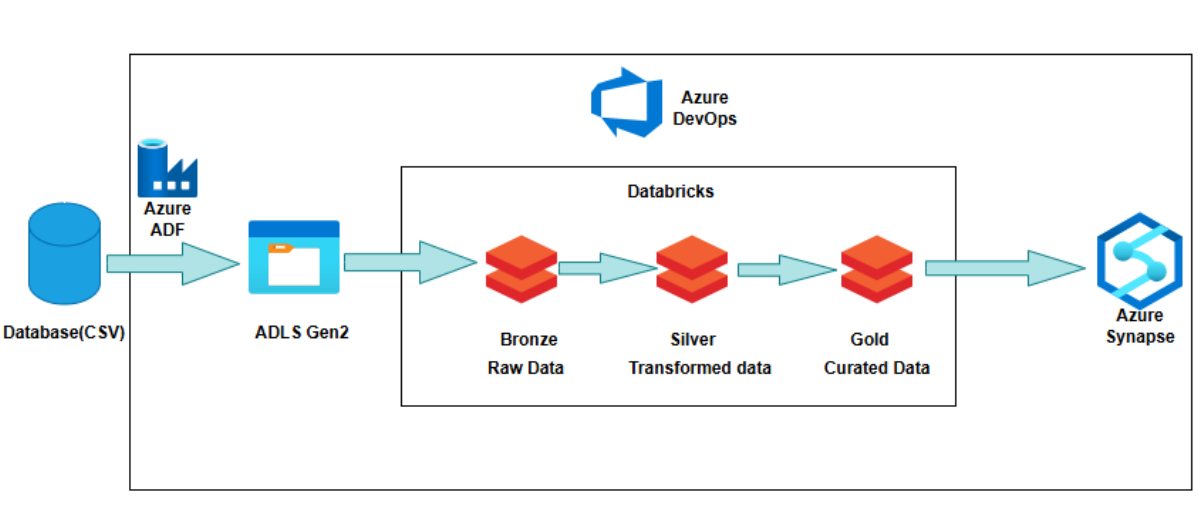
Data Pipeline for Wind Turbines

Introduction

This project focuses on building a scalable and testable data pipeline to process wind turbine data for a renewable energy company. The pipeline follows the Medallion Architecture with three layers: Raw (Bronze), Curated (Silver), and Gold and incorporates various data engineering best practices, including:

- CI/CD (Continuous Integration & Deployment)
- Version Control
- Data Governance
- Exception Handling & Error Logging
- Monitoring & Alerting

Pipeline Architecture



The data pipeline follows medallion architecture:

1. Bronze Layer

- Purpose: Stores unprocessed wind turbine data.
- Ingestion Method: Data is extracted from a relational database and stored in an Azure Data Lake Storage (ADLS) container using Azure Data Factory (ADF).
- Data Issues: Raw data may contain missing values, outliers, and anomalies.

2. Silver Layer

- Purpose: Cleans and processes data using Databricks and Spark.
- Processing Steps:
    - Handle missing values.
    - Detect outliers using Interquartile Range (IQR).
    - Apply transformations & calculations (e.g., standard deviations for anomaly detection).
- Output: Cleaned and structured data is stored in the Silver Layer.

3. Gold Layer

- Purpose: Stores cleaned and enriched data, along with summary statistics, in Azure Synapse Analytics for reporting and further analysis.

Data Ingestion (Bronze Layer)

Data Source & Extraction

- Data Generation: Each turbine group generates power output data, wind speed, and wind direction data in real-time. This data is batched and saved as CSV files (e.g., data_group_1.csv, data_group_2.csv, and data_group). Each file contains data from a specific set of turbines (e.g., turbines 1-5).
- Daily CSV Files: The turbine system produces these CSV files daily, each containing the last 24 hours of data for the corresponding turbine group.
- Sensor Connectivity: The sensors on the turbines are connected to a local on-premise server, which transfers the CSV files to a staging area in the cloud or local storage.
- ADF retrieves CSV and loads them into Azure Data Lake Storage (Bronze Layer).
- Each CSV file represents a daily batch of turbine data.

Azure Data Factory (ADF) Ingestion Workflow

- ADF extracts CSV data from the database and stores it in Azure Data Lake Storage (ADLS).
- Pipeline Components:
    - Source Linked Service: Connects to the database to extract data.
    - Sink Linked Service: Defines where ADF stores the data in the Bronze Layer.
    - Source Dataset: In the Copy Activity, the Source Dataset points to the staging area where the CSV files are generated daily
    - Sink Dataset: The Sink Dataset specifies the Azure Data Lake storage location in the Bronze Layer.

    - Copy Activity: Moves data from the database to Azure Data Lake (Bronze Layer).
    - Data Flow Activity: Validates schema and processes missing values.
- Monitoring & Error Handling:
    - Automatic retries in case of failures.
    - Alerts & notifications for missing records.
    - Pipeline scheduling (e.g., runs every 24 hours).

Data Transformation (Silver Layer)

Once raw data is ingested, the pipeline reads, processes, and cleans the data in Databricks (PySpark).

Processing Steps:

1. Read Raw Data (Bronze Layer)
2. Handle Missing Values
    - Fill numeric columns with median values.
    - Replace string columns with 'Unknown'.
3. Detect Outliers using the Interquartile Range (IQR).
4. Detect Anomalies in power output (values outside 2 standard deviations).
5. Testing is integral to this pipeline. Unit tests are written to validate individual transformations, while regression tests ensure that new changes don't break the existing logic.
6. Write Clean Data to the Silver Layer (Delta format).

Data Warehousing (Gold Layer)

Purpose:

The Gold Layer stores processed and transformed data for reporting and analytics.
In this layer, structured tables are created in Azure Synapse Analytics.

Exception Handling

- Missing Records: Automatic retries; alerts if database extract is incomplete.
- Schema Mismatch: Logs errors and moves corrupt data to quarantine storage.
- Transformation Failures: Uses Azure Monitor & Log Analytics for troubleshooting.

Testing Strategy

Testing is an essential part of this pipeline to ensure data accuracy and reliability.

Unit Tests

- Validate individual transformation functions.
- Ensure outlier and anomaly detection work correctly.

Regression Tests

- Ensure new updates do not break existing logic.

End-to-End Tests

- Simulate an entire pipeline run and validate outputs.

Further Enhancements

Visualisation

- Connect to Power BI for dashboards

CI/CD Pipeline

Version Control

- Code, ARM templates, and pipeline configurations are stored in Git (Azure Repos).
- Enables collaboration and change tracking.

Deployment (Azure DevOps)

- Automates deployment of ADF, Databricks, and Synapse configurations.
- Uses self-hosted agents and ARM templates.

Security

- Sensitive credentials (e.g., database connection strings, API keys) are stored in Azure Key Vault.
- Data encryption (SSL, KMS) is applied for security.

Monitoring & Alerting

- Azure Monitor & Log Analytics track pipeline health & failures.
- Alerts & notifications trigger when:
    - Data ingestion fails.
    - Processing errors occur.

- Missing or corrupted data is detected.

Data Governance & Security

- Data Quality Validation: Tools like Great Expectations / Deequ can be used for data validation and to enforce schema consistency before moving to Silver Layer.
- Data Security Measures: Azure key vault for encryption at rest & in transit
- Role-based access control (RBAC) ensures restricted data access.
- Audit Logging & Traceability: Azure purview tracks all data changes & transformations. Enables auditability for compliance.