Build Data Pipeline using Azure Medallion Architecture Approach

Business Overview

A well-incorporated pipeline is a critical data engineering practice designed to integrate data from various sources, such as on-premise databases, cloud-managed databases, CSV, XML, log files, Data Warehouses, and API endpoints. This type of data is then transformed under the comprehensive data pipeline (cloud/ on-premise), which is then used to generate meaningful insights for analytics. This approach has become fundamental for organizations aiming to manage large volumes of data efficiently to reduce manual intervention.

A comprehensive data pipeline empowers organizations to use their data to its fullest extent, facilitating quality data for analysis or model prediction. This capability has covered many applications in today's world, including e-commerce platforms, product companies, startups, and government agencies. By deploying the infrastructure of these data pipelines into their platform, these organizations can drive insights rapidly and effectively for their business growth.

Key features:

1. **Complex Workflow Handling:** Cloud-based pipeline manages complex data workflows seamlessly, enabling scaling and robustness.
2. **Data Quality Assurance:** Implementing medallion architecture in the Databricks workspace ensures data accuracy, cleaning, data transformation, and outlier detection
3. **Data Extraction:** Fetching data from Azure-managed SQL Database into a cohesive format.
4. **Analysis Efficiency:** By loading only high-quality, essential data into the final layer for visualization, data pipelines minimize the overhead of handling large datasets and enhance overall performance.

Wide-ranging and thoroughly planned ETL pipelines operate by applying predefined business rules and data transformation rules to the complex dataset to ensure high-quality outcomes, such as logistics optimization and user engagement rates. They also analyze sales data over several years to predict future sales trends. These pipelines also manage complex interactions with backend databases and systems to maintain throughput and data integrity for accumulated historical data.

One significant advantage of using Azure services is that integrated data management capabilities streamline data ingestion, transformation, and analysis across diverse

sources with seamless connectivity and orchestration. By leveraging different Azure services intended to change, we ensure the solution's availability, usability, and durability, which are crucial for maintaining a stable and efficient data pipeline.

Aim:

This project aims to analyze large sensor water data collected in European countries over different periods using Azure Services. The project's main purpose is to leverage Azure Services' capabilities to handle the workflow and get the raw data to be analyzed to generate insights as per requirements. This project leverages Azure services to uphold data management standards and efficiently handle large datasets throughout the pipeline, ensuring reliable performance without failures.

Tech Stack:

- **Programming:** SQL, Scala
- **Services:** Azure Managed SQL, Azure Logic app, Azure Blob Storage, Azure Data Factory, Azure Data Lake Storage Gen2, Azure Databricks, Power BI

Data description:

The dataset constitutes a complex view of aggregated water sensor data with 32 columns and more than one million rows collected across different European countries across the years. It includes detailed information on various aspects such as country, water body category, determinands obtained, concentration level (minimum, maximum, mean, and median) of determinands across particular time stamps, and quality samples conducted out of the total samples for each observation. Given the ongoing data collection at every timestamp, each recorded value is associated with a specific country, capturing different determinands content across various monitoring sites.

Approach:

1. The dataset was initially available in the SQL database bucket. The Azure Logic app was created to pull data from the Azure Managed SQL Database.

2. Azure Blob Storage was created to store the raw data from an Azure-managed SQL Server database. Upon dumping the raw data into the Azure Blob Storage account, ADLS (Azure Data Lake Storage Account Gen2) was created with hierarchical space enabled for Gen2.
3. After dumping raw data, an Azure Data Factory instance was created to orchestrate the data movement from blob storage to ADLS Gen2.
4. A detailed plan of medallion architecture was implemented, segregating it into three layers: a bronze layer, a silver layer, and a golden layer, which completes the underlying agenda of quality-proven data.
5. Cleaned and processed data was retrieved from a Hive metastore database and subsequently loaded into Power BI, which served as the foundation for generating valuable insights through visualizations.

Key Takeaways:

- In-depth understanding of Azure services
- Configuration and implementation of on-premise SQL Server
- Creation of Azure SQL database and server
- Development of data ingestion workflows using Logic Apps
- Extraction of data from Azure-managed SQL Server database
- Establishment of Azure Blob Storage account
- Setup of Azure Data Lake Storage Gen2 account
- Creation of Azure Data Factory workspace
- Implementation of data pipelines in Azure Data Factory
- Configuration of computation cluster in Databricks workspace
- Implementation of Medallion architecture for enhanced data quality
- Loading data from Databricks into Power BI
- Development of columns and measures using DAX in Power BI
- Creation of comprehensive dashboards in Power BI

Architecture Diagram: