Azure Data Engineering Project Documentation

SHR3YGO3L/prohectazure1

Project Overview

This project is designed as a comprehensive, end-to-end Azure data engineering solution aimed at mastering in-demand Azure tools and technologies such as Azure Data Factory (ADF), Azure Databricks, Azure Synapse Analytics, Managed Identities, API connections, and Power BI. The project simulates real-world data engineering scenarios, focusing on dynamic pipeline creation, data transformation, warehousing, and visualization. It helped the author secure a role as a data engineer by demonstrating expertise through practical application and addressing interviewer-asked scenarios.

Objectives

- Learn and implement Azure Data Factory pipelines from scratch to ingest data.
- Use Azure Databricks for big data transformations and analytics.
- Build a data warehouse solution using Azure Synapse Analytics.
- Understand and configure Managed Identities for secure data access.
- Create dynamic, parameterized pipelines to handle multiple data files efficiently.
- Establish connections between Azure Synapse Analytics and Power BI for data visualization.
- Gain hands-on experience with real-time use cases and interview-relevant questions.

Technologies & Tools Used

- Azure Data Factory (ADF): Orchestration tool used for data ingestion and pipeline management.
- Azure Databricks: Apache Spark-based analytics platform for data transformation.
- Azure Synapse Analytics: Data warehousing and analytics service for serving processed data.

- Azure Data Lake Storage Gen2: Storage solution to maintain raw (bronze), transformed (silver), and serving (gold) data layers.
- Managed Identities: Azure service for secure authentication and authorization.
- Power BI: Visualization tool to build dashboards from Synapse data.
- GitHub API: Source of raw data pulled dynamically via HTTP connectors.

Project Architecture

Medallion Architecture

The project follows the Medallion architecture pattern with three distinct data zones:

- Bronze Layer (Raw): Stores raw data ingested directly from the source APIs without transformation.
- Silver Layer (Cleaned/Transformed): Contains cleaned and transformed data prepared for analysis.
- Gold Layer (Serving): Final curated datasets stored in a data warehouse (Synapse Analytics) for business users and analytics.

Data Flow

- Data is fetched dynamically from a GitHub repository via REST API.
- Azure Data Factory pipelines ingest this data into the Bronze container in Azure Data Lake.
- Azure Databricks reads from Bronze, applies transformations, and writes to Silver container.
- Azure Synapse Analytics consumes data from Silver, creating views and external tables in the Gold layer.
- Power BI connects to Synapse to visualize and analyze the processed data.

Detailed Project Phases

Phase 1: Data Ingestion Using Azure Data Factory

Key Steps:

- Creation of Azure resources: Resource Group, Storage Account (with hierarchical namespace enabled → Data Lake Gen2), and Azure Data Factory instance.
- Creation of containers representing Bronze, Silver, and Gold layers.
- Building pipelines in ADF:
    - Static pipelines: Copying individual CSV files from GitHub (HTTP source) to Bronze container.
    - Dynamic pipelines: Parameterized pipelines using For Each loops and Lookup activities to iterate over multiple files, passing parameters such as relative URLs, folder names, and file names dynamically.
- Creation of linked services (connections) in ADF to GitHub (HTTP) and Azure Data Lake.
- Use of datasets to specify file formats and paths.
- Execution and monitoring of pipelines to ensure successful ingestion.

Highlights:

- Use of dynamic parameters and For Each activities to avoid redundant static pipelines.
- Demonstration of real-world API data ingestion and automated orchestration.
- Coverage of interview questions like data redundancy, linked services, and pipeline parameterization.

Phase 2: Data Transformation Using Azure Databricks

Key Steps:

- Provisioning an Azure Databricks workspace and cluster.
- Configuring secure access from Databricks to ADLS Gen2 using service principal (app registration in Azure AD) and managed identities.
- Reading raw data from Bronze container using Spark APIs.
- Performing various transformations in notebooks:
    - Date transformations (extracting month and year).
    - String operations (concatenation of customer names, splitting product SKUs).
    - Mathematical operations (multiplying columns).

    - Filtering, cleaning, and schema inference.
- Writing transformed data in Parquet format to Silver container.
- Performing exploratory data analysis and visualizations using built-in Spark UI (charts like bar and pie charts).
- Merging multiple years of sales data efficiently.

Highlights:

- Hands-on with PySpark DataFrame API and Spark SQL functions.
- Learning advanced Spark functions like `withColumn()`, `concat_ws()`, `split()`, and aggregation.
- Understanding file formats (Parquet) and their benefits.
- Incorporation of data validation and error handling.
- Visualization capabilities within Databricks notebooks.

Phase 3: Data Warehousing Using Azure Synapse Analytics

Key Steps:

- Creation of Synapse Analytics workspace and default storage account.
- Assigning managed identities and roles for secure access to ADLS Gen2.
- Introduction to Synapse components: Pipelines (integrated ADF), Spark Pools, and SQL Pools.
- Creation of Serverless SQL pool and databases for querying data lake files directly.
- Using `OPENROWSET` function to query Parquet files in Silver container (Lakehouse concept).
- Creating views in Synapse to abstract direct file queries.
- Creating external tables using:
    - Database scoped credentials (managed identities).
    - External data sources (pointing to ADLS containers).
    - External file formats (Parquet with compression).
    - Using `CREATE EXTERNAL TABLE AS SELECT (CETAS)` to materialize data in Gold container.

- Understanding the difference between views (virtual) and external tables (materialized).
- Validating data in Gold layer storage.

Highlights:

- Deep dive into lakehouse architecture: combining data lake storage with data warehousing capabilities.
- Serverless SQL pools for cost-effective querying without data duplication.
- Security best practices with managed identities and role assignments.
- Using SQL for data management and external table creation.
- Preparing data for consumption by business stakeholders.

Phase 4: Data Visualization Using Power BI

Key Steps:

- Download and installation of Power BI Desktop.
- Retrieving Synapse Serverless SQL endpoint URL.
- Establishing connection from Power BI to Synapse workspace using the SQL endpoint.
- Authentication with SQL credentials created during Synapse setup.
- Importing data from external tables and views.
- Creating simple visualizations:
    - Line charts to show trends over time.
    - Pie charts for category distributions.
    - Cards for KPI metrics.
- Formatting visuals and creating dashboards.
- Exporting reports for stakeholders.

Highlights:

- Demonstrates end-to-end data engineering workflow including delivery.
- Focus on establishing and managing connections between Azure services and BI tools.
- Building quick, insightful visualizations directly on cloud data without local storage.

Prerequisites & Environment Setup

- Hardware: Laptop or PC with stable internet.
- Accounts: Azure free account (with $200 credits for 30 days) for provisioning resources.
- Tools: Azure portal access, Azure Data Factory, Azure Databricks, Azure Synapse Analytics, Power BI Desktop.
- Software: VS Code or any IDE for JSON and notebook editing.
- Skills: Basic knowledge of Azure services, Python/PySpark, SQL.

Key Learnings & Best Practices

- Importance of architecture planning before implementation.
- Use of Medallion architecture (Bronze-Silver-Gold) for scalable, maintainable data pipelines.
- Building dynamic, parameterized pipelines in Azure Data Factory to handle multiple data files efficiently.
- Leveraging Azure Databricks for scalable big data transformations using Spark.
- Secure data access via Managed Identities and Service Principal authentication.
- Utilizing Lakehouse architecture via Synapse Serverless SQL pools for cost-effective querying.
- Creating external tables and views to organize and serve data for analytics.
- Integration of Power BI with Synapse for real-time visualization.
- Handling real-world interview scenarios and questions related to Azure data engineering.
- Debugging and iterative development mindset, embracing errors as learning opportunities.

Conclusion

This project offers a robust, practical guide to mastering Azure-based data engineering workflows, from data ingestion to transformation, warehousing, and visualization. It equips learners with real-time scenarios, dynamic pipeline construction, advanced Spark transformations, secure cloud resource management, and end-to-end data delivery skills.