LEARN

AZURE DATA FACTORY (ADF)

PART-1

C .R. Anil Kumar Reddy

www.linkedin.com/in/chenchuanil

WHAT IS AZURE DATA FACTORY?

Azure Data Factory (ADF) is a cloud-based data integration service provided by Microsoft as part of its Azure cloud platform. It allows users to create, schedule, and orchestrate data workflows and pipelines for moving, transforming, and integrating data from various sources to desired destinations.

KEY FEATURES OF ADF

1. Data Movement:

- ADF can connect to a wide range of on-premises and cloud-based data sources (e.g., databases, files, APIs) and move data between them.

2. Data Transformation:

- ADF integrates with services like Azure HDInsight, Azure Databricks, and SQL Server to allow complex data transformations during data integration workflows. It also has built-in transformation activities like data mapping, filtering, and aggregating.

3.ETL/ELT Process Automation:

- ADF is commonly used for Extract, Transform, Load (ETL) and Extract, Load, Transform (ELT) processes. It allows scheduling and automating these data workflows efficiently.

4.Integration with Other Azure Services:

ADF works seamlessly with other Azure services like Azure Data Lake, Azure Synapse Analytics, Azure Blob Storage, and Azure SQL Database, making it a powerful tool for handling large-scale data processing.

5. Pipeline Orchestration:

You can create data pipelines in ADF that define activities such as data extraction, transformation, and loading in a specific sequence. These can be scheduled to run on a specific trigger or time schedule.

6. Data Monitoring and Management:

ADF provides monitoring capabilities, where you can track the progress of your data workflows, identify failures, and troubleshoot issues with data integration.

USE CASES

- **Data Integration:** Consolidating data from different sources for analytics, reporting, or operational systems.

- **Big Data Processing:** Orchestrating large data workflows in big data environments, often in collaboration with tools like Azure Databricks.

- **Data Migration:** Moving data from on-premises databases to cloud storage or other databases during cloud adoption.

- **Data Transformation and Cleansing:** Preparing data for use in business intelligence, machine learning, or reporting applications.

ADVANTAGES OF ADF

- **Scalability:** It can handle small and large-scale data workflows.

- **No-Code and Code-Based Capabilities:** You can use ADF's drag-and-drop interface for simple workflows, or use custom code for more complex data transformations.

- **Cloud-Native:** As part of Azure, it seamlessly integrates with other Azure services, making it ideal for cloud data engineering solutions.

It's widely used by organizations that need to handle complex data integration tasks in the cloud.

KEY COMPONENTS OF ADF

1. Pipelines

- **Definition:** A pipeline is a logical grouping of activities that together perform a task. A pipeline allows you to manage and schedule workflows of data movement and transformation.

- **Use Case:** For instance, you may have a pipeline that takes data from a database, processes it in Azure Databricks, and then stores it in a data warehouse.

2. Linked Services

- **Definition:** Linked services are connections to external data stores or compute resources. They define the connection information needed to connect to external sources (databases, APIs, cloud storage) or services (Databricks, HDInsight, Machine Learning).

- **Use Case:** Linked services work as the configuration point to authenticate and connect ADF with sources such as Azure Blob Storage or an on-premises SQL Server.

3. Activities

Definition: Activities are the individual tasks that get executed within a pipeline. ADF supports different types of activities like data movement (Copy Activity), data transformation, control flow activities (like If-Else, ForEach, and Wait), and external services execution.

Types of Activities

- **Data Movement Activity:** Used to move data from one source to another (e.g., Copy Activity).
- **Data Transformation Activity:** Uses services like Azure Databricks, HDInsight, or SQL to transform the data.
- **Control Flow Activities:** Manage the logical flow of the pipeline (e.g., If, Switch, ForEach).
- **External Activities:** Allows running external processes like Azure Functions, Databricks notebooks, or HDInsight clusters.

4. Datasets

- **Definition:** A dataset represents the data you want to use in your activities. It specifies the data structure and location (e.g., a table in a database or a file in a storage account).

- **Use Case:** Datasets allow you to define where your input and output data will come from, such as Azure SQL Database, Blob Storage, or any other supported data store.

5. Linked Services

- **Definition:** Linked services are connections to external data stores or compute resources. They define the connection information needed to connect to external sources (databases, APIs, cloud storage) or services (Databricks, HDInsight, Machine Learning).

- **Use Case:** Linked services work as the configuration point to authenticate and connect ADF with sources such as Azure Blob Storage or an on-premises SQL Server.

6. Triggers

- **Definition:** Triggers allow you to automatically initiate the execution of a pipeline. Triggers can be time-based (scheduled) or event-based.

Types of Triggers

- **Schedule Trigger:** Initiates a pipeline at a predefined time.
- **Event Trigger:** Responds to events like the arrival of new data in a data store (e.g., Blob Storage).
- **Manual Trigger:** A pipeline can also be triggered manually from the Azure portal.

7. Integration Runtime (IR)

- **Definition:** The Integration Runtime is the compute infrastructure used by ADF to perform data movement, data transformation, and other activities.

There are three types:

- **Azure IR:** For data movement and transformation within the Azure environment.
- **Self-Hosted IR:** For connecting to on-premises data sources or to perform hybrid data movement.
- **SSIS IR:** To run SSIS packages in a fully managed Azure environment.
- **Use Case:** You can use a self-hosted IR to securely transfer data between an on-premises database and Azure storage.

8. Mapping Data Flows

- **Definition:** Mapping Data Flows allow you to visually design data transformations in ADF without needing to write code. They provide a graphical interface to create transformations like joins, aggregations, pivots, and data cleansing.

- **Use Case:** When you need to process large-scale data transformations without writing code, Mapping Data Flows can help you create data transformation logic that ADF will execute on Spark clus

9. Monitoring & Alerts

- **Definition:** ADF offers built-in monitoring capabilities that allow you to track pipeline executions, success rates, and any failures. You can set up alerts based on custom conditions.

- **Use Case:** You can monitor the progress of your pipelines and get notified if any failure occurs or if execution is delayed.

10. Parameters

- **Definition:** Parameters allow you to pass values into a pipeline at runtime. This enables dynamic behavior within pipelines and activities.

- **Use Case:** For instance, you could use parameters to specify the file path for data loading, making your pipelines reusable for multiple data sources.

11. Variables

- **Definition:** Variables are used to store temporary values within a pipeline. They can be set or modified during pipeline execution.

- **Use Case:** Variables can be used to store intermediate results, like a computed value or loop index, for use in subsequent activities.

Example of How It All Comes Together:

Imagine you want to move data from an on-premises SQL Server to Azure Blob Storage, transform it using Databricks, and then load it into an Azure SQL Database.

- Pipelines would orchestrate the entire process.
- Activities would move, transform, and load the data.
- Datasets would define the data sources and destinations.
- Linked Services would connect ADF to the SQL Server, Blob Storage, and Azure SQL Database.
- Integration Runtime would allow you to securely access the on-premises SQL Server.
- Triggers would schedule the pipeline to run daily.

These components work together to build a complete data integration workflow in Azure Data Factory.

WHAT IS A BLOB STORAGE

A blob is a short form of a Binary Large Object. It allows users to store large amounts of unstructured data on Microsoft's data storage platform. which includes objects such as images, videos, pdf files, documents, text files, etc. These are known as unstructured data because they don't follow any particular data model.

·Serving images or documents directly to a browser.
·Storing files for distributed access.
·Streaming video and audio.
·Storing data for backup and restore, disaster recovery, and archiving.
·Storing data for analysis by an on-premises or Azure-hosted service.

Azure offers three types of blob service:

Block blob: Block blob stores data in the form of blocks. One block blob can store only 50000 blocks with the size of each block blob up to 4000 MB. The size of each block can be different from the other blocks in the same block blob. Each block is assigned with a unique block id. The block blobs are suitable for a large amount of data that needs frequent accessibility with optimal performance like video or audio streaming websites.

Append Blob: Append blobs are specifically designed for use with append operations. The most common use of an append blob is for storage and updating of log files. Blocks may be appended to the end of an append blob, but previously existing blocks may not be modified or deleted, just as with block blobs, an append blob may contain up to 50,000 blocks, each up to 4 MiB.

Page blob: Data stored in page blob in the form of 512-byte pages and not in the form of blocks. The page blob is suitable for creating disk subsystems for Azure virtual machines on top of Azure blob storage in premium storage account types where we can run transactional workloads that require frequent read and write access. The maximum size of a page blob is 8TB. This blob type supports only the hot access tier and not cool or archive access tiers.

Blob storage pricing is divided into 3 tiers

1. HOT: Hot storage is for the most frequently accessed data. It has the highest storage cost but the lowest storage cost of the three tiers. Hot blob storage is always online.

2. COOL: Cool storage is for data that is periodically accessed, but not with great frequency. Specifically, cool storage is appropriate when data is not accessed more than once every 30 days. Cool storage has a lower storage cost than hot storage, but higher transaction costs. As with hot storage, cool storage data is always online.

3. ARCHIVAL: Archival storage is for data that requires long-term storage and very infrequent access (i.e., less than once every 180 days). It has the lowest storage cost and the highest transactional cost. Archival data storage is always offline.

KEY VAULTS

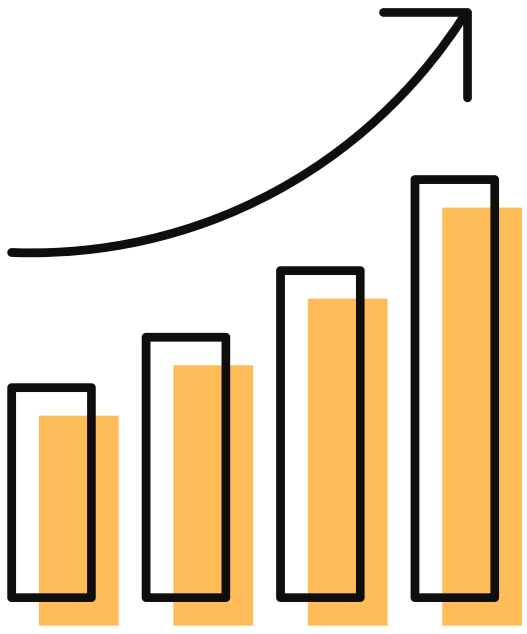
Key vault used to maintain our secrets, Azure Key Vault is a tool for securely storing and accessing secrets. A secret is anything that you want to tightly control access to, such as API keys, passwords, or certificates. Key vault is a logical group of secrets.

What are Keys and Secrets

The "key" is your user ID, and the "secret" is your password. They just use the "key" and "secret" terms because that's how they've implemented it.

ANIL REDDY CHENCHU

***Torture** the data, and it will confess to anything*



DATA ANALYTICS

Azure Data Factory

SQL

databricks

APACHE Spark

Azure Synapse Analytics

Power BI

SHARE IF YOU LIKE THE POST

Lets Connect to discuss more on Data

www.linkedin.com/in/chenchuanil