MICROSOFT FABRIC GLOBAL ONLINE CONFERENCE

Sep 10-11

Utilizing Microsoft Fabric for Metadata-Driven Pipelines

Alpa Buddhabhatti

https://microsoftfabric.global

About Me

Alpa Buddhabhatti

She/Her

Azure Data Engineer/Trainer

alpabuddhabhatti

@AlpaB7

alpaBuddhabhatti/MetadataDrivenPipeine

@meetalpa

@alpabuddhabhatti

Microsoft Data Platform (MVP)

Microsoft Certified Trainer(MCT)

Azure Developer(AZ-204)

Azure Data Engineer(DP-203)

Azure Data Scientist(DP-100)

Azure Administrator(AZ-104)

Agenda

1. Overview of a Metadata-Driven Pipeline Approach
2. Demo
3. Conclusion
4. Q&A

1. Overview of a Metadata-Driven Pipeline

Metadata Driven Approach – What

- Data about data OR the structure of business output

- It answers :
  - What information is contained in the business outcome or data?
    1. Source and Sink data source (e.g. SFTP, Blob, OneLake, etc)
    2. Data sources Types (e.g. file, table)

Metadata Driven Approach – What

- What processes were applied to the data related to your business outcome or data?

    1. Operations like
        - Delta loads (only loading changes)
        - Full loads (loading all data)
        - Deletes, Updates, and any other data transformation
    2. Email Functionality

Control information regarding a specific Business output or requirement for your workflows, pipelines, SSIS Packages, etc

- How is the data processed and transformed?

    Where data store after each steps during ETL or ELT. (e.g Data storage location or system)

Metadata Driven Approach – When?

1. Incremental Load and transformation
2. Delta Load and transformation
3. Based on operation - Executing store procedure, update table, delete data
4. How to load multiple tables ,files, APIs, etc using a single or few pipeline

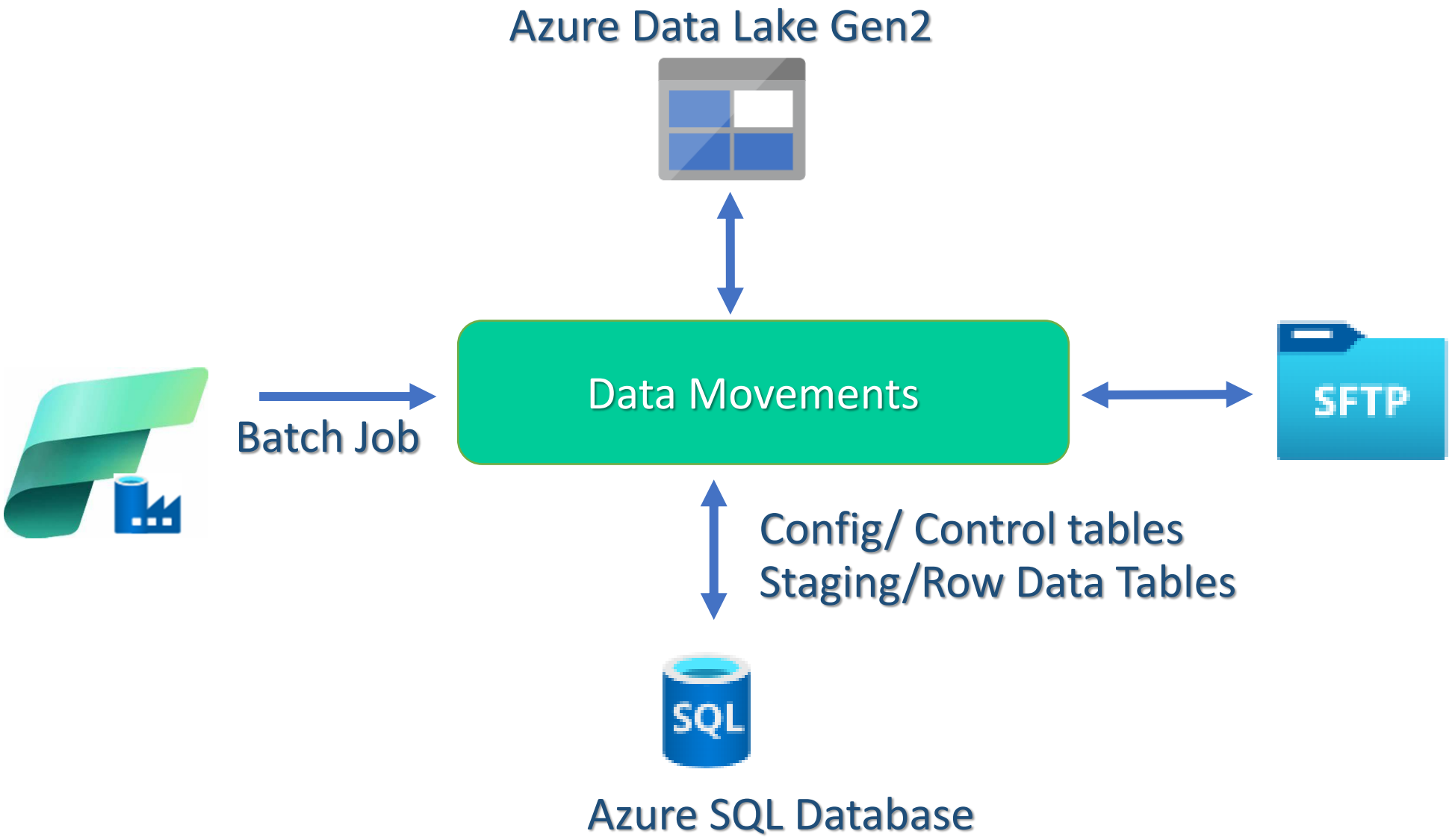
Metadata-driven approaches provide a way to define, manage, and automate these processes, making data integration and transformation more flexible and efficient

Metadata Driven Approach – Why

1. Easy to Maintain
2. Ready for future Features
3. Acceleration/ Quick development cycle
4. Reduces Mistakes
5. Uniformity
6. Scalability
7. Auto Documents

2.DEMO

Demo

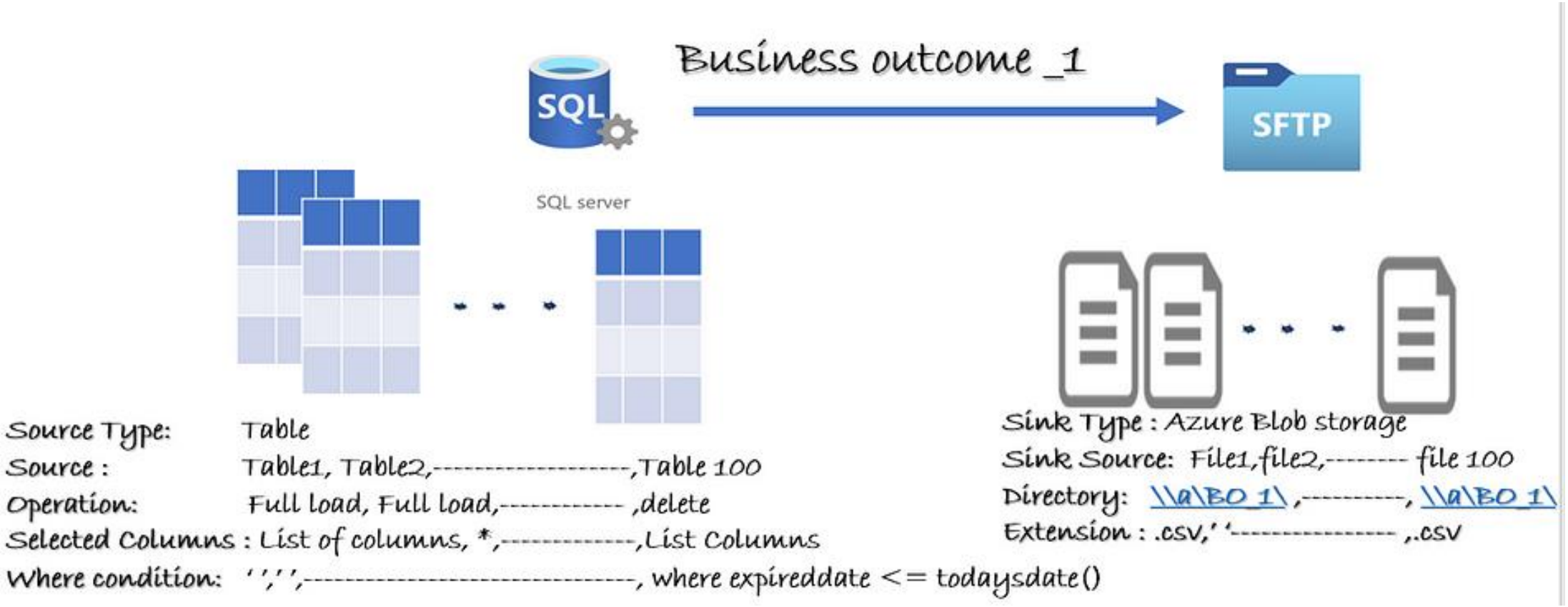

Azure Data Lake Gen2
Batch Job
Data Movements
SFTP
Config/ Control tables
Staging/Row Data Tables
SQL
Azure SQL Database

Metadata Driven Approach – BO1?

BO1 : Football

- ✓X Number of tables
- ✓Tables saved as .csv to SFTP
- ✓Full Load for tables
- ✓Load only selected columns
- ✓Different target locations
- ✓Delete operation based on filter conditions (for Y Number of table)
- ✓Should be scalable with minimal code/config
- ✓No email functionality

Metadata Driven Approach – BO1

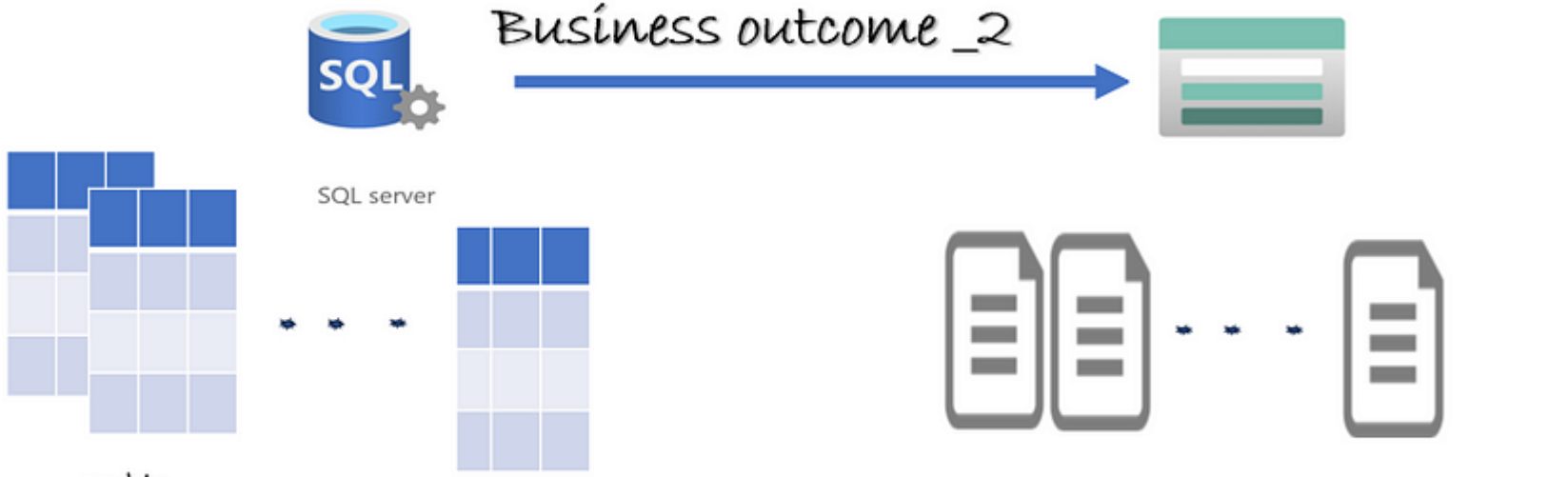


Metadata Driven Approach – BO2

BO2 : Cricket

- ✓X Number of tables
- ✓Tables saved as .csv to Blob Storage
- ✓Full Load
- ✓Should be scalable with minimal code/config
- ✓No Email Functionality

Metadata Driven Approach : BO2



Source Type: Table
Source : Table1, Table2,----------,Table 100
Operation: Full load, Full load,---------- ,Full load
Selected Columns :List of columns, *,----------,List Columns
Where condition: ' ',' ',----------,' '

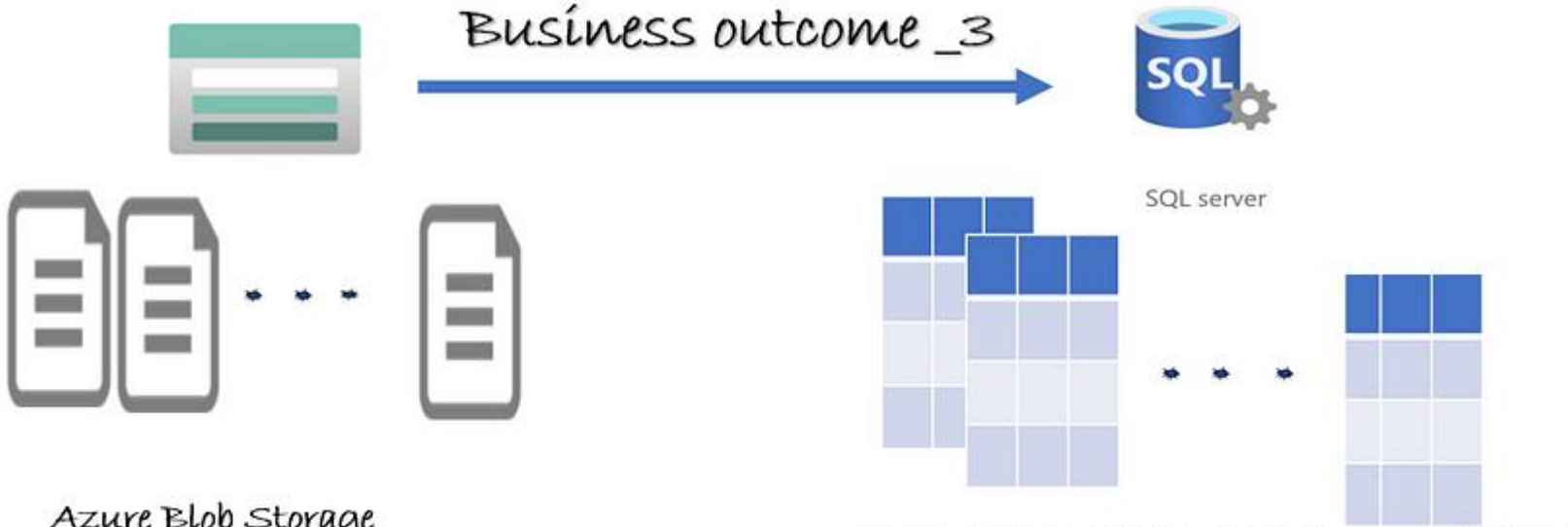
Sink Type : Azure Blob storage
Sink Source: File1,file2,-------- file 100
Directory: ' ',' ',----------,' '
Extention : .csv,----------,.csv

Metadata Driven Approach – BO3

BO3 : Movies

- ✓X Number SQL table /Files
- ✓Tables saved as .csv to Blob Storage
- ✓Load only selected columns
- ✓Delta Load
- ✓Should be scalable with minimal code/config
- ✓Email Functionality for Delta Load

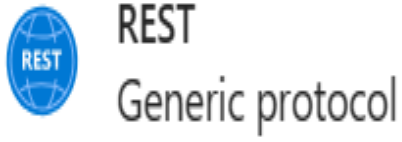
Metadata Driven Approach : BO3



Let's see a quick demo using Microsoft Fabric

Metadata Driven Approach : BO4

- DELTA Load
- Email Functionality
- bo4-in
- URL



Source Data Source

Azure Blob Storage

Sink Data Source

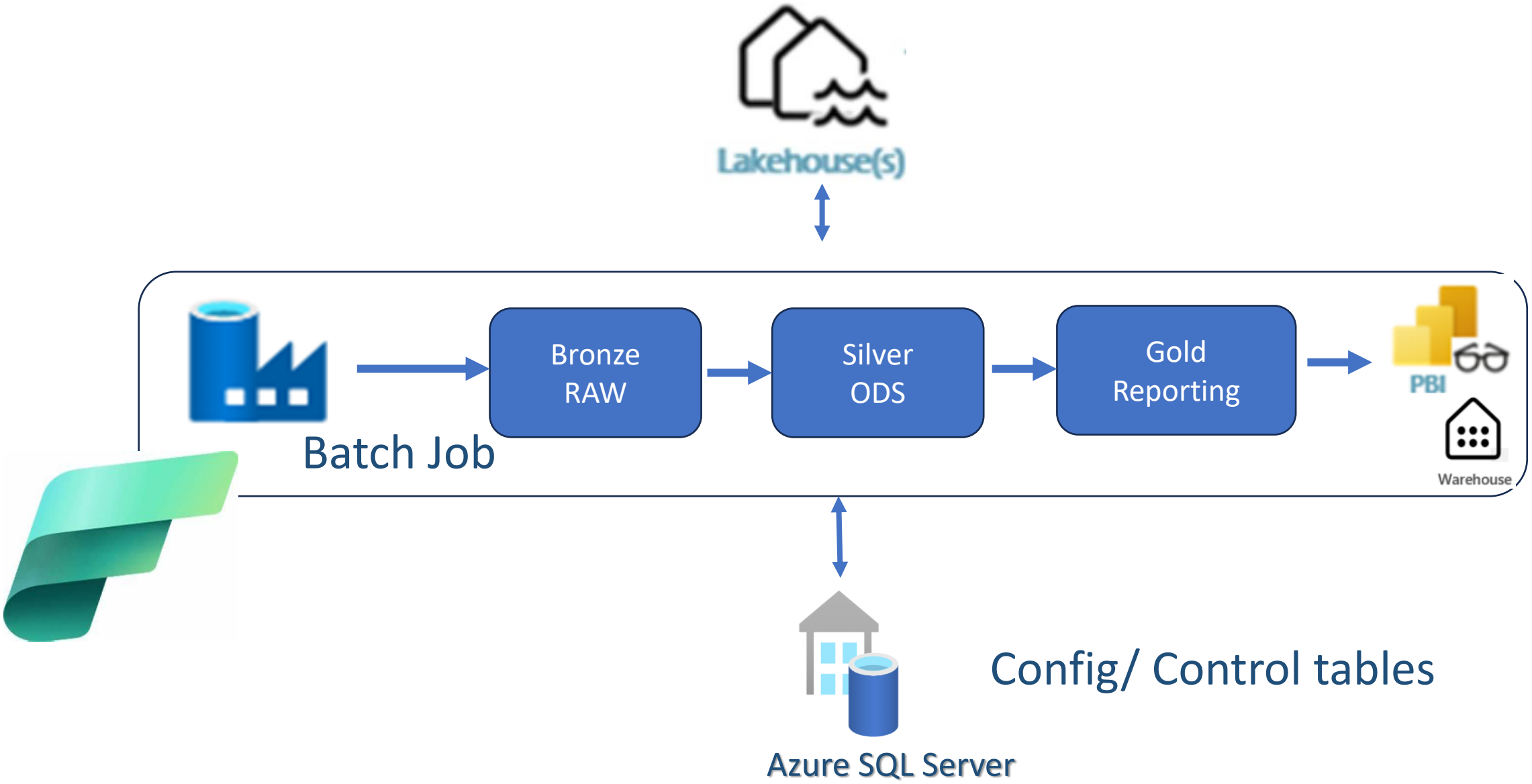
What should you do to make it work? Should you create a new pipeline or use the existing pipeline? Do you need to change the config table? Do you need to modify the control table for the delta load?

Metadata Driven Approach : BO4

```
CREATE TABLE Delta_Load_Control (
  SourceSystem VARCHAR(100),   -- E.g., 'REST_API'
  LastRunTimestamp DATETIME,  -- Last successful extraction timestamp
  MaxProcessedID INT,                -- Max ID fetched for incremental fetch (if API uses IDs)
  Status VARCHAR(50),                -- Status of the last extraction (Success, Failed, In-Progress)
  URL VARCHAR(500),                  -- The API endpoint (if dynamic)
  Notes TEXT                         -- Any additional notes or error details
);
```

Pipeline modification (if it is not handling before this senaris)

Metadata Driven Approach – BO5



Metadata Driven Approach – BO5

Create new config table for Silver and Gold Layer having a specific information related to that layer

e.g.

Silver layer :

SpeccodeId, Source , Target, ValidationRules, CleansingRules, more etc

Gold Layer:

SpeccodeId, Source, LastRefreshed, AggregationRules, CalculationRules, KPI_Definitions, etc.

Azure Resources

1. Microsoft Fabric Azure Data Factory
2. Azure SQL Database (or Fabric Warehouse)
3. Azure Blob Storage (or Fabric Lakehouse)

Conclusion

✓**Metadata Driven pipeline**

- Save your time & cost
- Scalability
- Reusability
- Unified code

Useful Resources Links

Environment Set UP:

Create your own Microsoft Fabric environment - Kevin Chant (kevinrchant.com)

A Beginner’s Guide to Microsoft Fabric | by alpa buddhabhatti | Sep, 2023 | Medium

Microsoft Fabric

Get started with Microsoft Fabric - Training | Microsoft Learn

Metadata Driven Pipeline in Fabric

Step-by-Step Guide for Building Metadata Driven Pipelines in Microsoft Fabric - Microsoft Community Hub

DEMO Resources

GitHub Code Repo:

alpaBuddhabhatti/MetadataDrivenPipeine (github.com)

Blog:

alpa buddhabhatti – Medium

Thank you

Alpa Buddhabhatti

She/Her

Azure Data Engineer/Trainer

alpabuddhabhatti

@AlpaB7

alpaBuddhabhatti/MetadataDrivenPipeine

@meetalpa

@alpabuddhabhatti

Microsoft Data Platform (MVP)

Microsoft Certified Trainer(MCT)

Azure Developer(AZ-204)

Azure Data Engineer(DP-203)

Azure Data Scientist(DP-100)

Azure Administrator(AZ-104)