Striking Gold: Mastering Medallion Architecture for Bronze, Silver, and Gold Tiers

RK Iyer · Follow

7 min read · Dec 17, 2023

Listen Share More

BEST PRACTICES & TIPS FOR TO DESIGN MEDALLION ARCHITECHTURE

✐ Co-Author — Sen Sayantani

Photo by dirk von loen-wagner on Unsplash

❑ Background

Data is the driver for any modern business offering insights into customer interactions, enabling optimization for both end-users and increased profitability, necessitating the creation of Data Lake/Lakehouse. While constructing a Lakehouse architecture, **meticulous planning** of the data architecture is crucial to logically **organize data** before it lands in the Lakehouse. This strategic **organization and governance** play a pivotal role in preventing data swamps.

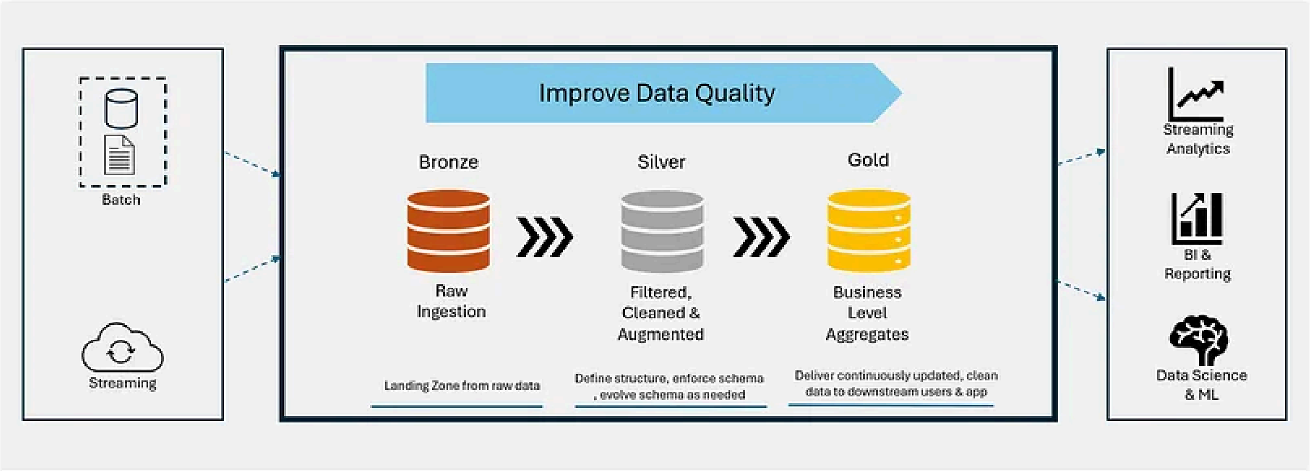
Several critical questions need answers during the planning phase: How to **segment data into multiple layers** such as Bronze (raw), Silver (validated), and Gold (enriched)? Which **file formats** are optimal for each layer? What should be the **structure within each layer**? How can the Lakehouse be effectively **secured and governed**?

Many organizations have adopted the **Medallion architecture** in various forms to get answers to above questions. While the concept seems **straightforward** in theory, the **nuances become apparent during implementation**. In this blog, I aim to offer **guidance to those embarking on their Lakehouse journey**. Drawing from my experiences in implementing data lakes, I will cover fundamental concepts and considerations to assist you in building a robust data foundation.

❑ What is a medallion architecture?

A medallion architecture is a **data design pattern** used to **logically organize data in a lake house**, with the **goal of** incrementally and progressively **improving the structure and quality of data** as it flows through each layer/zone of the architecture.

> *It is a "multi-hop architecture" organized from **Bronze ⇒ Silver ⇒ Gold** so is also referred to as **"multi-hop"** architectures.*



Medallion architecture

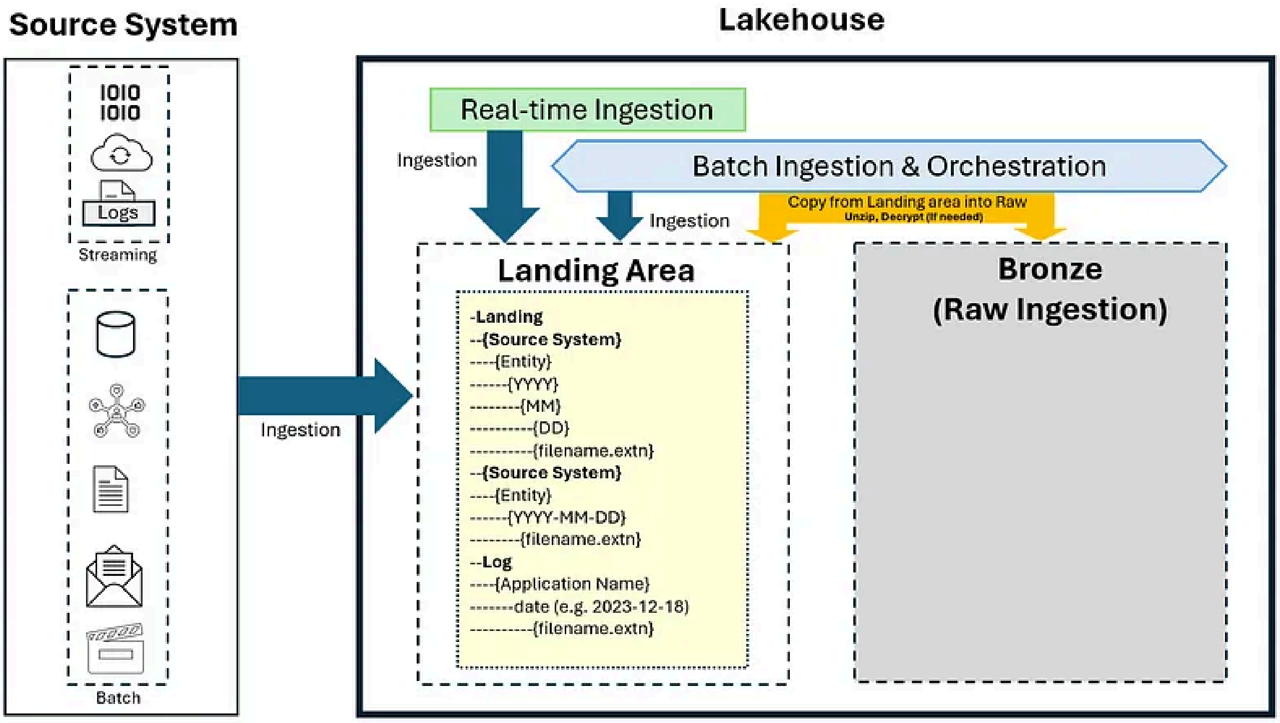
This architecture guarantees **ACID (Atomicity, Consistency, Isolation, and Durability)** as data passes through multiple layers of validations and transformations before being stored in a layout optimized for efficient analytics.

A layer/zones do **not** always need to reside in the **same physical data lake** and could also reside as separate **filesystems, storage accounts, or even in different subscriptions**. e.g. Multiple storage accounts in different subscriptions may be a good idea for large petabyte-scale data lake or with throughput requirements exceeding a request rate of 20k per second.

❑ Landing/Staging Zone (Optional)

A landing/Staging zone serves as a **temporary/transient storage** location for data gathered from various sources before transferring into the bronze layer. While ingesting the data from source system, the data is not directly pushed into raw layer but follows a 2-step process -

1. Push data into landing zone.
2. Pull data from landing zone to bronze zone after decrypting, unzipping etc. if needed.



Landing/Staging Zone

This layer **decouples the direct ingestion from the source system into the raw layer**, providing an **additional control** for pulling the data into the raw layer on a particular time schedule or based on a particular event **(e.g. only load if a group of 3–4 files arrive).** It provides a buffer in the **event of any intermittent failure or incorrect data being sent from the source system**. It also **facilitates reprocessing into the raw layer when necessary**.

This layer plays an important role when the **responsibility of sending data** to the data lake is of **source system** through daily extract e.g. external client pushing the daily/hourly data or data is pulled using a Rest API from 3rd party applications or if the data is encrypted and needs to be pushed to the bronze layer after decrypting it into Lakehouse.

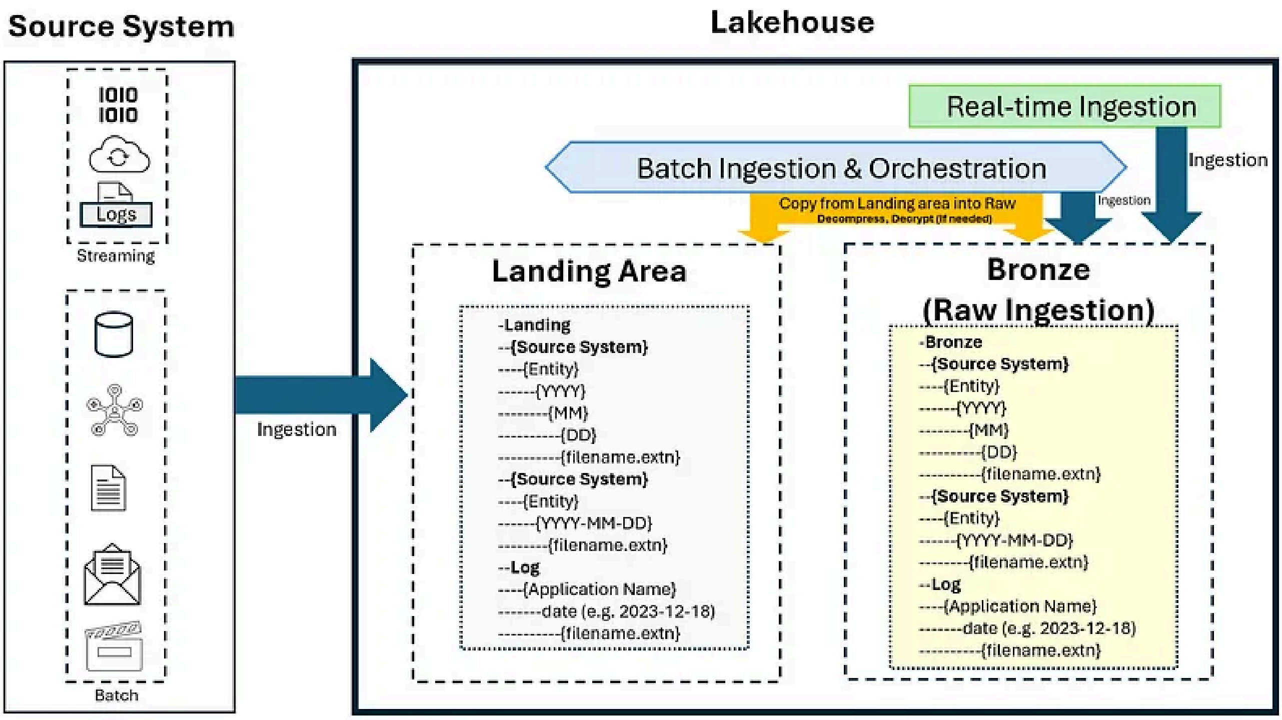
Data that is stored in Landing/Staging zone has usually the following **key characteristics:**

- Typically, the data for **last 3-7 days** are stored in this layer. Lifecycle management can be used to automate the removal of data more than 3–7 days.
- This is useful for **batch as well as streaming ingestion**. e.g. of streaming ingestion — An event hub capture pushing data using a capture feature creates day-wise hour wise folders. This data is pushed periodically into “Bronze” layer and audited.

- Different file formats ranging from CSV, JSON, XML, Parquet, mp3, jpg, zip etc.

□ Bronze (Raw) Zone

The Bronze (Raw) layer is where we land all the data from external source systems in its **natural and original state.** The table structures in this layer correspond to the source system table structures **AS-IS** along with any additional metadata columns that capture the load date/time, process ID, etc. In this layer you either get data using **full loads or delta loads.**



Bronze (Raw) Zone

The main purpose of this layer is to provide an **historical archive** of source system. It is also useful for **reprocessing**, if needed without rereading the data from the source system.

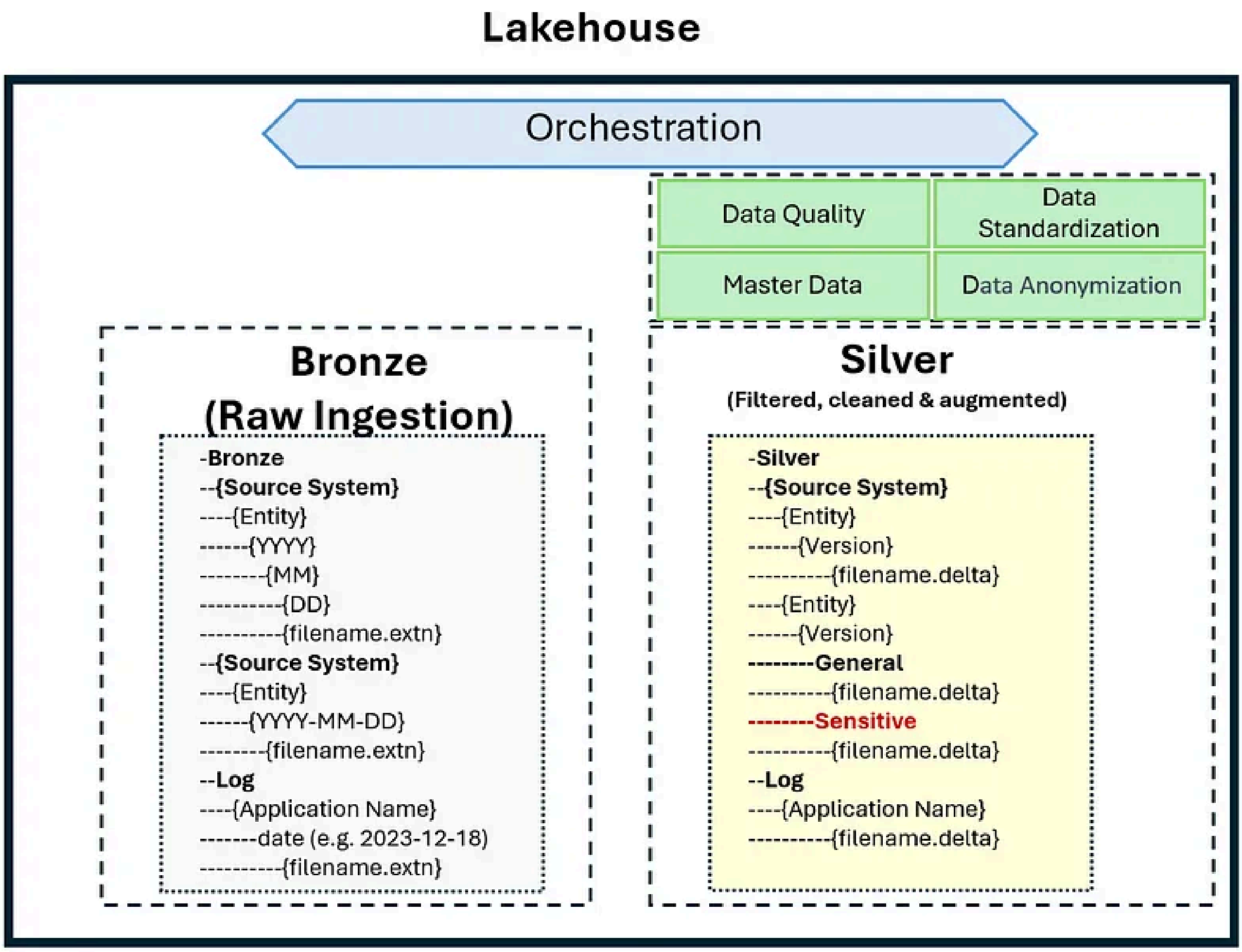
Data that is stored in bronze has usually the **following key characteristics:**

- It contains **unvalidated & immutable data.**
- It contains **structured, semi-structured, or unstructured data** in different file formats ranging from CSV, JSON, XML, Parquet, mp3, jpg, zip etc.
- Managed using **interval partitioned tables**, for example, using a YYYYMMDD or datetime folder structure.

- Retains the **full (unprocessed) history of each dataset** in an efficient storage format, for example, Parquet or Delta.
- It may include **extra metadata, such as schema information, source file names or recording the time data** was processed which can be useful for data lineage.

❑ Silver (Filtered, Cleaned and Conformed data) Zone

The silver zone comprises **validated and enriched data**, prepared **for further analysis**. The organization of this zone is primarily influenced by business considerations, offering an ‘**Enterprise view**’ that encompasses **key business entities, concepts, and transactions** (e.g., master customers, stores, non-duplicated transactions, and cross-reference tables), rather than being dictated solely by the source system. Typically, data is organized into folders based on departments or projects.



Silver (Filtered, Cleaned and Conformed data) Zone

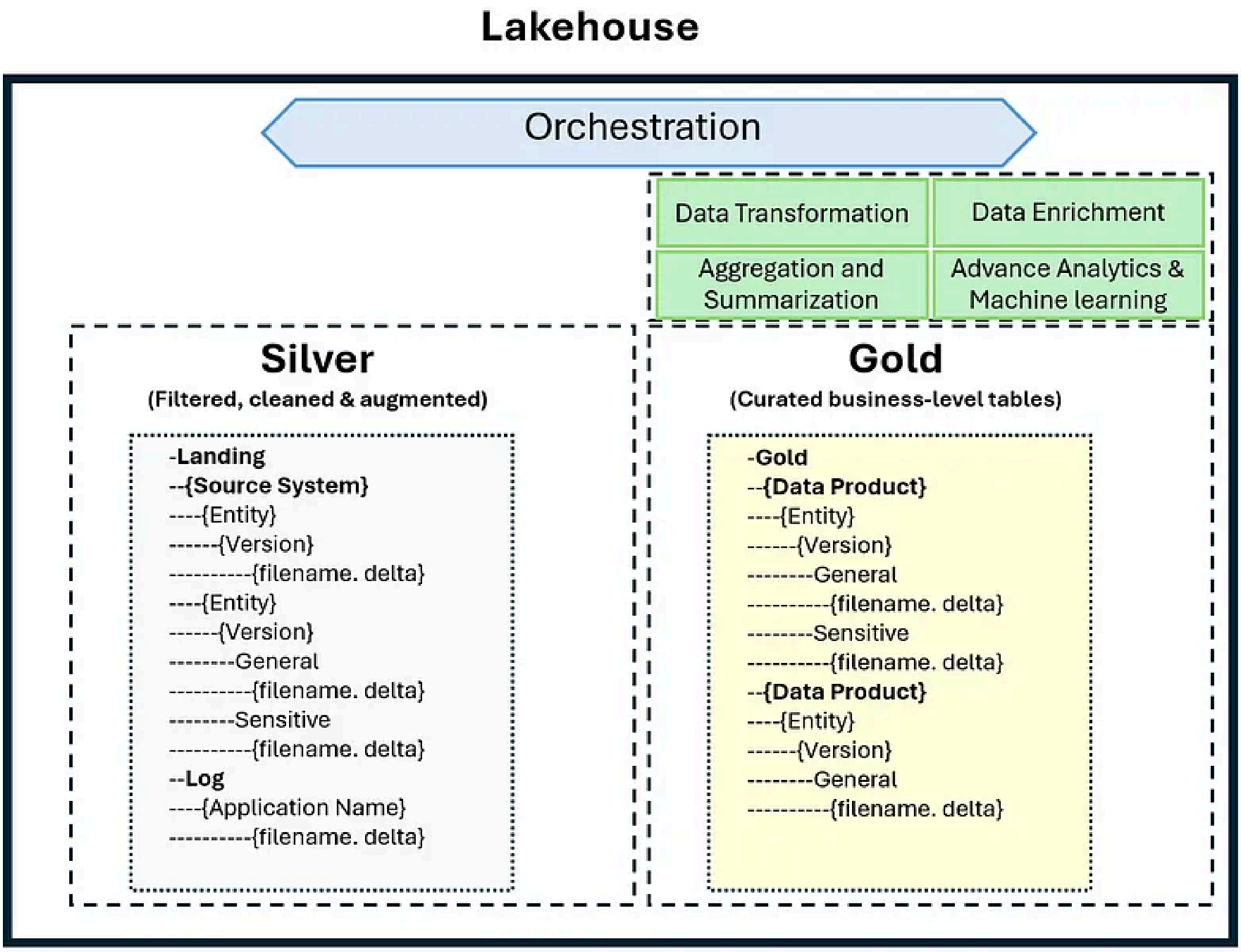
Data that is stored in Silver has usually the following **key characteristics:**

- It contains data in an **optimal storage format**, preferably **Delta** or, as an alternative, **Parquet**.
- Common tasks within this layer include **defining schemas and data types**, **removing unnecessary columns**, **deduplicating** raw data, and applying cleaning (data quality) rules for **validation and standardization,** modifying or **removing personally identifiable information (PII)** from datasets to protect individuals' privacy.
- Additionally, **enrichment** processes may involve merging datasets to enhance the overall value of insights.
- It serves as a **source for Business users, Data Engineers and Data Scientists to answer business problems** via enterprise and departmental data projects in the Gold Layer.

❑ Gold (Curated business-level tables) Zone

The Gold layer within the Lakehouse consists of **meticulously curated and aggregated** data, **formatted into consumption-ready 'project/domain/use case-specific' datastore**. Data from your Silver layer is **transformed into high-value data products with a structure** that are served to your data consumers. It can be served to the consumers as-is, such as data science notebooks, or through another read data store.

This specialized layer is dedicated to reporting and utilizes denormalized, read-optimized data models with a minimized number of joins.



Gold (Curated business-level tables) Zone

Various project-specific presentation layers, such as Customer Analytics, Product Quality Analytics, Inventory Analytics, Customer Segmentation, Product Recommendations, Marketing/Sales Analytics, etc., are housed within this layer. Kimball-style star schema-based data models or Inmon-style Data marts are frequently integrated into the Gold Layer of the Lakehouse.

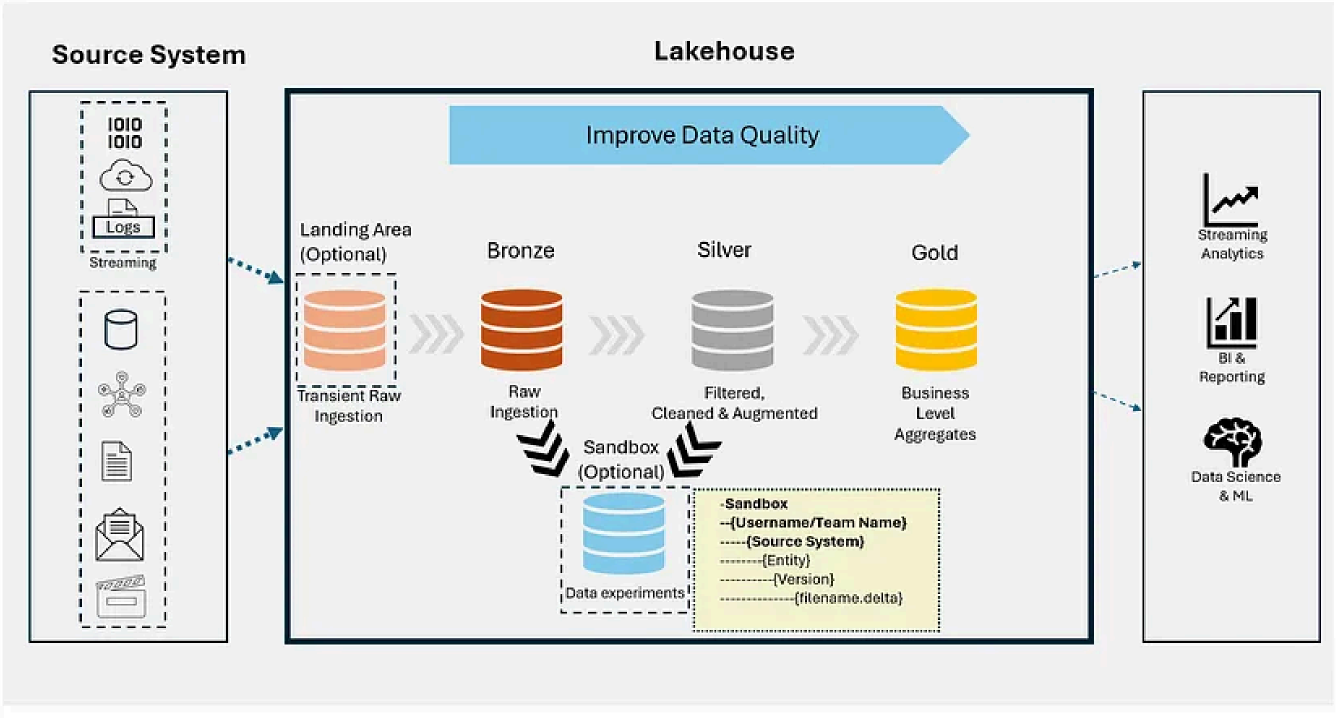
Data that is stored in gold has usually the following **key characteristics:**

- Tables in the Gold layer **encapsulate data** that has undergone transformation for consumption or specific use cases.
- While all tables in the Lakehouse should serve an important purpose, **gold tables represent data that has been transformed into knowledge, rather than just information**.
- It contains data in an optimal storage format, preferably Delta.
- Within the **Gold layer, intricate business rules are implemented**, involving **numerous post-processing activities, calculations, enrichments, and**

optimizations tailored to specific use cases.

- Data in this layer is **highly governed and well-documented.**

❑ Sandbox Zone (Optional)



Sandbox Zone

A Sandbox zone is working area for an individual or a small group of collaborators (**advanced analysts and data scientist's)** to carry out their experiments when looking for patterns or correlations. As an illustration, suppose a data science team aims to identify the most effective product placement strategy for a new region.

In this scenario, they **can integrate additional entities** such as customer demographics and usage data, derived from analogous products in that region. Leveraging the valuable sales insights obtained from this data, the team can then assess the product's market fit and devise an optimal offering strategy.

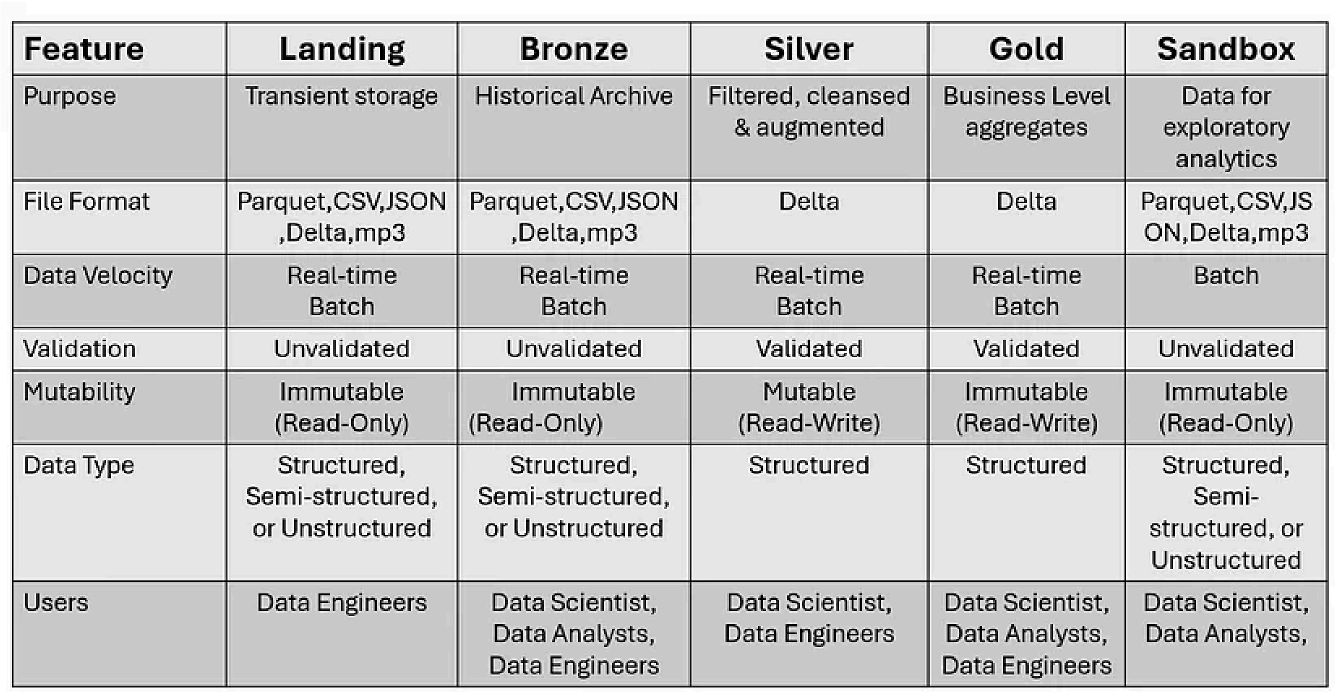
Data that is stored in Sandbox has usually the following **key characteristics**:

- The sandbox layer **provides a space for experimentation and innovation,** fostering a culture of continuous improvement and exploration within the data team.
- These policies **limit the total available storage and how long data can be stored.**

- Bronze & Silver zone provide input to the Sandbox zone.

❑ Comparison of various zones

Please find below the table comparing various zones which we discussed.



Feature	Landing	Bronze	Silver	Gold	Sandbox
Purpose	Transient storage	Historical Archive	Filtered, cleansed & augmented	Business Level aggregates	Data for exploratory analytics
File Format	Parquet,CSV,JSON ,Delta,mp3	Parquet,CSV,JSON ,Delta,mp3	Delta	Delta	Parquet,CSV,JSON,Delta,mp3
Data Velocity	Real-time Batch	Real-time Batch	Real-time Batch	Real-time Batch	Batch
Validation	Unvalidated	Unvalidated	Validated	Validated	Unvalidated
Mutability	Immutable (Read-Only)	Immutable (Read-Only)	Mutable (Read-Write)	Immutable (Read-Write)	Immutable (Read-Only)
Data Type	Structured, Semi-structured, or Unstructured	Structured, Semi-structured, or Unstructured	Structured	Structured	Structured, Semi-structured, or Unstructured
Users	Data Engineers	Data Scientist, Data Analysts, Data Engineers	Data Scientist, Data Engineers	Data Scientist, Data Analysts, Data Engineers	Data Scientist, Data Analysts,

Comparison of various zones

❑ Reference

What is the medallion lakehouse architecture? — Azure Databricks | Microsoft Learn

What is a Medallion Architecture? (databricks.com)

I trust this blog has aided in your comprehension of the fundamental concepts and considerations for constructing a resilient Lakehouse.

Till then … Happy Learning!!!

Please Note — All opinions expressed here are my personal views and not of my employer.

Thought of the moment-

Dream is not that which you see while sleeping it is something that does not let you sleep — Dr. A.P.J Abdul Kalam

Medallion Architecture Data Databricks Lakehouse Architecture Data Lake

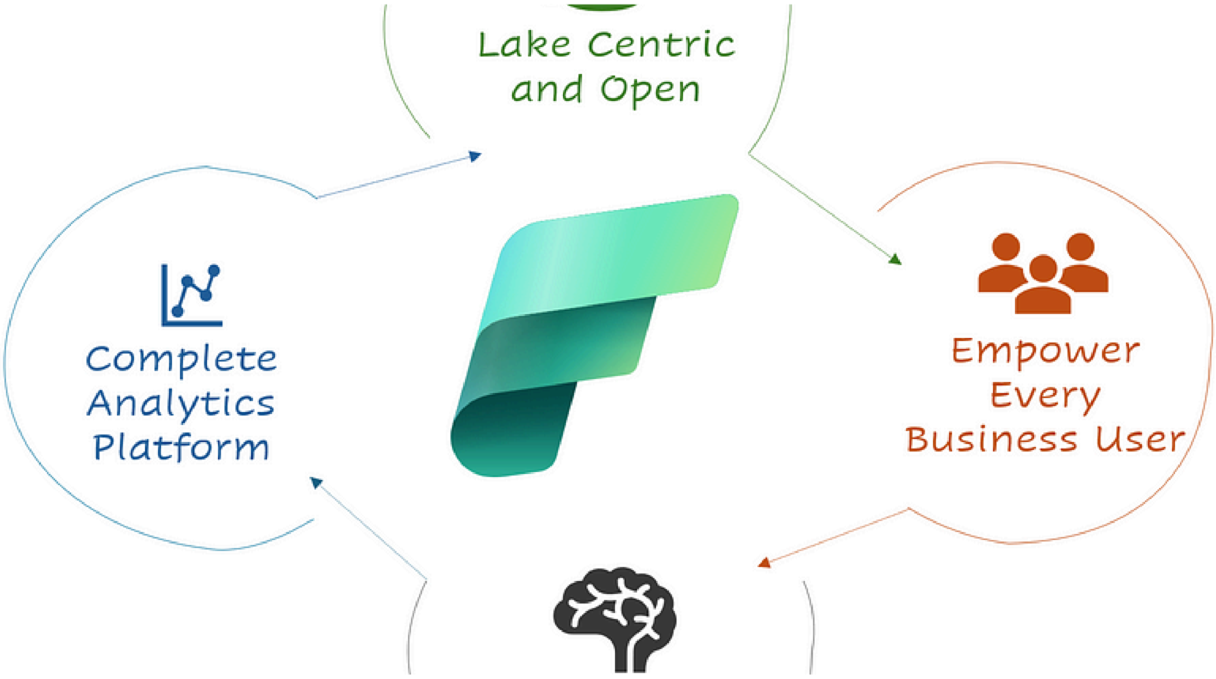
Follow

Written by RK Iyer

210 Followers

Architect@Microsoft, Technology Evangelist, Sports Enthusiast! All opinions here are my personal thoughts and not my employers.

More from RK Iyer



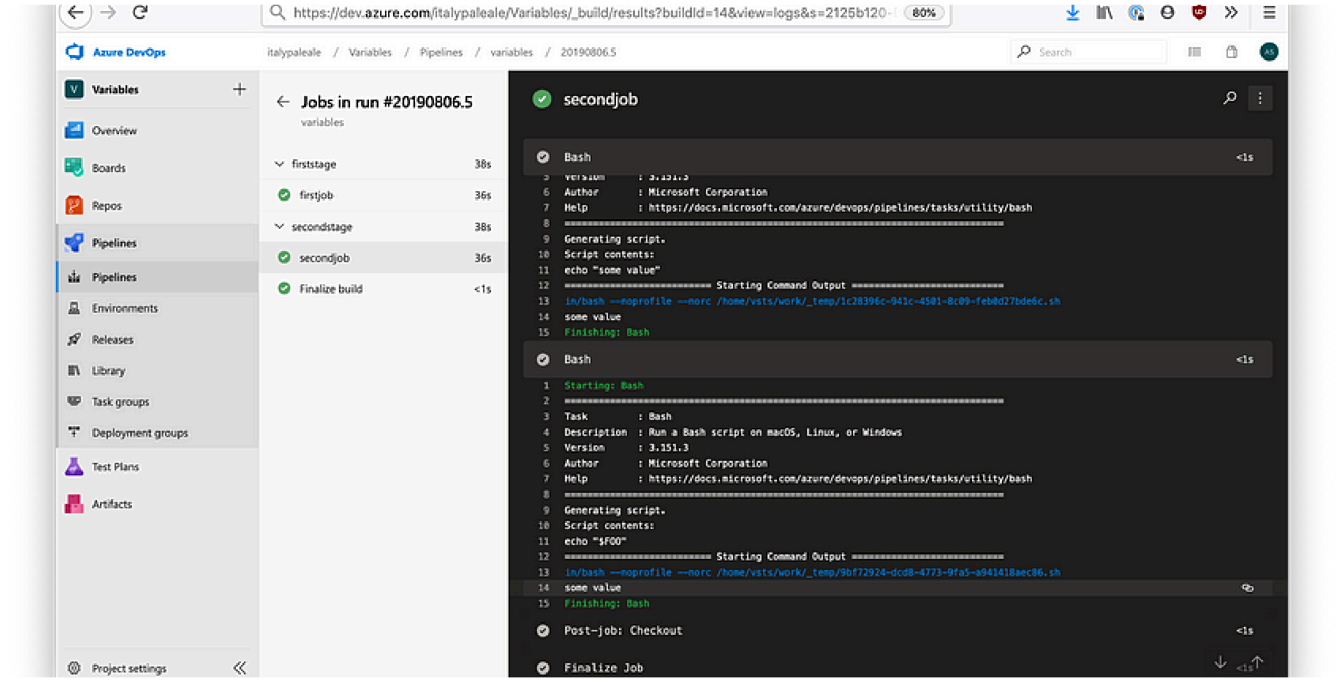
RK Iyer in Microsoft Azure

The Core and Essence of Microsoft Fabric

FABRIC SERIES: FUNDAMENTALS 01— INTRODUCTION TO MICROSOFT FABRIC, WHAT, WHY FABRIC?

6 min read · Mar 3, 2024

80 1



```
Version      : 3.151.3
Author       : Microsoft Corporation
Help         : https://docs.microsoft.com/azure/devops/pipelines/tasks/utility/bash
==============================================================================
Generating script.
Script contents:
echo "some value"
========================== Starting Command Output ===========================
in/bash --noprofile --norc /home/vsts/work/_temp/1c28396c-941c-4501-8c09-feb0d27bde6c.sh
some value
Finishing: Bash
```

```
Starting: Bash
==============================================================================
Task         : Bash
Description  : Run a Bash script on macOS, Linux, or Windows
Version      : 3.151.3
Author       : Microsoft Corporation
Help         : https://docs.microsoft.com/azure/devops/pipelines/tasks/utility/bash
==============================================================================
Generating script.
Script contents:
echo "$FOO"
========================== Starting Command Output ===========================
in/bash --noprofile --norc /home/vsts/work/_temp/9bf72924-dcd8-4773-9fa5-a941418aec86.sh
some value
Finishing: Bash
```

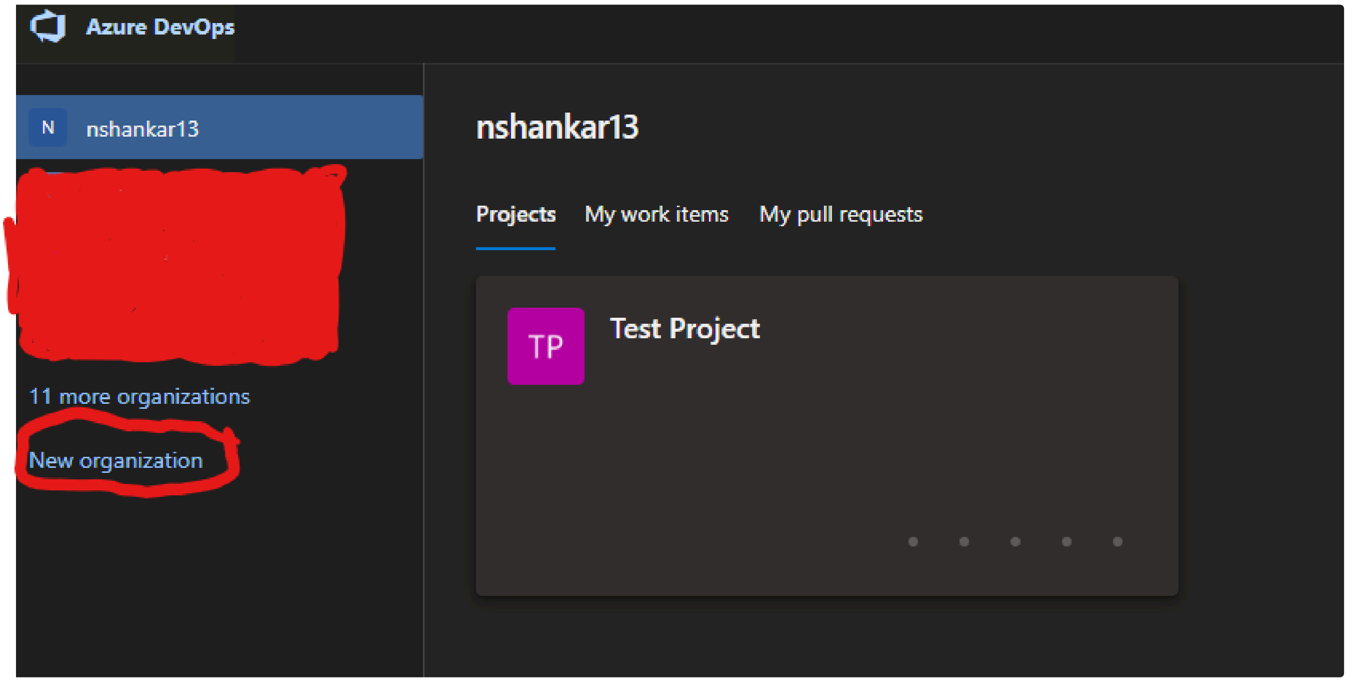
Alessandro Segala in Microsoft Azure

How to pass variables in Azure Pipelines YAML tasks

Passing variables between steps, jobs, and stages: explained

5 min read · Aug 5, 2019

520 9



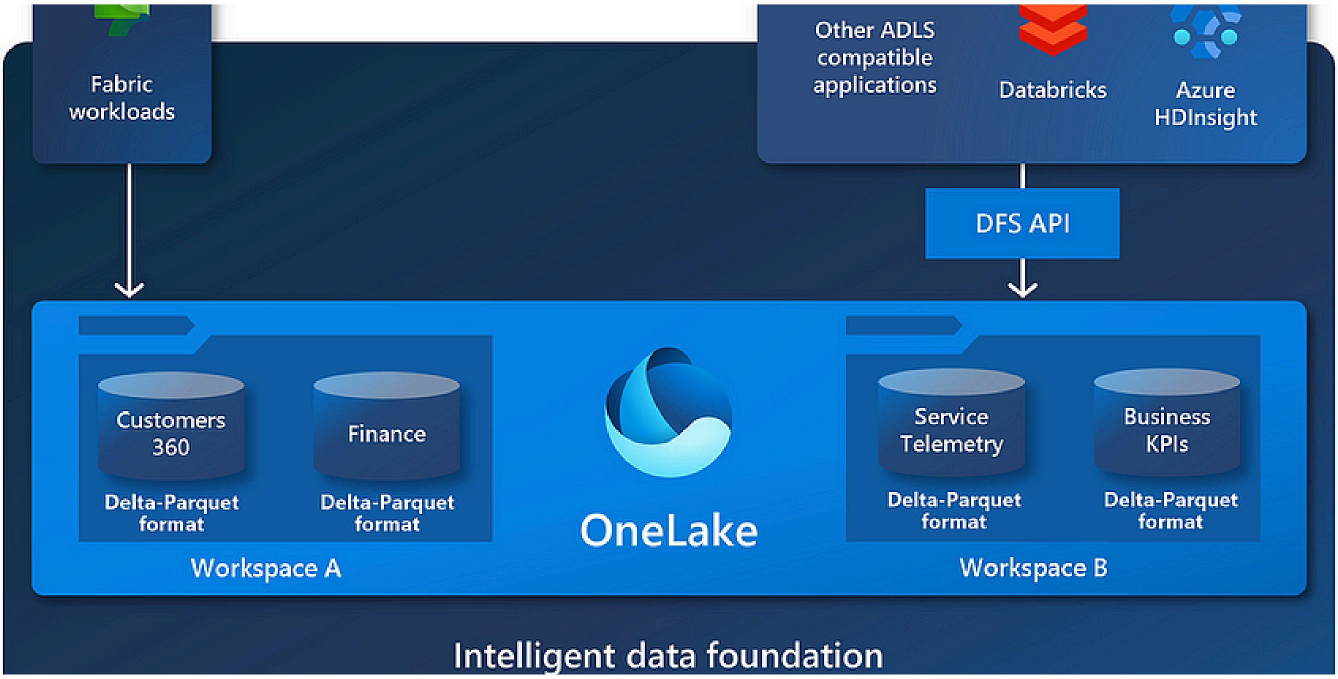
Niranjan Shankar in Microsoft Azure

Creating an Azure DevOps CI/CD Pipeline for your Kubernetes Microservice Application

(Though this guide is tailored towards microservice applications, many of the steps outlined — setting variables, writing the pipeline…

19 min read · Mar 6, 2023

113



RK Iyer

Decoding Microsoft Fabric: An Introduction to Fabric OneLake

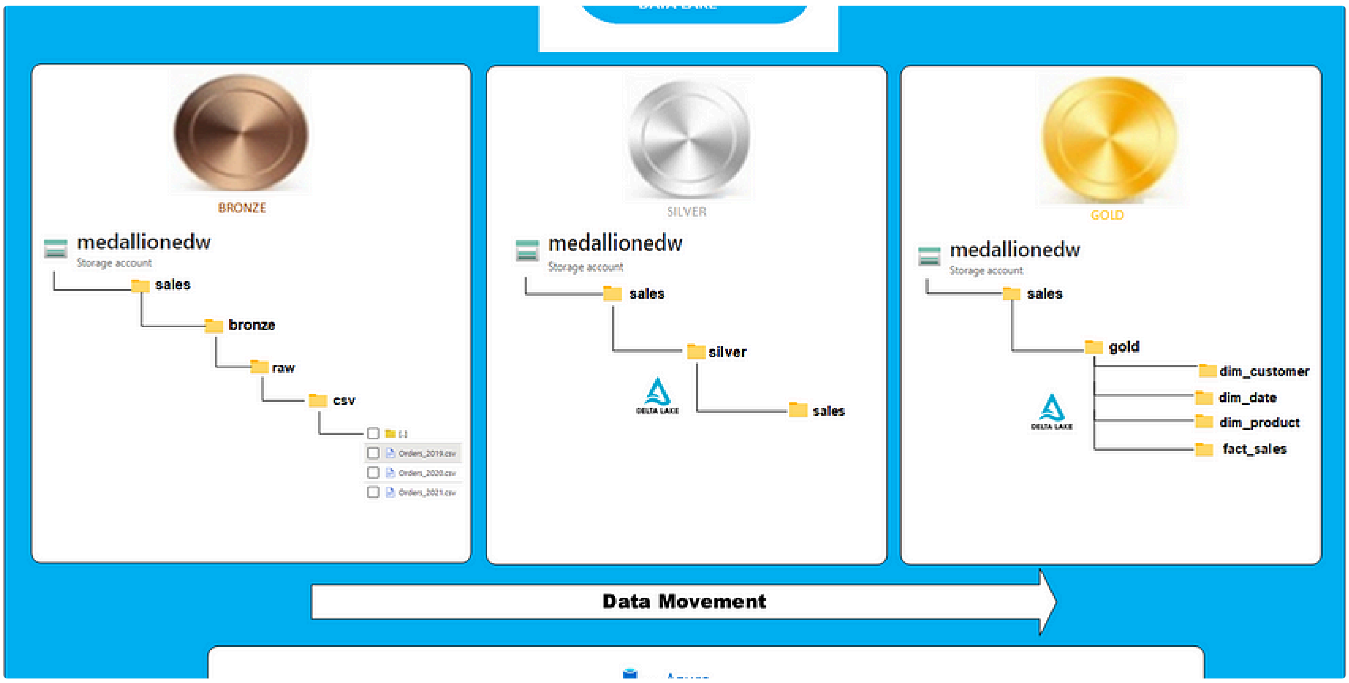
FABRIC SERIES: FUNDAMENTALS 03—INTRODUCTION TO MICROSOFT FABRIC LINGOS—OneLake

6 min read · Apr 13, 2024

5

See all from RK Iyer

Recommended from Medium



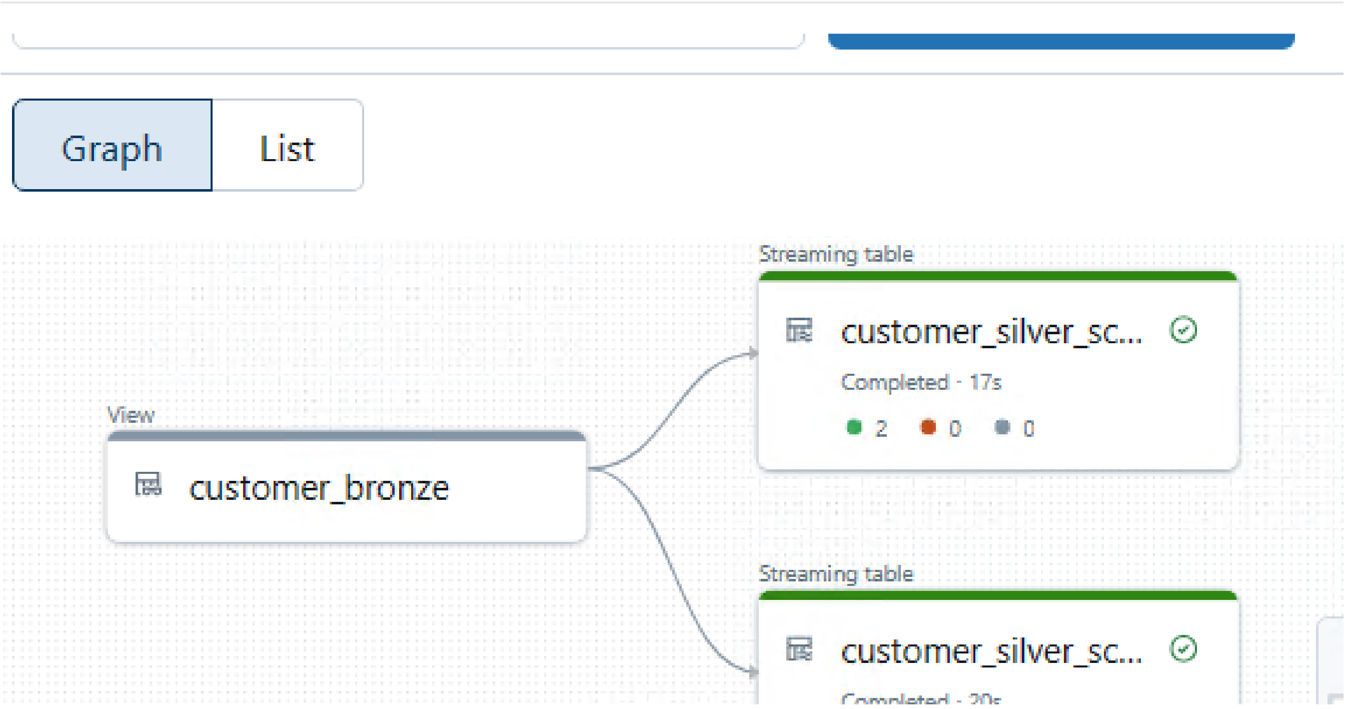
Valentin Loghin

Medallion Architecture in Data Lakehouse with Delta Lake and Databricks

What is a medallion architecture?

9 min read · Feb 15, 2024

150



Samarendra Panda

Working with Change Data Feed and Delta Live Tables in Azure Databricks

Introduction

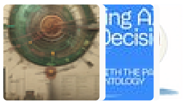
6 min read · Apr 23, 2024

9

Lists

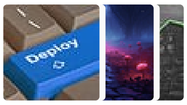
General Coding Knowledge

20 stories · 1188 saves



data science and AI

40 stories · 146 saves

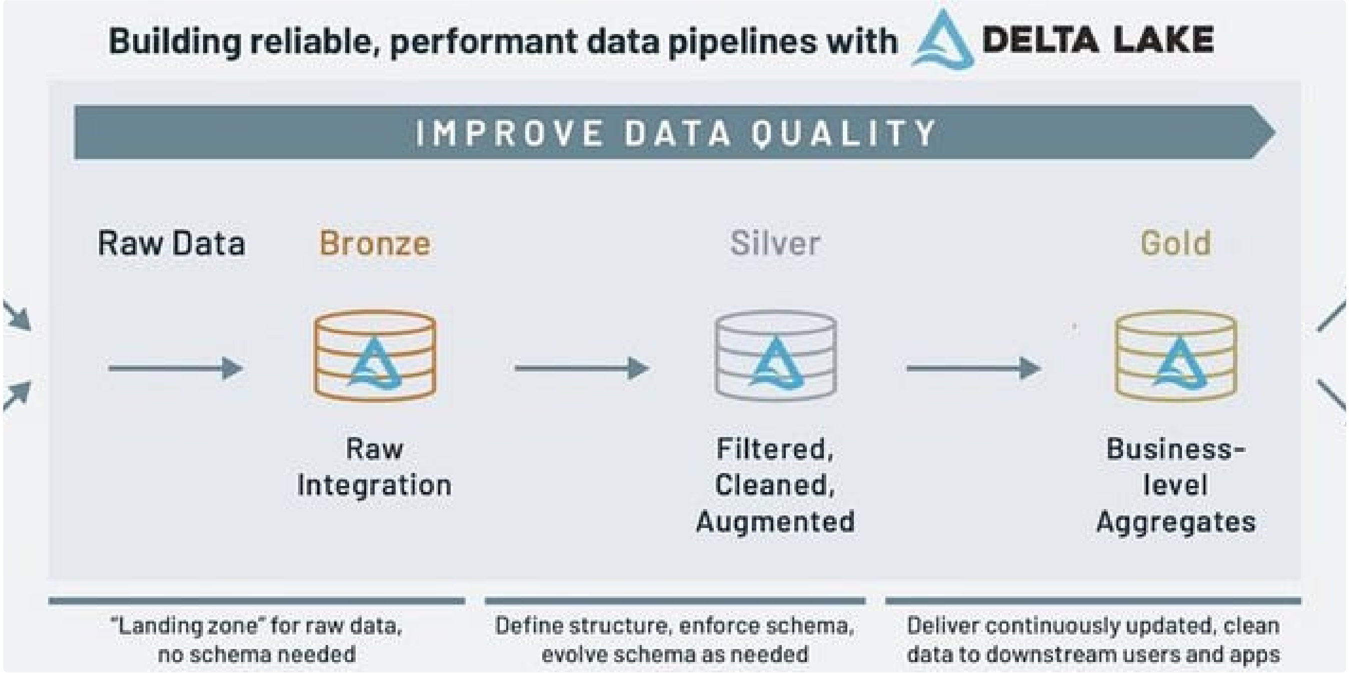


Predictive Modeling w/ Python

20 stories · 1154 saves

ChatGPT

21 stories · 616 saves


Building reliable, performant data pipelines with DELTA LAKE
IMPROVE DATA QUALITY
Raw Data
Bronze
Silver
Gold
Raw Integration
Filtered, Cleaned, Augmented
Business-level Aggregates
"Landing zone" for raw data, no schema needed
Define structure, enforce schema, evolve schema as needed
Deliver continuously updated, clean data to downstream users and apps

Bharanidharan M

Brass layer in front of Bronze in Medallion?

Medallion design pattern formulated by Databricks is to logically organize data in the Lakehouse, with the goal of improving quality of...

3 min read · Jan 25, 2024

3

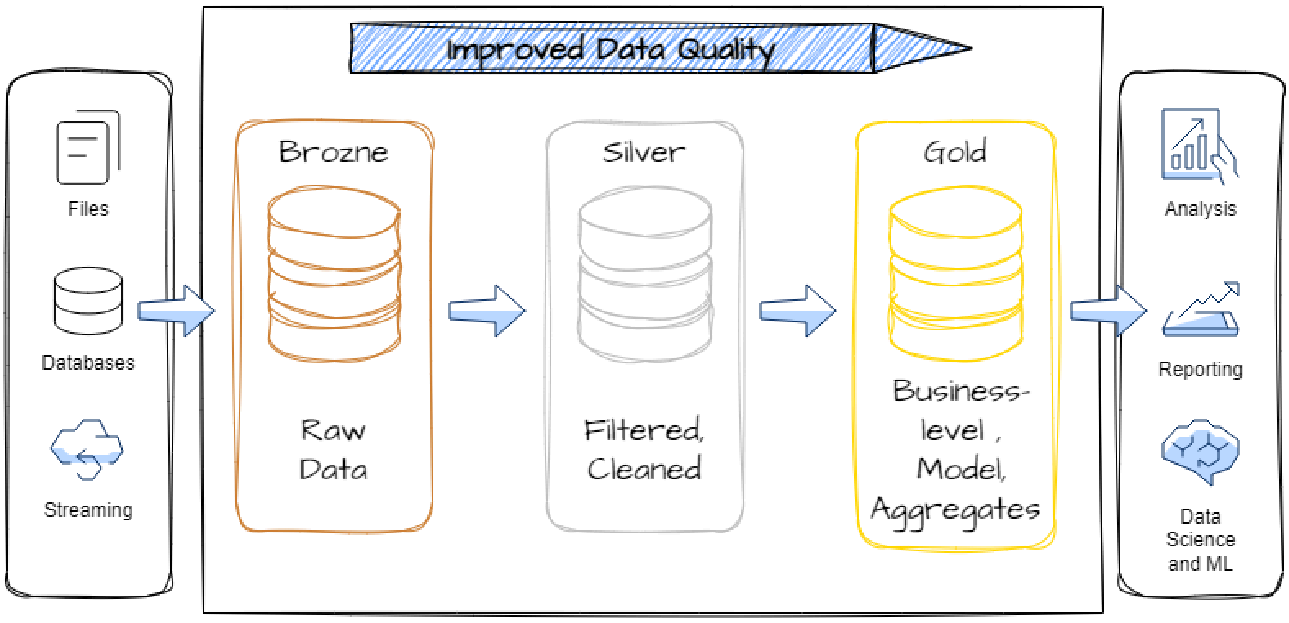
lineage
cataloging
security
discovery
sharing
auditing
quality
classification
data governance

Josemanuelgarciagimenez

Data Governance with Databricks

13 min read · Mar 26, 2024

14



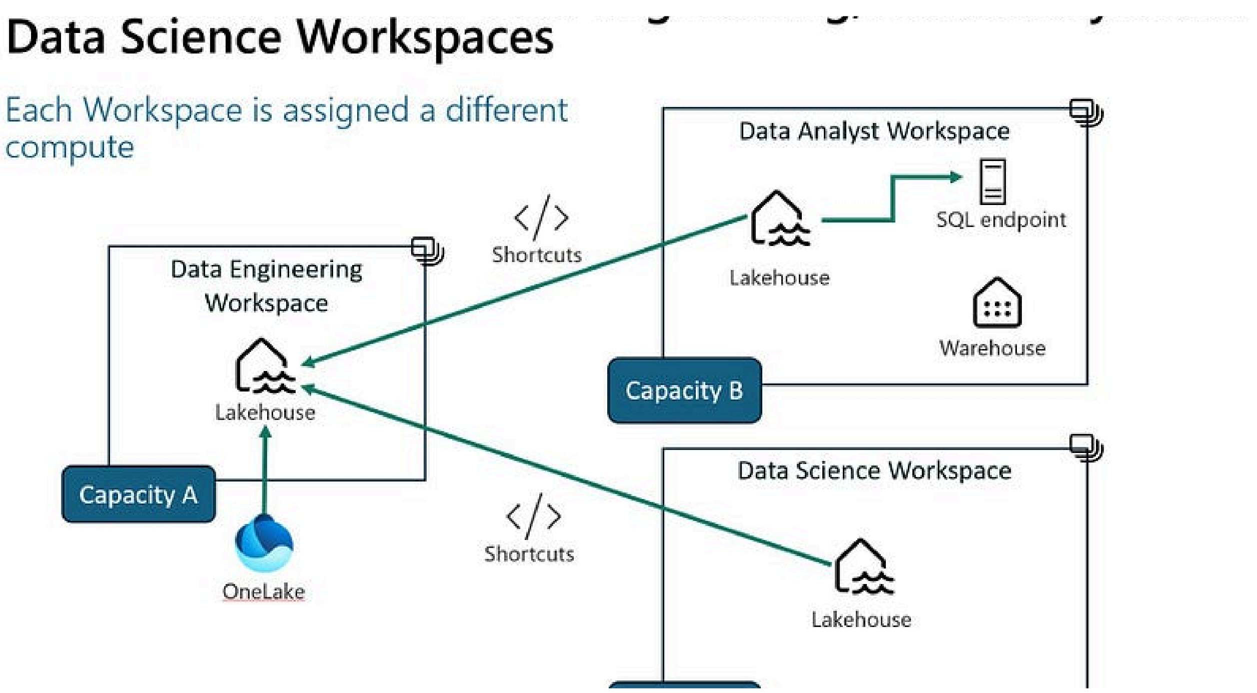
Mariusz Kujawski

Exploring the Medallion Architecture in Microsoft Fabric

The Medallion architecture stands out as one of the most popular frameworks for constructing a data lake or lakehouse. Its core concept…

9 min read · Feb 5, 2024

188 2



Inderjit Rana in Microsoft Azure

Microsoft Fabric — Data Sharing between Data Engineering, Data Analyst and Data Science Teams

Microsoft Fabric is an end to end analytics platform with capabilities for Data Engineers, Data Analysts as well as Data Scientists. A...

11 min read · Apr 29, 2024

80

See more recommendations