What goes into bronze, silver, and gold layers of a medallion data architecture?

Here’s a four-layer medallion architecture that explicitly addresses data governance and separation-of-responsibility

Lak Lakshmanan
·
Follow
9 min read
·
Sep 18, 2024

The standard way to build data lakehouses in an enterprise is to use the medallion architecture, where data is taken from the source system and transformed in stages. These stages are referred to as bronze, silver, and go layers. However, I find that a lot of data teams are very unclear on what goes into each of these stages. Also, the three-tier architecture fails to take into account that different teams do different tasks, and this makes governance difficult.

In this article, I’ll explain the medallion architecture that I recommend to data teams that are building an enterpris data platform that needs to satisfy the needs of multiple functional areas such as marketing, ML, FP&A, etc. I finc that it is helpful to design the enterprise data platform in terms of four layers, not three. This four-layer medallion architecture (I introduce a “platinum layer” between silver and gold) explicitly addresses data governance, separation of responsibility, and cost efficiency.

The Traditional 3-layer Medallion Architecture

Here’s how Databricks, for example, describes what each of these layers consists of:

The three tiers of a medallion architecture, as depicted by Databricks.
https://www.databricks.com/glossary/medallion-architecture [Image © Databricks, fair use for criticism]

According to Databricks, the bronze layer is for capturing data from the source systems as-is and no schema is needed; the silver layer brings data from different sources, defines structure, evolves schema, and enables self-service analytics; while the gold layer is denormalized project-specific databases containing business-level aggregates.

This medallion 3-layer architecture works for teams that are building data applications. If you are part of such a project team, it helps to refine your data stage-by-stage. Your end-goal is a project-specific database. But the reason it works is that you don’t have to worry about inter-team collaboration or governance of intermediate datasets.

Problems with the 3-layer medallion architecture

Problems with the 3-layer medallion architecture

The 3-layer medallion is an application architecture masquerading as an enterprise data architecture. While it works as the architecture for a single data application, it completely falls flat when you take this architecture and t to organize your company's data platform on it.

The reason is that it is an extremely engineering-focused view of the world, and tempts data engineering teams somehow land the data and throw it over the wall to a ▮business▮ team and let them figure it out. To be clear, thi is not a problem with Databricks ▮ it's a problem with how the medallion approach is generally understood and implemented in many organizations. I'm referencing Databricks above only because they have the best documentation on the current medallion approach.

There are several problems with the 3▮layer medallion approach:

- The gold layer is completely unorganized, and pretty much unusable. There are hundreds of data projects i a typical company, and they will all have different needs. Even something as simple as a ▮shipped product▮ will have different definitions depending on whether the use case is logistics or accounting. Coordination is impossible, and the net result is a chaotic gold layer.
- The silver layer does too much. It brings in data from the source systems, cleans them up, organizes them f self-service analytics, and provides an enterprise view of key business entities. Really? With this kind of scop the silver layer has to be created by a central engineering team, and few central teams understand busines requirements enough to be able to do this effectively.
- There is an implicit assumption of progress-by-stage. Gold data is ▮better▮ than silver data, which is itself bett than bronze data. This lends itself to an architecture where a dataset is simply copied (or at least views created) from layer-to-layer because someone has mandated that dashboards can only read from the gold layer.
- The data modeling requirements for the layers are unclear. A good data model will define the logical, physical, and transformation aspects of your data. The bronze tier matches the source system. This gets you to the logical, but what are the physical mapping and transformations allowed? The silver and gold layers ha goals (▮self-service analytics▮ and ▮project-specific databases▮) but not the path to get there.
- Data governance is an afterthought. Who governs the data in each layer is left unaddressed.

Recommended 4-layer architecture

I find that an enterprise data architecture becomes a lot simpler if you add a fourth layer (I call it ▮platinum▮, and p it between the silver and gold layers), and carefully define the roles and responsibilities in each layer.

4-layer medallion architecture for an enterprise data platform. Image by author. © V Lakshmanan, CC-BY-4.0.

Let's walk through the above diagram layer-by-layer.

Bronze Layer

The bronze layer contains a lossless replication of the source data, but in a format that is conducive to loading a an external table into your data platform. In practice, this means exporting data from source systems into cloud storage in formats such as Parquet or JSON.

Bronze layer. Diagram by author.

For example, you could export the data from your source system as JSON files on AWS S3 if your enterprise da platform is Snowflake because Snowflake is able to create a table off the data stored there without having to mak another copy.

The schema is whatever the source system exports it into. You'll partition the data by ingest timestamp, so that i cost-effective to query for the latest data and to automatically passivate old partitions.

This can be done either in batch mode or in streaming mode.

In the diagram above, you see light-green and dark green boxes in the bronze layer. The light-green boxes consist of artifacts (such as export scripts and exported files) that are conceptually part of the bronze layer. The dark-green boxes are the data products that are available to other layers. The data products are stored in the da platform; but here, the product may be just an external table definition.

Silver Layer

It is not enough to just export the data from Salesforce or SAP or wherever. The data has to be cleaned (for example, invalid SKUs removed) and deduped (so that updated rows reflect only the latest data). The data also has to be conformed I this means that if you have a specific name and formatting for a field of some type across the organization, you have to do the renaming and reformatting in the silver layer.

Silver Layer. Image by author.

Optionally, you can also join some of the tables, and/or denormalize the data to make it easier to use. Type 2 slowly changing dimensions is commonly handled in the silver layer. Because of this, all the tables in the silver layer are IasofI a certain date.

The governance responsibility for the silver layer belongs with the source team. In practice, this is the data engineering team that operates the source system. For example, suppose that your organization mantains the s of customers under contract in Salesforce. Then, data engineers in the sales org (or the central data engineerin team in IT, if there are no engineers in the sales org) are responsible for creating the table on contracted customers, renewal dates, etc. That table will, however, be aligned to the information in Salesforce only.

The source-aligned clean datasets in the silver layer are data products and will be used by other organizations. is essential that the data engineers creating silver layer data products understand what the users of their dataset are doing with them. They need to ensure that when the source systems change, the ensuring changes to the clean datasets do not break downstream applications.

Platinum Layer

The platinum layer consists of data marts created by functional teams (Marketing, Membership, Logistics, etc.) th want to use data to solve their business problems.

In our example, a marketing team that wants to market to an audience of customers whose renewal term is coming up will get the cleaned data of customers under contract from the sales teams' table. But they may also need product usage data by those customers, and perhaps that data comes from data exported from MixPanel, and made available by the product team.

Platinum layer. Image by author.

This join of silver data across the enterprise to satisfy functional needs is done in the Platinum layer. The key thi to note is that the data model of the data within the data marts meets the requirements of the functional team. As such, it will include aggregates that are helpful in multiple use cases within the functional area. This is because the purpose is to support the applications of that functional team.

The governance of the data in the platinum layer is limited to the functional area within which the use cases are being developed.

Gold Layer

A small subset of the modeled data in the Platinum data marts is useful to the rest of the company. This data is used to create conformed specifications of the key entities and relationships.

The most important thing to realize about the gold tier is that it is a single source of truth. Entities, KPIs, aggregat available in the gold layer are those that have to be used consistently in dashboards and applications across th company.

Gold layer. Image by author.

Because of this, the governance of the gold tier requires stakeholders across the company to agree on the shared definitions. Because the gold layer requires agreement between different divisions, it is important that yo keep it as small as possible. Often, I find it helpful to start from the company's Financial Planning and Analysis (FP&A) and note down the key entities and metrics needed. These cannot change willy-nilly, and so they provi a forcing function for what actually requires centralized governance, and what can remain at the Platinum level.

The gold layer needs to capture the shared definition of important entities (who counts as a customer, what counts as a new product line, etc.). In addition, the gold layer also needs to capture the important KPIs and how they are calculated. For example, if you are reporting monthly active users, you need to clearly capture how thi calculated and make sure that a single shared definition is used throughout the company.

Data Governance

I recommend a 4-layer medallion architecture to data teams that are building an enterprise data platform that needs to satisfy the needs of multiple functional areas. This four-layer medallion architecture explicitly addresse data governance, separation of responsibility, and cost efficiency.

There are data products at each layer, and as long as the user of the data understands the limitations (e.g., the schema of the data in the silver layer could be changed when the source system evolves), they can use the products available in that layer. This promotes agility.

At the same time, data products in the Platinum are much more business-focused. They meet the needs of a functional area, but are still more usable because they conform to enterprise-wide definitions.

The physical requirements in each layer (partitions, external tables, ELT, materialized views, data mesh) all support the end-goal of reducing the number of copies and delegating data management responsibilities wherever possible to the data platform.

Suggested Reading

- For your zooming and panning pleasure, here's a link to the architecture diagram https://excalidraw.com/#json=lkVCEaWH0t3JPM5n_QiSF,HaSz-0fYZLCtebYKDWDYfw
- Read our O'Reilly book for detailed guidance on architecting data and ML platforms: https://www.amazon.com/Architecting-Data-Machine-Learning-Platforms/dp/1098151615
- The LinkedIn version of this post has a number of comments/questions/answers that might be worth readin https://www.linkedin.com/pulse/what-goes-bronze-silver-gold-layers-medallion-data-lakshmanan-r93nc/

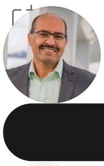
Data Platforms

Data Governance

Data Lakehouse

–

9



Follow

Written by Lak Lakshmanan

10.9K Followers

·

88 Following

articles are personal observations and not investment advice.

Responses (9)

See all responses