

Step 1: Data Lake Fundamentals

What is a Data Lake?

A **Data Lake** is a centralized, scalable repository that stores vast amounts of raw data in its native format (JSON, CSV, Parquet, Logs, etc.) until it is needed.

- **Purpose:** To break down data silos and provide a single source of truth for diverse workloads, including big data processing, real-time analytics, and machine learning.
- **Comparison:** Unlike a traditional Data Warehouse (which requires "Schema-on-Write"), a Data Lake allows "**Schema-on-Read.**" You ingest first and define the structure later.

Why not store everything in one folder?

- **Bottlenecks:** Searching through a single folder with millions of files is computationally expensive and slow.
- **Governance Nightmare:** You cannot easily apply security policies (e.g., "HR only") if all files are in one "bucket."
- **Naming Conflicts:** Different source systems might produce files with the same name (e.g., `data.csv`).

Problems Zones Solve

1. **Data Quality:** Zones separate "dirty" raw data from "validated" business data.
 2. **Processing Efficiency:** By partitioning and using optimized formats (like Parquet) in higher zones, query speeds increase.
 3. **Access Control:** You can grant Data Engineers access to Raw data, while Business Analysts only see the Gold zone.
-

Step 2: Zone Responsibilities

| Feature | Raw / Bronze | Cleaned / Silver | Curated / Gold |
|---------|-------------------------|---------------------------|---------------------------|
| Format | Native (JSON, CSV, XML) | Optimized (Parquet/Delta) | Optimized (Parquet/Delta) |

| | | | |
|-------------------|---------------------------|-----------------------------|--|
| Validation | None (As-is) | Schema & Type checking | Business logic & referential integrity |
| Consumers | Data Engineers | Data Scientists / Engineers | Business Analysts / Power BI |
| Purpose | Historical record/Archive | Filtered, joined, & cleaned | "Feature-ready" / Reporting |

Zone Characteristics

- **Raw:** Immutable. We never edit data here. If a load is bad, we fix the code and re-ingest.
 - **Silver:** The "Workhorse." We remove duplicates, handle nulls, and standardize formats (e.g., all dates become `YYYY-MM-DD`).
 - **Gold:** Highly aggregated. Instead of individual sales, it might store `monthly_sales_by_region`.
-

Step 3: Folder Structure Design

The Hierarchy

```
/data-lake
  /raw
    /source_system_erp
      /sales
        /year=2026/month=02/day=21/
          sales_rec_001.json
      /customers
        /year=2026/month=02/day=21/
          cust_export.csv
  /silver
    /sales
      /year=2026/month=02/
        sales_cleansed.parquet
    /customers
```

```
/customers_cleansed.parquet  
/gold  
/finance  
/monthly_revenue_summary/  
rev_2026_02.parquet  
/marketing  
/customer_360/  
active_users.parquet
```

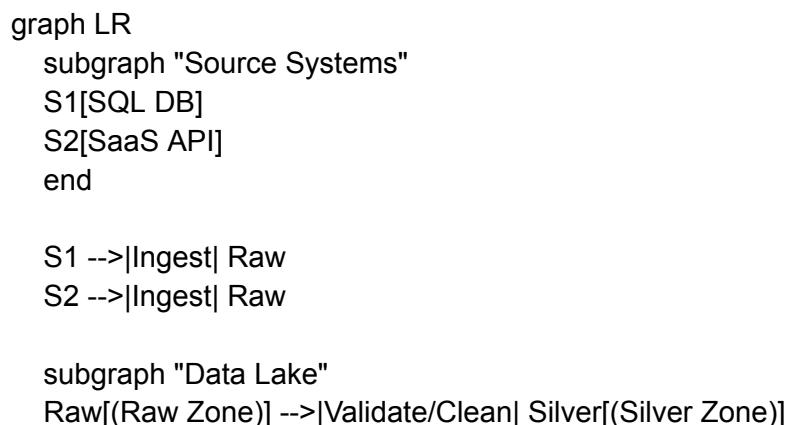
Design Choices

- **Partitioning (year=X/month=Y):** This uses "**Hive-style partitioning.**" Tools like Snowflake, Spark, and Athena can "prune" partitions, meaning if you query for February, the engine ignores the other 11 months entirely, saving time and money.
- **Naming Conventions:** Lowercase only, no spaces, and using underscores. This prevents issues when moving data between Linux-based storage and SQL-engines.
- **File Formats:** We move from **Row-based** (JSON/CSV) in Raw to **Columnar** (Parquet) in Silver/Gold. Parquet is much faster for analytical queries that only need a few columns.

Step 4: Data Flow Explanation

1. **Ingestion (Source → Raw):** Data is pulled via API or Batch and dumped into the Raw zone. We capture **Metadata** like `ingestion_timestamp` and `source_filename` to ensure we can trace data back to its origin.
2. **Cleansing (Raw → Silver):** An ETL job (like dbt or Spark) reads the Raw data. It enforces a schema, casts data types (e.g., converting "10.5" string to a decimal), and removes "corrupt" records.
3. **Curation (Silver → Gold):** This is where **Business Logic** lives. We join the `sales` table with the `customers` table to create a "Rich" dataset. We aggregate totals and calculate KPIs (Key Performance Indicators).

Data Flow Diagram



Silver -->|Aggregate/Join| Gold[(Gold Zone)]
end

Gold -->|Report| BI[Power BI / Tableau]
Silver -->|Explore| DS[Data Science/ML]