

Data Lakehouse Full-Stack BI Project

- From Databricks to Power BI Reporting

Presented by: Nitin Kunigal

Agenda

- Problem Statement & Strategic Context
- Business Objectives & Success Criteria
- Solution Approach
- Modern Data Lakehouse Architecture
- Estimated Benefits and Expected Outcomes
- Limitations, Constraints and Design Trade-offs
- Future Enhancements

Problem Statement & Strategic Context

- Hypothetical mid-sized retail company
- Fragmented reporting across ERP and CRM systems
- Leadership lacked timely visibility
- Lack of unified view of KPIs across business functions

Business Objectives & Success Criteria

Primary Objectives

- Consolidate ERP and CRM data into a unified analytical model
- Standardize KPI definitions across sales, customer, and product teams
- Enable leadership to analyze performance drivers and trade-offs

Success Criteria

- Clean, reliable Silver-layer datasets with consistent definitions
- Business-ready Gold views supporting a star-schema model
- Power BI reports that answer stakeholder questions without manual work

Solution Approach

The solution is structured around three layers:

- Bronze Layer: Raw ingestion of source data
- Silver Layer: Data quality fixes and standardization
- Gold Layer: Business transformations, dimensional modeling, and reporting views.

High Level Data Architecture



Modern Data Lakehouse

Sources



CRM



ERP

Object Type: CSV Files

Interface: Files in
Databricks (Volume)

E, L

Bronze Layer

Notebook (Spark SQL)



Raw Data

Object Type: Delta Tables

Load:

- Full Load
- Insert & Overwrite

No Transformations

Data Model: None (as-is)

T, L

Silver Layer

Notebook (Spark SQL)



Cleaned, Standardized
Data

Object Type: Delta Tables

Load:

- Full Load
- Insert & Overwrite

Transformations:

- Data Cleansing
- Data Standardization
- Data Normalization
- Derived Columns
- Data Enrichment

Data Model: None (as-is)

T, L

Gold Layer

Notebook (Spark SQL)



Business-Ready Data

Object Type: Views

No Load

Business Transformations:

- Data Integrations
- Aggregations
- Business Rules

Data Model:

- Star Schema
- Aggregated Table

Consume



BI & Reporting



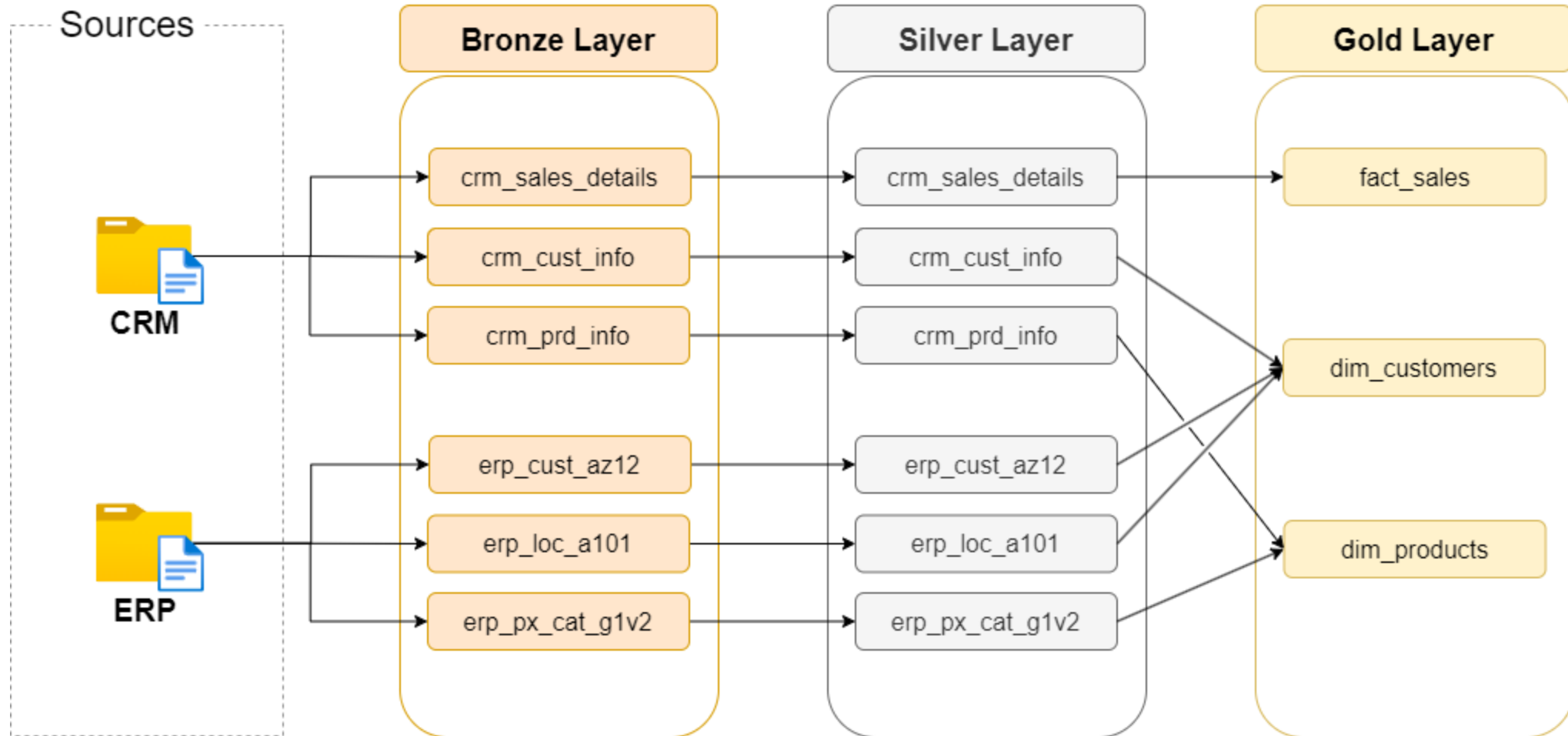
Ad-Hoc
SQL Queries



Machine Learning

End-to-End Data Pipeline Orchestration

Data Flow (Lineage) Diagram



Estimated Benefits and Expected Outcomes

If implemented in a real organization, this solution would:

- Reduce manual reporting and reconciliation efforts
- Improve trust in leadership dashboards
- Enable faster, data-driven decision-making
- Support scalable analytics as data volumes grow

Limitations, Constraints and Design Trade-offs

- Databricks Free Edition does not support External source connections
- Orchestration was simplified to focus on core BI concepts rather than enterprise tooling
- Microsoft Fabric was not used due to access constraints, though the architecture is directly transferable

Future Enhancements

- Automated ingestion from live source systems
- Incremental loading and change data capture
- Role-level security and governance enhancements
- Migration to Fabric or other cloud-native orchestration tools