https://www.linkedin.com/in/chinedu-amagwara-0a7a508a/

Article on Designing a Lakehouse Cloud Architecture for a Retail Bank

Prepared by

Chinedu Patrick Amagwara

https://www.linkedin.com/in/chinedu-amagwara-0a7a508a/

Table of Contents

https://www.linkedin.com/in/chinedu-amagwara-0a7a508a/

Introduction

Retail banks today face intense competition and rapidly evolving customer expectations. To thrive, banks must harness the power of their data to drive decisions that enhance customer experiences, optimize operations, and ensure security. Azure Lakehouse architecture—a unified platform combining the strengths of data lakes and data warehouses—empowers retail banks to transform raw data into actionable insights.

This article explores how Azure Lakehouse architecture supports retail banks' mission to deliver data-driven decisions and achieve operational excellence.

Mission Statement

Empowering data-driven decision-making by leveraging Azure Lakehouse architecture as a centralized, secure, and scalable platform for data ingestion, storage, and analysis. By integrating and analyzing data, banks can enhance customer experiences, optimize operations, and foster innovation across the financial ecosystem.

Objectives

1. Centralized Data Collection
   Raw data from ATMs, card wallets, POS systems, marketing platforms, and more into Azure Data Lakehouse for seamless integration and analysis.

2. Data Integration
   Create enriched datasets that unify customer behavior, operational performance, and market trends.

3. Operational Efficiency & Real-Time Analytics
   Optimize ATM uptime, manage cash more effectively, and reduce costs with predictive analytics. Detect and prevent fraud by monitoring transaction patterns and device health in real time.

4. Targeted Marketing
   Use customer insights to design personalized campaigns and improve ROI.

5. Compliance and Security
   Ensure data privacy and regulatory compliance through robust governance frameworks.

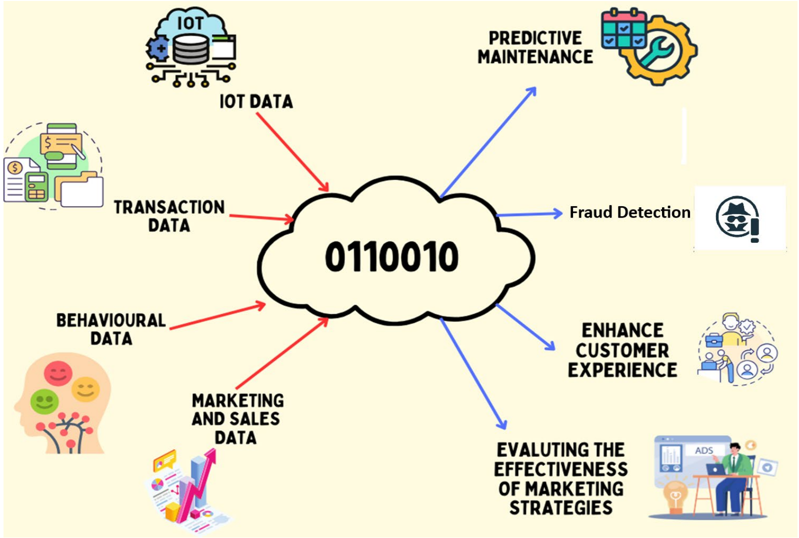
6. Innovation

Develop new products, enhance services, and identify emerging market opportunities through actionable insights.

Vision Diagram

The purpose of a vision diagram is to ensure that all stakeholders—technical teams, business managers, and data users—have a shared understanding of the data flow.

It clarifies how different systems interact and aligns the architecture with the organization's goals, such as enhancing decision-making and operational efficiency.



Vision Diagram

The vision diagram is essential for identifying data sources and sinks:

1. Maps Data Sources and Their Relationships
   Purpose: It identifies and categorizes the diverse data sources (e.g., IoT data from ATMs, transactional data, behavioral data, marketing data) that feed into the Lakehouse.
   Benefit: By visually mapping the sources, teams can better plan ingestion methods, prioritize integration efforts, and ensure no critical data is overlooked.

2. Defines Data Destinations (Sinks)
   Purpose: The diagram highlights where the processed data will be delivered, such as analytics dashboards, machine learning models, or reporting systems.
   Benefit: This ensures that the Lakehouse delivers actionable insights effectively to operational systems or decision-makers.

Data Sources

1. IoT Data from ATM Machines
    - **Data Type**: Structured and Semi-structured data.
    - **Ingestion Method**: Real-time streaming using Azure IoT Hub to capture telemetry and send it to Azure Stream Analytics for continuous processing. **Examples**: Device health metrics (e.g., uptime, errors), cash level indicators, operational logs in JSON or telemetry formats. ATM Environmental (Temperature, Humidity, Vibration, Shock/Impact) sensors, Security (Camera Modules, Door, Motion, Biometric, Card Skimmer), Operational (Cash Level, Receipt Paper, Currency Validation, Jam Detection) Sensors.
    - **Storage**: Data is ingested into Azure Data Lake as semi-structured files (e.g., JSON, Avro) for further analysis

2. **Transactional Data**
    - **Data Type**: Structured data.
    - **Ingestion Method**: Real-time ingestion via Azure Event Hubs (streaming).
    - **Examples:** Withdrawals, deposits, check deposits, wallet transactions stored in relational databases.
    - **Storage:** Data is stored in tabular format (e.g., Parquet or Delta) in Azure Data Lake for querying and transformation.

3. **Marketing Data**
    - **Data Type**: Structured and unstructured data.
    - **Ingestion Method:**
        - ➢ Structured Data: Use Azure Data Factory for Batch ingestion to extract marketing metrics from databases or APIs.
        - ➢ Unstructured Data: Use Azure Cognitive Services to process chatbot logs into text files or sentiments.
    - **Examples:**
        - ➢ Structured: Campaign engagement metrics, app and website usage statistics from marketing platforms.
        - ➢ Unstructured: Chatbot logs containing free-text conversations.
    - **Storage:** Data is stored in its respective formats—structured data as tables, unstructured data as text or JSON.

4. **Behavioral Data**
    - **Data Type**: Semi-Structured and Unstructured data.
    - **Ingestion Method**: Batch ingestion via Azure Data Factory.
    - **Examples**: Website navigation patterns, app usage, email engagement.
    - **Storage:** Store behavioral data in Azure Data Lake as semi-structured JSON for flexibility.

Storage and Processing

Storage

The Azure Lakehouse architecture integrates Azure Data Lake Storage Gen2 with Delta Lake for unified storage and processing:

1. Bronze Layer (Raw Data)
    - Stores ingested data in its original format.
    - Use Case: Retaining historical data for compliance and audit.

2. Silver Layer (Curated Data)
    - Stores curated datasets enriched with business logic and metadata.
    - Use Case: Unified customer profiles, fraud detection models.

3. Gold Layer (Aggregated Data)
    - Optimized for querying and reporting.
    - Use Case: Operational dashboards, targeted marketing insights.

Processing

- Azure IoT Hub: Captures and preprocesses IoT data in real time.
- Azure Stream Analytics: for filtering, aggregation, and anomaly detection in real time.
- Azure Event Hub: Streams high-throughput transactional and behavioral data.
- Azure Data Factory: Orchestrates batch ETL pipelines for diverse datasets.
- Azure Synapse Analytics: Enables advanced analytics and large-scale data processing.

Insights and Sink

Data insights are made actionable using Azure analytics and visualization tools:

1. Operational Dashboards
    - Tool: Power BI connected to Azure Stream Analytics and Azure Data Warehouse.
    - Purpose: Real-time monitoring of ATM uptime, cash levels, and device performance. Monitor transaction trends, Revenue and Cost Analysis KPIs.

2. Fraud Detection Alerts
    - Tool: Azure Machine Learning integrated with the silver layer.
    - Purpose: Detect anomalies in transaction patterns and alert fraud prevention teams.

3. Customer Personalization
    - Tool: Power BI and Azure Synapse.
    - Purpose: Deliver personalized product recommendations and offers.

4. Marketing Campaign Analysis
    - Tool: Azure Cognitive Services for chatbot data analysis.
    - Purpose: Evaluate campaign effectiveness and improve customer engagement.

Compliance and Security

Azure ensures data security and compliance through:

1. Data Governance
    - Tool: Azure Purview for data lineage and cataloging.
    - Purpose: Ensure transparency and compliance with financial regulations like PCI DSS.

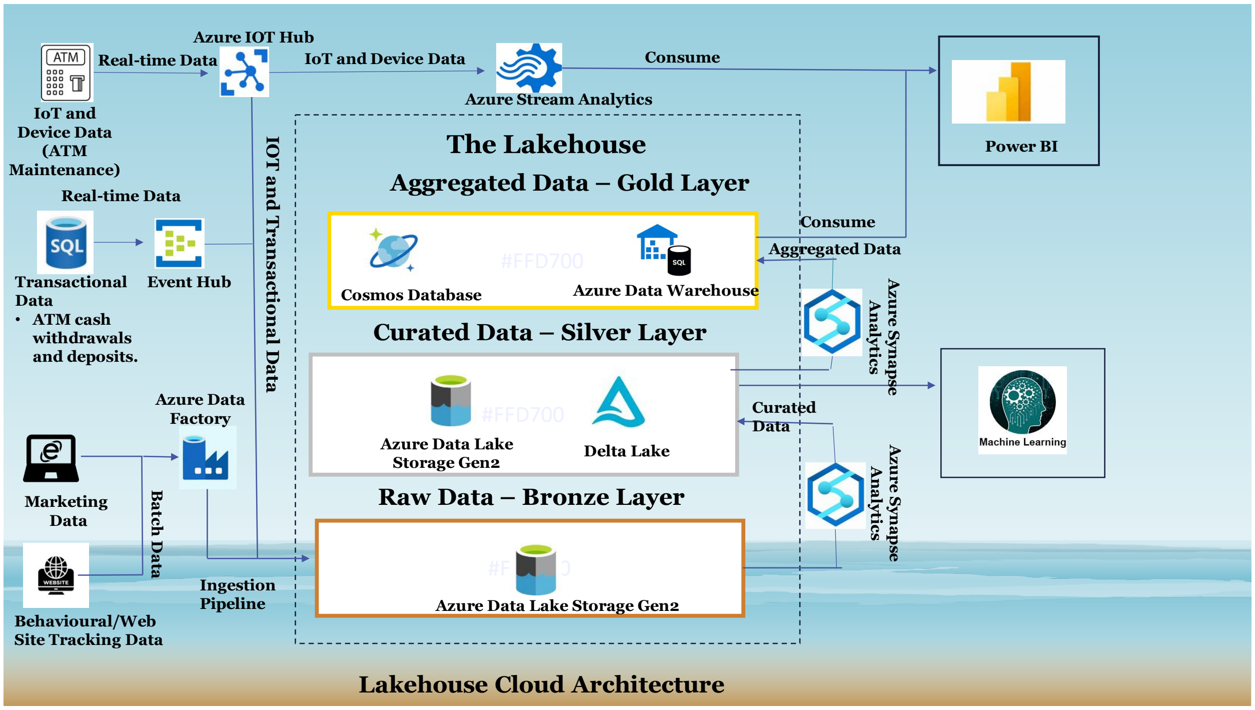
2. Security
    - Tool: Azure Active Directory for role-based access control.
    - Purpose: Protect sensitive customer and transaction data.

3. Encryption
    - Ensures all data is encrypted both at rest and in transit.

Lakehouse Cloud Architecture

Designing the Lakehouse Architecture for a Retail Bank

The Lakehouse Architecture for a retail bank combines the flexibility of a data lake with the analytical power of a data warehouse, enabling a seamless environment for real-time and batch processing. It facilitates centralized storage, transformation, and analysis of data while supporting advanced use cases like predictive analytics, fraud detection, and targeted marketing.



Data Flow Overview

1. Ingestion: Data ingested via IoT Hub, Event Hubs, and Data Factory.
2. Storage:
    - ➢ Raw Data: Stores ingested data in its native format (immutable) in Data Lake Gen2.
    - ➢ Delta Lake Processed Zone: Stores cleaned, structured, and enriched data.
    - ➢ Curated Data: Holds cleaned, transformed, and optimized datasets for analytics and reporting.
    - ➢ Gold Layer: Stores aggregated data that can directly support business operations, analytics, reporting, and strategic decision-making.
3. Processing: Transformation using Azure Synapse Analytics for analytical processing.
4. Querying: Azure Synapse and ML pipelines enable ad hoc analysis and predictions.
5. Consumption: Insights delivered to stakeholders through Power BI and APIs.

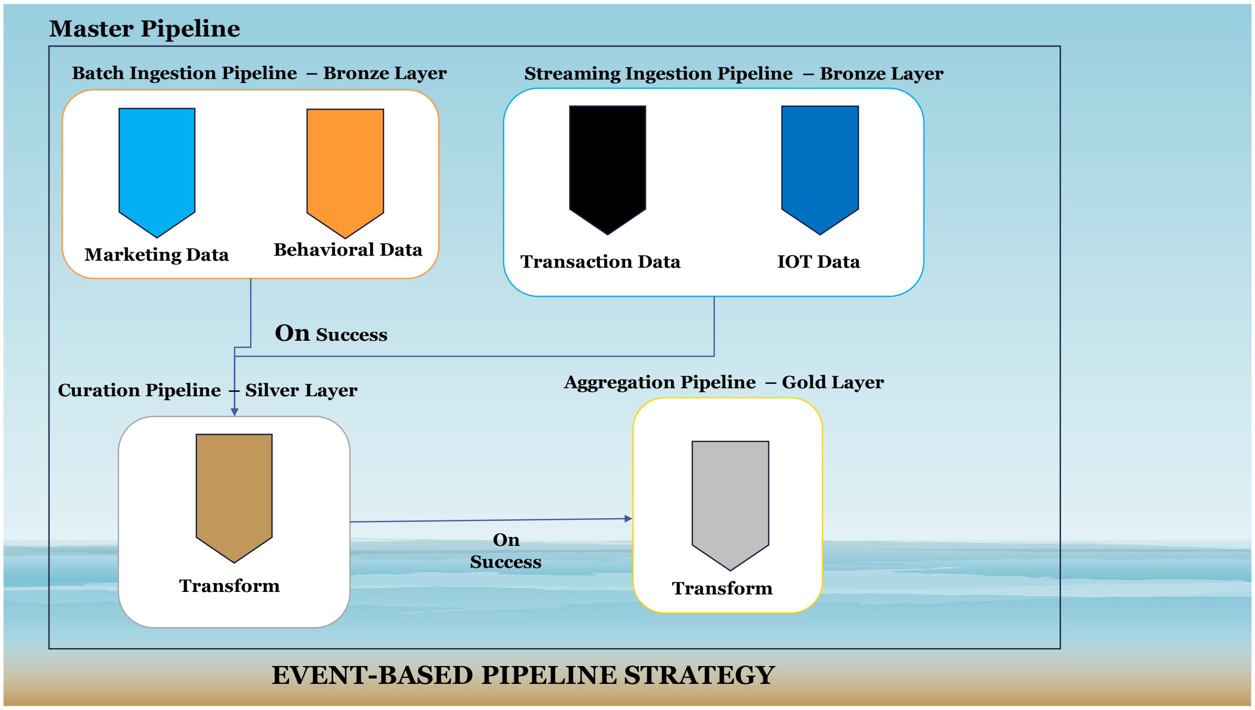
Features of the Lakehouse Architecture

- **Centralized Data Platform:** Combines raw and processed data in a single location for easy access and management.
- **Scalability:** Azure's scalable resources accommodate growing data volumes and complexity.
- **Real-time and Batch Analytics:** Simultaneous support for streaming and batch processing.
- **Enhanced Governance:** Use Azure Purview for data cataloging, lineage, and governance, ensuring compliance with banking regulations.
- **Integration with AI and ML:** Supports advanced analytics, such as fraud detection, predictive cash management, and customer personalization. Features of the Lakehouse Architecture

Pipeline Strategy

Event-based Approach

The event-based approach is a powerful way to design data pipelines in a retail bank, where the execution of processes depends on the success or failure of preceding tasks. This method ensures that the data processing flow happens in a controlled, logical sequence and helps avoid unnecessary resource consumption by triggering subsequent pipelines only when previous ones succeed. The event-based strategy also enhances operational efficiency by ensuring the bank reacts swiftly to data events, driving real-time insights and decisions.


Master Pipeline
Batch Ingestion Pipeline – Bronze Layer
Marketing Data
Behavioral Data
Streaming Ingestion Pipeline – Bronze Layer
Transaction Data
IOT Data
On Success
Curation Pipeline – Silver Layer
Transform
Aggregation Pipeline – Gold Layer
On Success
Transform
EVENT-BASED PIPELINE STRATEGY

Master Pipeline Overview

In the event-based approach, a master pipeline orchestrates the execution of multiple downstream pipelines. This master pipeline is the central controller, invoking specific tasks based on the outcomes (success or failure) of prior tasks. The master pipeline ensures that processes are carried out in a logical sequence and without interruptions, optimizing the entire data workflow.

Pipeline Failure Strategy

Time-Out Setting

Time-out settings are configured at various stages of the pipeline. This ensures that long-running operations do not stall indefinitely and provide quick feedback on potential bottlenecks.

Time-out limits are set based on the expected load and response times from external data sources (e.g., ATM machines, POS systems) and the processing capabilities of your pipeline.

Retry Interval

Configuring an appropriate retry interval ensures that the system does not overwhelm itself with repeated retries, but also allows for recovery.

Set an increasing three times retry interval (exponential backoff) where the system retries at progressively longer intervals (e.g., 5 minutes, 15 minutes, 30 minutes). This strategy is especially useful for system recovery from temporary resource shortages.

Terminate the Entire Pipeline and Handle Failure Separately

In cases where a failure cannot be rectified by retries, the pipeline is terminated. This approach prevents cascading failures that might impact other stages of the data processing flow and ensures that the issue is isolated and handled separately.

Alerting Features

To promptly respond to pipeline failures, it is essential to establish an alerting mechanism that notifies relevant stakeholders (e.g., IT teams, data engineers, business analysts) about the failure.

Actions:

- Send email/SMS notifications to the relevant stakeholders (e.g., operations team, data engineers).
- Trigger automated remediation workflows, such as reinitializing failed tasks, retrying data ingestion, or initiating manual intervention.
- Log failure details in an error tracking system (e.g., Azure Application Insights or Azure Log Analytics) for deeper analysis and reporting.

https://www.linkedin.com/in/chinedu-amagwara-0a7a508a/

Conclusion

Azure Lakehouse architecture provides retail banks with a powerful platform for data-driven decision-making. By centralizing data collection, enabling real-time analytics, and ensuring compliance, banks can enhance customer experiences, optimize operations, and stay ahead of emerging challenges. This architecture is not just an investment in technology but a strategic enabler for innovation and sustainable growth in a competitive financial landscape.