**Day 18 Notes**

**Name : Tejaswini Gokanakonda- DE142**

**Date : 03-12-2024**

### 

### **Overview of Delta Live Tables (DLT)**

* Delta Live Tables is a declarative ETL framework within Databricks Lakehouse that simplifies building and managing pipelines for streaming and batch processing.
* It emphasizes automation and data quality, helping engineers reduce the operational burden by automatically handling aspects such as dependencies, orchestration, and monitoring.
* Key Objective: Allow data engineers to focus on data value rather than pipeline maintenance.

### **Medallion Lakehouse Architecture**

This architecture divides the data pipeline into three distinct layers:

1. **Bronze Layer:**
   * Purpose: Stores raw, unprocessed data directly from multiple sources.
   * Features:
     + Maintains data in its original format.
     + Useful for audit purposes or tracking back to the data source.
     + Examples: Logs, raw JSON files, unstructured data.
2. **Silver Layer:**
   * Purpose: Cleans, filters, and standardizes the Bronze data for further use.
   * Features:
     + Handles missing data and type conversions.
     + Flattens nested structures for easier querying.
     + Renames columns or organizes data based on business requirements.
   * Examples: Transforming JSON into tabular formats, renaming columns like emp\_id to employee\_id.
3. **Gold Layer:**
   * Purpose: Provides aggregated, business-specific data models.
   * Features:
     + Optimized for reporting and analytical workloads.
     + Contains facts (metrics) and dimensions (descriptive attributes) for analysis.
   * Examples: Sales dashboards, revenue metrics by region.

### **Complexity in Data Delivery**

Organizations face several challenges in managing large-scale ETL pipelines:

* Dependency Management:
  + Difficult to manage dependencies between tables and handle transitions from streaming to batch processing.
* Error Recovery:
  + Errors in upstream processes can affect downstream systems, making recovery time-consuming.
* Operational Overhead:
  + Data engineers often spend excessive time on operational tasks rather than development.
* Impact on Teams:
  + Pipeline failures disrupt the workflows of data scientists, analysts, and other dependent teams.

### **What Makes Delta Live Tables (DLT) Unique?**

DLT addresses these complexities through several key capabilities:

1. **Declarative Pipeline Creation:**
   * Engineers define what needs to be done (e.g., transformations, data flows) rather than how.
   * Code written in SQL or Python specifies data sources, transformations, and target states.
2. **Automation:**
   * Automatically orchestrates and manages the ETL lifecycle, reducing manual intervention.
   * Handles task dependencies and retries for failed operations.
3. **Unified Batch and Streaming:**
   * Integrates batch and streaming data using a single API, reducing complexity in managing real-time data.

### **Key Benefits of DLT**

1. ETL Acceleration:
   * Simplifies and accelerates the creation of data pipelines.
   * Automatically orchestrates the execution of pipelines by constructing DAGs (Directed Acyclic Graphs).
   * Retries failed tasks and handles changes in data sources dynamically.
2. Automated Infrastructure Management:
   * Takes care of recovery, auto-scaling, and optimization without manual intervention.
   * Improves performance through automatic cluster tuning.
3. Data Quality Assurance:
   * Provides built-in data validation rules and monitoring capabilities.
   * Enforces quality controls such as dropping bad records or quarantining problematic data.
   * Logs quality metrics for future audits.
4. Declarative APIs:
   * SQL and Python APIs allow engineers to focus on business logic rather than error handling or system-level configurations.
   * Features automatic lineage tracking, highlighting relationships between data sources and transformations.
5. Observability and Monitoring:
   * Generates detailed lineage diagrams to visualize data flow and dependencies.
   * Provides row-level logging for troubleshooting.
   * Enables proactive monitoring with notifications and alerts.
6. Automated Error Handling:
   * Reduces downtime by automating error detection, recovery, and retries.
   * Includes easy replay capabilities for failed tasks, minimizing impact on downstream processes.
7. Scalable Workflow Orchestration:
   * Seamlessly integrates with tools like Apache Airflow, Azure Data Factory, and Databricks Jobs to manage workflows in multi-cloud environments.
   * Supports running pipelines as part of broader data workflows.

### **How Delta Live Tables Differentiate from Traditional ETL Tools**

1. Continuous Data Ingestion:
   * Uses an Auto Loader to efficiently load data into Delta Lake as it arrives in cloud storage.
   * Automatically detects and evolves the schema of incoming data, eliminating manual updates.
2. Integrated Ecosystem:
   * Fully integrates with the Databricks platform, making debugging and inspecting results easier.
   * Reduces the need for separate tools for orchestration, testing, and monitoring.
3. Declarative Design:
   * Encourages a simple declarative approach instead of procedural coding.
   * Engineers can focus on business insights, not the tools or infrastructure.

### **Example Use Cases**

* Streaming Analytics:
  + Real-time fraud detection by continuously ingesting transaction data.
* Batch Analytics:
  + Building dashboards for sales trends and customer segmentation.
* Data Science Pipelines:
  + Preparing clean datasets for machine learning models by automating data transformations.

### **Conclusion**

Delta Live Tables offer a comprehensive, automated solution for modern data engineering needs. It bridges the gap between data lakes and data warehouses, enabling faster, more reliable, and quality-driven ETL pipelines with minimal manual effort.