Delta Live Tables UI & SQL Pipelines

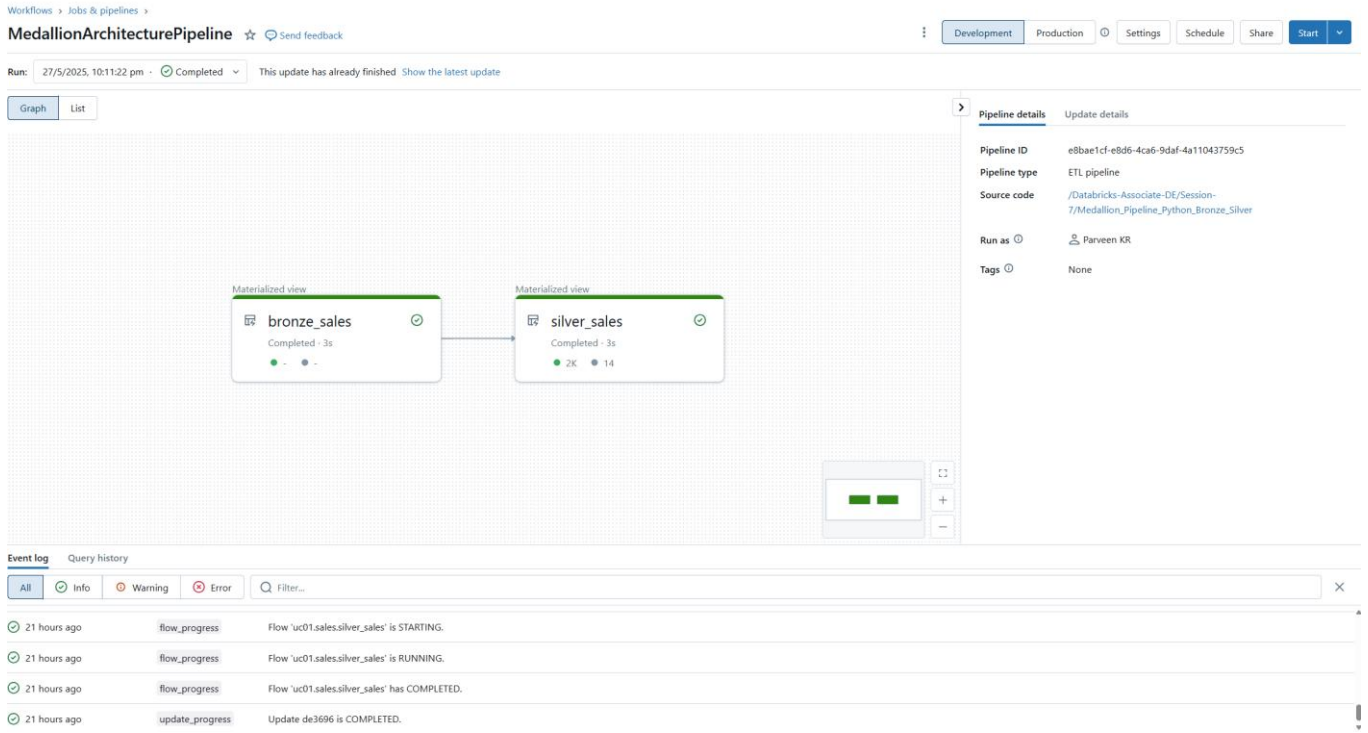
Navigating UI and managing data pipelines effectively

MC ID: 7124506 | Parveen KR | Parveen.R@hotmail.com

Agenda

- Introduction to Delta Live Tables UI
- Navigating the DLT UI
- Creating & Configuring Pipelines in UI
- Scheduling Pipelines
- Monitoring Pipelines
- Configuring Data Quality Checks
- SQL Declarative Pipelines

Understanding Delta Live Tables UI



Navigating the DLT UI

Learn how to effectively navigate the Delta Live Tables user interface to access various features.

Creating DLT Pipelines

Understand the steps involved in creating DLT pipelines using the user interface for data processing.

Configuring DLT Pipelines

Learn how to configure DLT pipelines via the UI to optimize data workflows based on specific requirements.

Scheduling DLT Pipelines

Explore how to schedule DLT pipelines using the UI to automate data processing tasks efficiently.

Creating and Configuring DLT Pipelines

Monitoring Pipeline Progress

Effective monitoring of pipeline progress, logs, and alerts is essential for smooth operation.

Creating SQL Pipelines

Write SQL-based DLT pipelines using the CREATE LIVE TABLE syntax for efficient data processing.

Incremental and Streaming Pipelines

Implementing incremental and streaming SQL pipelines allows real-time data processing and analysis.

Pipeline Lifecycle Management

Manage the pipeline lifecycle effectively by starting, stopping, and updating as needed.

Data Quality Checks

Apply data quality checks and expectations in DLT pipelines to ensure reliable outputs.

Introduction to Delta Live Tables UI

MC ID: 7124506 | Parveen KR | Parveen.R@hotmail.com

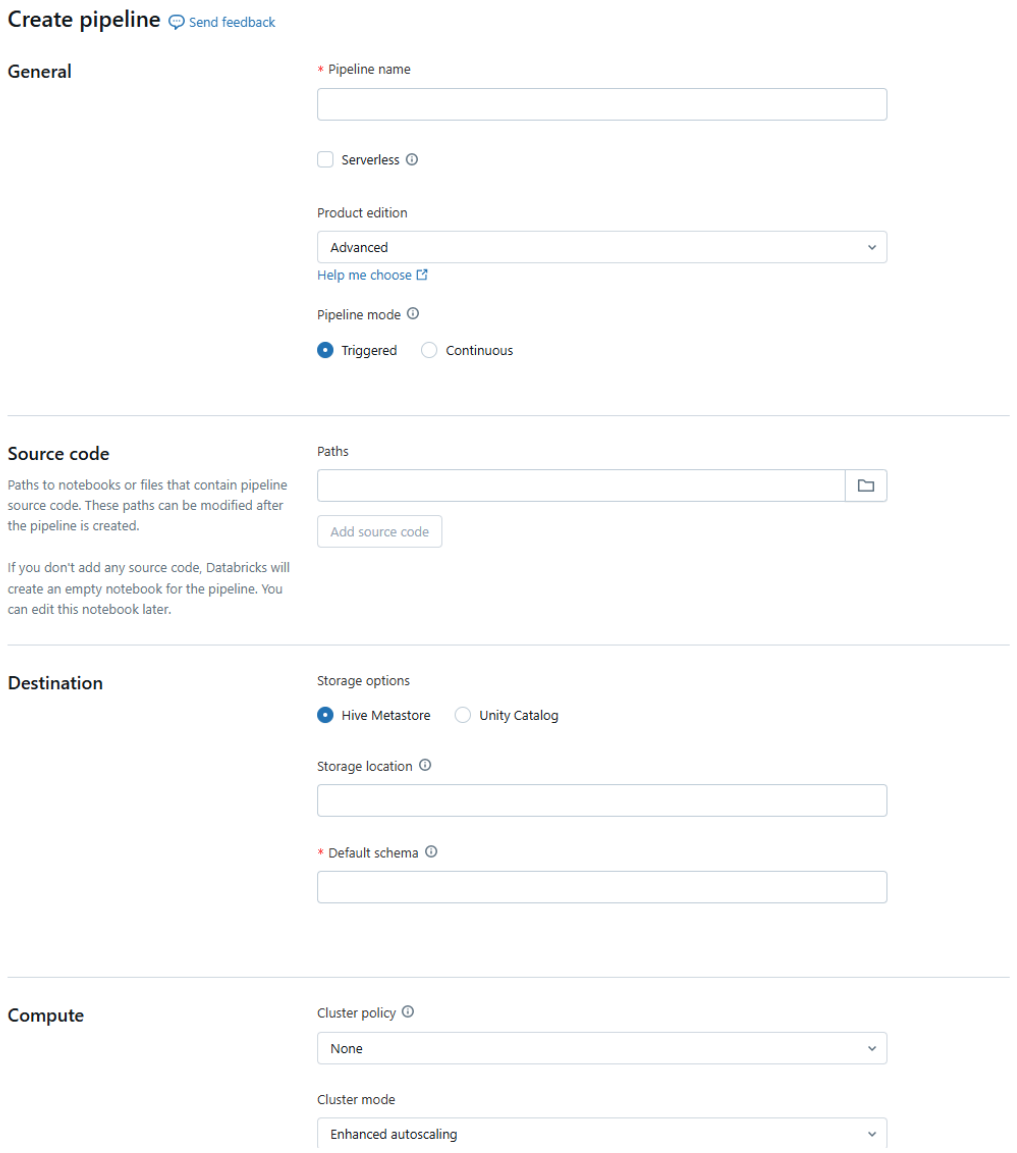
Overview of Delta Live Tables UI

Delta Live Tables UI

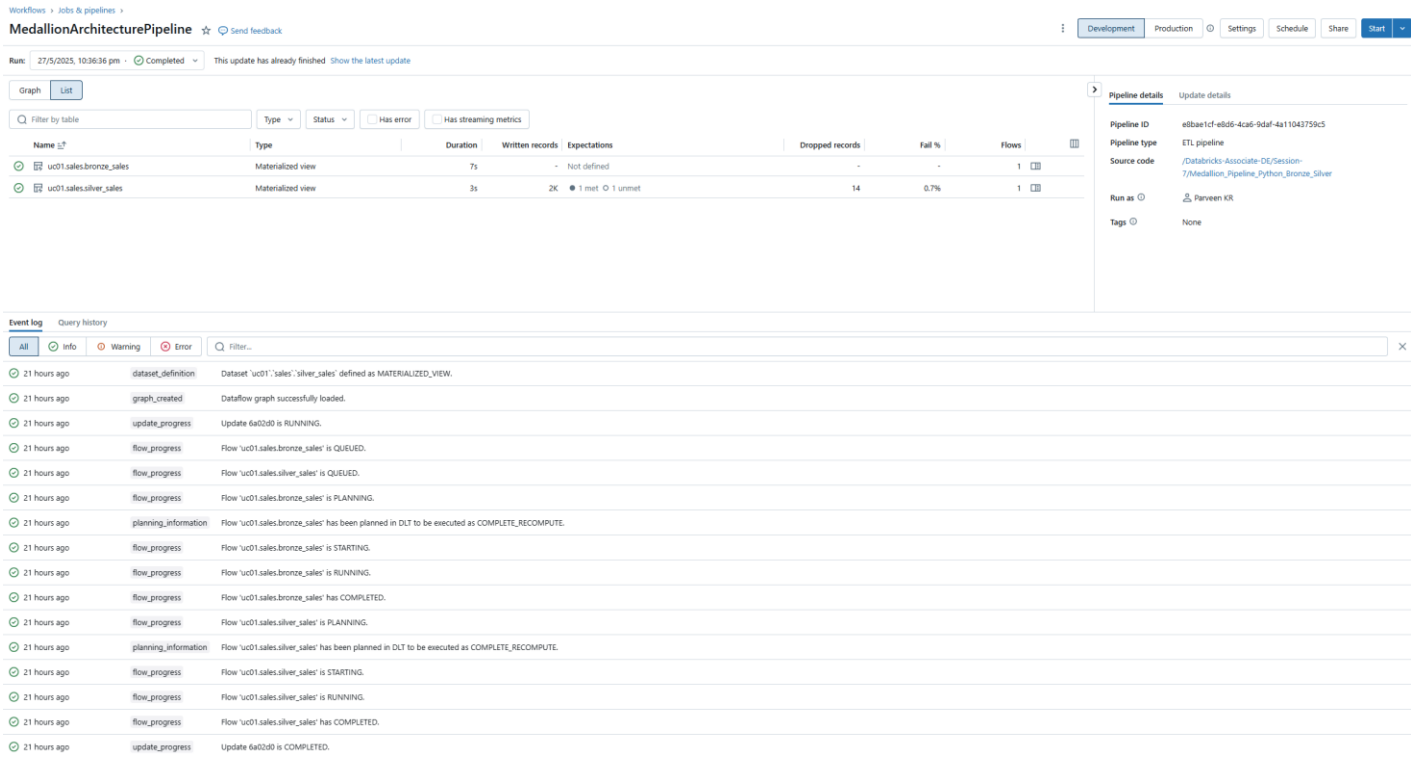
Delta Live Tables UI is a web interface designed for managing data pipelines effectively in Databricks.

Simplified Pipeline Management

This UI simplifies the building, monitoring, and troubleshooting of data pipelines, enhancing efficiency.



Key UI Components



Pipelines List

The Pipelines List is a comprehensive overview of all available data processing pipelines, essential for project management.

Pipeline Overview & Details

This component provides detailed insights into specific pipelines, including their performance and configurations.

Logs & Alerts

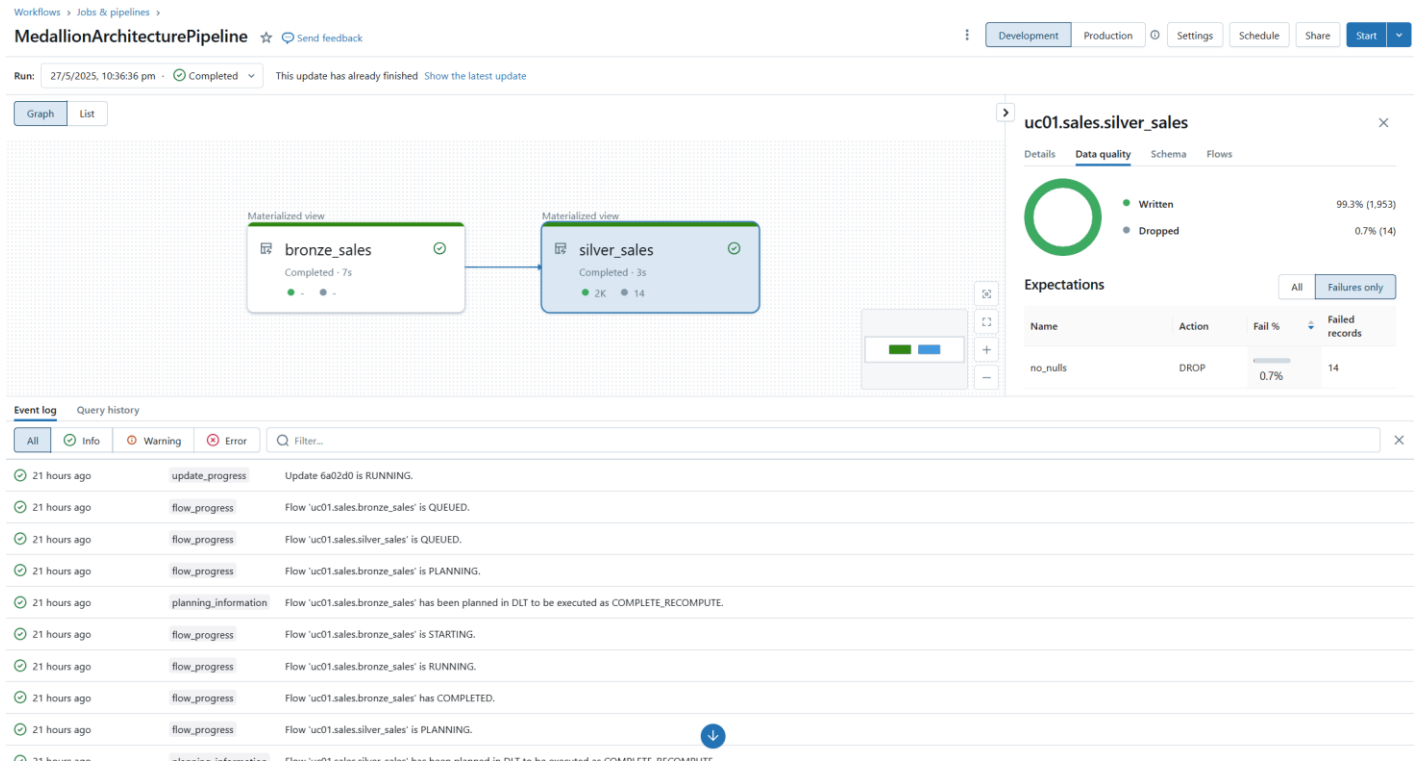
Logs & Alerts monitor system activity and notify users of important events or errors, ensuring timely responses.

Quality Checks Dashboard

The Quality Checks Dashboard tracks the performance metrics of pipelines, ensuring data quality and reliability.

Navigating the DLT UI

Pipeline List & Details



Accessing DLT UI

Access the DLT UI from the Databricks Workspace sidebar for an overview of your pipelines.

Pipeline List Overview

The Pipeline List provides an overview of all pipelines, their statuses, and last run information.

Pipeline Details Page

The Pipeline Details page summarizes configurations, progress, lineage, and logs for each pipeline run.

Alerts and Notifications

Stay informed with alerts on pipeline failures or quality check violations through the dedicated alerts tab.

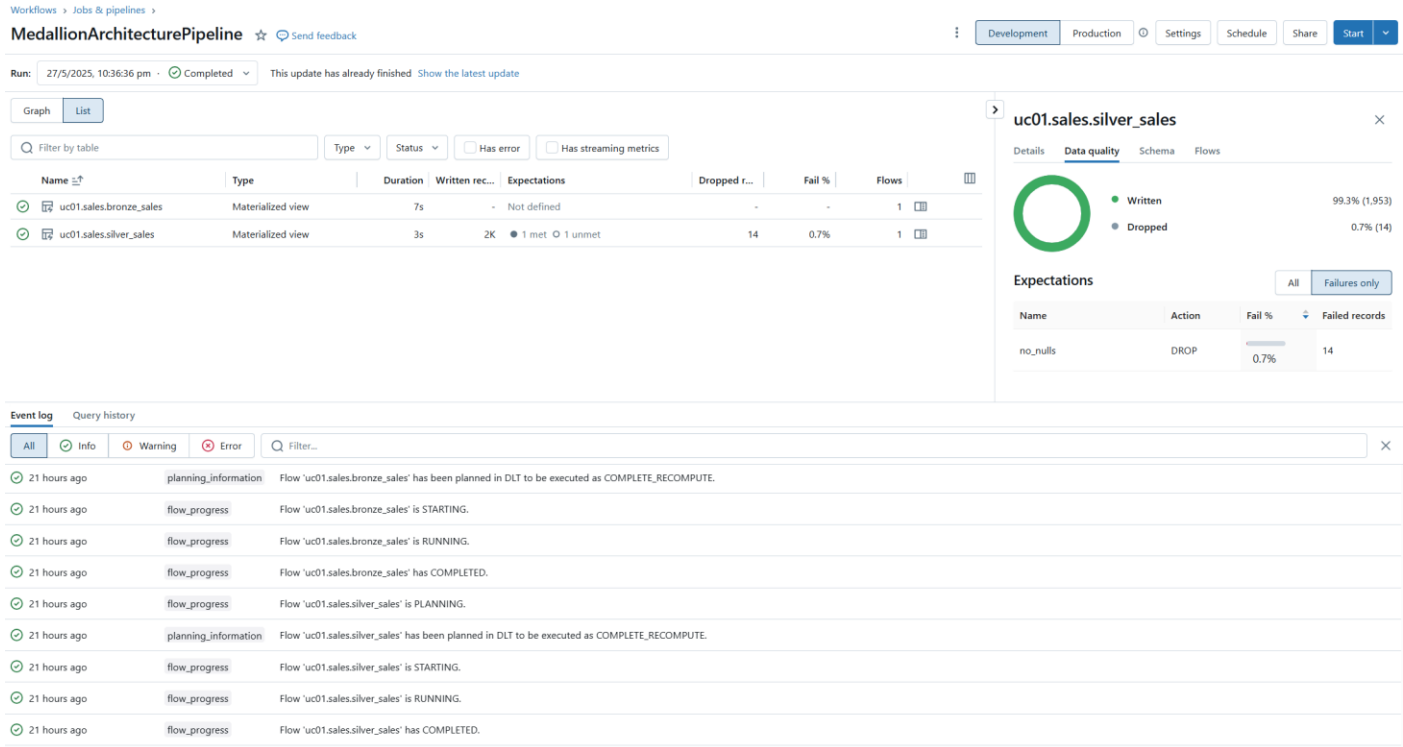
Accessing DLT UI

Accessing DLT UI

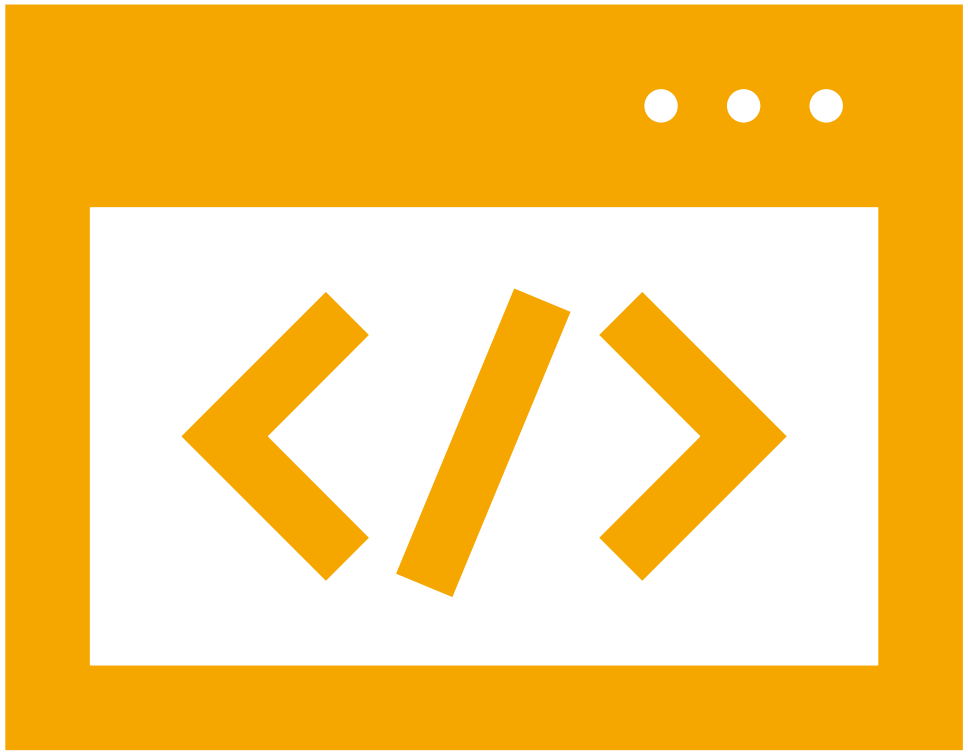
Access the DLT UI directly from the Databricks Workspace sidebar to manage your data pipelines.

Viewing Pipeline List

The Pipeline List displays all pipelines, their current status, and information about the last run.

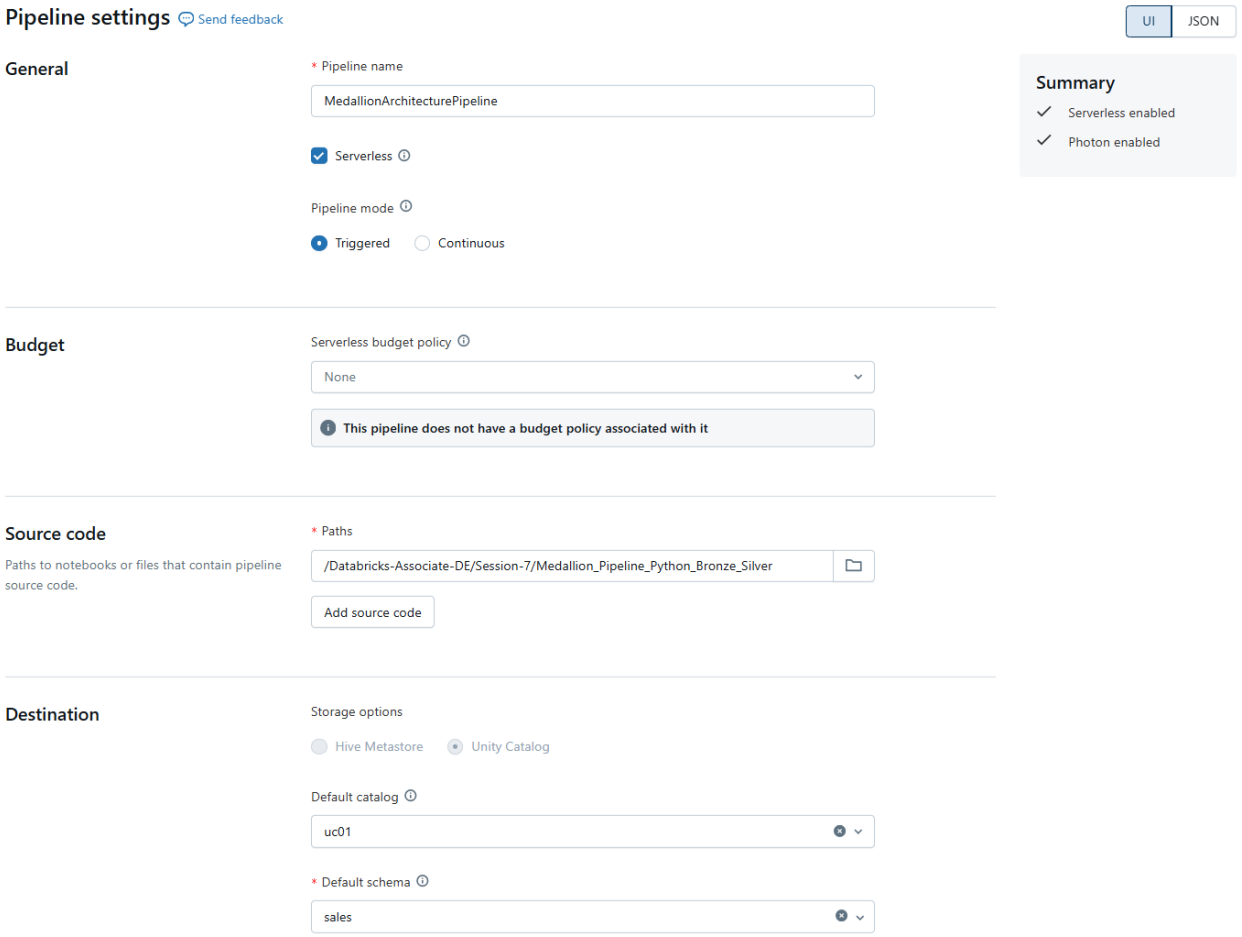


Creating & Configuring Pipelines in UI



MC ID: 7124506 | Parveen KR | Parveen.R@hotmail.com

Steps to Create Pipeline



Create a Pipeline

To start, click on the 'Create Pipeline' button in the user interface to initiate the process.

Pipeline Configuration

Provide a unique pipeline name and select the appropriate cluster configuration based on your needs.

Select Pipeline Mode

Choose between Batch or Continuous mode to define how the pipeline will operate.

Define Storage Location

Set a storage location for pipeline checkpoints and tables to ensure data integrity.

Best Practices for Pipeline Configuration

Auto-Scaling Clusters

Setting up auto-scaling clusters can significantly optimize costs by adjusting resources based on workload demands.

Latency Requirements

Choosing the right processing mode is crucial based on the required data latency to ensure timely data delivery.

Scheduling Pipelines

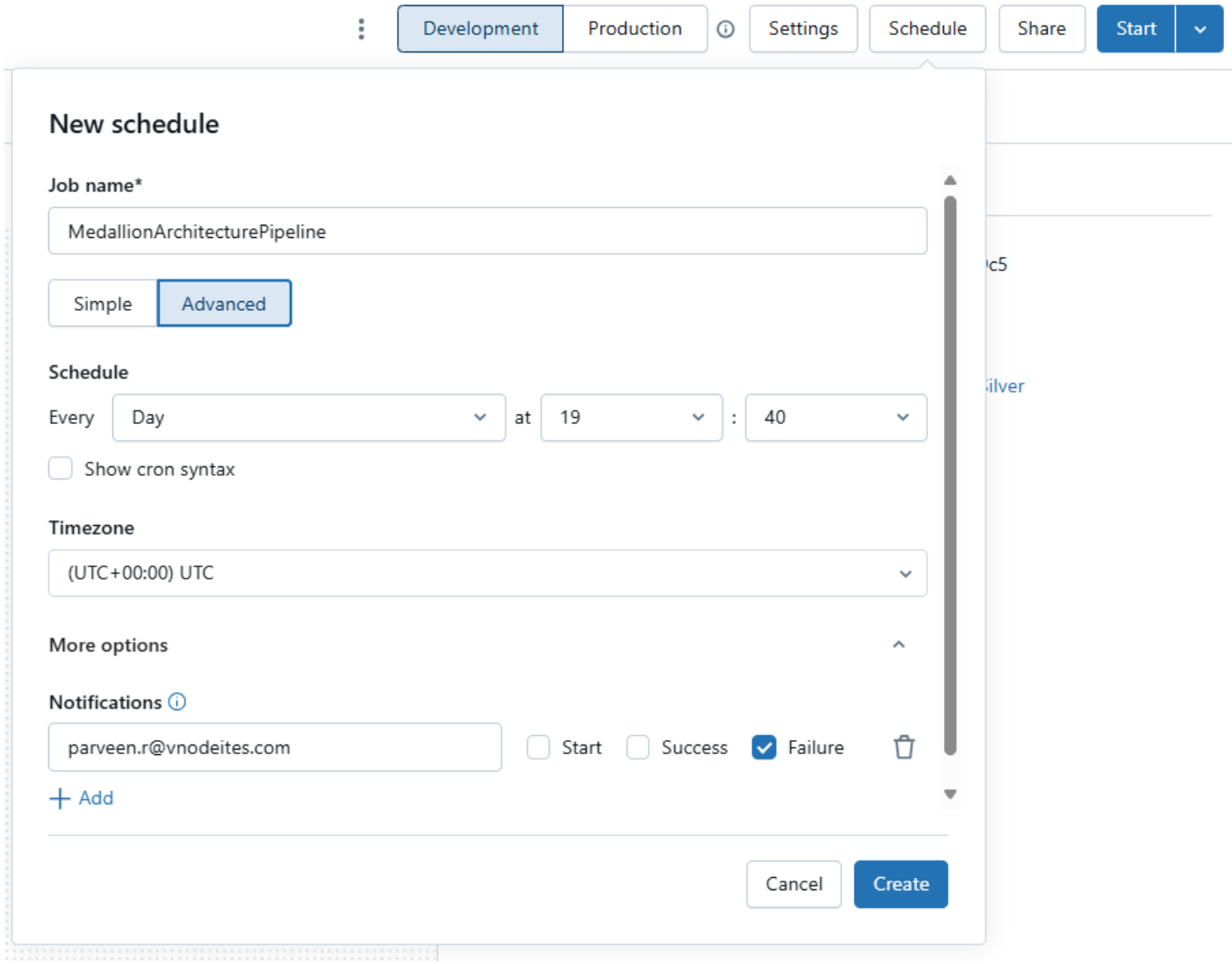
Batch Scheduling Options

Batch Scheduling

Batch scheduling allows users to run tasks manually or at predefined intervals, improving workflow efficiency.

Cron Syntax Support

Users can employ cron syntax to set fixed intervals for batch processes, enhancing flexibility and control.



Streaming Pipeline Options

Continuous Streaming

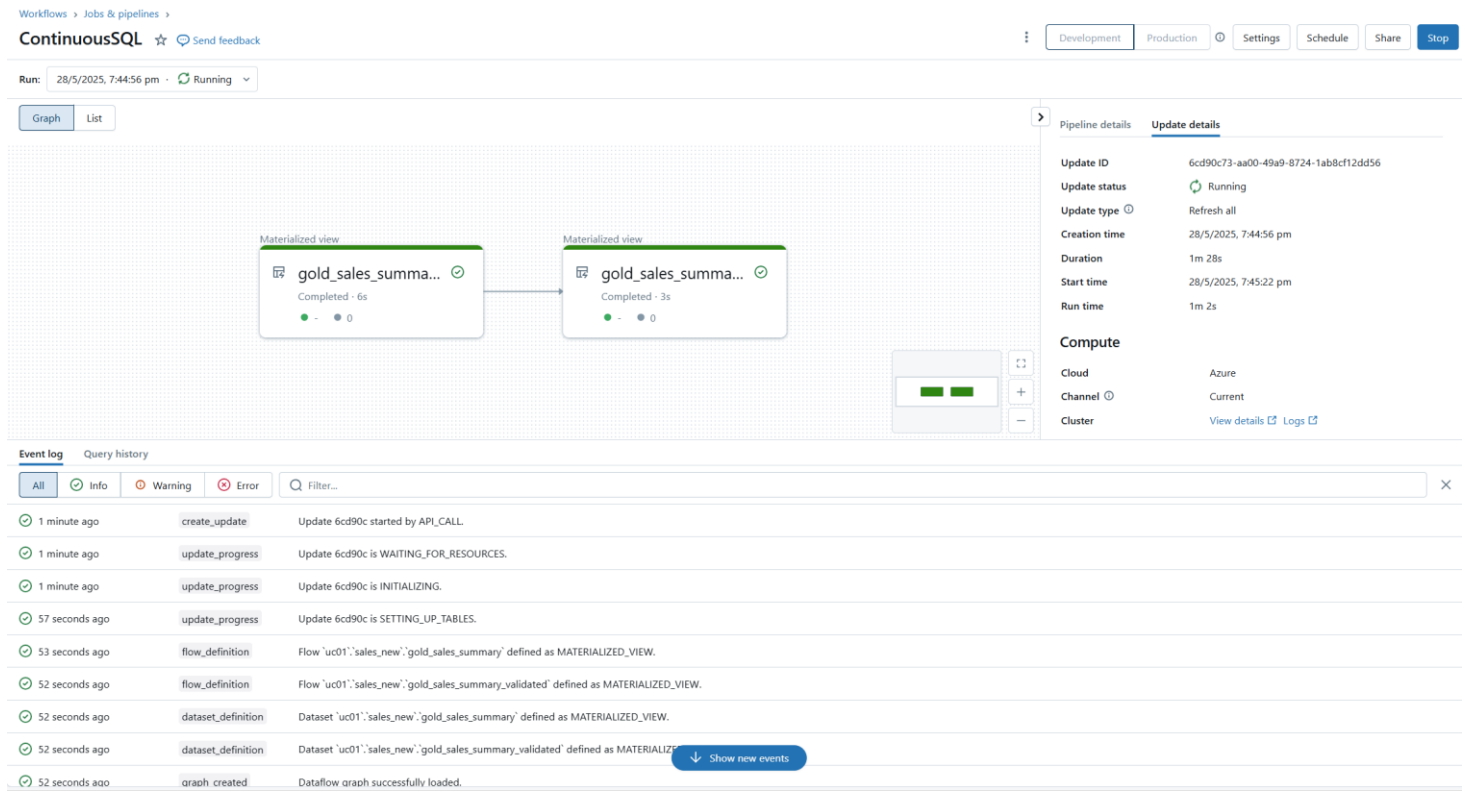
Streaming pipelines operate continuously, allowing real-time data processing with minimal latency for immediate insights.

User Interface Control

The user interface offers simple trigger controls to start, stop, and restart streaming pipelines efficiently.

Monitoring Metrics

Users can monitor job duration and resource usage metrics through the UI, ensuring optimal performance.



Monitoring Pipelines

Progress and Logs

Progress Monitoring

The progress bar displays the percentage completion of the current run, providing clear visibility into processing status.

Detailed Logs

Logs capture detailed Spark events, errors, and data validation results, essential for troubleshooting and analysis.

Alerts and Notifications

Configured alerts notify users of failures or expectation violations, enabling timely responses to issues.

Overview Metrics

Metrics such as rows processed, execution time, and resource utilization provide insight into processing efficiency.

Configuring Data Quality Checks

Common Expectations

Enforcing Data Quality Rules

Expectations are vital for enforcing data quality rules during data pipeline execution, ensuring reliable data processing.

Common Expectations

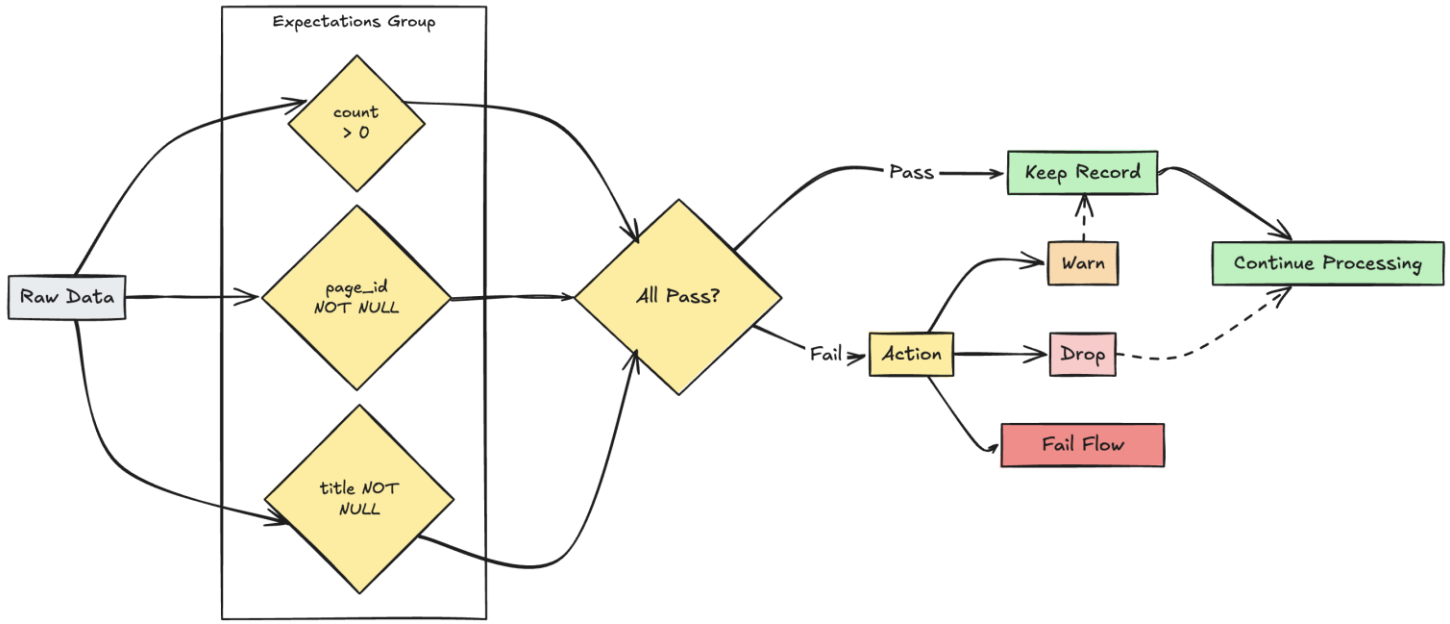
Common expectations include checks for non-null values, uniqueness, and value ranges to maintain data integrity.

Handling Failures

Configuring how failures are handled can trigger alerts or halt pipeline runs based on specific conditions.

Configuration Methods

Expectations can be configured through user interfaces or declarative SQL, allowing flexibility in implementation.



SQL Declarative Pipelines

MC ID: 7124506 | Parveen KR | Parveen.R@hotmail.com

Overview of SQL Pipelines

Declarative SQL Syntax

DLT allows for the creation of pipelines using easy-to-understand SQL syntax, streamlining the coding process.

Focus on What, Not How

This approach simplifies development by emphasizing the desired outcome instead of the implementation details.

Table Creation Command

The key command 'CREATE LIVE TABLE' defines tables in the pipeline, ensuring structured data management.

Dependency Management

DLT automatically handles dependencies and execution order, making the pipeline efficient and user-friendly.

CREATE LIVE TABLE Syntax

Basic Syntax

The basic syntax for creating a live table involves using CREATE LIVE TABLE followed by the table name and a SELECT statement.

Data Transformations

CREATE LIVE TABLE can include transformations, filters, and aggregations to manipulate data effectively.

Defining Expectations

You can define expectations inline to ensure data integrity, such as NOT NULL and UNIQUE constraints.

Creating Views

Views can also be created using CREATE LIVE VIEW for simplified data access and management.

```
CREATE LIVE TABLE gold_sales_summary
COMMENT "Aggregated daily sales totals"
AS
SELECT
  OrderDate,
  SUM(Quantity * UnitPrice) AS total_sales,
  COUNT(DISTINCT CustomerId) AS unique_customers
FROM LIVE.silver_sales
GROUP BY OrderDate;
```

Hands-on-Labs