Release Notes

erwin Data Quality V3.1.3

Prepared by

Features

1. Remediation - Dedupe

The deduplication functionality enables users to detect and merge duplicate or similar records across spreadsheets and databases by leveraging machine learning techniques. It is especially valuable for organizations handling messy or inconsistent data, such as variations in names, addresses, or other key fields.

Functionality

- **ML-Based Matching Engine:** Automatically learns how to detect duplicates and similar records based on user-provided examples.
- **Guided Workflow:** A step-by-step interface for uploading data, training the model, reviewing results, and exporting clean datasets.
- **Customizable Field Matching:** Supports complex logic for comparing names, emails, addresses, business data, and more.
- **Scalable Integration:** Compatible with large-scale datasets from CRM, ERP, and marketing systems.

Use Cases:

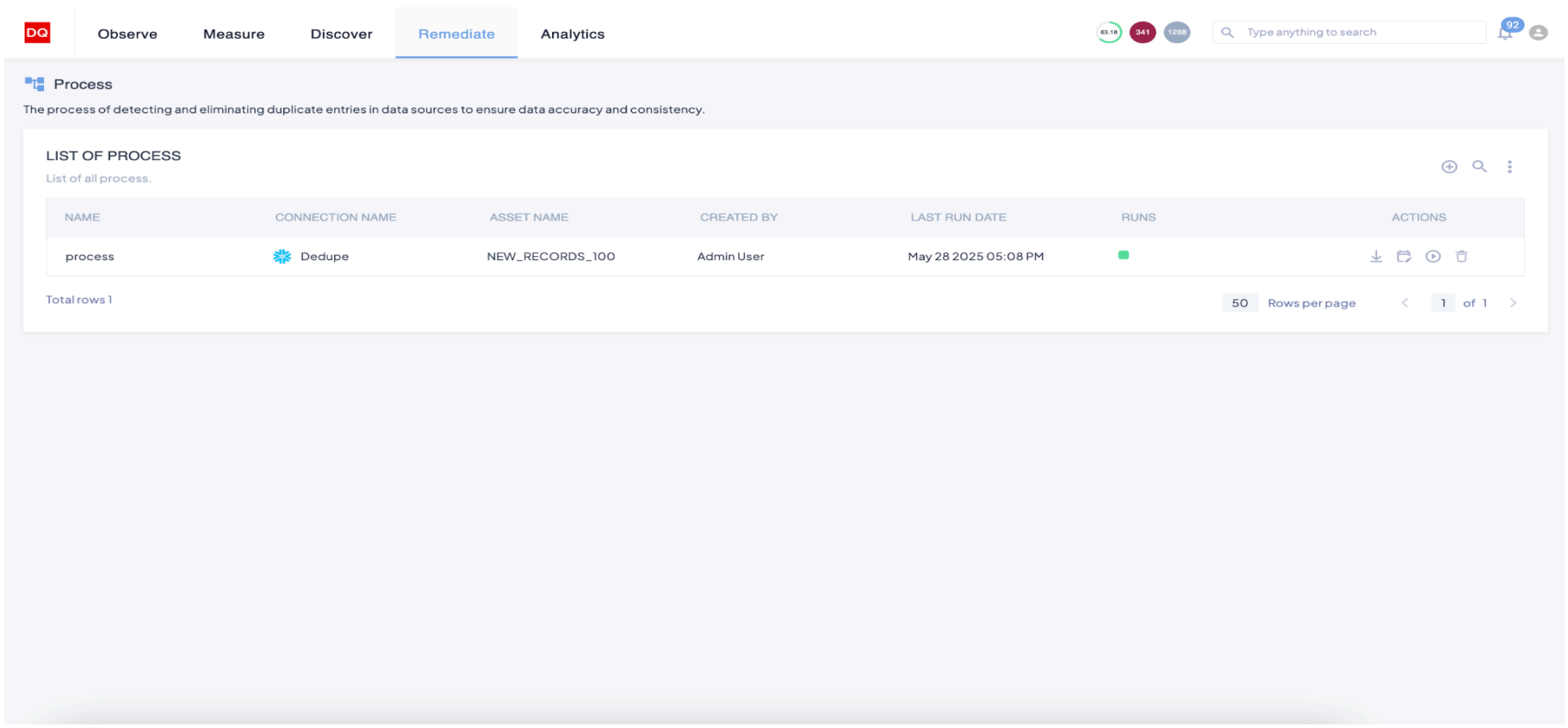
- Merging customer records from multiple sources
- Cleaning and enriching mailing lists
- Consolidating vendor and business data
- Preparing data for analytics, reporting, or CRM migration
- Resolving duplicates across merged databases

Business Problems Solved:

- Eliminates redundant or duplicate entries, improving data quality
- Saves time and labor through automation
- Enhances reporting accuracy and customer insights
- Reduces the cost of poor data quality in operations and marketing

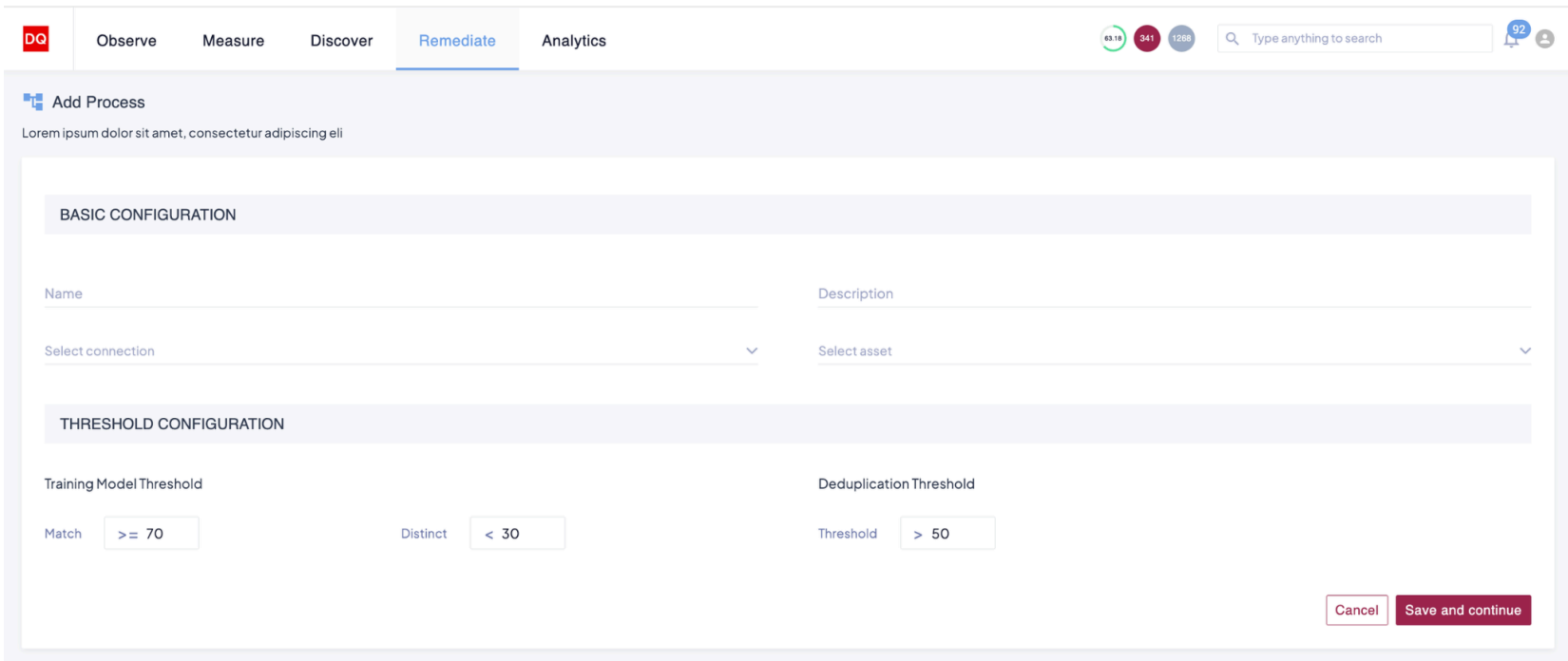
Follow the steps below to create a dedupe process in DQLabs:

Step 1: Navigate to Remediate → Process



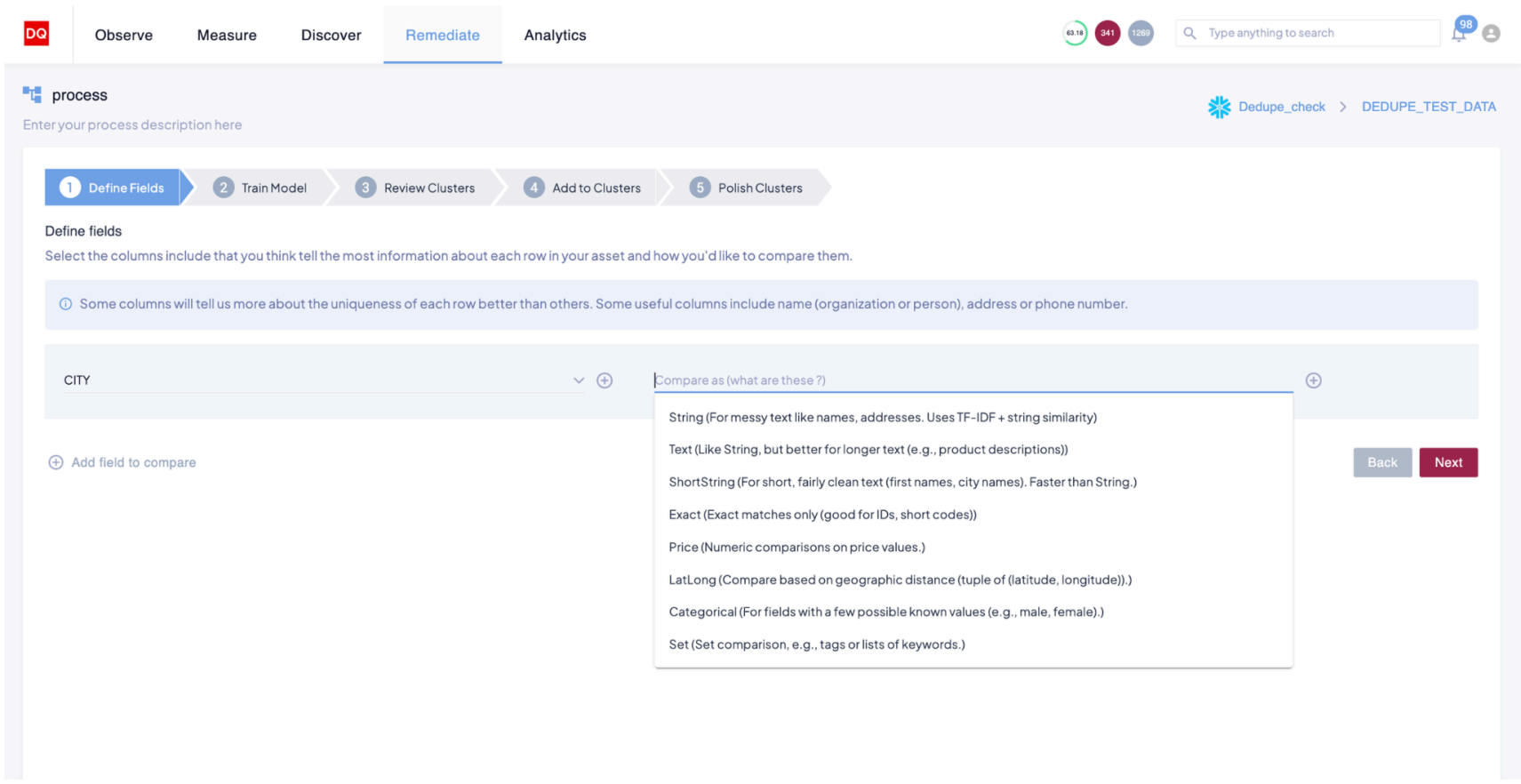
Step 2: Click on the "+" icon and provide the following details, save, and continue

- Basic Configuration
  - Name
  - Description
  - Select Description
  - Select Asset
- Threshold Configuration
  - **Match Percentage** - Refers to the confidence score that two records refer to the same entity. Helps prioritize high-confidence matches for automated deduplication or user review.
  - **Distinct Percentage** - The complement of Match Percentage: this is the confidence that two records are different. Example: 15% match score = 85% distinct = low likelihood of duplication.
  - **Threshold Percentage** - A cutoff value used to decide what is considered a match.



Step 3: Define fields in this section by selecting the attributes and selecting the compare as an option as described below, and proceed to the next step.

- Choose the attributes (columns) to consider when identifying duplicates.
- Typical fields: Name, Email, Phone, Address, Company, etc.
- For each field, specify the type:
    - **String**: For names, business names, etc.
    - **Address**: For structured postal addresses.
    - **Set**: For unordered lists like tags or categories.
    - **Exact**: For IDs or categories that must match exactly.
    - **Text:** For longer texts, such as descriptions
    - **Price:** For numeric comparison
    - **Latlong:** For geographic distance
    - **Categorical:** For possible known values

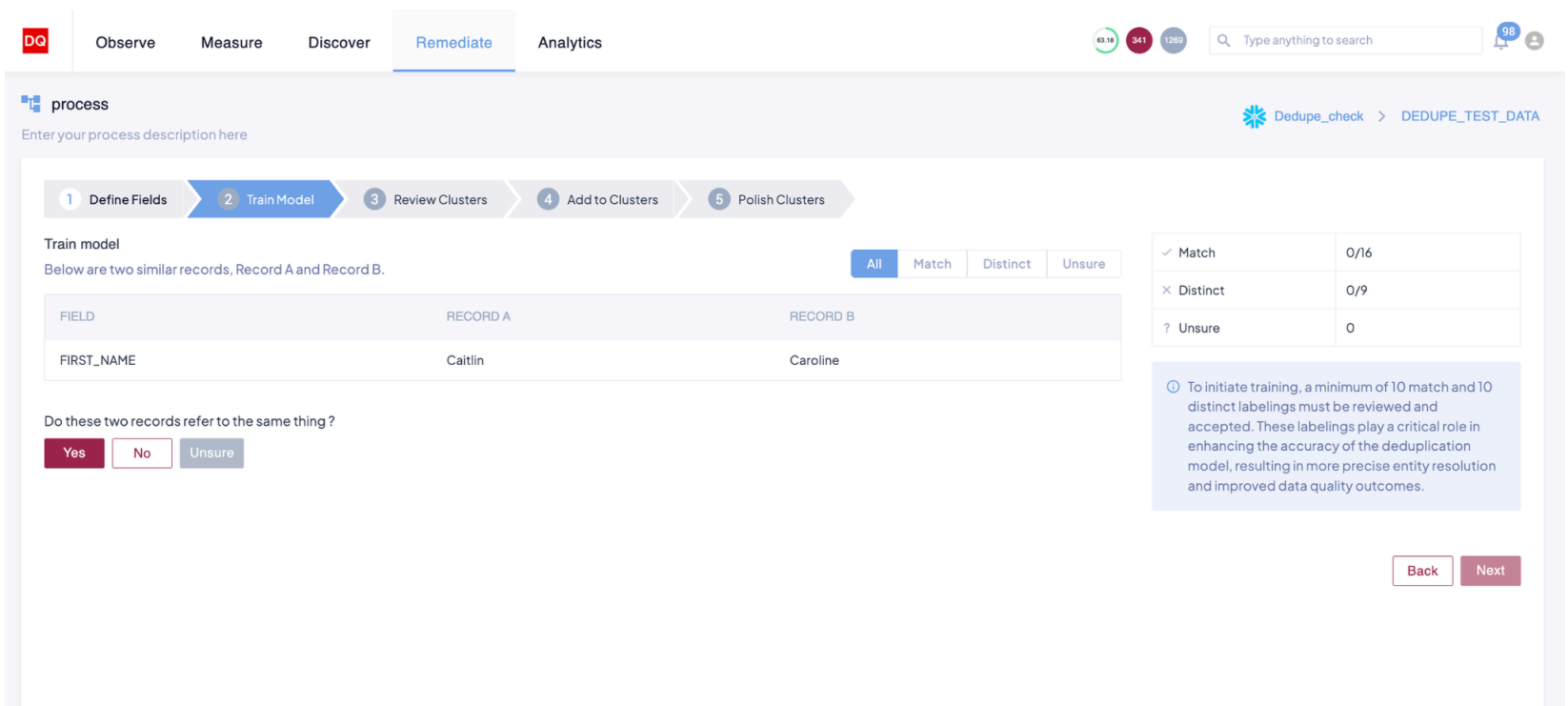


The user can add multiple columns to compare.

Step 4: This section will ask the users to match the records for Distinct and duplicate

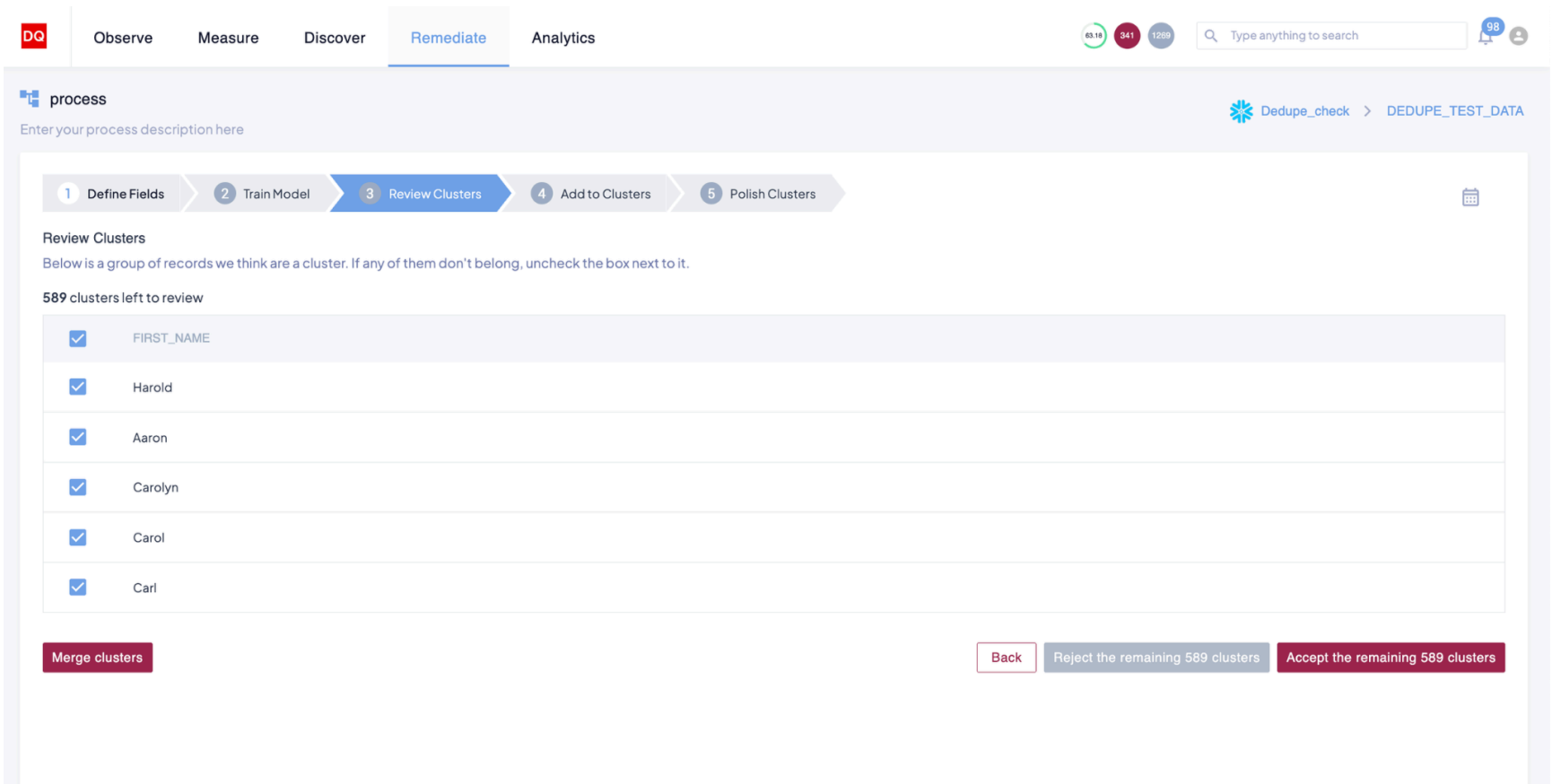
- The system will show you pairs of records and ask: Are these the same?
- You manually label pairs as:
    - **Match** (same entity)
    - **Distinct** (different entities)
    - **Unsure** (skip or decide later)
- The more you label, the smarter the model gets.
- DQLabs uses active learning to focus on the most informative examples.

Click “Yes” to Match and “No” to Distinct. A total of 10 records should be mapped for both Match and Distinct to proceed further.

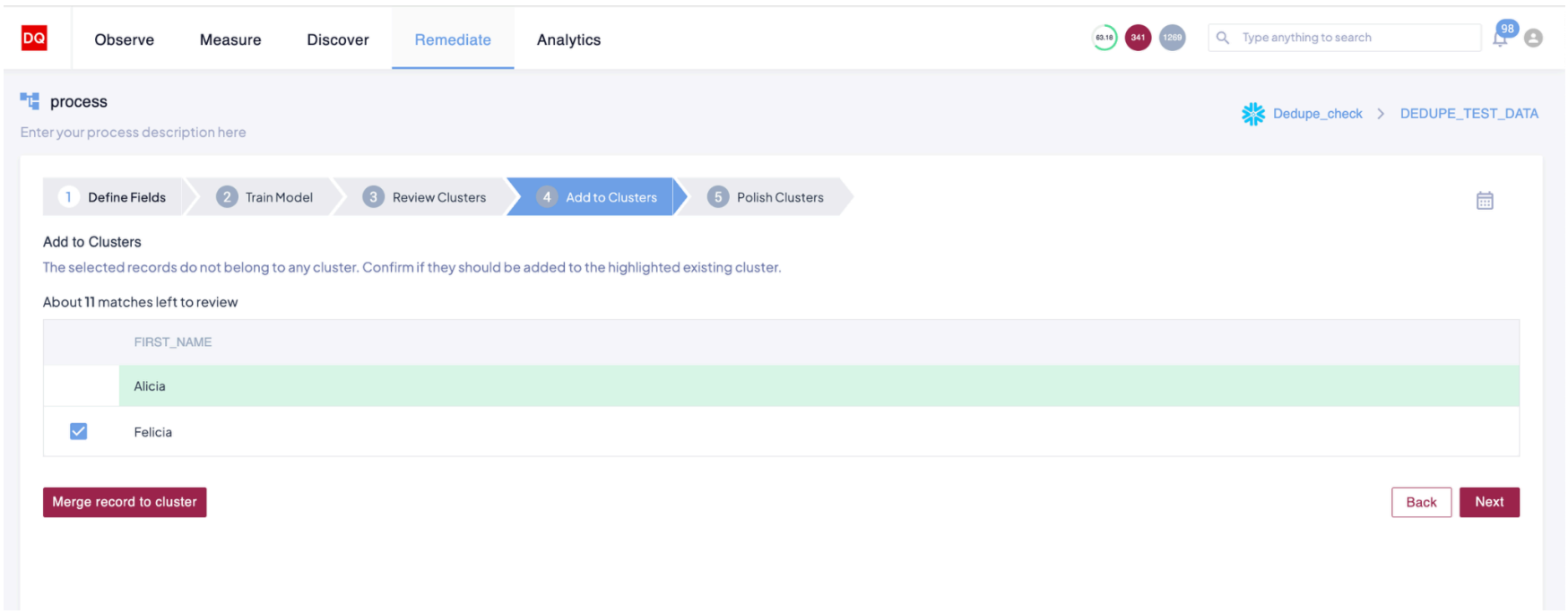


Step 5: The records that match are grouped as a cluster, and the user can review and confirm the records

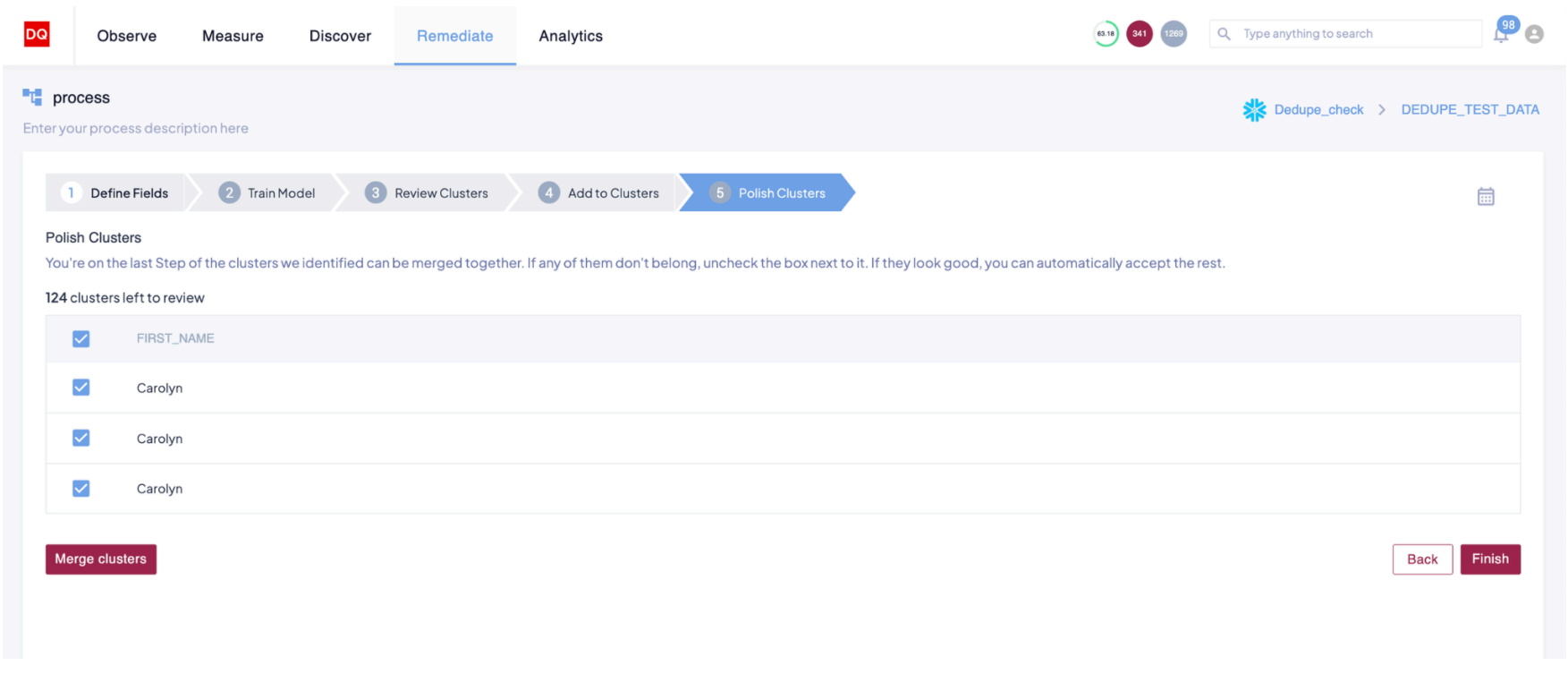
- After training, DQLabs shows **clusters of records** that are the same entity.
- Review and adjust clusters:
  - Merge or split as needed.
  - Confirm or reject system decisions.



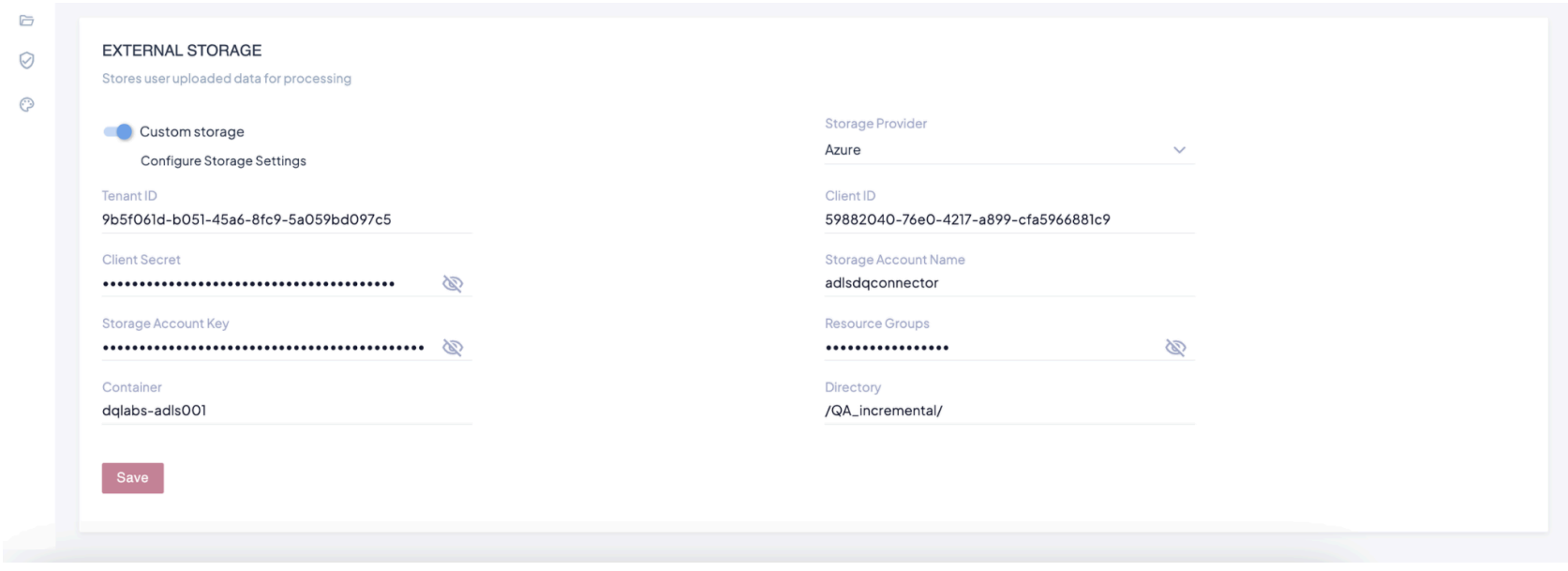
Step 6: The records that do not belong to any clusters are displayed in the "Add to cluster" section. The users can either add the records to the cluster or proceed to the next step



Step 7: The “Polish Cluster“ section allows the users to fine-tune and manually review the clusters of matched records generated by the machine learning model. The users can merge clusters manually and click on **finish**



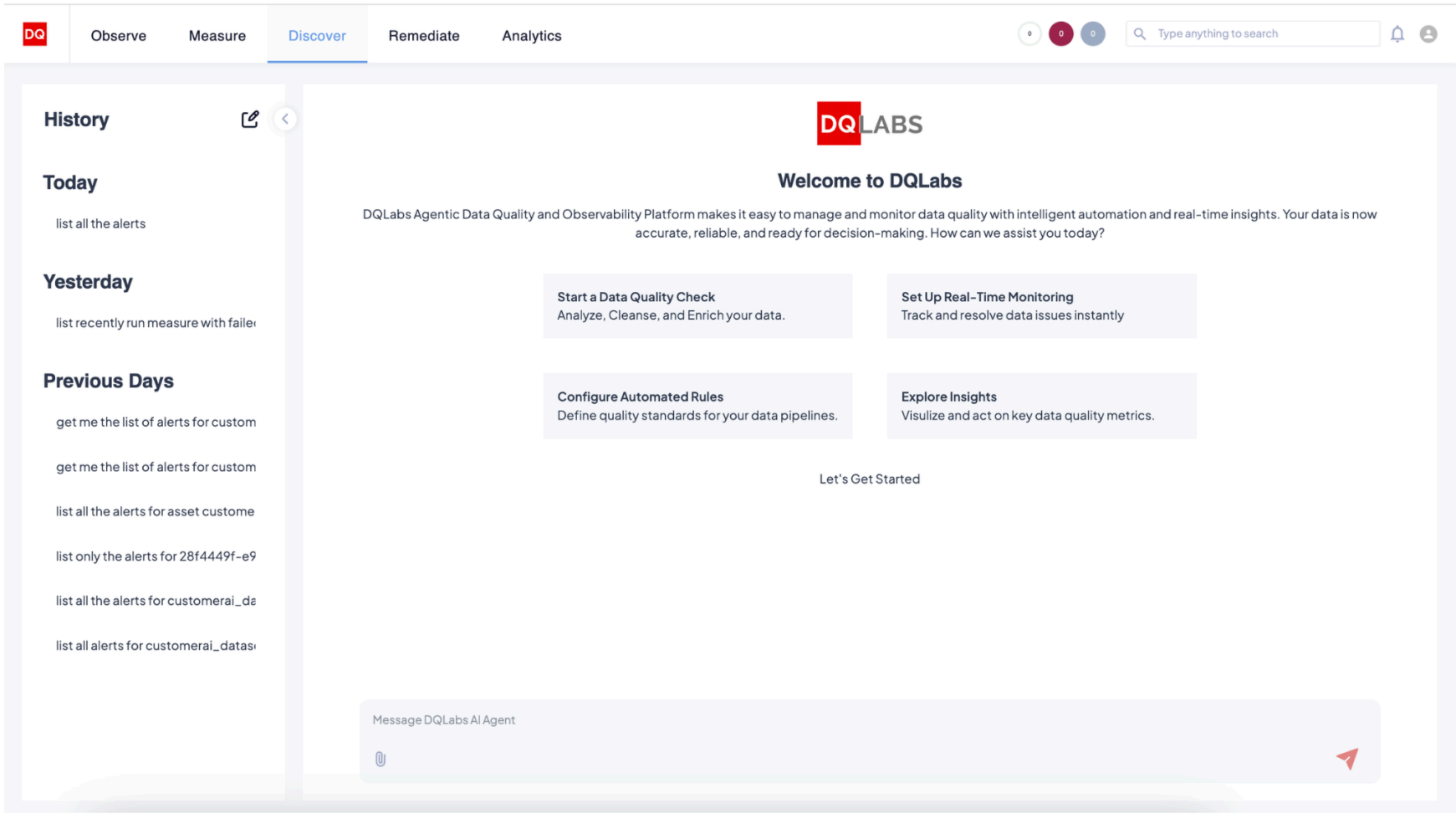
Once the setup is complete, sample records will be downloaded to the local machine where the user can review the results. Also, the user can create schedules to run the process, and then processed files are located at the storage specified under the external storage section under settings



2. Converse AI

DQLabs empowers users to leverage agentic AI to create and manage actions within the platform. Through natural language commands, users can explore data assets and attributes, receive intelligent suggestions for data quality checks, and automatically generate SQL-based validation rules. This simplifies workflows by allowing users to run checks, preview failed records, and receive automated recommendations for corrective actions, minimizing manual effort and speeding up issue resolution.

Additionally, DQLabs offers robust capabilities for impact analysis and data lineage, helping users understand how data quality issues affect downstream systems. The platform supports threshold recommendations, alert management, and both bulk and individual issue creation, making it easy to monitor and address data problems. By combining automation, AI-driven insights, and an intuitive natural language interface, DQLabs streamlines end-to-end data quality operations and fosters greater confidence in enterprise data. The Converse AI feature is accessible via Discover → Converse.



The following actions are supported in DQLabs:

Discovering Assets & Attributes

- **List Top Assets**
  Command: list top 10 assets
  Displays the most used or critical data assets in the environment.
- **View Asset Attributes**
  Command: show attributes related to <ASSET_NAME>
  Retrieves schema or column-level metadata for a specified asset.

Data Quality Rules & Measures

- **Recommend Quality Checks**
  Command: recommend data quality check for <ATTRIBUTE_NAME>
  Suggests applicable validation rules (e.g., null check, range check, uniqueness).
- **Show SQL for Rules**
  Commands:
  - Show Range Check in SQL format using DDL
  - Show all rules in SQL format using DDL
    Generates SQL expressions for enforcing data quality rules.
- **Create Measures**
  Commands:
  - Create a measure for this
  - Create a measure for all
    Defines and stores measures to monitor data quality over time.

Execution & Monitoring

- **Run Checks**
  Command: run this
  Executes the selected rule or measure on the data.
- **Preview Failed Records**
  Command: preview failed records for the above
  Displays data entries that violated the applied quality checks.
- **Suggest Corrective Actions**
  Command: Suggest corrective actions for this
  Recommends ways to fix data quality issues (e.g., clean, replace, ignore).

Issue Management

- **Threshold Recommendation**
  Command: recommend threshold for a measure null on <ASSET_NAME> and attribute <ATTRIBUTE_NAME>
  Suggests a suitable threshold based on data profiling.
- **Issue Creation Options**
  Commands:

- Grouped Issue:
  Create issue for all alerts within the measure <MEASURE_NAME> for <ATTRIBUTE_NAME> <ASSET_NAME> and tag all alerts to the same issue
- Per-Asset Issue:
  Create an issue per alert in the asset <ASSET_NAME>
- Per-Attribute Issue:
  Create an issue per alert in the attribute <ATTRIBUTE_NAME> from asset <ASSET_NAME>
  Facilitates structured tracking of data issues.

Alerts Management

- **Manage Alert Statuses**
  Commands:
  - Mark as normal for all alerts with Priority High
  - Unmark as normal for all alerts with Priority High
    Used to suppress or reactivate specific alerts.
- **Alert Insights**
  Command: Show insights for alert <ALERT_ID>
  Returns a summary including root cause and downstream impact.

---

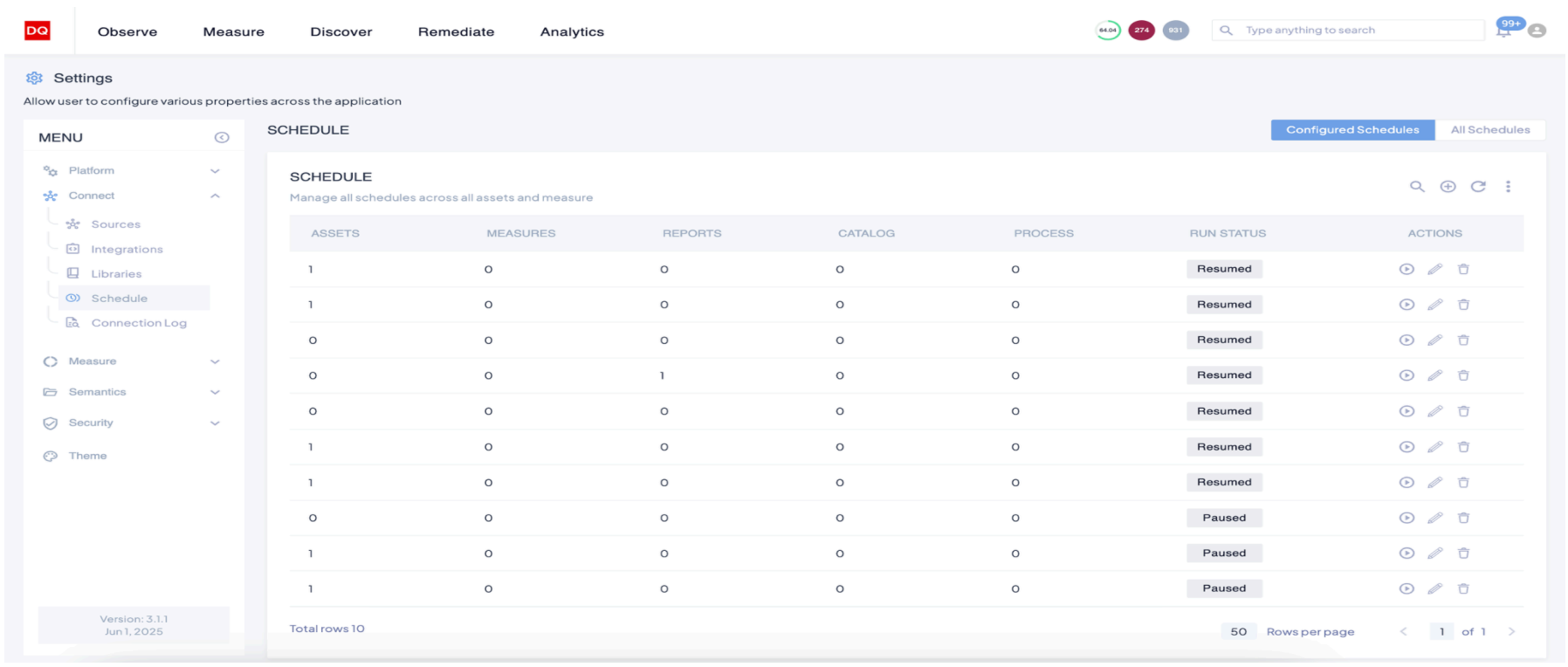
3. Ability to Pause and Resume Job Schedules Without Backfilling Past Runs

This feature allows users to pause and resume job schedules from the Schedule Management page, particularly useful during maintenance periods. It prevents unnecessary job runs during downtimes and ensures the system resumes smoothly without backfilling skipped executions. Bulk update functionality simplifies status management across multiple schedules. The following are the benefits of the new feature

- Prevent unnecessary job executions during planned maintenance or downtimes.
- Ensure that job schedules resume cleanly from the next valid occurrence.
- Improve operational efficiency through bulk update capabilities.
- Enhance transparency and traceability with audit logs.

Pause a Schedule

- Navigate to the **Settings → Schedule Page**.
- Locate the desired schedule and select the **Pause** action.
- Once paused:
  - The schedule status will update to **"Paused"**.
  - No job executions will be triggered while the schedule is paused.

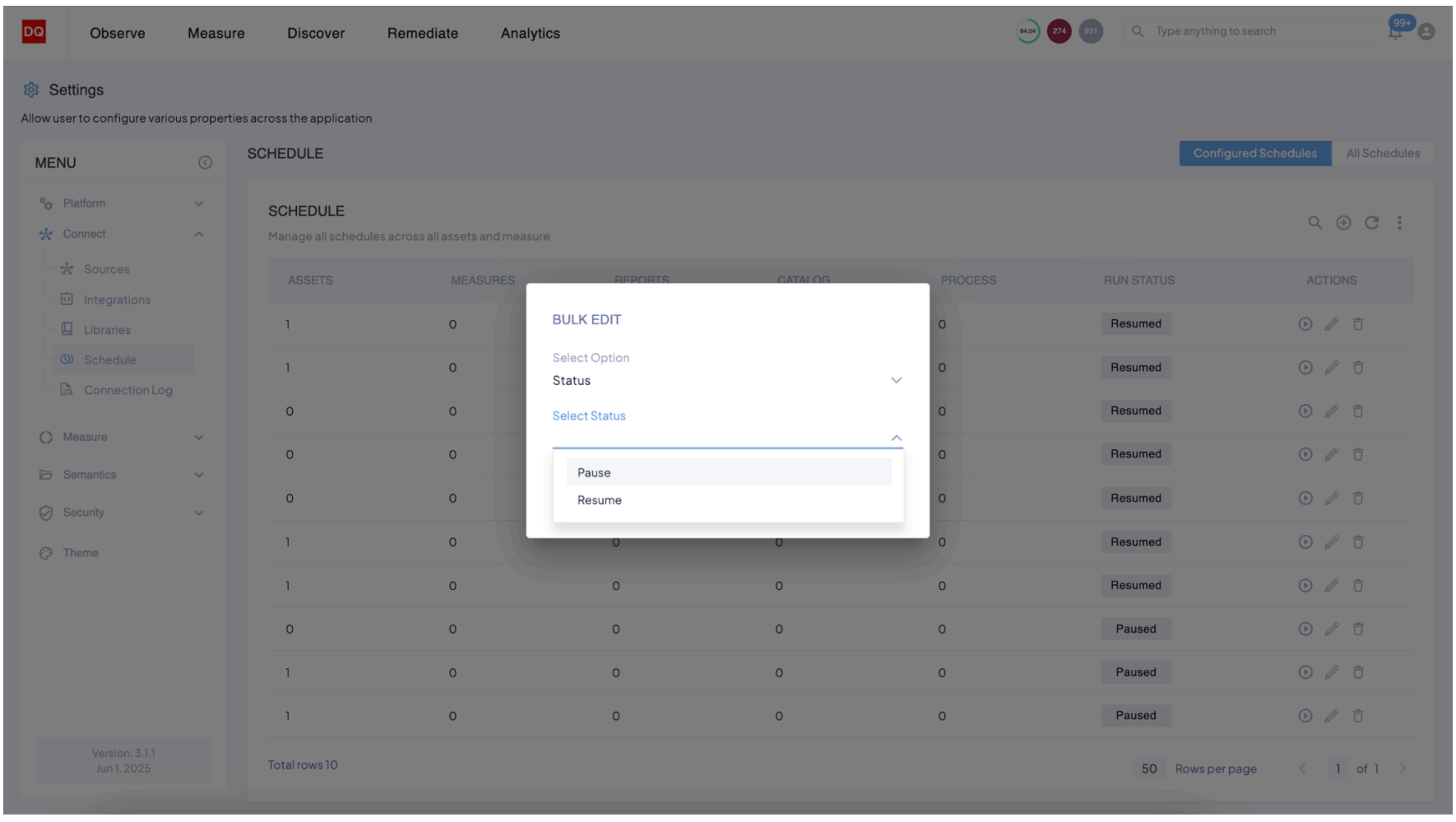


Resume a Schedule

- Select the paused schedule and choose the **Resume** action.
- Upon resuming:
  - The system will automatically update the **next run time** to the **next valid scheduled occurrence**, skipping all past missed executions.
  - No backfill jobs will be executed for the missed interval.
  - The schedule status will return to **"Resumed"** in the UI.

Bulk Update Support

- The Schedule table allows selection of multiple schedules.
- Use **bulk actions** to pause or resume multiple schedules at once.
- This is useful for managing jobs across teams or environments during global maintenance windows.



Audit Logging

- All **Pause** and **Resume** actions are logged with the following metadata:
  - User ID
  - Timestamp
  - Schedule Name/ID
  - Action Type (Pause/Resume)
- Logs are accessible via the **Audit Logs** or **Activity History** section for compliance and review purposes.

---

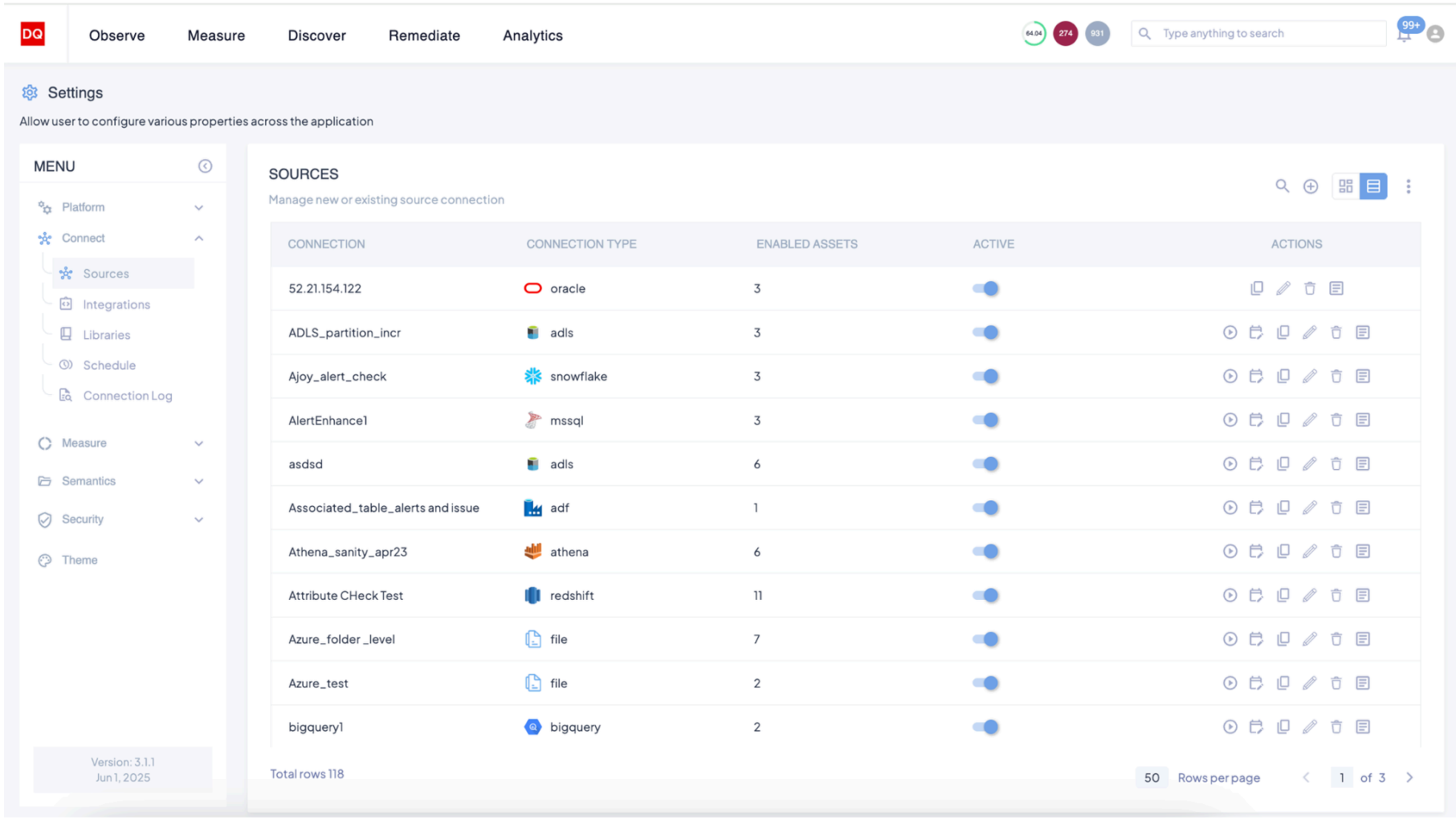
4. Health Tab for Observability(Databricks)

DQLabs has introduced new observability features that allow users to configure observability for newly added assets. Enhanced measures have been implemented to provide greater visibility into asset performance and quality. The following changes have been introduced as part of this update:

- The connection page will now allow users to include/exclude the database and schemas in the connection to be enabled for health measures
- The users can schedule the observe job to run across all selected databases and schemas.
- The user can also select individual assets using the “Only use selected Assets toggle
- The following new measures have been implemented for the health tab, categorized as observed under the All Measures page.
  - Freshness
  - Volume
  - Schema
- The users will be able to view a new tab under the asset detail page called “Health“ with the following charts
  - Alerts
  - Issues
  - Freshness
  - Volume
  - Schema
- The existing volume, freshness, and schema measure is renamed to Row Count, Last updated, Column Count
- The measure summary chart, measure list at the “All“ selection of the attribute, has now been removed.

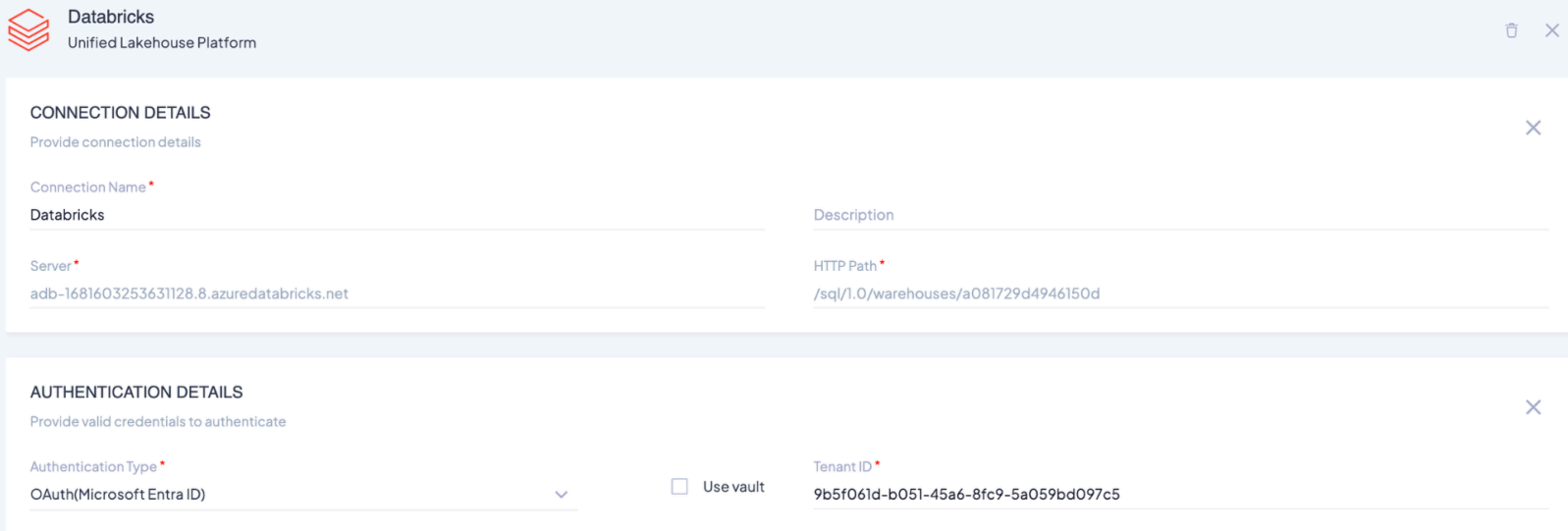
Follow the steps below to set up Observe functionality for a Databricks asset:

Step 1: Navigate to settings → Connect → Source and click on the “+“ icon.



Step 2: Click on the Databricks connector and provide the following details

- Connection Details
  - Connection Name
  - Description
  - Server
  - HTTP Path
- Authentication Details
  - Authentication Type
    - Token
    - OAuth(m2m)
    - OAuth(Microsoft EntraID)
  - Provide details based on the authentication type to observe
- Include - Allows selection of multiple databases & schemas
  - Allows wildcard-based selection for databases. For example: PRD_* (will select all the databases starting with “PRD_”
- Exclude - Allows selection of multiple databases & schemas to exclude from Observe
  - Allows wildcard-based selection for databases. For example: PRD_* (will select all the databases starting with “PRD_”
- Configuration
  - Observe - Enable health measure
  - Schedule - A schedule for Observability will be selected by default to run for every hour
  - Only use selected Assets - For individual selection of assets
- Pipeline configuration
  - Runs
  - No.of days
  - Status
  - Tasks
- Event propagation
  - Score, alert, and issue propagation settings



Step 3: Once the connection is established, the user will be able to view all the assets in the catalog page. Only one job will be run for the observability across all configured assets in the connection

Step 4: Navigate to the asset page to view the configured asset details. The admin/privileged user will now be able to see the “Health Tab“ with the mentioned graphs

Limitations/Constraints:

- Currently implemented only for Snowflake and Databricks connectors
- The users cannot select tables using the wildcard patterns

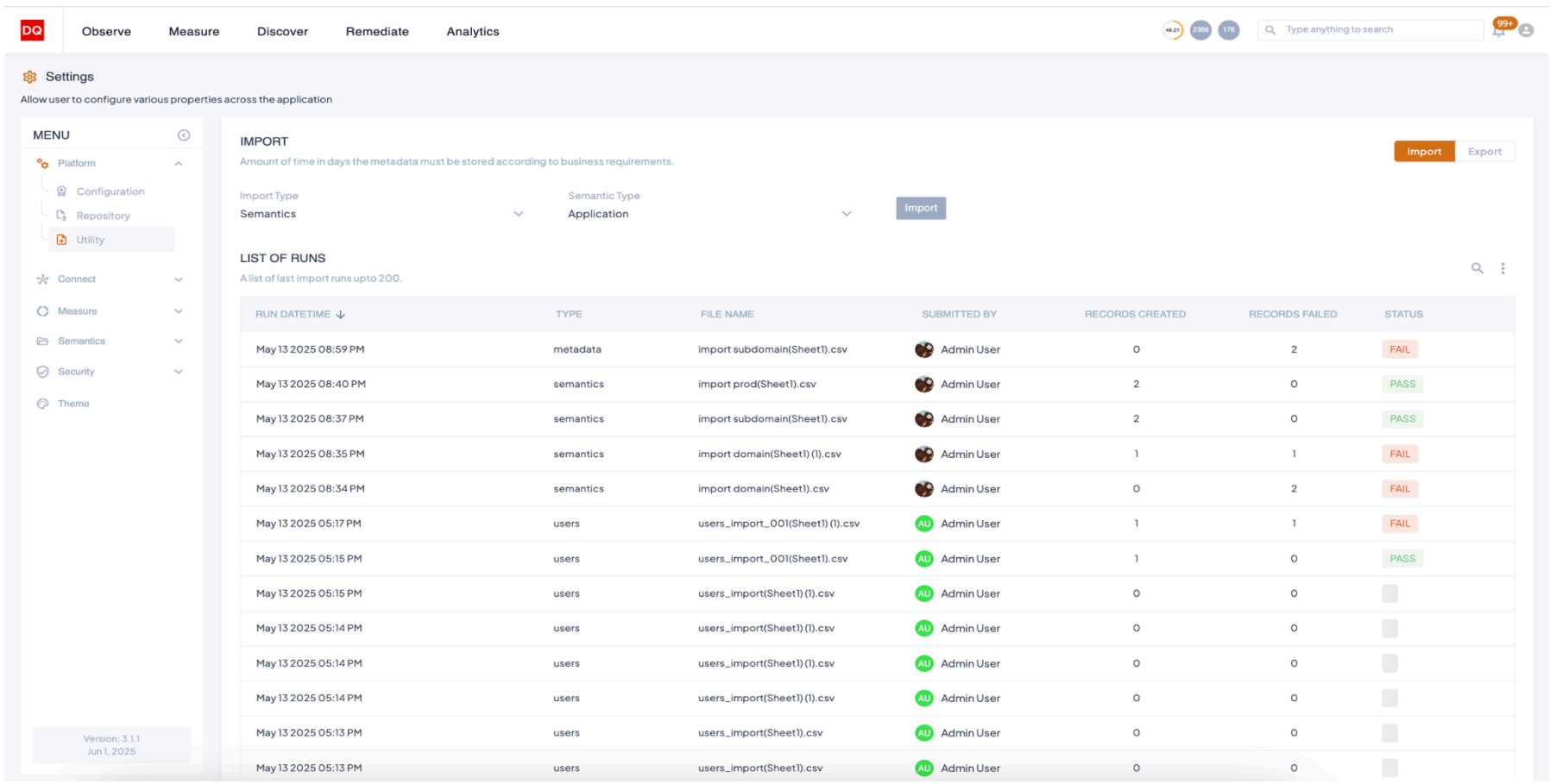
---

5. Bulk upload business Glossary, Domain, Tag, Products

This feature allows Admin or Privileged Users to import and export semantic layer definitions directly from the Settings > Utility section. It supports bulk data operations for metadata such as domains, tags, terms, applications, and product references. Users can select the desired components and export them in CSV/Excel file formats or import definitions from structured files.

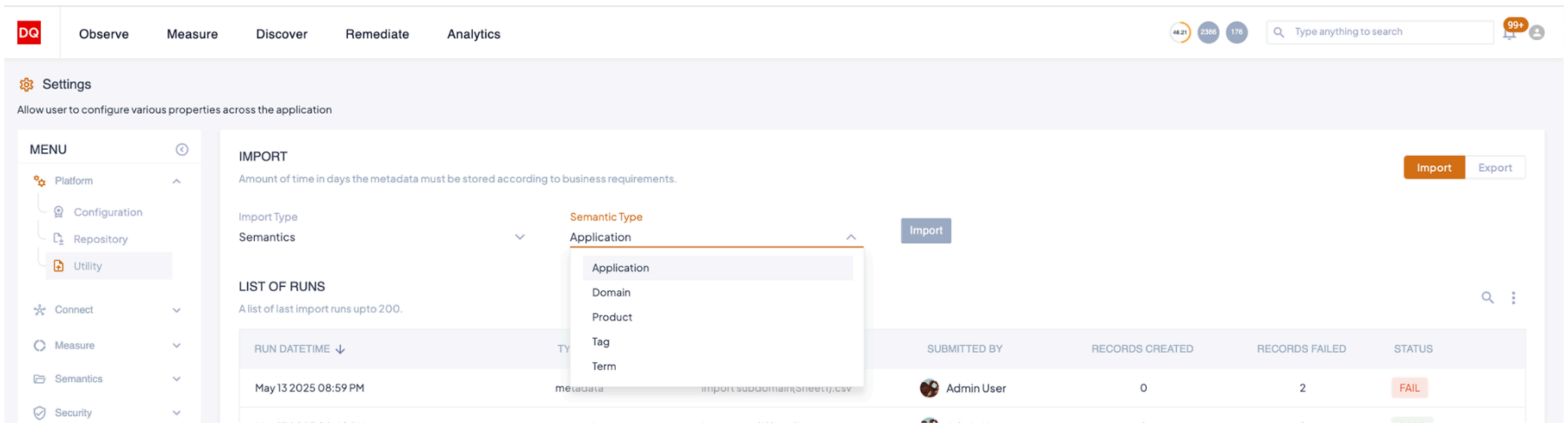
Navigating to the Utility

- Log in with Admin or Privileged credentials.
- Go to the top navigation menu and click on **Settings**.
- Under **Utility**, select the **Semantics** option from the dropdown.



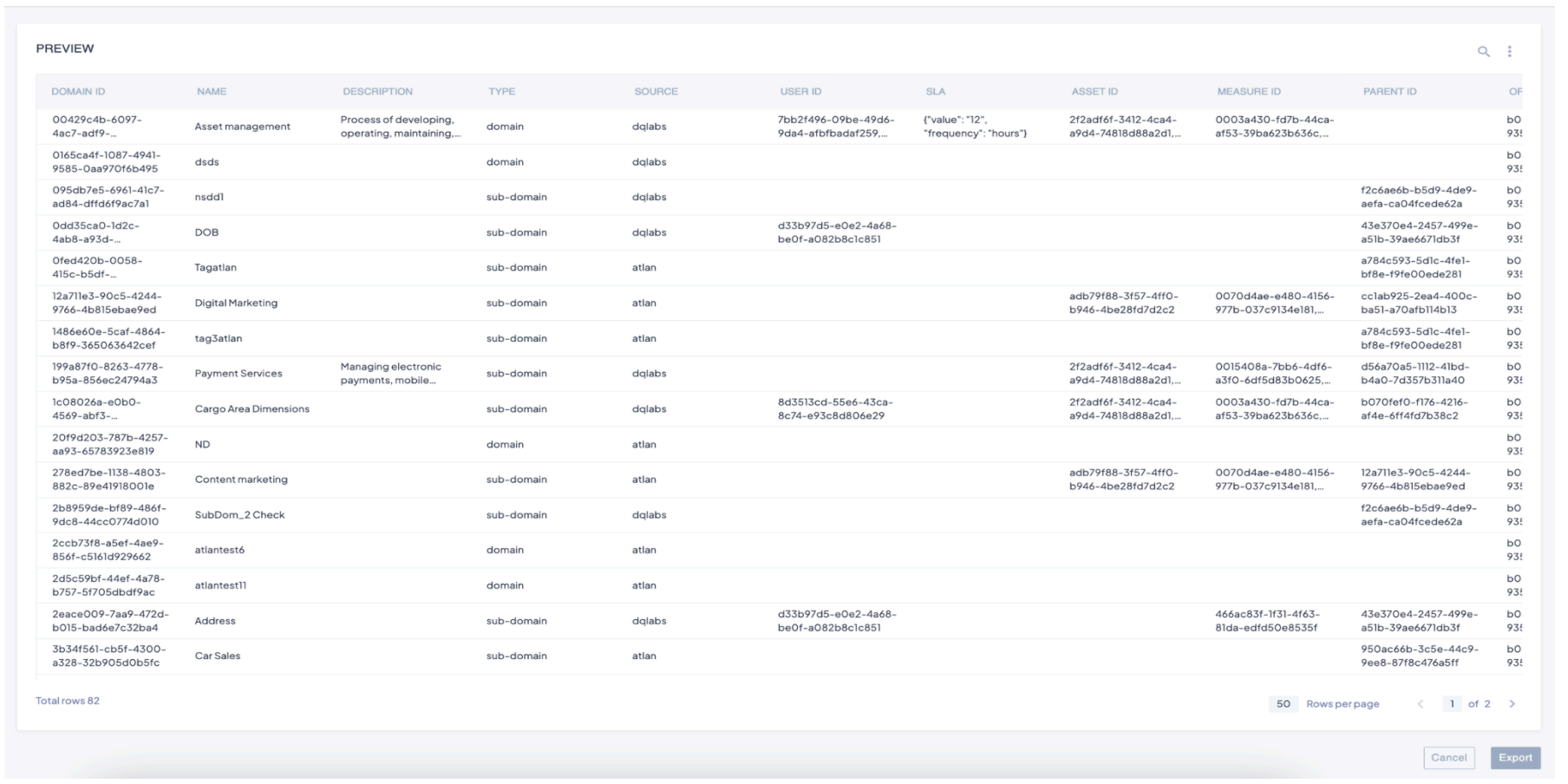
Selecting Semantic Layer Definitions

- Once on the Semantics Utility page, a **multi-select dropdown** will appear.
- You can select one or more of the following semantic components:
  - **Domains**
  - **Tags**
  - **Terms**
  - **Applications**
  - **Product**



Exporting Semantic Definitions

- After selecting the desired semantic components, click on the **Export** button.
- You will be prompted to choose the export file format:
  - **CSV**
  - **Excel (.xlsx)**
- The system will generate a downloadable file containing the selected definitions.



Importing Semantic Definitions

- To import, click on the **Import** button.
- Choose the appropriate file (CSV or Excel) containing valid semantic definitions.
- Ensure the file matches the expected schema/template format.
- Upon successful upload, the system will validate and update the semantic layer accordingly.

Recommendations

- Ensure exported files are saved securely if they contain sensitive metadata.
- Always validate data format before import to prevent schema errors.
- Use this functionality to back up, migrate, or synchronize semantic definitions across environments.

6. Teams Integration Using Service Account for Direct Messaging

Microsoft Teams (MS Teams) is a collaboration platform that combines chat, video meetings, file storage, and app integrations, enabling seamless communication across organizations. It's widely used by data, engineering, and business teams to centralize conversations and streamline workflows.

When using DQLabs, data quality incidents, such as audits, alerts, or issues, require prompt attention. Without direct integration, these alerts typically arrive via email or platform dashboards, which can be slow or overlooked.

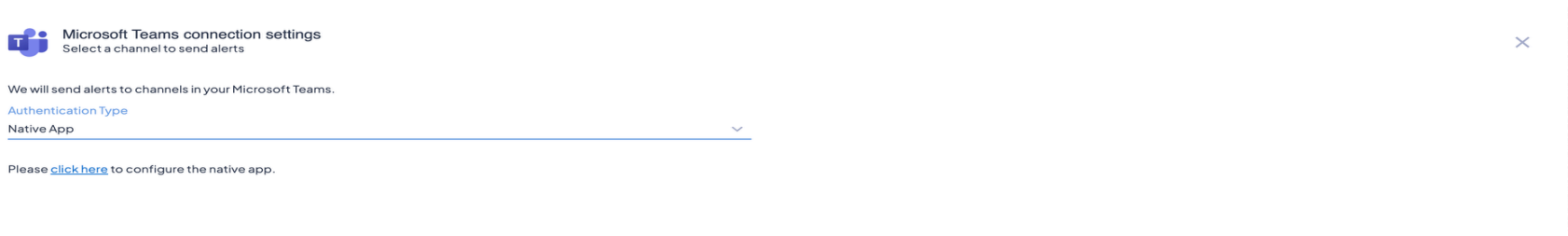
An MS Teams app bridges this gap by allowing the observability platform to send real-time notifications directly into Teams channels where relevant teams are already collaborating. This ensures:

- Faster incident response: Engineers and analysts see issues as they arise.
- Collaborative triage: Teams can discuss, assign, and resolve incidents within their existing chat workflow.
- Auditability: Notifications and discussions remain logged in a shared space for accountability.

Follow the steps below to integrate MS Teams:

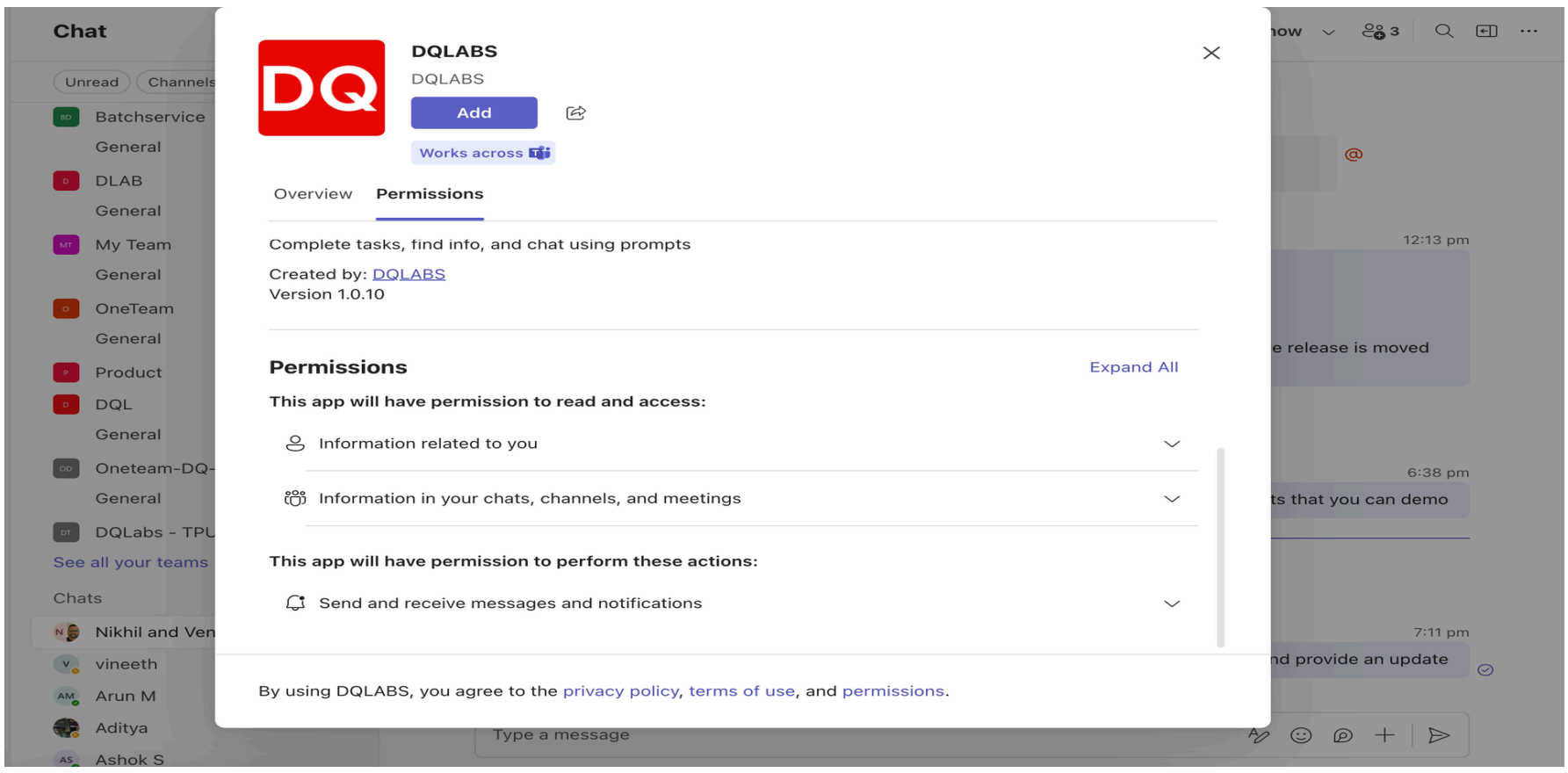
Enable the DQLabs–Teams Integration

1. Navigate to settings → Integrations
2. Locate Microsoft Teams in the list of integrations
3. Click Enable to begin linking DQLabs to Teams.
4. Choose Authentication Type as “Native App“ and click on the link to redirect to Teams to set up the integration

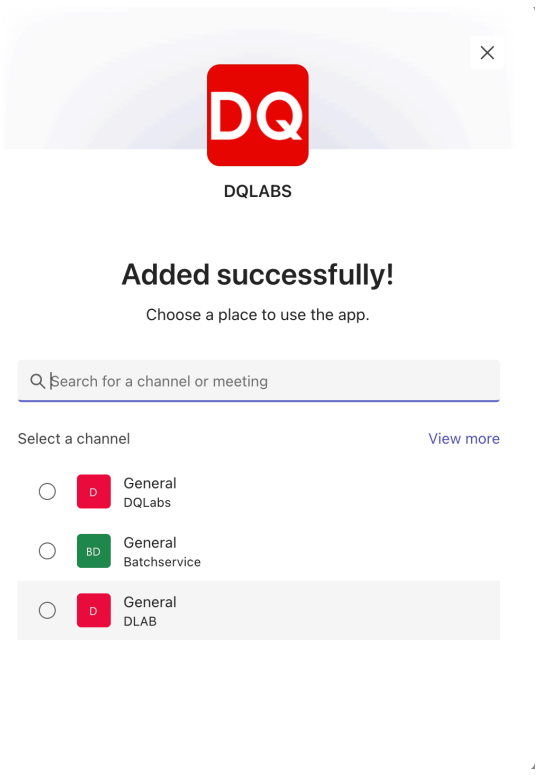


Install the DQLabs App in Teams

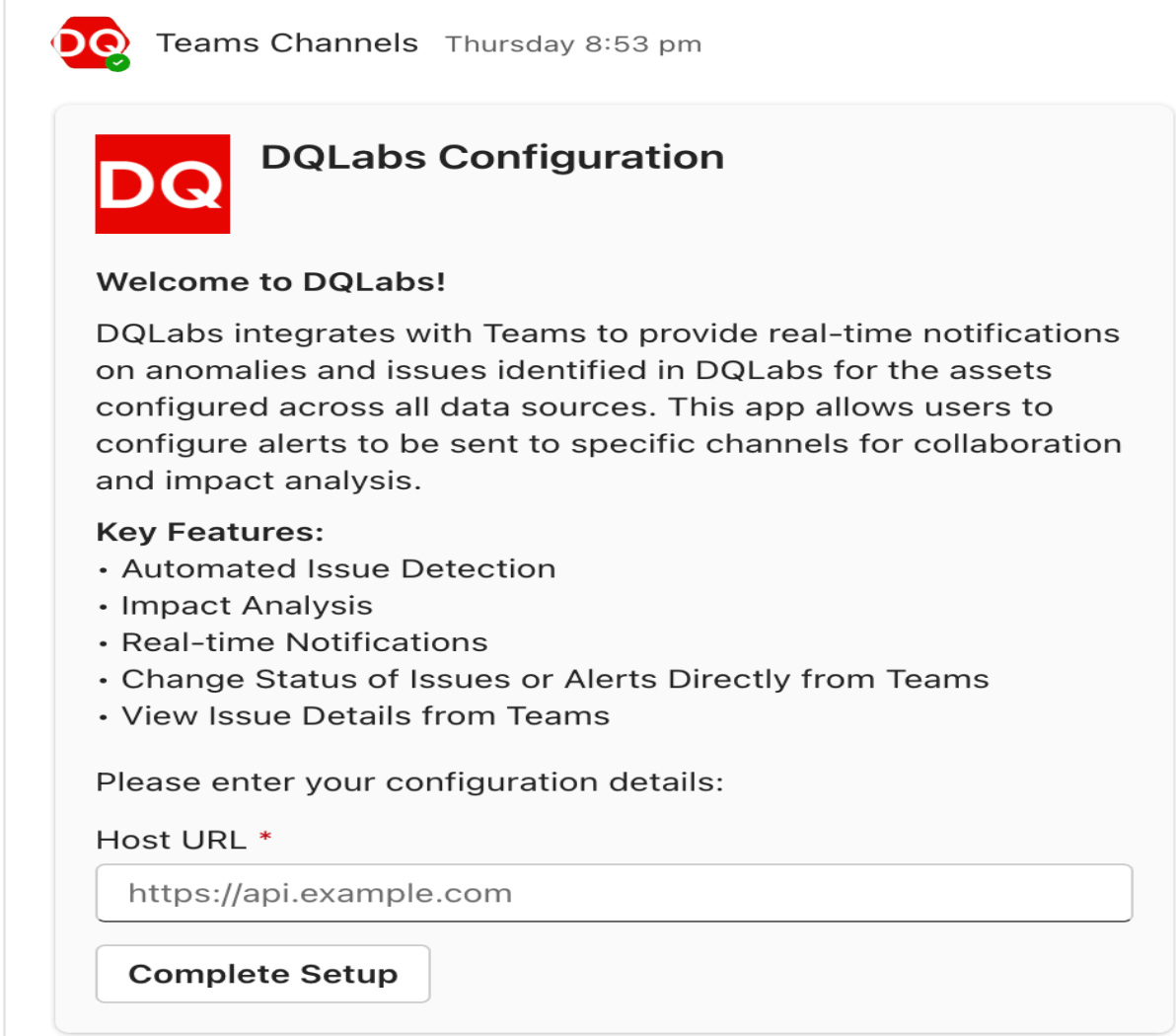
1. You’ll be redirected to the Microsoft Teams App Store.



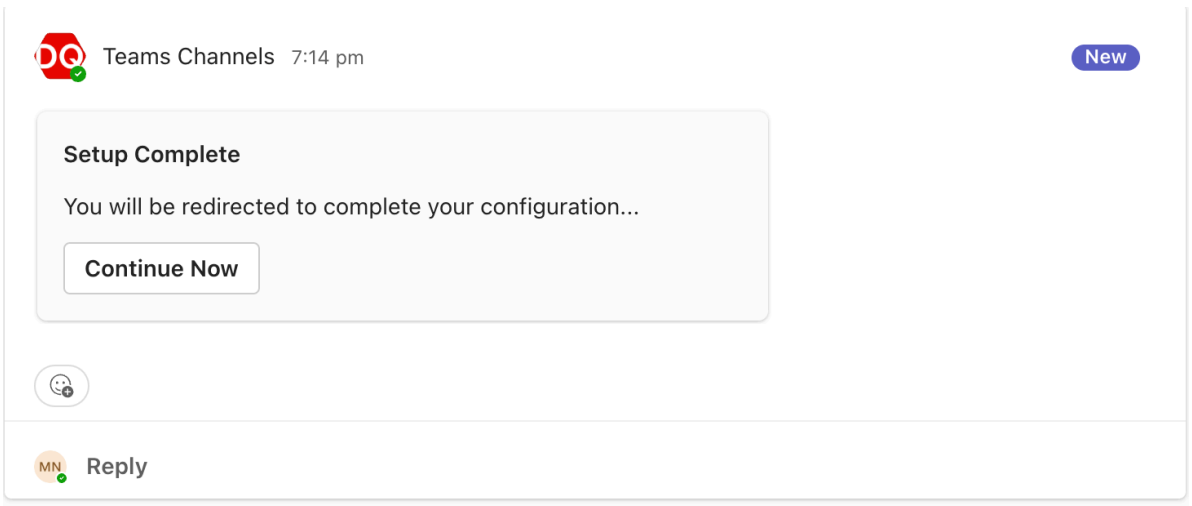
2. Click Add, then select the appropriate team channel.



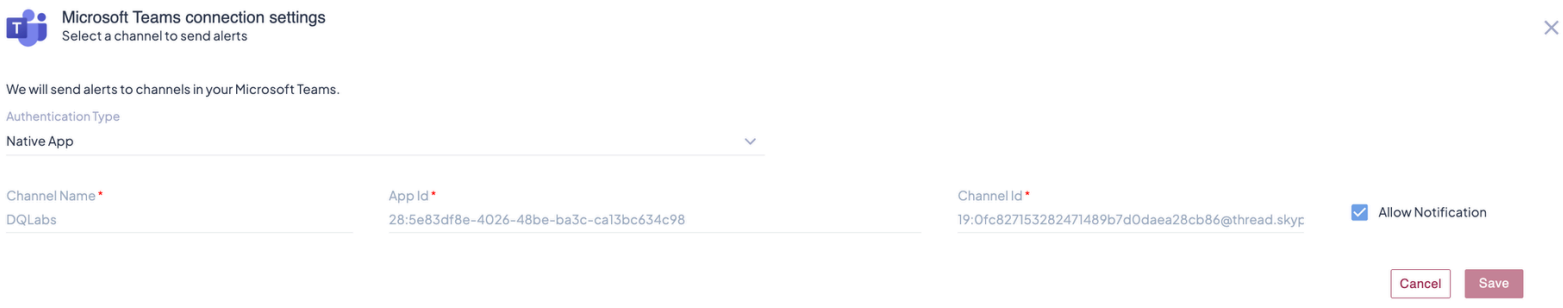
3. Upon successful installation, a welcome message or confirmation post appears in the channel, where the user can provide the URL to the DQLabs platform



4. Once provided, the team will be prompted to continue to complete the setup in DQLabs, click on “Continue Now. “



5. Click on save to complete the integration



Once integrated, the user will now receive notifications from DQLabs in Teams. To set up multiple channels, go to the DQLabs app in Teams and select the required channels.

To create an app from the local file, navigate to Apps → manage your Apps and click on Upload an App, and select the following file: DQlabs_Teams_Nativeapp.zip

---

7. Integrate SCIM for User Provisioning and Role-Based Access

DQLabs now supports seamless Single Sign-On (SSO) integration combined with automated user provisioning and role-based access control through SCIM (System for Cross-domain Identity Management). SCIM allows organizations to automatically manage user accounts and roles in DQLabs, synchronized with their corporate identity systems, reducing manual effort and improving security.

DQLabs can be integrated with SCIM-enabled SAML Identity Providers (IdPs) such as Okta and Azure Active Directory. When SCIM is configured, user accounts are automatically created, updated, or deactivated in DQLabs based on their status in the IdP. In addition, SCIM supports the dynamic assignment of roles, enabling precise role-based access control in alignment with organizational policies.

For example, when a new employee is onboarded and added to a group in Okta or Azure AD, they can be automatically provisioned in DQLabs with the appropriate role (e.g., Data Steward, Data Analyst). Similarly, when an employee leaves or changes roles, their DQLabs access is automatically updated or revoked.

This integration improves operational efficiency, strengthens compliance, and enhances security posture by eliminating the risk of orphaned accounts and ensuring real-time synchronization of access rights.

DQLabs SCIM support is currently certified with Okta and Azure AD, with additional providers planned in future releases.

OKTA SCIM Setup

Prerequisites

- An Okta admin account.
- A target application that supports SCIM (and provides a SCIM endpoint + bearer token or basic auth).
- SCIM Base URL and API Token from the target application.

Steps to Configure SCIM in Okta

Step 1: Add the Application to Okta

- Log in to Okta Admin Console.
- Go to Applications → Applications.
- Click Browse App Catalog.
- Search for your target app (if it's pre-integrated) OR
- Click Create App Integration → OIDC / SAML (for SSO) → Save.

Step 2: Enable SCIM Provisioning

- Inside the app settings → go to the Provisioning tab.
- Click Configure API Integration.
- Check Enable API Integration.
- Enter SCIM Base URL (provided by the app).
- Enter API Token (provided by the app).
- Click Test API Credentials → should show success.
- Click Save.

Step 3: Configure Provisioning Features

- Go to the App tab under Provisioning.
- Enable:
    - Create Users.
    - Update User Attributes.
    - Deactivate Users.
- Map attributes as needed (Okta → App attribute mappings).

Step 4: Assign Users

- Go to the Assignments tab.
- Assign users or groups.
- SCIM will now auto-provision users based on your settings.

AZURE SCIM Setup

Prerequisites

- Azure AD Premium P1 or P2 license.
- An Azure Global Admin or Privileged Role Admin role.
- Target app with SCIM API URL and Token.

Steps to Configure SCIM in Azure AD

Step 1: Add the Application to Azure AD

- Go to Azure Portal → Azure Active Directory.
- Go to Enterprise applications → + New application.
- Choose an app from the gallery (if available) OR create a non-gallery app.
- Name and create the application.

Step 2: Configure SCIM-based Provisioning

- In your app → go to the Provisioning blade.
- Click Get started.
- Set Provisioning Mode → Automatic.
- Enter Admin Credentials:
  - SCIM API Endpoint → Enter SCIM base URL.
  - Secret Token → API Token.
- Click Test Connection → should return success.

Step 3: Configure Mappings

- Go to the Mappings section.
- Configure:
  - User mappings → Map Azure AD attributes → SCIM attributes.
  - Group mappings (optional).
- Save mappings.

Step 4: Start Provisioning

- Go to Provisioning blade → click Start Provisioning.
- Azure will start the initial sync → monitor status under Provisioning logs.

Step 5: Assign Users

- Go to Users and groups → Assign users/groups to the app.
- Assigned users will get auto-provisioned to the target app via SCIM.

Once the above setup is complete, configure the SAML in DQLabs in order to support SSO authentication. The SCIM is currently supported only at the user level, and groups are not supported

---

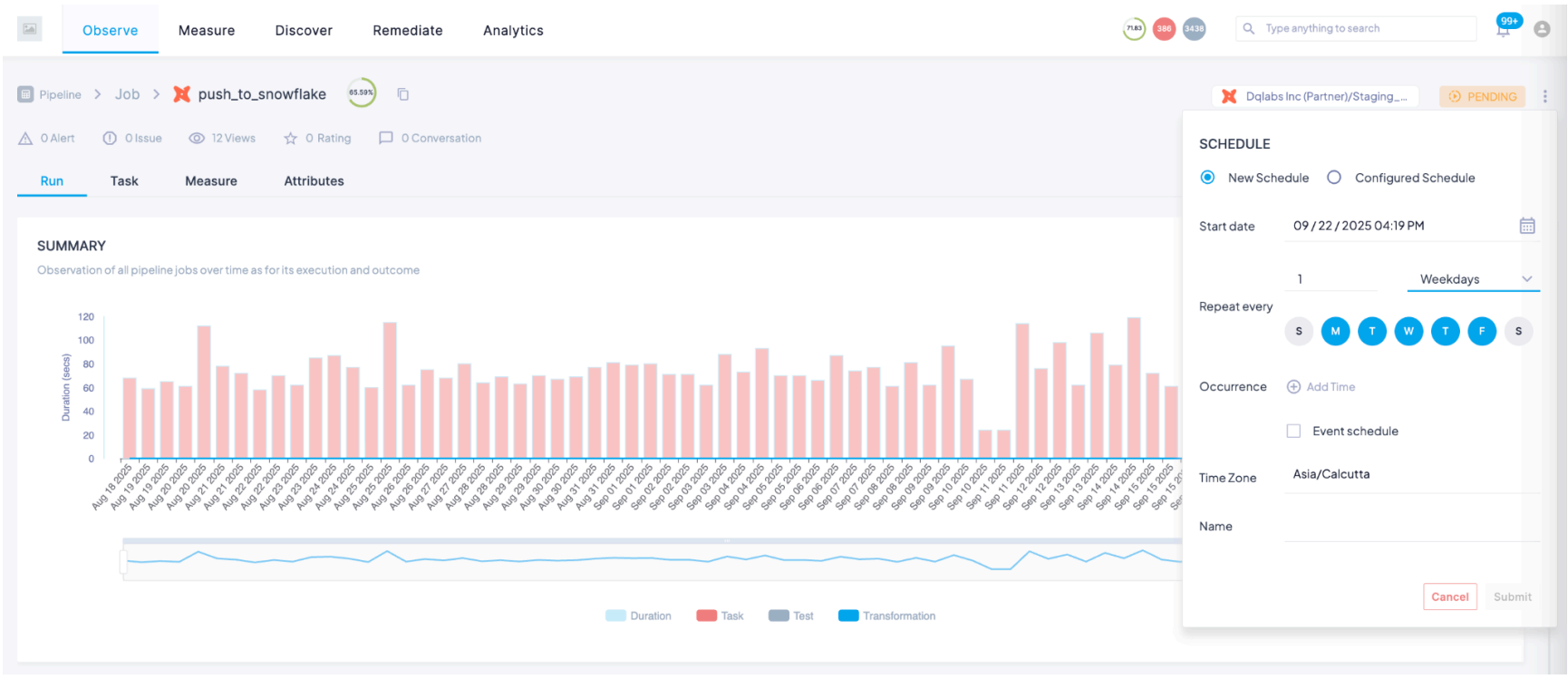
8. Ability to schedule objects in DQLabs by Weekdays

Previously, schedules were executed based on selected days of the week, requiring admin/privileged users to manually exclude Saturdays and Sundays if weekend runs were not desired. To simplify this, the system now supports weekday-only scheduling.

With this enhancement, admin/privileged users can configure schedules to run exclusively from Monday to Friday, eliminating unnecessary weekend runs.

To create a weekday schedule:

1. Navigate to the Schedules section.
2. Click Create Schedule (or edit an existing one).
3. In the Schedule Configuration form, open the Weekdays dropdown.
4. Select the desired weekdays (default: Monday–Friday).
5. Save the schedule.



Once configured, the associated object (job, task, or process) will run only on the selected weekdays. For example, if Monday – Friday is chosen, no runs will occur on Saturdays or Sundays. The weekday scheduling option is available wherever schedules can be created or updated.

---

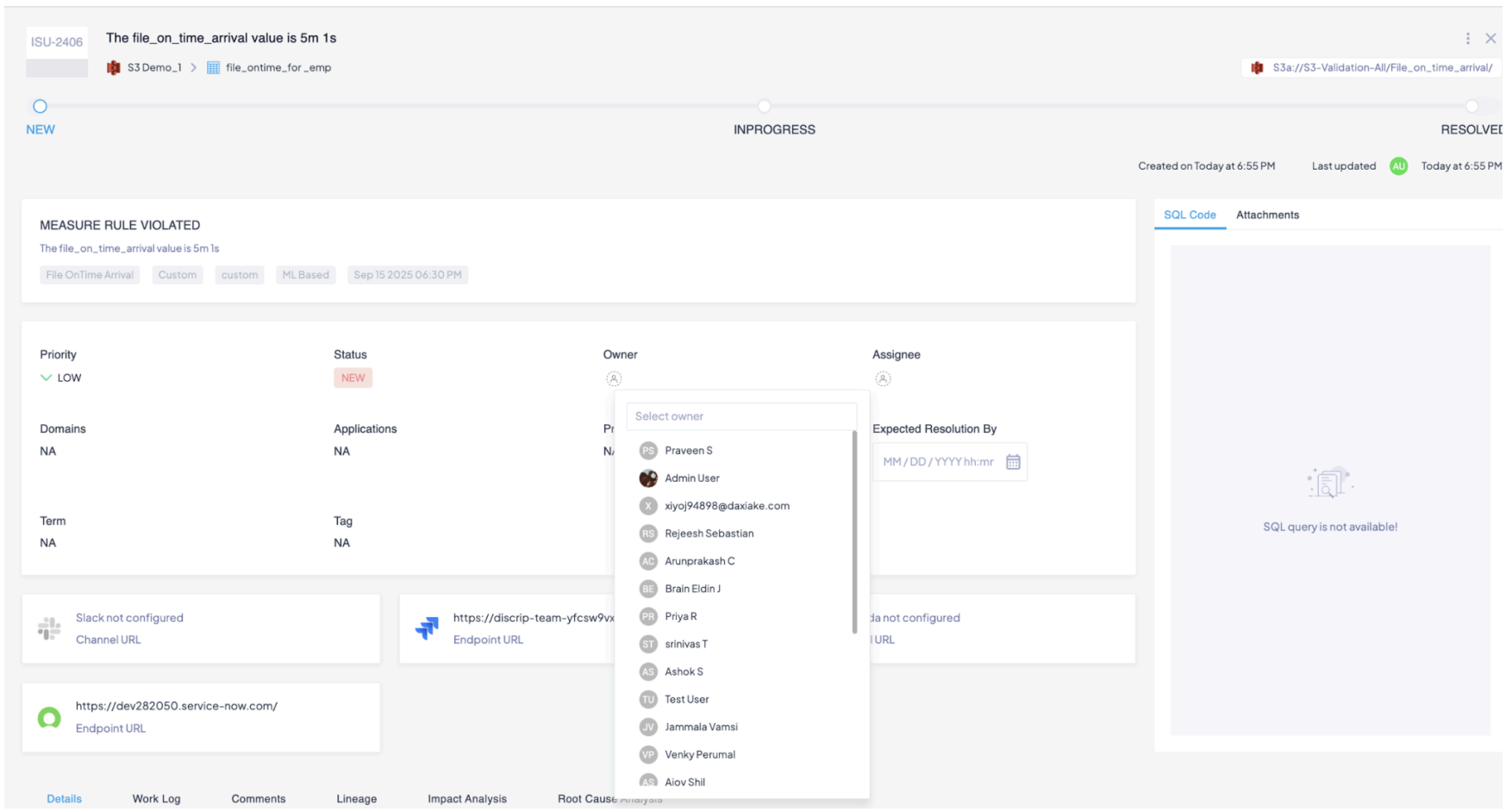
9. Flexible Issue Ownership Management

In DQLabs, issue ownership now supports flexible reassignment and removal, ensuring accountability aligns with real-world operating models such as domain-driven ownership.

Previously, an issue's owner was automatically assigned to the user who created it or the one defined in a workflow. This often led to mismatched ownership, making tracking and remediation less effective. With this enhancement, authorized users can change or remove issue owners, ensuring accountability is always assigned to the right party.

To change an issue owner:

1. Navigate to the Issues section in DQLabs.
2. Select the issue you want to update.
3. Click on the Owner field.
4. Choose Change Owner.
5. Select the new user or role from the dropdown.



All changes are recorded in the audit logs of the measure. Users with permission to edit issues will be able to update ownership as needed.

---

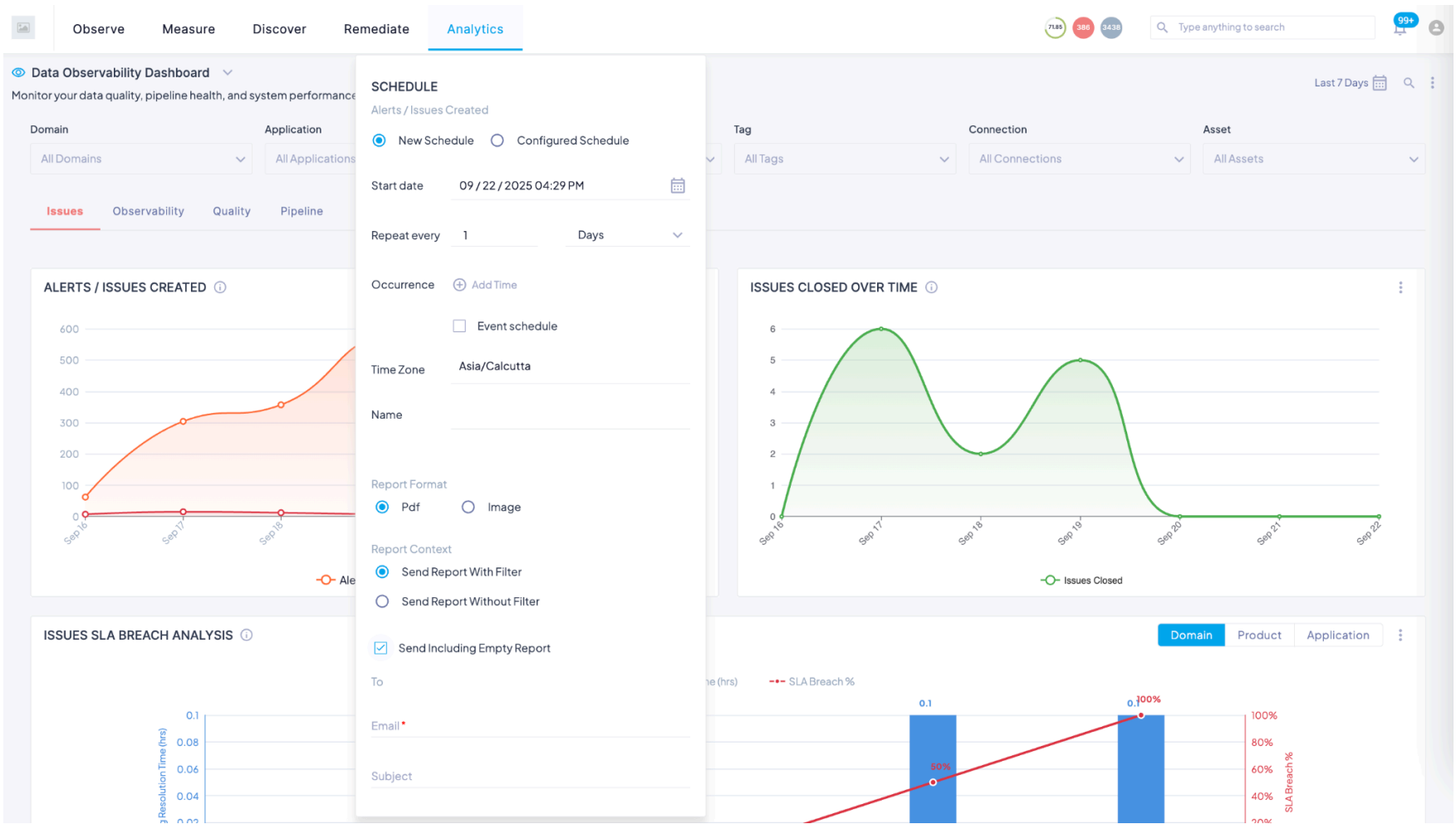
10. Ability to send blank reports when data is not present for the current run

Exception reports in DQLabs can now be tailored to customer preferences. Previously, emails were sent even when no records were available, often resulting in unnecessary blank reports.

With this enhancement, customers can now decide whether to receive emails when exception reports are empty. If an empty report is sent, the subject line will clearly indicate it.

To configure empty report handling:

1. Navigate to the Report Scheduling section in the DQLabs UI.
2. Use the Send Including Empty Report toggle:
    a. Turn it on to allow blank reports.
    b. Leave it off to skip sending blank reports.
3. Save the schedule.



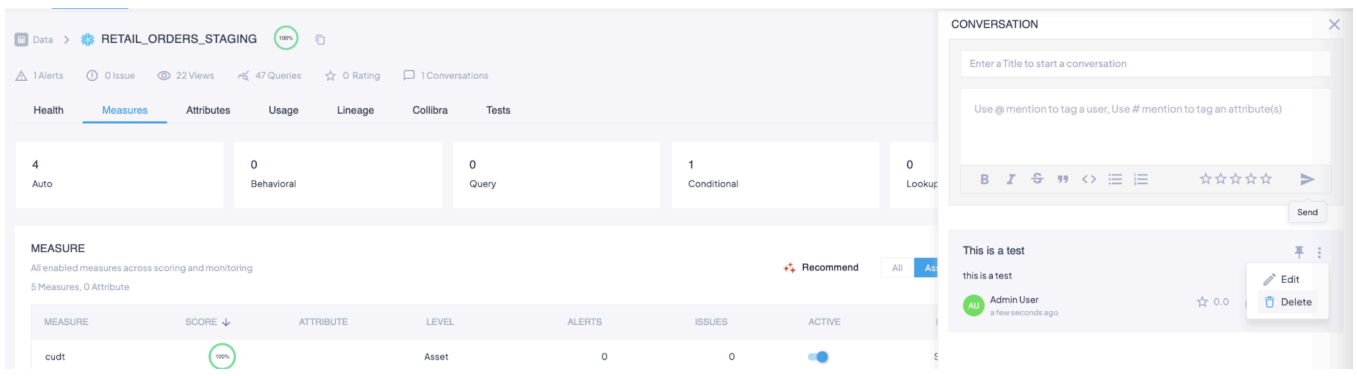
---

11. Admin Ability to Manage All Conversations in DQLabs

Previously, conversation management within assets in DQLabs was limited to the creator. Regardless of role—Admin, Owner, or User—each user could only edit or delete conversations they had personally created.

With this enhancement, Admin users can now manage conversations across all assets, providing full visibility and moderation capabilities. Admin users can:

- Edit or delete their own conversations.
- Delete conversations created by any user across all assets.
- To delete a conversation as an Admin:
    - Log in as an Admin.
    - Navigate to an asset and open the Conversations section.
    - Hover over or select the conversation you wish to remove.
    - Click Delete to remove the conversation, regardless of who created it.

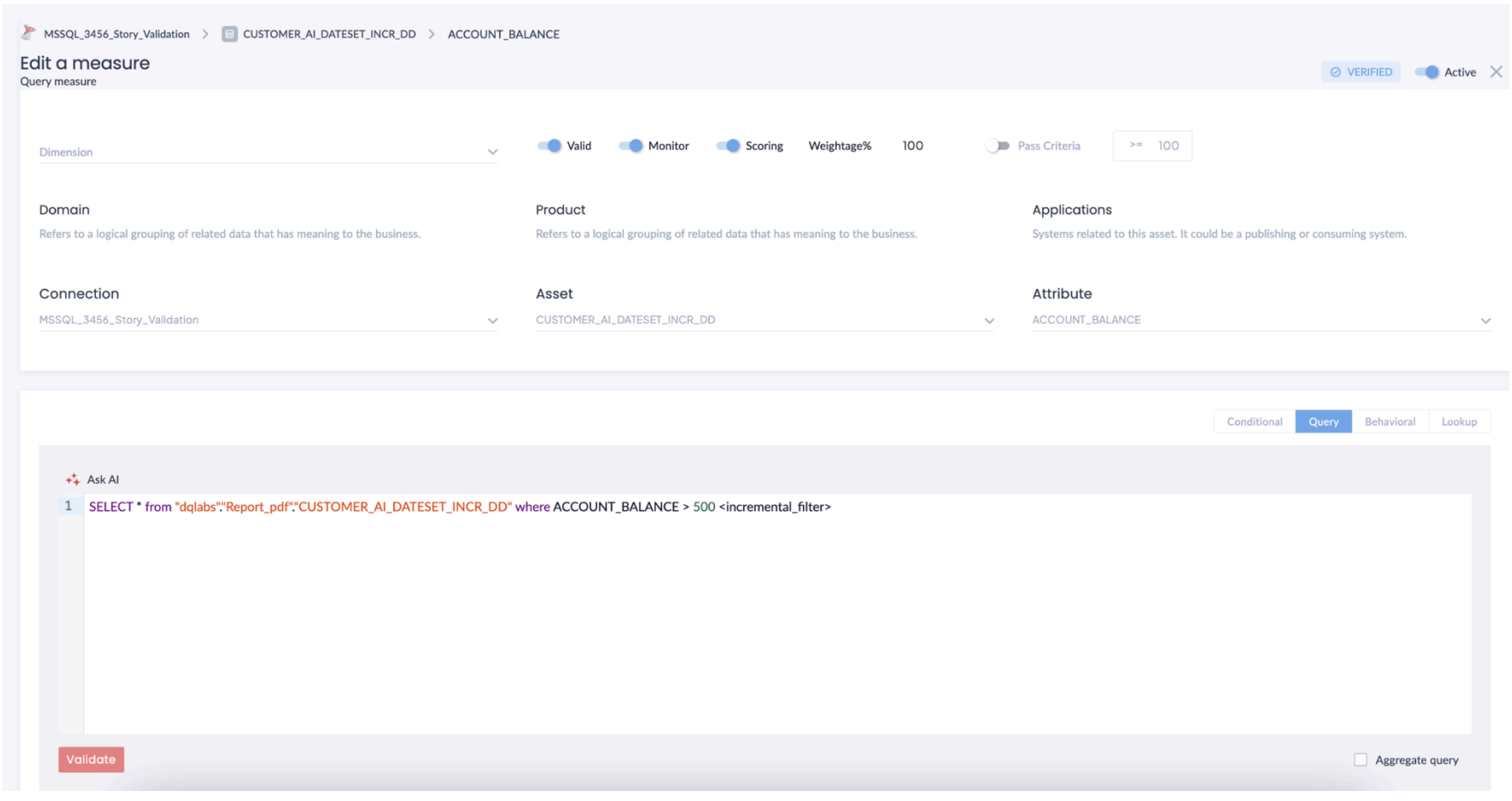


12. Parametric Base Queries for Custom Query-Based Measures

Previously, incremental logic was applied automatically only to Out-of-the-Box (OOB) and conditional measures. For custom query-based measures, users had to manually embed incremental logic (e.g., using fingerprint columns or depth).

This manual approach was both error-prone and rigid—any change to an asset's incremental depth or fingerprint configuration required updating every custom query individually.

With this enhancement, DQLabs introduces parametric base queries for custom query-based measures. Users can now leverage a reusable <incremental_filter> placeholder that automatically applies the asset's incremental logic, providing the same flexibility and consistency available in OOB measures.



Limitations

This enhancement is currently supported only for the following connectors:

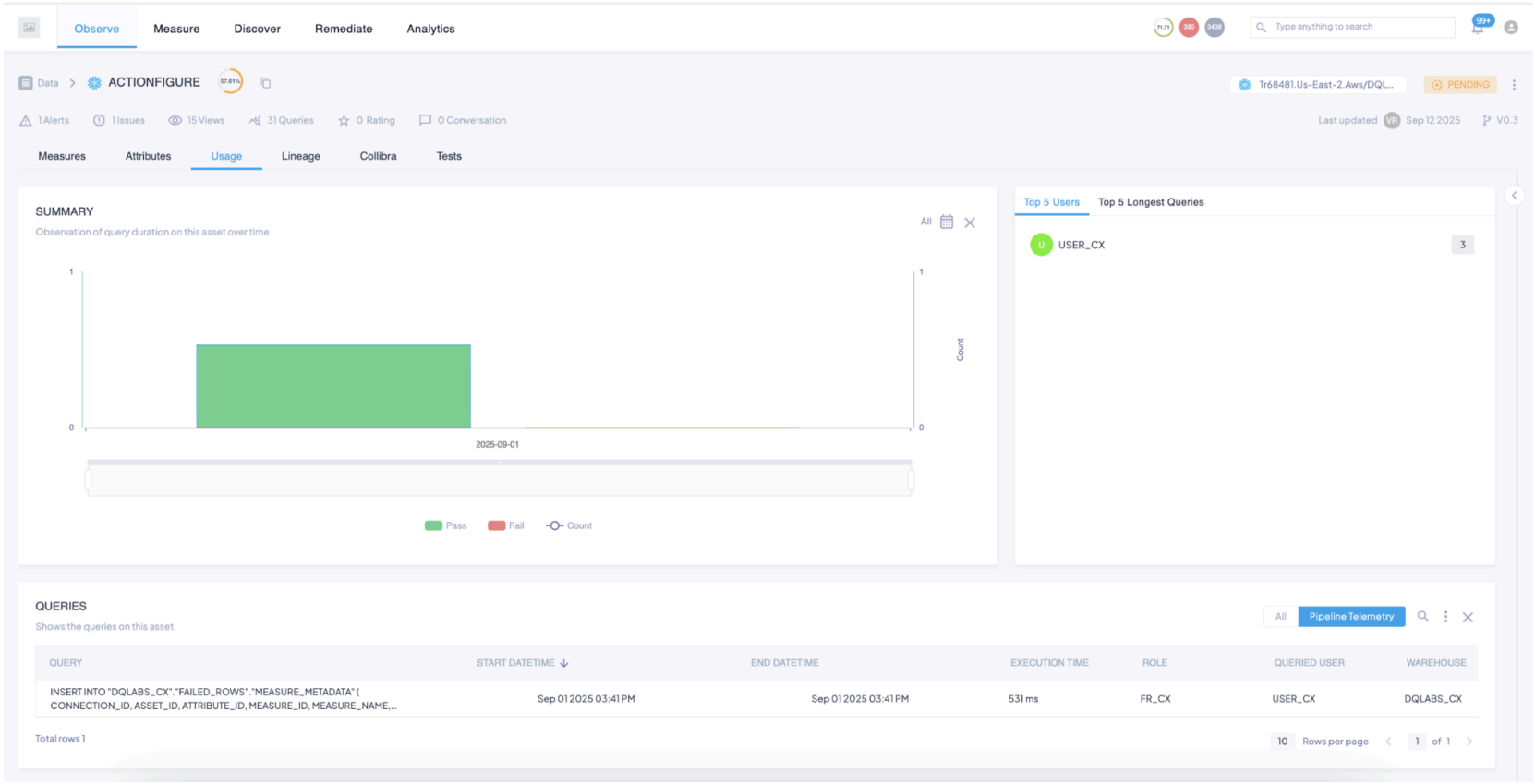
- Snowflake
- Microsoft SQL Server (MSSQL)
- Amazon Redshift
- Oracle
- SAP HANA
- Databricks

13. Pipeline Telemetry for Snowflake Queries

Administrators and privileged users can now access the Pipeline Telemetry tab under Usage → All Queries to monitor transformation-related queries executed in Snowflake. This feature enhances visibility and supports better monitoring and governance of transformation activities.

Key Highlights:

- A dedicated Pipeline Telemetry tab is now available in the All Queries section.
- Accessible only to Administrators and Privileged Users.
- Displays transformation queries executed on Snowflake assets.
- Supported transformation operations include:
  - INSERT
  - UPDATE
  - DELETE
  - MERGE
  - COPY
  - TRUNCATE
- A date filter is available to refine results and focus on queries within a specific time range.
- Navigation: Usage → All Queries → Pipeline Telemetry.



---

14. Dashboard Enhancements

We've introduced a streamlined dashboard experience to improve organization, usability, and visibility.

Deprecation of Unused Dashboards

- Dashboards that are no longer in use have been **deprecated**, reducing clutter and ensuring that only relevant dashboards are available.
- The following widgets have been **removed**:
  - Connections
  - Assets
  - Attributes
  - Measures
  - Attribute Coverage
  - Asset Coverage
  - Score
  - Organizational Score
  - Asset Score by Dimension
  - Score by Domain Hierarchy
  - Historical

Classic Dashboards Category

- All currently active dashboards have been moved into a new "Classic" category.
- This ensures easy access to existing dashboards while distinguishing them from newly created ones.

New Dashboard Experience

- A redesigned dashboard builder is now available, following modern layout and visualization best practices.
- Key improvements:
  - More flexible layouts
  - Modern visualization options
  - Better usability and navigation

Data Observability Dashboard *(for Admin/Privileged Users)*

A new Data Observability Dashboard has been introduced, providing visibility into data quality and operational health.

Categories:

1. Issues
2. Observability
3. Quality
4. Pipeline

Issues Dashboard

Provides visibility into the lifecycle of alerts and issues, helping track data quality problems from creation through resolution.

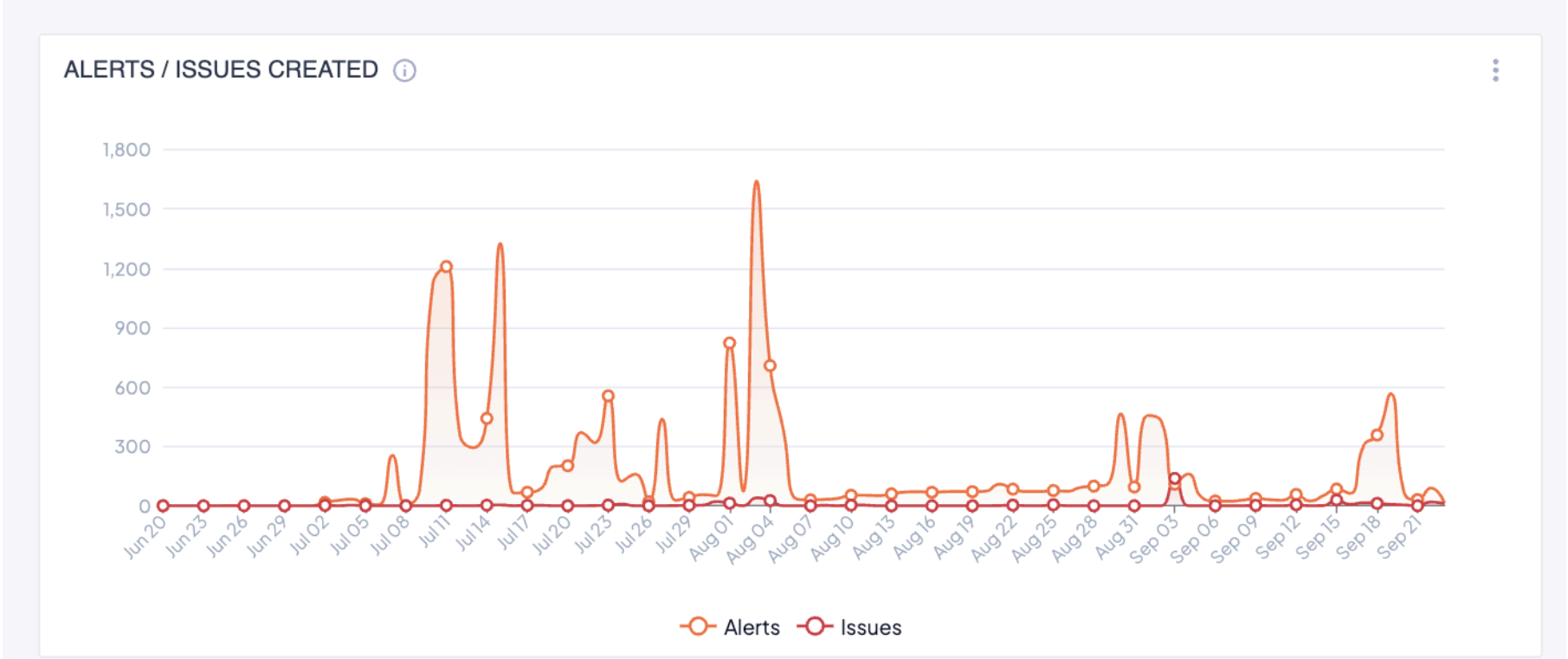
Widgets:

1. **Alerts / Issues Created**
   - **Type:** Line chart

- **X-axis:** Date
- **Y-axis:** Count of alerts and issues
- **Legend:** Alerts (orange), Issues (red outline)
- **Interaction:** Toggle legend items to filter view

Key Insights:

- Large spikes in alerts may indicate recurring data quality issues.
- Gaps between alerts and issues may highlight under-triaging of problems.



2. **Issues Closed Over Time**

- **Type:** Line chart
- **X-axis:** Date
- **Y-axis:** Count of issues closed
- **Legend:** Issues Closed (green)

Key Insights:

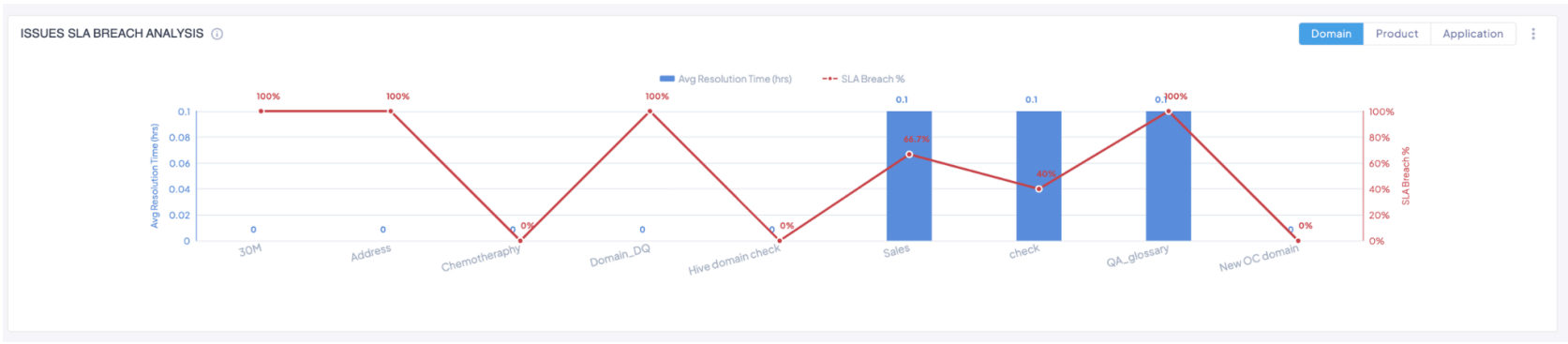
- Consistent closure of issues reflects effective remediation processes.
- Peaks in closure may indicate backlog clearance rather than steady resolution.
- Low closure rates alongside high alert creation suggest growing unresolved issues.

3. **Issues SLA Breach Analysis**

- **Type:** Combo chart (bars + line)
- **Y-axis:** Avg. Resolution Time (hrs) – blue bars; SLA Breach % – red line
- **X-axis:** Domain / Products / Application (filter-based)

Key Insights:

- High SLA breach % = recurring delays.
- Long resolution times suggest resource/process misalignment.
- Zero resolution + high breach % = tracking misalignment.

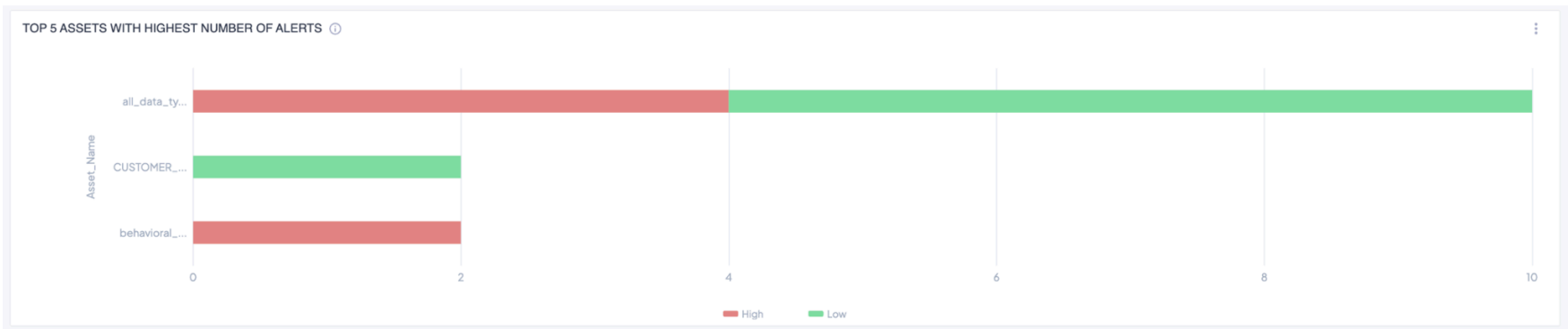


4. **Top 5 Assets with the Highest Number of Alerts**

    - **Type:** Horizontal stacked bar chart
    - **Segments:** High (red), Medium (yellow), Low (green)

Key Insights:

- Persistent top assets = systemic issues.
- High volumes (even low severity) can still impact downstream processes.

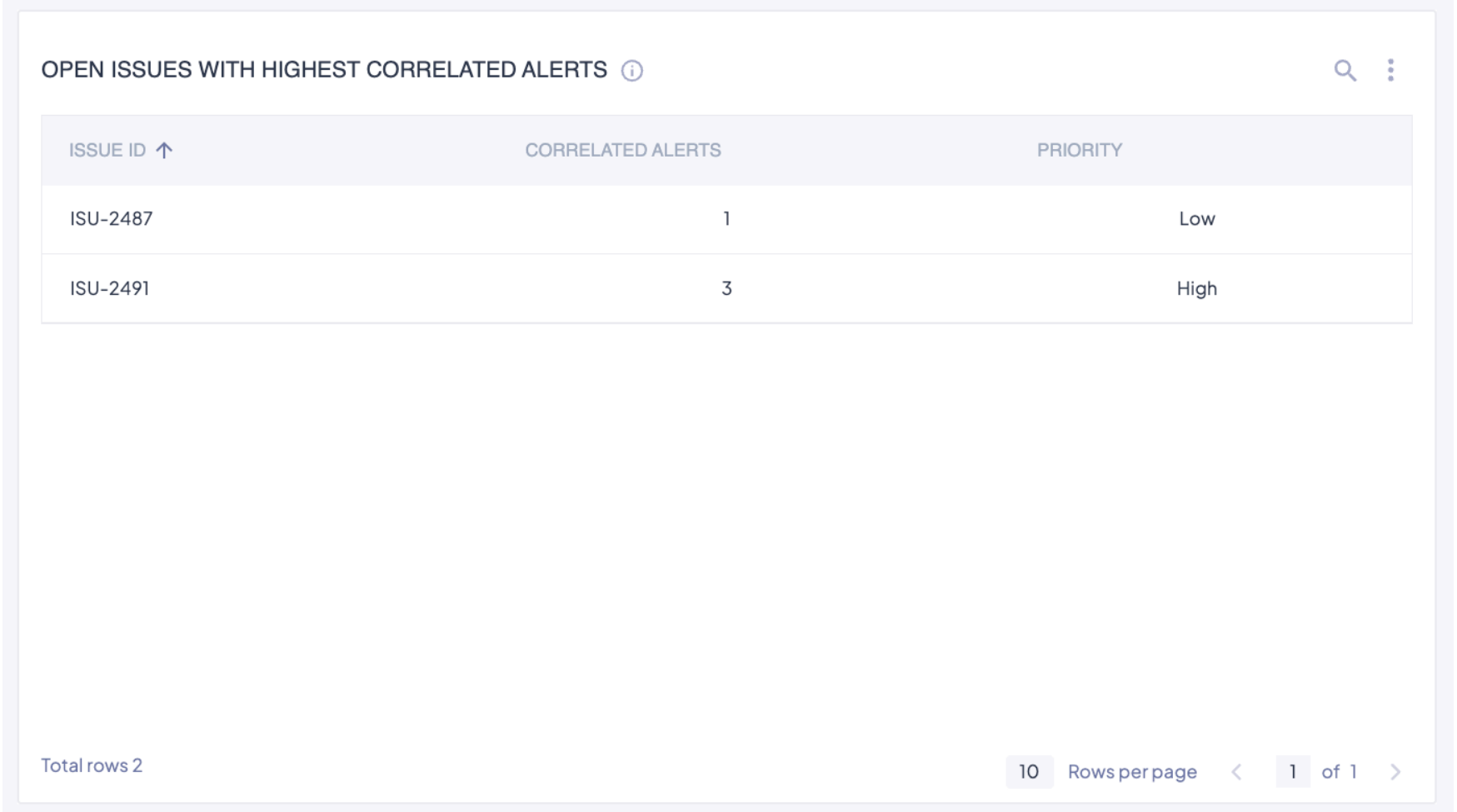


5. **Top 5 Assets with the Highest Number of Alerts**

    - **Type:** Table
    - **Columns:** Issue ID, Correlated Alerts, Priority

Key Insights:

- High alert counts may signal systemic problems.
- Even low-priority issues with many alerts shouldn't be ignored.



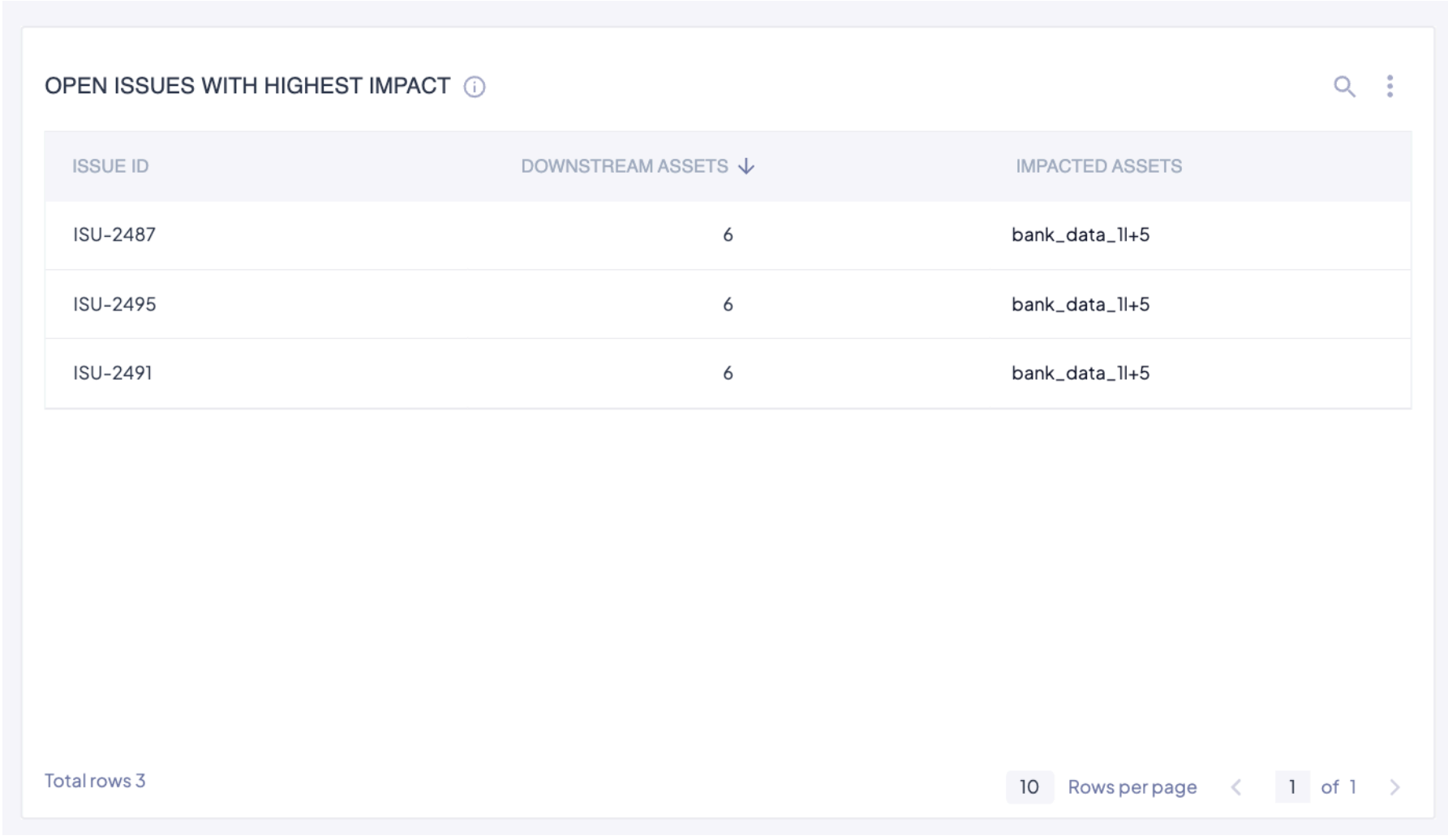
6. **Open Issues with Highest Impact**

    A table listing open issues that have the greatest potential impact based on downstream dependencies. For each issue, you can see:

    - **Issue ID:** the unique identifier of the issue.
    - **Downstream Assets:** the number of assets that depend on the data linked to this issue.
    - **Impacted Assets:** the specific asset(s) affected by the issue.

Key Insights:

- High downstream impact = higher business risk.
- Impact can outweigh alert volume in prioritization.



Observability Dashboard

Provides real-time monitoring of data quality across the environment, surfacing issues with data volume, freshness, and custom metrics.

Widgets:

- **Tables with Volume Issues**
  - Count & proportion of tables with abnormal record volumes.
  - Trend vs. previous week.
  - Table columns: Asset, Connection, Anomaly Count, Last Occurred.

- **Tables with Freshness Issues**
  - Shows the number of tables with delayed or outdated data.
  - Weekly trend included.
  - Table columns: Asset, Connection, Anomaly Count, Last Occurred.

- **Tables with Custom Measure Issues**
  - Tracks breaches of user-defined rules/thresholds.
  - Trend since last week.
  - Table columns: Asset, Connection, Anomaly Count, Last Occurred.

Volume Alerts

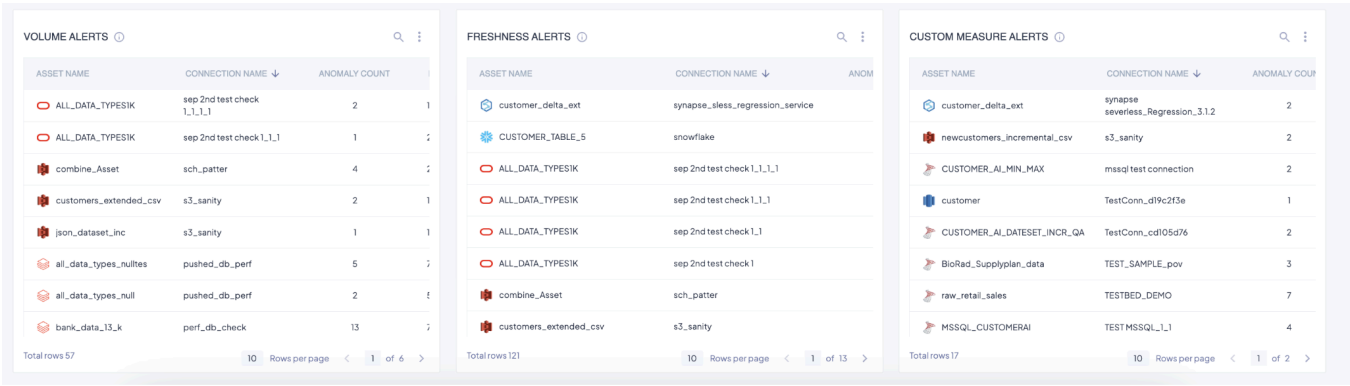
- Asset Name: The table with volume anomalies.
- Connection Name: The data pipeline or source where the table resides.
- Anomaly Count: Number of distinct volume issues detected for the asset.
- Last Occurred: Date and time for the last detected alert
- Sort, search, or page through all affected assets for granular investigation.

Freshness Alerts

- Asset Name: Table found with delayed or stale data.
- Connection Name: Data source for context and tracing.
- Anomaly Count: Number of detected freshness violations.
- Last Occurred: Date and time for the last detected alert
- Review at a glance which assets are not up-to-date, and drill down as needed.

Custom Measure Alerts

- Asset Name: Table breaching user-defined business logic or statistical thresholds.
- Connection Name: Source or system for targeted debugging.
- Anomaly Count: Total custom metric anomalies per asset.
- Last Occurred: Date and time for the last detected alert
- Search for specific tables or connections to quickly check the health of critical assets



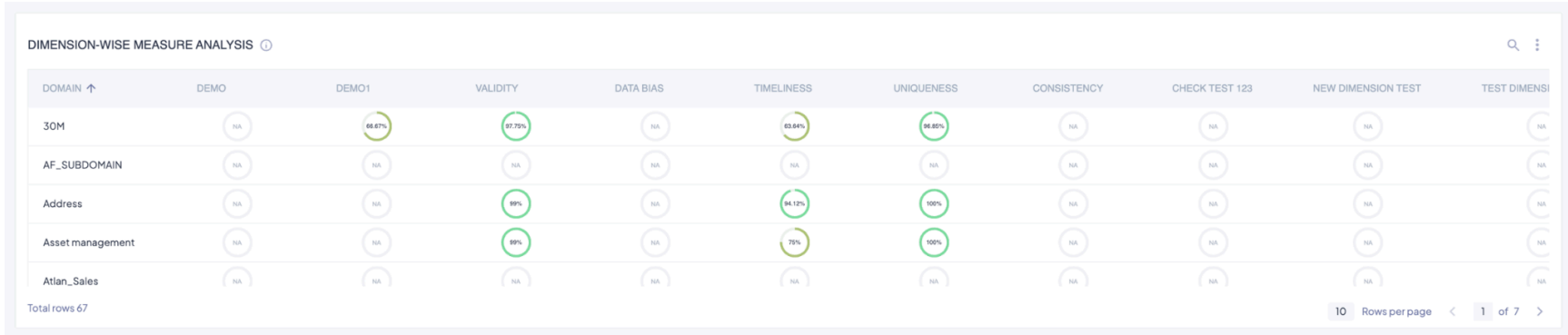
Quality Dashboard

The quality dashboard covers the monitoring and analysis of custom and out-of-the-box (OOTB) data quality measures applied across different business data domains. It provides a detailed view of how these measures perform over time, helping users understand data quality coverage, identify anomalies, and track alert generation trends.

- **Custom Measures Count:** Shows the total number of user-defined custom measures active in the system, along with the weekly trend (increase or decrease).
- **OOTB Measures Count:** Displays the number of automated, built-in quality checks running, with trend data to understand shifts in coverage or alerting.

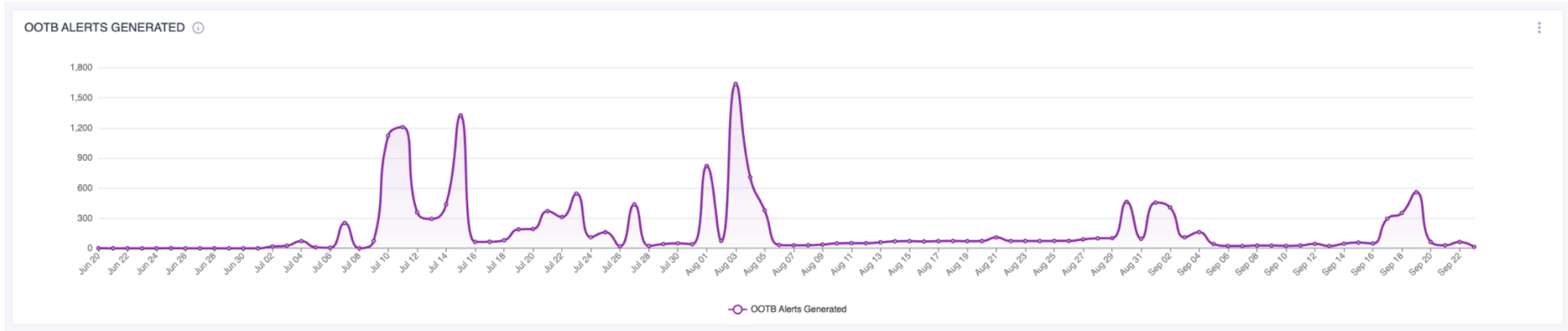
Dimension-Wise Measure Analysis:

- Breaks down quality check results across different data domains, showing percentages for validity, timeliness, uniqueness, and other quality dimensions.
- Helps users quickly identify domains with good data quality versus those needing attention.
- This detailed view supports granular analysis and targeted improvements.



OOTB Alerts Generated Chart:

- Visualizes alert volume over time from built-in checks, highlighting spikes or anomalies for operational investigation.
- Provides insight into alerting trends, assisting in resource planning, and risk management.



---

15. S3 - Connectors (Iceberg Spark Connector)

Overview

Amazon S3 (Simple Storage Service) is a cloud-based object storage service from AWS that allows storing and retrieving unlimited amounts of data in buckets. It is highly durable, scalable, and commonly used for backups, data lakes, static websites, and application storage. DQLabs enables users to connect to S3 and profile their data.

Workflow

The S3 Connector provides seamless integration between Amazon S3 and Apache Iceberg (via Spark), enabling end-to-end data validation, catalog management, and monitoring. This ensures secure, reliable, and high-quality data ingestion into Iceberg tables, similar to the ADLS connector.

Key workflow steps:

1. DQLabs uses a Spark Cluster to process incoming files from Amazon S3.
2. Files undergo multiple validation checks:
    a. Structure validation
    b. Content hash verification
    c. Duplicate detection
    d. Freshness monitoring
3. Validated records are written directly into Iceberg tables stored in S3.
4. DQLabs can manage these Iceberg tables directly from S3, with support for:
    a. Schema evolution
    b. Metadata tracking
    c. Optional AWS Glue Catalog integration
5. This ensures seamless interoperability between S3, Iceberg, and downstream analytics.

The following new measures are implemented for the S3 connector

File Structure Validation - Standalone measure

The S3 Connector validates the structure of files stored in the configured S3 bucket. It checks whether a file is empty, corrupted, or in an unsupported/invalid format. Any file that fails this validation is automatically flagged as invalid, ensuring that only usable and correctly formatted files are processed further. Scoring is supported only for the file structure validation measure

Content Hash Verification - Standalone measure

To guarantee data integrity, the connector computes and compares the content hash of each file. This ensures that the file content has not been altered or corrupted during transfer or storage. A mismatch in hash values immediately indicates integrity issues.

Duplicate File Detection - Standalone measure

The connector detects duplicate files by comparing their content hashes rather than relying only on file names. If two or more files share the same hash, they are marked as duplicates, even if their names differ. This prevents accidental reprocessing of identical files.

File On-Time Arrival (Advanced Options only) - Asset level custom measure

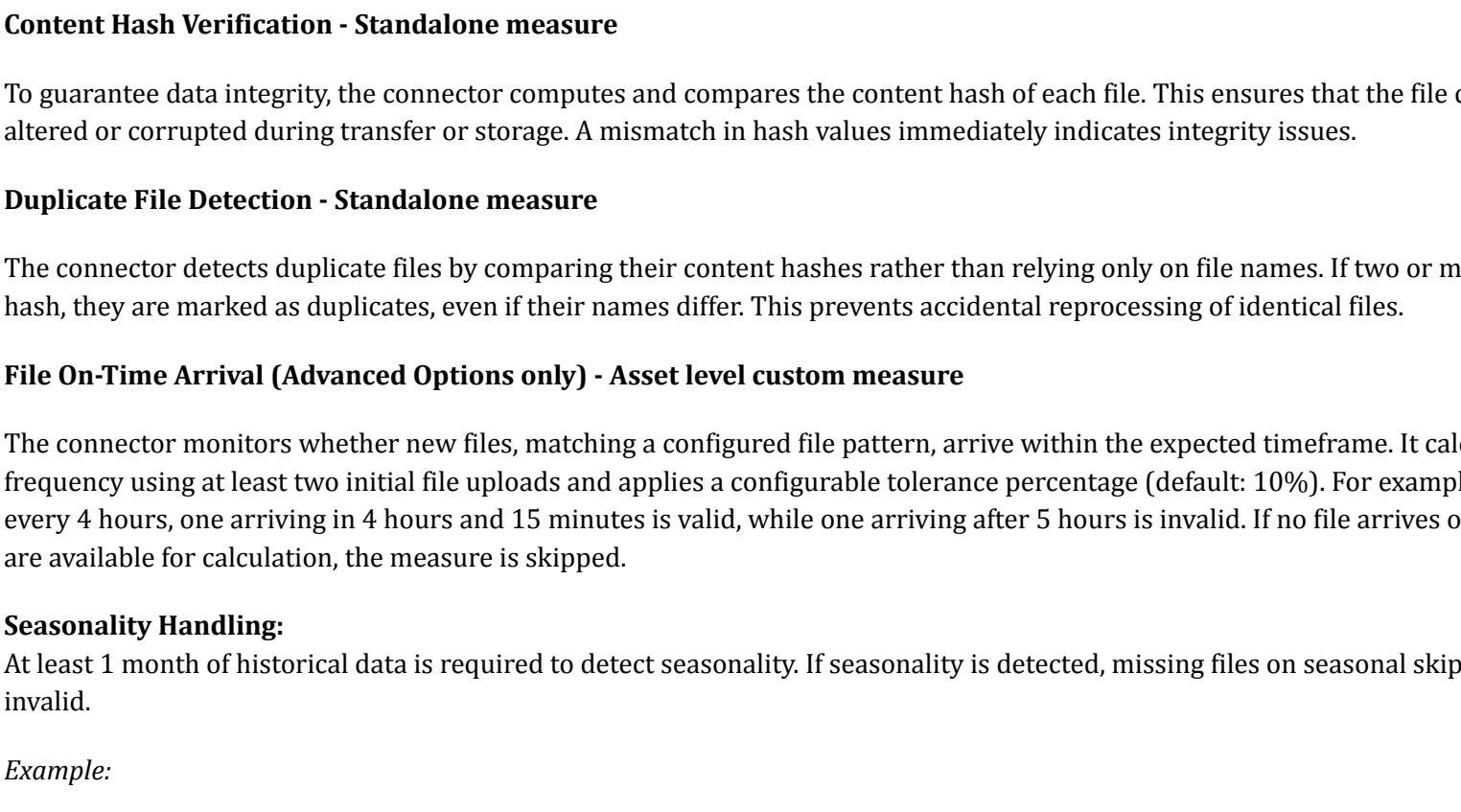
The connector monitors whether new files, matching a configured file pattern, arrive within the expected timeframe. It calculates the expected frequency using at least two initial file uploads and applies a configurable tolerance percentage (default: 10%). For example, if a file is expected every 4 hours, one arriving in 4 hours and 15 minutes is valid, while one arriving after 5 hours is invalid. If no file arrives or fewer than two files are available for calculation, the measure is skipped.

Seasonality Handling:
At least 1 month of historical data is required to detect seasonality. If seasonality is detected, missing files on seasonal skip days are not treated as invalid.

Example:

- Files arrive daily at 10 AM.
- Every Sunday, no file is uploaded.
- On Monday, the system detects Sunday as a seasonal skip → File considered valid.

All of the above measures are automatically created for each S3 bucket and will appear in the Measure section in DQLabs.



Prerequisites

Whitelisting

If your organization uses a whitelist to manage AWS S3 access, reach out to customersupport@dqlabs.ai to assist with whitelisting

Permissions

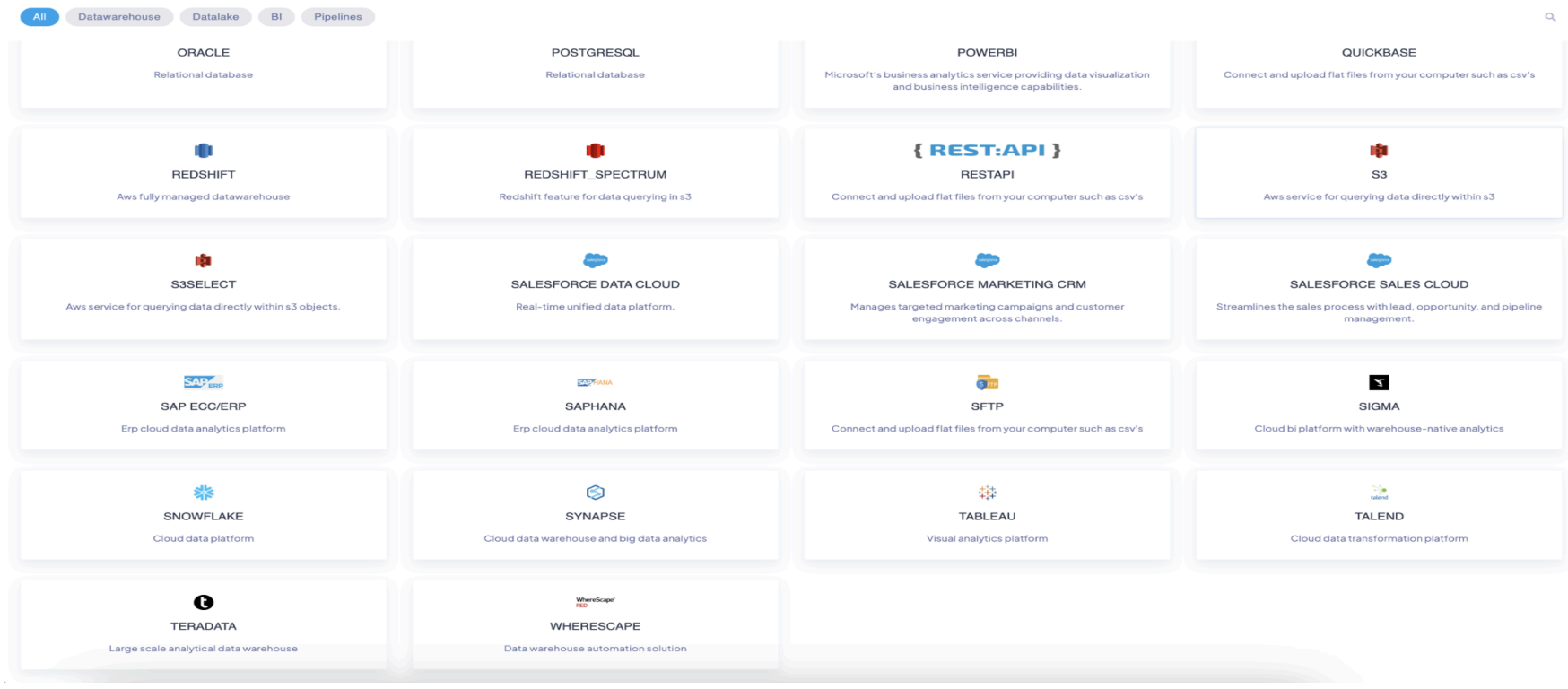
The following permissions are needed in s3 for the account to setup export failed using iceberg tables

Permission	Equivalent SQL Operation	Description	Notes
glue:CreateDatabase	CREATE NAMESPACE	Create a namespace for Iceberg tables (similar to Schema).	The Namescpase can be created before if this permission is not provided
glue:CreateTable	CREATE TABLE	Create a table (for creating icebreg table creation in s3).	
glue:DeleteTable	DROP TABLE	Delete a table in a namespace(To delete the created iceberg tables, including metadata in the catalog)	
glue:UpdateTable	INSERT TABLE, UPDATE TABLE, DELETE RECORDS	Insert, update, or delete records in a table.(To modify the records in the iceberg tables)	
glue:GetTable	GET TABLE	Read data from a table (To read metadata from the iceberg tables)	
s3:GetObject	SELECT	Allows querying of data in the iceberg tables	

Connect to S3

Follow the steps below to connect to s3 connector

Step 1: Navigate to Settings → Connect → Source and click on the "+" icon



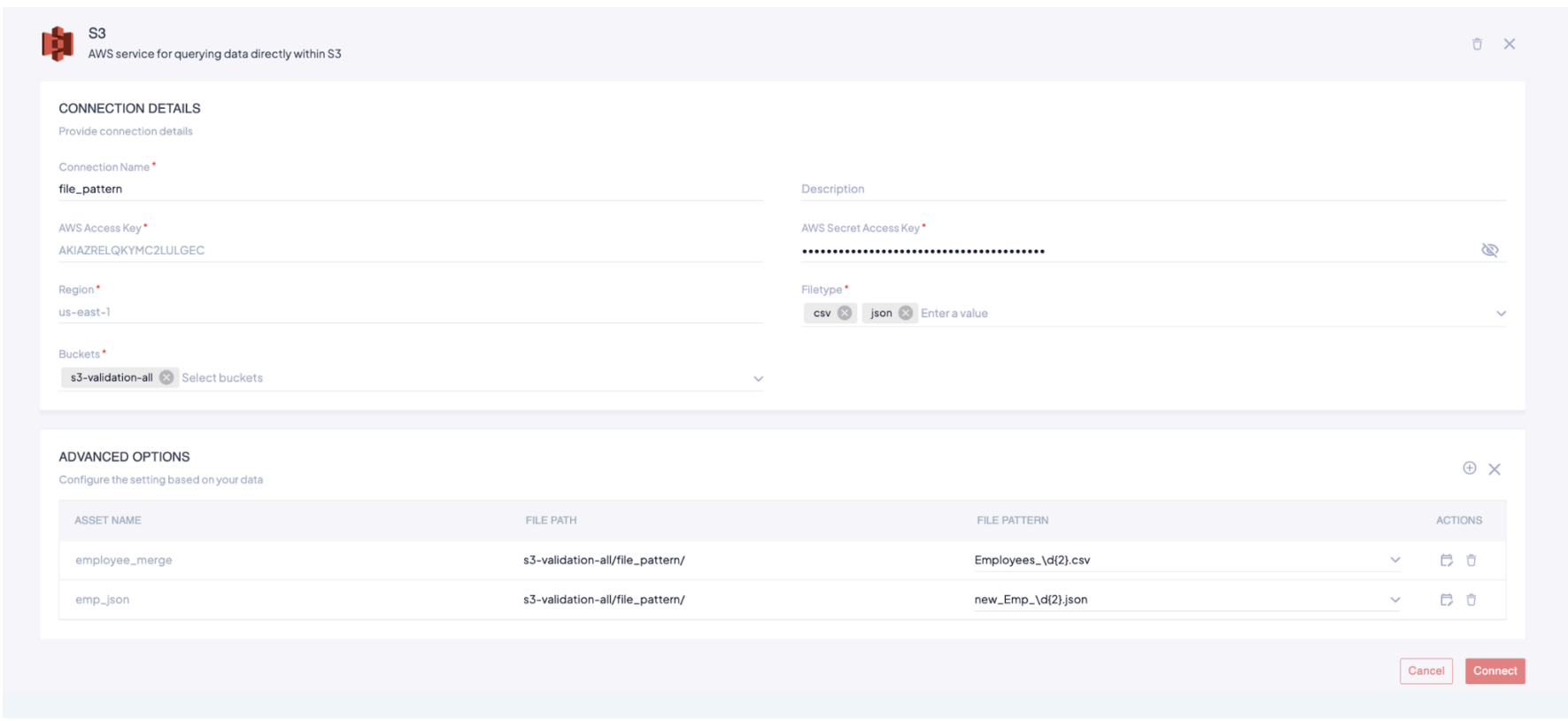
Step 2: Select "S3" and provide the following details

Field Name	Description
Connection Name	A unique name for the S3 connection, used to identify it within the platform.
Description	An optional text field to describe the purpose or usage of this connection.
AWS Access Key	The AWS access key ID used for authentication.
AWS Secret Access Key	The AWS secret access key, paired with the access key ID, is used for authentication.
Region	The AWS region where the S3 bucket is hosted (e.g., us-east-1).

Field Name	Description
Buckets	The target S3 bucket(s) to be connected for data validation and ingestion.
Filetype	The supported file types (CSV, JSON, Parquet, and XML) will be validated.

Supported Regex

Pattern	Description
.*	Matches any file name in the bucket.
.*\.csv	Matches all CSV files.
.*\.json	Matches all JSON files.
.*\.parquet	Matches all Parquet files.
^data_.*	Matches files starting with data_.
.*_2025\.csv	Matches files ending with _2025.csv.
^sales_[0-9]+\.csv$	Matches files like sales_001.csv, sales_202.csv.
^.*\d{4}-\d{2}-\d{2}\.csv$	Matches files with YYYY-MM-DD.csv format (e.g., 2025-09-18.csv).
^.*\d{8}\.csv$	Matches files with YYYYMMDD.csv format (e.g., 20250918.csv).
^.*\d{14}\.parquet$	Matches files with full timestamp YYYYMMDDHHMMSS.parquet.
.*_v[0-9]+\.csv	Matches versioned files like file_v1.csv, file_v10.csv.
.*_[0-9]{3}\.json	Matches files ending with 3-digit numbers like report_001.json.
^folder1/.*\.csv	Matches all CSV files under folder1/.
^raw/[0-9]{4}/[0-9]{2}/.*\.parquet	Matches partitioned files like raw/2025/09/data.parquet.



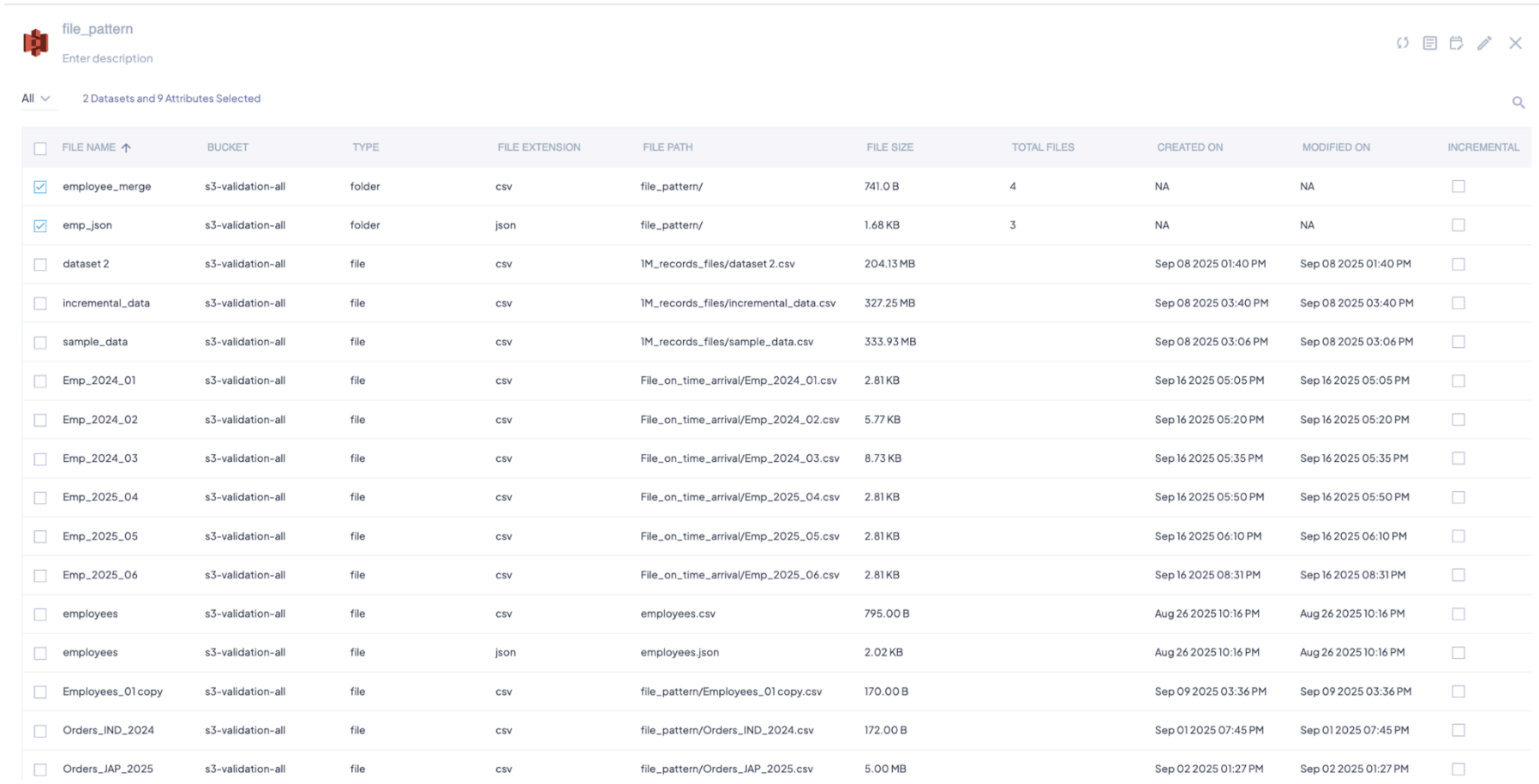
Advanced options

The Advanced Option allows admin or privileged users to select files based on specific file patterns within a batch and configure them as assets. Using this option, users can also set up incremental loads by leveraging file patterns, enabling efficient and automated data processing.

Field Name	Description
Asset Name	A user-defined name for the asset created from the S3 files.
File Path	The S3 path (within the bucket) where the files are stored.
File Pattern	A regex-based pattern used to detect and validate matching files in the path.
Actions	Schedule or delete the asset configuration

Once you provide the above details, click on connect.

Step 3: The users can now see the list of assets in the s3 bucket, including the assets configured by advanced options, select the assets to be configured, and click on connect.



FILE NAME	BUCKET	TYPE	FILE EXTENSION	FILE PATH	FILE SIZE	TOTAL FILES	CREATED ON	MODIFIED ON	INCREMENTAL
employee_merge	s3-validation-all	folder	csv	file_pattern/	741.0 B	4	NA	NA	
emp_json	s3-validation-all	folder	json	file_pattern/	1.68 KB	3	NA	NA	
dataset 2	s3-validation-all	file	csv	1M_records_files/dataset 2.csv	204.13 MB		Sep 08 2025 01:40 PM	Sep 08 2025 01:40 PM	
incremental_data	s3-validation-all	file	csv	1M_records_files/incremental_data.csv	327.25 MB		Sep 08 2025 03:40 PM	Sep 08 2025 03:40 PM	
sample_data	s3-validation-all	file	csv	1M_records_files/sample_data.csv	333.93 MB		Sep 08 2025 03:06 PM	Sep 08 2025 03:06 PM	
Emp_2024_01	s3-validation-all	file	csv	File_on_time_arrival/Emp_2024_01.csv	2.81 KB		Sep 16 2025 05:05 PM	Sep 16 2025 05:05 PM	
Emp_2024_02	s3-validation-all	file	csv	File_on_time_arrival/Emp_2024_02.csv	5.77 KB		Sep 16 2025 05:20 PM	Sep 16 2025 05:20 PM	
Emp_2024_03	s3-validation-all	file	csv	File_on_time_arrival/Emp_2024_03.csv	8.73 KB		Sep 16 2025 05:35 PM	Sep 16 2025 05:35 PM	
Emp_2025_04	s3-validation-all	file	csv	File_on_time_arrival/Emp_2025_04.csv	2.81 KB		Sep 16 2025 05:50 PM	Sep 16 2025 05:50 PM	
Emp_2025_05	s3-validation-all	file	csv	File_on_time_arrival/Emp_2025_05.csv	2.81 KB		Sep 16 2025 06:10 PM	Sep 16 2025 06:10 PM	
Emp_2025_06	s3-validation-all	file	csv	File_on_time_arrival/Emp_2025_06.csv	2.81 KB		Sep 16 2025 08:31 PM	Sep 16 2025 08:31 PM	
employees	s3-validation-all	file	csv	employees.csv	795.00 B		Aug 26 2025 10:16 PM	Aug 26 2025 10:16 PM	
employees	s3-validation-all	file	json	employees.json	2.02 KB		Aug 26 2025 10:16 PM	Aug 26 2025 10:16 PM	
Employees_01 copy	s3-validation-all	file	csv	file_pattern/Employees_01 copy.csv	170.00 B		Sep 09 2025 03:36 PM	Sep 09 2025 03:36 PM	
Orders_IND_2024	s3-validation-all	file	csv	file_pattern/Orders_IND_2024.csv	172.00 B		Sep 01 2025 07:45 PM	Sep 01 2025 07:45 PM	
Orders_JAP_2025	s3-validation-all	file	csv	file_pattern/Orders_JAP_2025.csv	5.00 MB		Sep 02 2025 01:27 PM	Sep 02 2025 01:27 PM	

Once connected, the files will be available as assets in the DQLabs platform

Limitations

- Secondary fingerprint incremental is not supported
- Look up and behavioural measures are not supported
- Exporting failed rows is not supported

---

16. Snowflake Cost and Performance

The Cost and Performance Dashboard has been streamlined to focus on the most valuable performance, cost, and storage insights. It now includes simplified filters, curated metrics, and redesigned tabs for improved usability and decision-making. The users can sync the data to get the updated metrics

Cost

The Cost Monitoring Dashboard provides comprehensive visibility into data warehouse and compute resource usage costs, helping teams optimize spending and track consumption patterns. The Cost dashboard enables monitoring and analysis of resource consumption costs across different data warehouse connections, providing insights into spending patterns and contract utilization to support financial planning and cost optimization. The cost dashboard has the following categories:

- **Insights**: Currently active tab showing cost analysis summaries
- **Compute**: View compute-specific resource usage and costs
- **Storage**: Monitor storage consumption and related expenses
- **Measure**: Access custom cost measurement configurations

The admin/privileged user will be able to select the connection and date range for each category to view the details

Insights

- The contract section provides the following details:
    - Contract Remaining: Shows unused contract balance
    - Contract Used: Displays total consumed amount
    - % Contract Used: Percentage of total contract consumed
    - Start Date: Contract beginning date
    - End Date: Contract expiration date
- High contract usage percentage indicates near-complete utilization of the allocated budget
- Contract timeline helps plan for renewals and budget adjustments
- Remaining balance tracking supports cost control and spending decisions

CONTRACT DETAILS

CONTRACT REMAINING($)	CONTRACT USED($)	% CONTRACT USED	START DATE	END DATE
$82.00	$9,918.00	99.18%	09/08/2025	04/22/2026

- The “Top Warehouses by Credit Consumption“ table provides the following details
  - Warehouse: Lists compute warehouse names (e.g., COMPUTE_WH)
  - Date: Shows specific dates when credits were consumed
  - Credits Used: Displays exact credit amounts consumed on each date
- Click column headers to sort by warehouse, date, or credit usage

Insights

- Daily Tracking: Monitor day-by-day credit consumption patterns
- Warehouse Comparison: Identify which warehouses consume the most resources
- Trend Analysis: Spot usage spikes or unusual consumption patterns
- Cost Attribution: Understand which workloads drive the highest costs

TOP WAREHOUSES BY CREDIT CONSUMPTION

WAREHOUSE	DATE	CREDITS USED ↓
COMPUTE_WH	Aug 26 2025	43.02
COMPUTE_WH	Aug 25 2025	39.21
COMPUTE_WH	Aug 27 2025	33.59
COMPUTE_WH	Sep 02 2025	31.16
COMPUTE_WH	Aug 28 2025	28.61
COMPUTE_WH	Aug 31 2025	25.95
COMPUTE_WH	Sep 03 2025	25.51

Total rows 167 — 50 Rows per page — 1 of 4

- Warehouse Cost Spikes shows individual warehouses on specific dates where credit consumption spiked compared to the previous day.
  - Warehouse: Name of the compute warehouse.
  - Usage Date: Date of the observed spike.
  - Credits Used: Total credits consumed on that date.
  - Previous Day Credits: Credits consumed the day before.
  - Percentage Increase: Relative spike in usage (e.g., 292.43% for SVC_AIRFLOW_DEV_WH on Aug 27).

WAREHOUSE COST SPIKES

WAREHOUSE	USAGE DATE	CREDITS USED	PREVIOUS DAY CREDITS	PERCENTAGE INCREASE
FIVETRAN_WAREHOUSE	Aug 25 2025	0.03	0.03	1.4
COMPUTE_WH	Aug 25 2025	39.21	24.71	58.7
SVC_REPORTING_DEV_WH	Aug 25 2025	0.11	0.05	110.43
FIVETRAN_WAREHOUSE	Aug 26 2025	0.04	0.03	3.42
COMPUTE_WH	Aug 26 2025	43.02	39.21	9.71
SVC_REPORTING_DEV_WH	Aug 26 2025	0.12	0.11	7.53
SVC_AIRFLOW_DEV_WH	Aug 27 2025	0.18	0.04	292.43

Total rows 61 — 50 Rows per page — 1 of 2

Insights

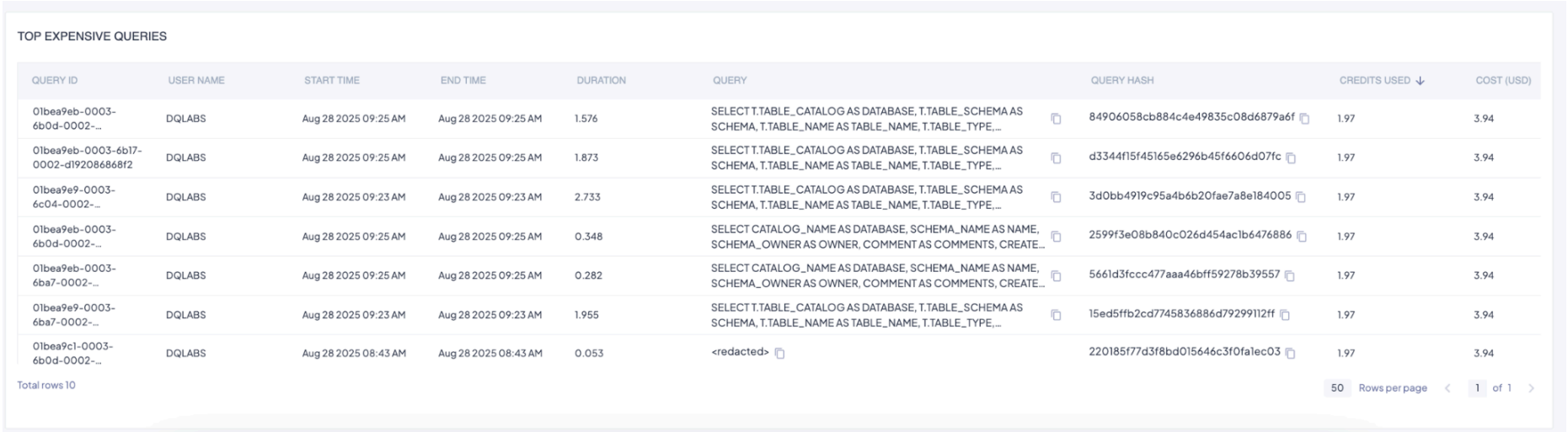
- Warehouses with high percentage increases may indicate runaway queries or misconfigurations.
- Even small absolute credit jumps (e.g., from 0.04 to 0.18 credits) can signal unexpected development activity or testing gone awry.
- Average Long-Running Queries by Warehouse helps to determine which warehouses suffer the most from slow queries.
- The table contains
  - Date: When the measurement was taken.
  - Long-Running Query Count: Number of queries exceeding a defined execution time threshold.
  - Average Execution Time (sec): Mean runtime of those long-running queries.
  - Max Execution Time (sec): Slowest query duration observed.

AVERAGE LONG RUNNING QUERIES BY WAREHOUSE

WAREHOUSE	DATE	LONG RUNNING QUERY COUNT ↓	AVG EXECUTION TIME (SEC)	MAX EXECUTION TIME (SEC)
COMPUTE_WH	Aug 27 2025	306	753.310882352941	3601.658
SVC_REPORTING_DEV_WH	Sep 22 2025	18	74.574666666667	87.087
COMPUTE_SERVICE_WH_USER_TASKS_POOL_XSMALL_0	Sep 16 2025	12	147.976416666667	232.675

Total rows 3 — 50 Rows per page — 1 of 1

- Comparing average vs. max runtimes helps distinguish consistent slowdowns from one-off outliers.
- Top Expensive Queries table shows a detailed table listing the queries that consumed the most credits (and incurred the highest USD cost) during the selected period. Columns include:
    - Query ID: Unique identifier for each query execution.
    - User Name: The account or service that ran the query.
    - Start Time / End Time: Timestamps marking query execution window.
    - Duration: Total runtime in seconds.
    - Query: The SQL text (truncated) so you can identify the operation.
    - Query Hash: Unique fingerprint for grouping identical query patterns.
    - Credits Used: Compute credits charged for the query.
    - Cost (USD): Dollar cost derived from credit usage.


TOP EXPENSIVE QUERIES

QUERY ID	USER NAME	START TIME	END TIME	DURATION	QUERY	QUERY HASH	CREDITS USED ↓	COST (USD)
01bea9eb-0003-6b0d-0002-...	DQLABS	Aug 28 2025 09:25 AM	Aug 28 2025 09:25 AM	1.576	SELECT T.TABLE_CATALOG AS DATABASE, T.TABLE_SCHEMA AS SCHEMA, T.TABLE_NAME AS TABLE_NAME, T.TABLE_TYPE,...	84906058cb884c4e49835c08d6879a6f	1.97	3.94
01bea9eb-0003-6b17-0002-d192086868f2	DQLABS	Aug 28 2025 09:25 AM	Aug 28 2025 09:25 AM	1.873	SELECT T.TABLE_CATALOG AS DATABASE, T.TABLE_SCHEMA AS SCHEMA, T.TABLE_NAME AS TABLE_NAME, T.TABLE_TYPE,...	d3344f15f45165e6296b45f6606d07fc	1.97	3.94
01bea9e9-0003-6c04-0002-...	DQLABS	Aug 28 2025 09:23 AM	Aug 28 2025 09:23 AM	2.733	SELECT T.TABLE_CATALOG AS DATABASE, T.TABLE_SCHEMA AS SCHEMA, T.TABLE_NAME AS TABLE_NAME, T.TABLE_TYPE,...	3d0bb4919c95a4b6b20fae7a8e184005	1.97	3.94
01bea9eb-0003-6b0d-0002-...	DQLABS	Aug 28 2025 09:25 AM	Aug 28 2025 09:25 AM	0.348	SELECT CATALOG_NAME AS DATABASE, SCHEMA_NAME AS NAME, SCHEMA_OWNER AS OWNER, COMMENT AS COMMENTS, CREATE...	2599f3e08b840c026d454ac1b6476886	1.97	3.94
01bea9eb-0003-6ba7-0002-...	DQLABS	Aug 28 2025 09:25 AM	Aug 28 2025 09:25 AM	0.282	SELECT CATALOG_NAME AS DATABASE, SCHEMA_NAME AS NAME, SCHEMA_OWNER AS OWNER, COMMENT AS COMMENTS, CREATE...	5661d3fccc477aaa46bff59278b39557	1.97	3.94
01bea9e9-0003-6ba7-0002-...	DQLABS	Aug 28 2025 09:23 AM	Aug 28 2025 09:23 AM	1.955	SELECT T.TABLE_CATALOG AS DATABASE, T.TABLE_SCHEMA AS SCHEMA, T.TABLE_NAME AS TABLE_NAME, T.TABLE_TYPE,...	15ed5ffb2cd7745836886d79299112ff	1.97	3.94
01bea9c1-0003-6b0d-0002-...	DQLABS	Aug 28 2025 08:43 AM	Aug 28 2025 08:43 AM	0.053	<redacted>	220185f77d3f8bd015646c3f0fa1ec03	1.97	3.94

Total rows 10 — 50 Rows per page — 1 of 1

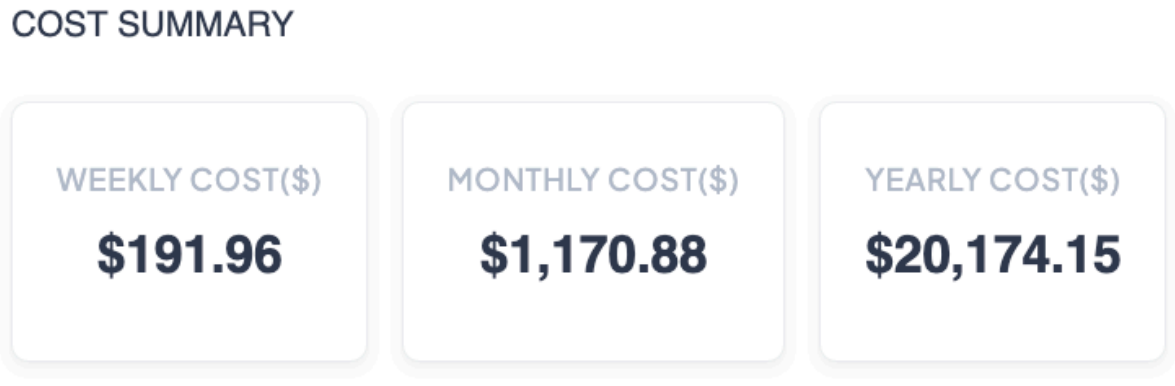
Insights

- High-cost queries often indicate full table scans, large data shuffles, or unbounded result sets.
- Short-duration but expensive queries can still incur significant costs if they touch large volumes of data.
- Identical Query Hashes with varying runtimes or costs signal changes in data volume or query plan inefficiencies over time.

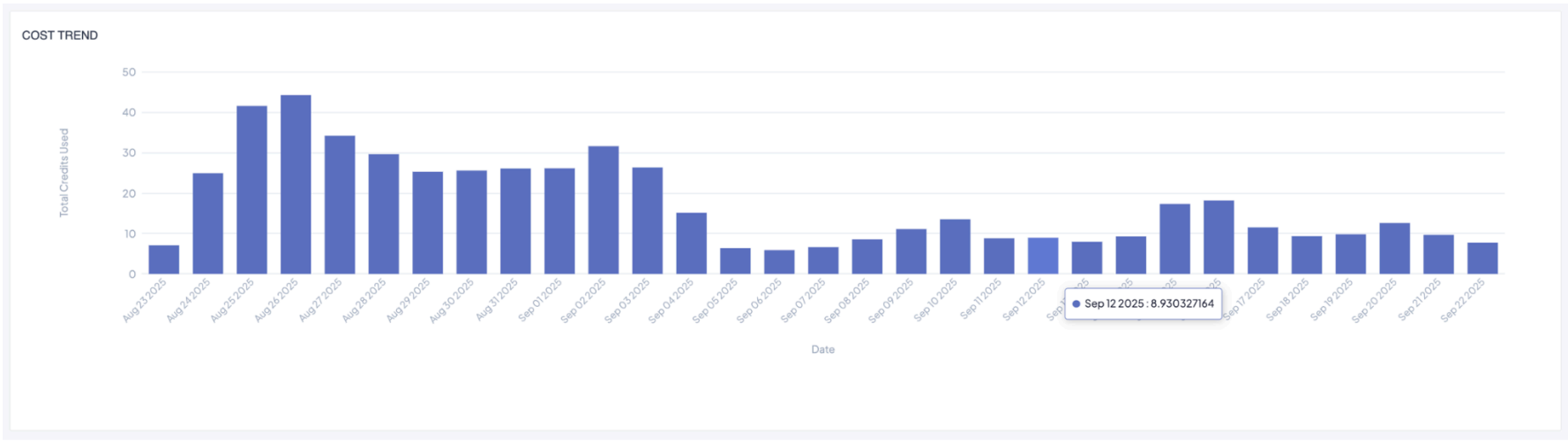
Compute

The Compute tab in the Cost dashboard provides a detailed view of compute resource spending over time, helping teams understand weekly, monthly, and annual costs as well as daily credit consumption trends.

- Cost summary shows high-level metrics enable rapid assessment of short-term and long-term compute expenditure
- Weekly Cost ($): Total credits converted to dollars spent over the last seven days
- Monthly Cost ($): Total credits converted to dollars spent over the last 30 days.
- Yearly Cost ($): Extrapolated or actual spend over the last 365 days



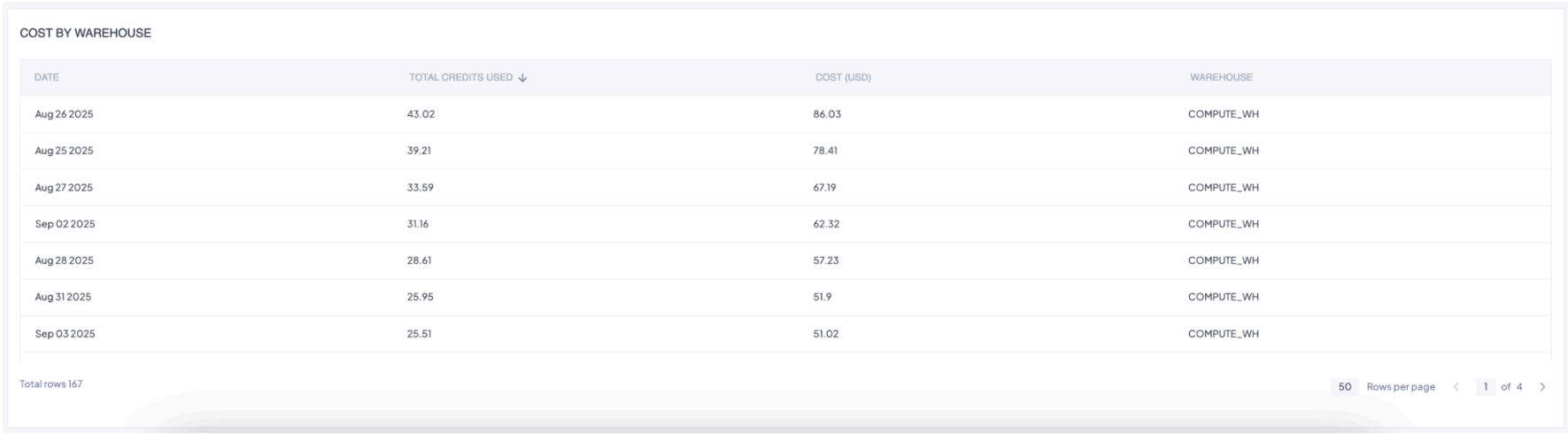
- The Cost Trend Bar Chart shows daily bars that display the exact credits consumed each day within the selected period
- A vertical bar chart plotting Total Credits Used on the Y-axis against Date on the X-axis.



Insights

- Identify Spikes and Valleys: Locate dates with unusually high or low credit usage.
- Correlate with Events: Map spikes to heavy ETL jobs, analytics queries, or pipeline runs scheduled on those dates.
- Assess Stability: Consistent bar heights indicate stable workload demand; large fluctuations may signal inefficiencies or unplanned jobs.

- Budget Forecasting: Use daily patterns to predict future spend and adjust warehouse sizing or scheduling to smooth peaks
- Cost by Warehouse lists each day's compute usage per warehouse within the selected date range. Columns include:
    - Date: The specific day of usage.
    - Total Credits Used: Number of compute credits consumed by that warehouse on that day.
    - Cost (USD): Dollar amount equivalent of the consumed credits, calculated using your agreed rate (e.g., 43.02 credits → $86.03).
    - Warehouse: Name of the Snowflake compute warehouse or service cluster



COST BY WAREHOUSE

DATE	TOTAL CREDITS USED	COST (USD)	WAREHOUSE
Aug 26 2025	43.02	86.03	COMPUTE_WH
Aug 25 2025	39.21	78.41	COMPUTE_WH
Aug 27 2025	33.59	67.19	COMPUTE_WH
Sep 02 2025	31.16	62.32	COMPUTE_WH
Aug 28 2025	28.61	57.23	COMPUTE_WH
Aug 31 2025	25.95	51.9	COMPUTE_WH
Sep 03 2025	25.51	51.02	COMPUTE_WH

Total rows 167 — 50 Rows per page — 1 of 4

Insights

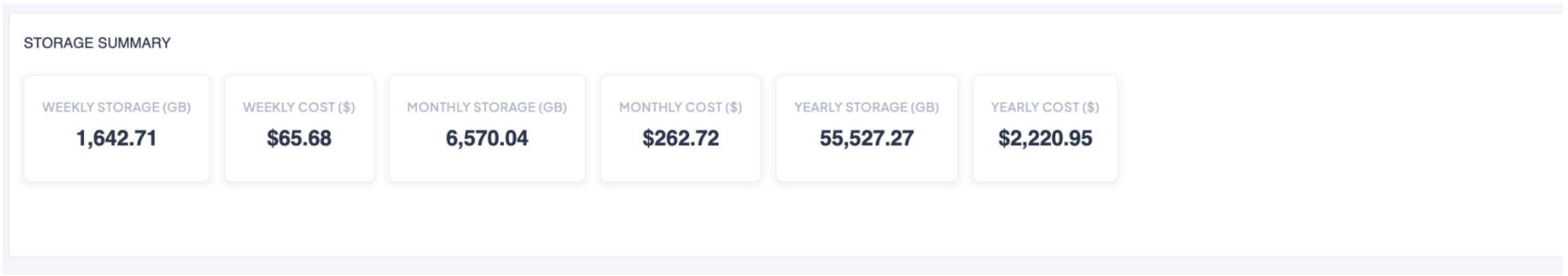
- Heavy Consumers: Warehouses repeatedly appearing at the top of this list signal hotspots of compute activity that may need rightsizing or schedule adjustments.
- Cost Efficiency: Warehouses with similar credit usage but different costs may reflect tiered pricing or different runtime characteristics.
- Usage Patterns: Consistent daily consumption suggests stable workloads, while erratic spikes warrant investigation.

Storage

The Cost dashboard's Storage tab gives a clear summary of your organization's storage usage and related costs, as well as a breakdown of storage consumption by database for the selected connection and date range

- Weekly Storage (GB): Quantity of storage consumed in the last 7 days
- Weekly Cost ($): Storage cost for the week
- Monthly Storage (GB): Storage used in the last 30 days
- Monthly Cost ($): Storage cost for the month
- Yearly Storage (GB): Storage accumulated over the past year
- Yearly Cost ($): Total annual storage cost

These cards provide an at-a-glance context for rapid assessment of data footprint and trending storage expenses



STORAGE SUMMARY

WEEKLY STORAGE (GB)	WEEKLY COST ($)	MONTHLY STORAGE (GB)	MONTHLY COST ($)	YEARLY STORAGE (GB)	YEARLY COST ($)
1,642.71	$65.68	6,570.04	$262.72	55,527.27	$2,220.95

- The Top Databases by Storage Cost tables show each database by its storage cost
    - Database: Each line shows a database name with the greatest daily storage cost.
    - Date: When the storage usage was measured
    - Storage Cost (USD): Dollar cost of the storage for that database on this date.
    - Storage (GB): How much storage (in GB) was used.

Insights

- Most storage costs are driven by a handful of large databases.
- Tracking these costs frequently enables early detection of anomalous growth in database size or cost

TOP DATABASES BY STORAGE COST

DATABASE	DATE	STORAGE COST (USD)	STORAGE (GB)
DQLABS_PROSPECTS	Sep 22 2025	97.39	2434.72
DQLABS	Sep 22 2025	78.54	1963.51
DQLABS_COST_COMPUTE	Sep 22 2025	59.95	1498.87
DQLABS_PROSPECTS	Sep 22 2025	25.13	628.31
DQLABS	Sep 22 2025	20.27	506.77
DQLABS_COST_COMPUTE	Sep 22 2025	15.48	386.94
DQLABS_REPL_VERSION1	Sep 22 2025	6.09	152.23

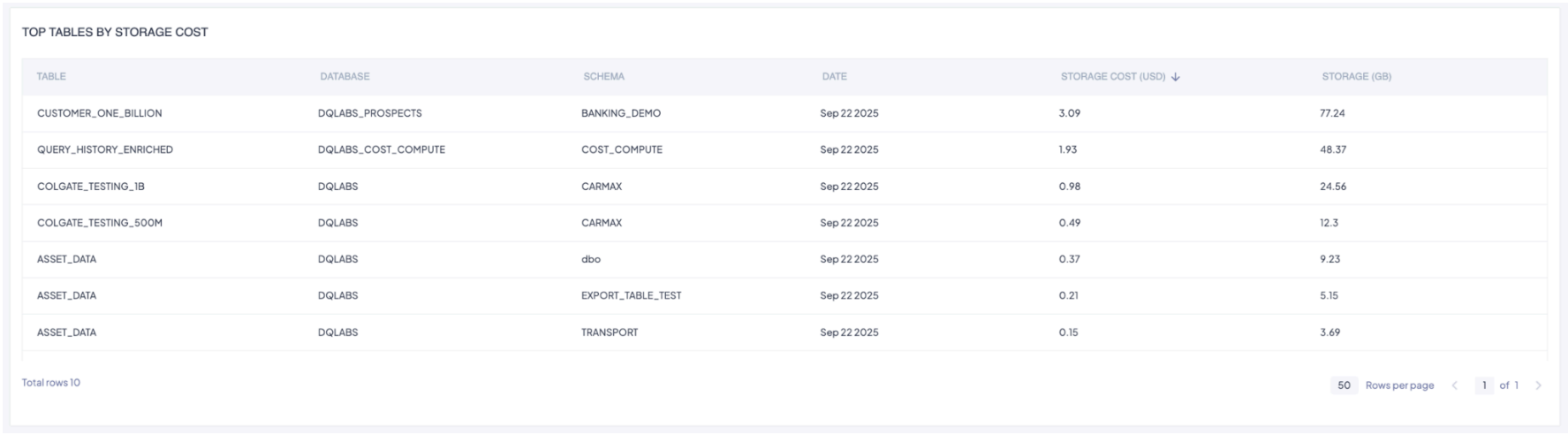
Total rows 10 — 50 Rows per page — 1 of 1

- The "Top Tables by Storage Cost" shows which individual tables within your data platform are consuming the most storage and incurring the highest associated storage costs. It provides information on the table name, the database and schema it belongs to, the date of the measurement, the storage size in gigabytes (GB), and the estimated storage cost in USD for the given period.
    - Table: Name of each high-cost table
    - Database: The owning logical database
    - Schema: The schema containing the table

- Date: Storage measurement date
- Storage Cost (USD): Daily dollar cost for storage.
- Storage (GB): Gigabytes consumed by the table.

Insights

- A small set of tables typically drives the majority of storage costs.
- Large, rarely accessed tables (e.g., historical test data, enriched query logs) may be good candidates for cost reduction.



TOP TABLES BY STORAGE COST

TABLE	DATABASE	SCHEMA	DATE	STORAGE COST (USD)	STORAGE (GB)
CUSTOMER_ONE_BILLION	DQLABS_PROSPECTS	BANKING_DEMO	Sep 22 2025	3.09	77.24
QUERY_HISTORY_ENRICHED	DQLABS_COST_COMPUTE	COST_COMPUTE	Sep 22 2025	1.93	48.37
COLGATE_TESTING_1B	DQLABS	CARMAX	Sep 22 2025	0.98	24.56
COLGATE_TESTING_500M	DQLABS	CARMAX	Sep 22 2025	0.49	12.3
ASSET_DATA	DQLABS	dbo	Sep 22 2025	0.37	9.23
ASSET_DATA	DQLABS	EXPORT_TABLE_TEST	Sep 22 2025	0.21	5.15
ASSET_DATA	DQLABS	TRANSPORT	Sep 22 2025	0.15	3.69

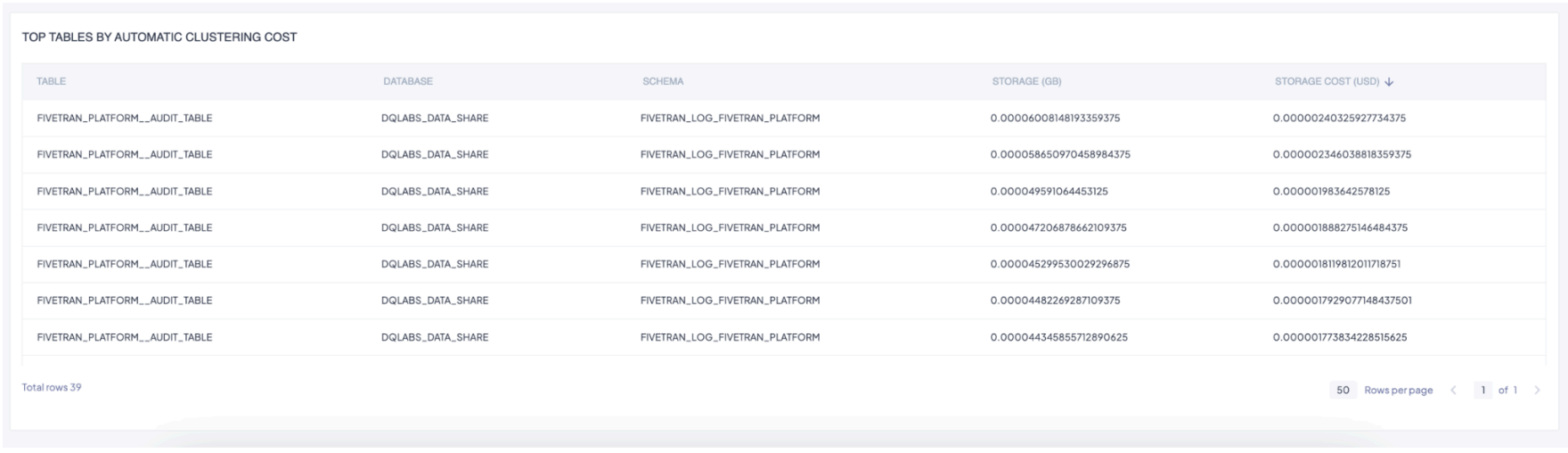
Total rows 10

50 Rows per page 1 of 1

- The "Top Tables by Automatic Clustering Cost" shows specific tables where Snowflake's automatic clustering feature is incurring storage-related costs. Automatic clustering continuously reorganizes underlying data to improve query performance, but this reclustering process uses compute resources and storage, generating additional credits spent.
- This section lists tables with clustering costs broken down by database and schema, showing storage footprint (in gigabytes) and the associated clustering cost in dollars. It helps identify tables where clustering overhead is significant, enabling teams to evaluate if clustering benefits outweigh costs, tune clustering policies, or disable clustering where not cost-effective.
  - Table: Name of clustered table
  - Database: Database containing the table
  - Schema: Associated schema
  - Storage (GB): Table's cluster-driven storage footprint.
  - Storage Cost (USD): Daily cost incurred by clustering.

Insights

- Automatic clustering can add a nontrivial cost for large or frequently updated tables.
- Monitor the clustering cost table to maintain a balance between query performance benefits and storage spend



TOP TABLES BY AUTOMATIC CLUSTERING COST

TABLE	DATABASE	SCHEMA	STORAGE (GB)	STORAGE COST (USD)
FIVETRAN_PLATFORM__AUDIT_TABLE	DQLABS_DATA_SHARE	FIVETRAN_LOG_FIVETRAN_PLATFORM	0.0000600814819335937 5	0.000002403259277343 75
FIVETRAN_PLATFORM__AUDIT_TABLE	DQLABS_DATA_SHARE	FIVETRAN_LOG_FIVETRAN_PLATFORM	0.000058650970458984375	0.000023460388183593 75
FIVETRAN_PLATFORM__AUDIT_TABLE	DQLABS_DATA_SHARE	FIVETRAN_LOG_FIVETRAN_PLATFORM	0.000049591064453125	0.000001983642578125
FIVETRAN_PLATFORM__AUDIT_TABLE	DQLABS_DATA_SHARE	FIVETRAN_LOG_FIVETRAN_PLATFORM	0.000047206878662109375	0.000001888275146484375
FIVETRAN_PLATFORM__AUDIT_TABLE	DQLABS_DATA_SHARE	FIVETRAN_LOG_FIVETRAN_PLATFORM	0.000045299530029296875	0.000001811981201171875
FIVETRAN_PLATFORM__AUDIT_TABLE	DQLABS_DATA_SHARE	FIVETRAN_LOG_FIVETRAN_PLATFORM	0.00004482269287109375	0.000001792907714843750 1
FIVETRAN_PLATFORM__AUDIT_TABLE	DQLABS_DATA_SHARE	FIVETRAN_LOG_FIVETRAN_PLATFORM	0.000044345855712890625	0.000001773834228515625

Total rows 39

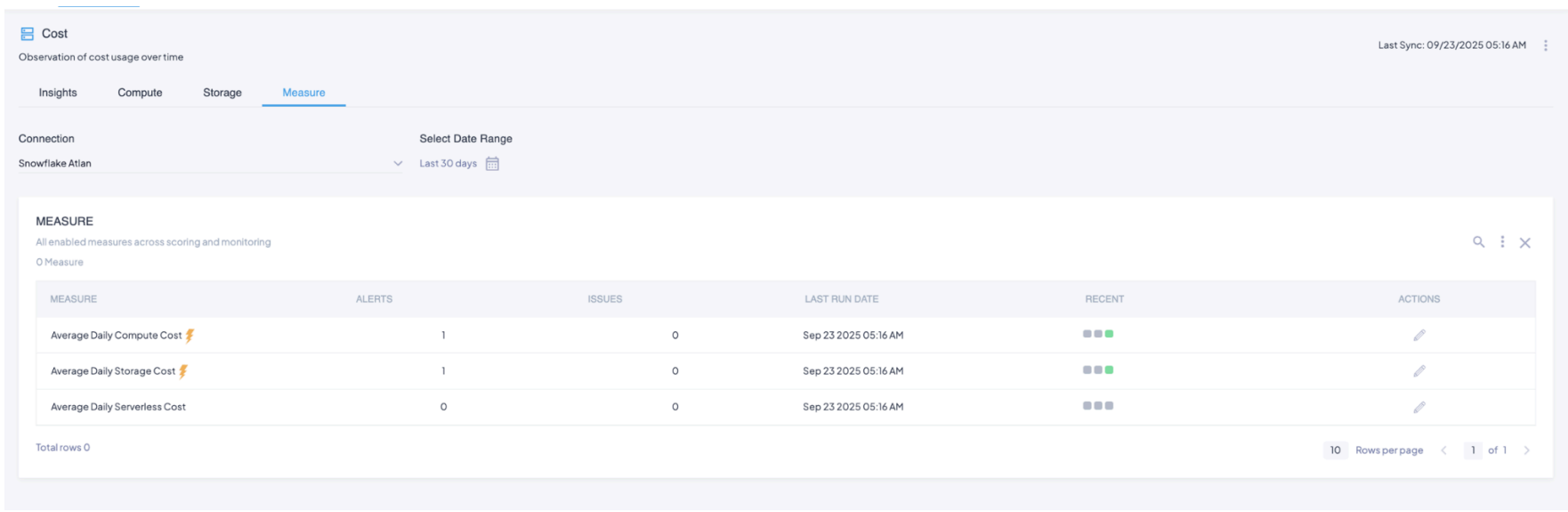
50 Rows per page 1 of 1

Measure

The Measure tab summarizes the status of enabled cost monitoring measures applied to the connection.The Measure Table shows a list of custom-defined cost measures enabled for monitoring and scoring within the data platform. It displays:

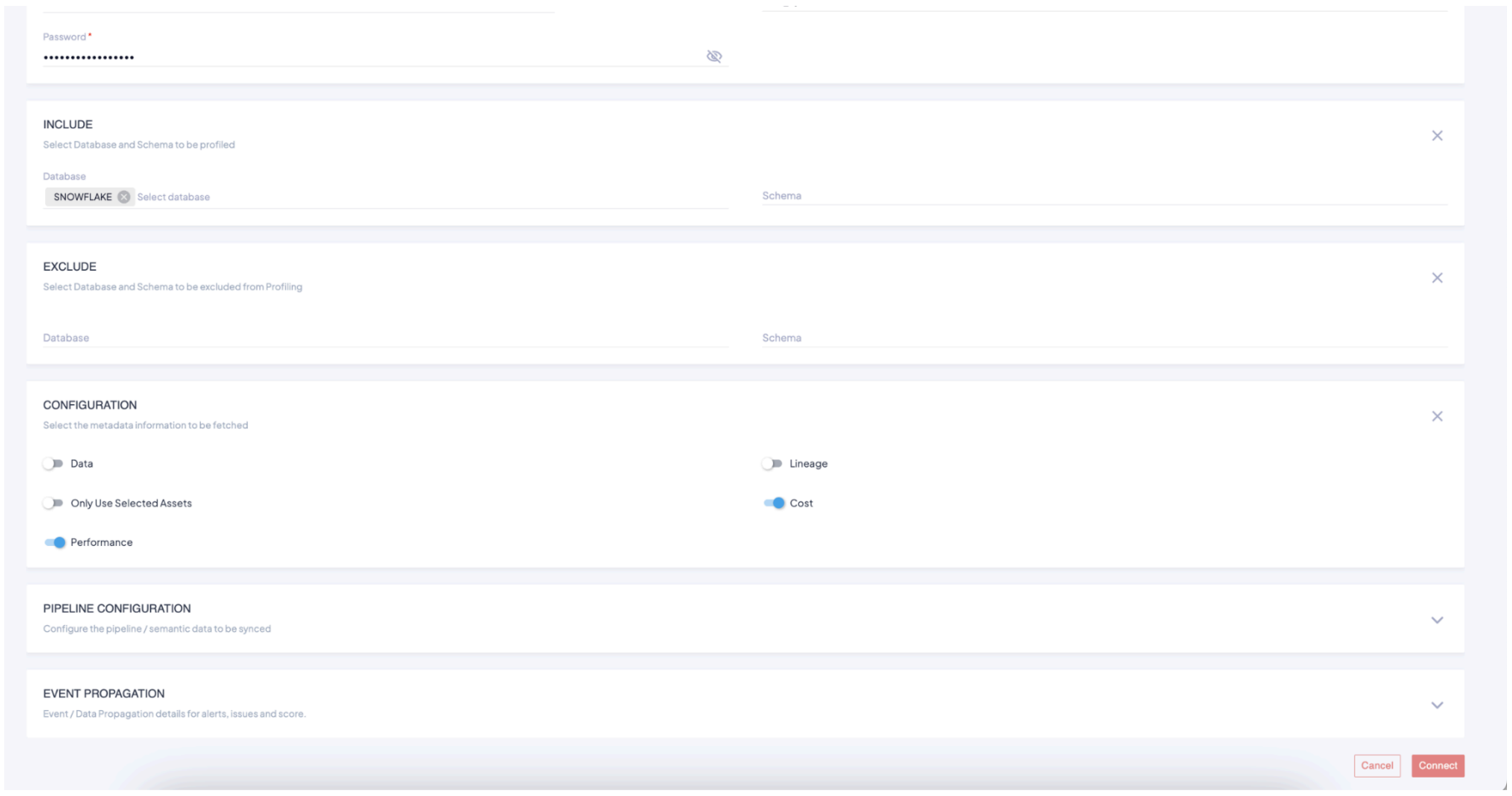
- The name of each measure
- The number of alerts triggered for that measure
- The count of associated issues
- The timestamp of the last time the measure was run
- A visual recent status indicator of the measure's health or activity
- An action button to edit the measure settings
- The admin/privileged user will able to able to select other applicable columns for the select columns list

This table helps users track key financial KPIs, see any anomalies or alerts related to cost spikes or trends, and maintain oversight over cost measure health across their data environment. It supports proactive optimization and governance of compute and storage expenses.



The admin/privileged user can enable the following measures from the connection configuration page by selecting the cost toggle.

- Average Daily Compute Cost
- Average Daily Storage Cost
- Average Daily Serverless Cost



Performance

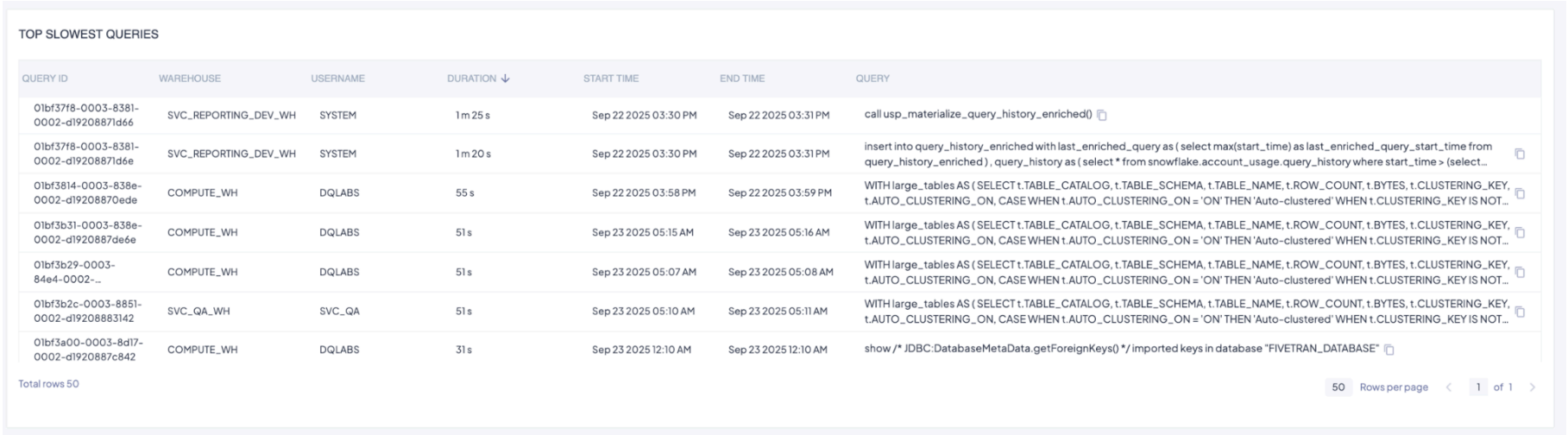
The Performance Monitoring Dashboard provides operational visibility into the efficiency and responsiveness of your data warehouse environment. It enables teams to track, diagnose, and optimize system performance by surfacing key metrics, bottlenecks, and usage patterns across SQL queries, compute resources, and user activity.

The admin/privileged user will be able to select the connection and date range for each category to view the details

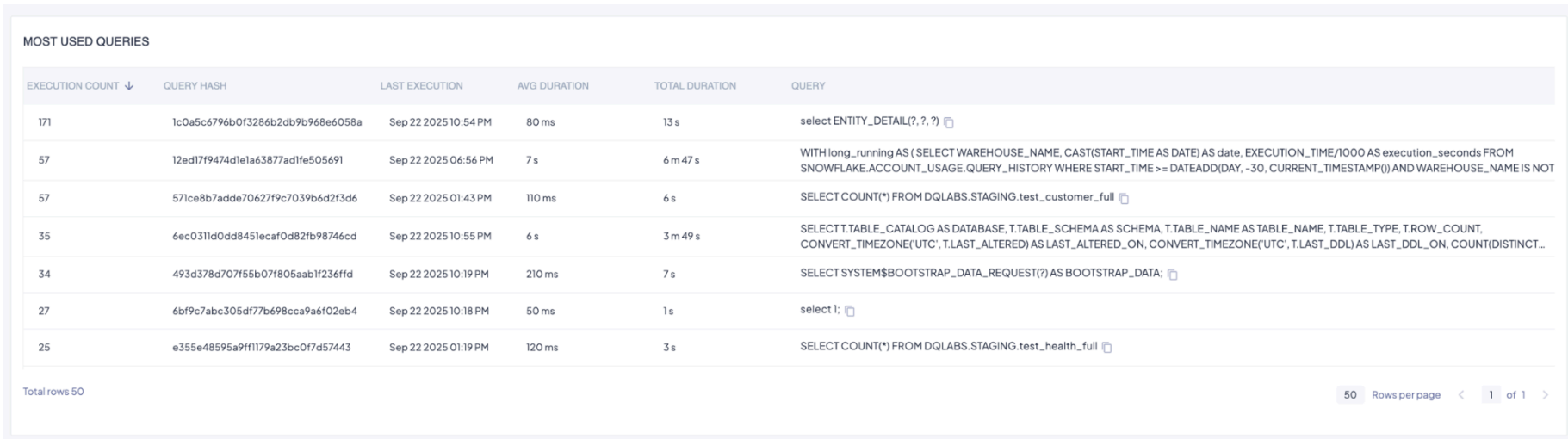
Query

The query tab provides a focused view of query execution health, highlighting bottlenecks and heavy usage patterns to support troubleshooting, optimization, and efficient resource management.

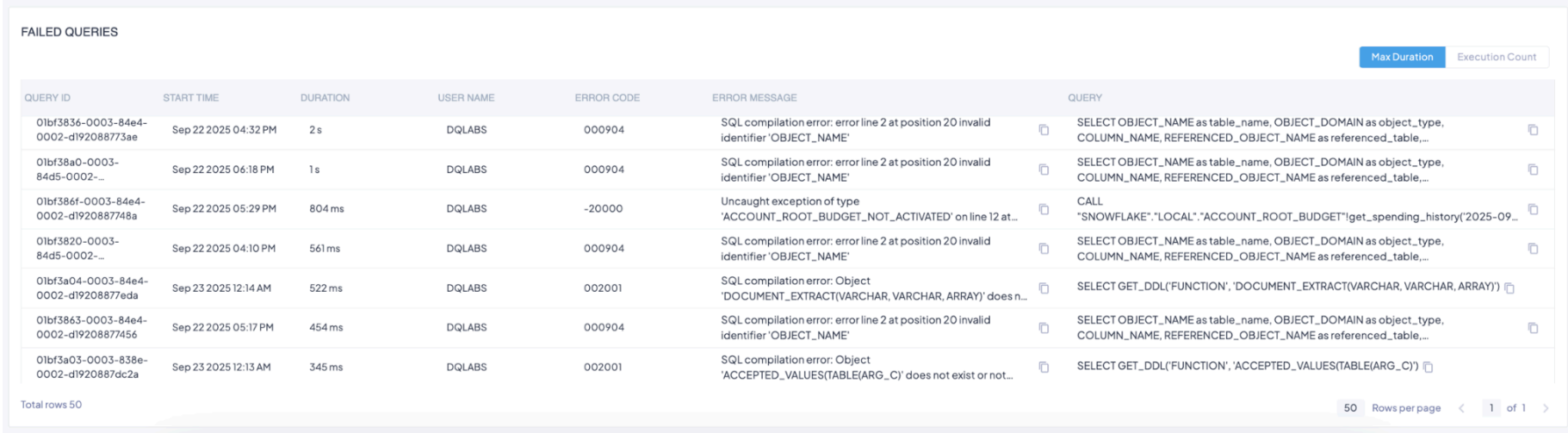
- Top Slowest Queries shows the list of the slowest-running SQL queries for the selected connection and date range, with the following columns:
  - Query ID: Unique identifier for each SQL execution.
  - Warehouse: The compute cluster where the query ran.
  - Username: Who triggered the query (user or service).
  - Duration: Elapsed time taken to complete the query.
  - Start Time / End Time: Exact timestamps for when the query started and finished.
  - Query: The SQL statement executed (truncated for preview).



- Most Used Queries shows queries with the highest execution count, signaling workload patterns and repeat operations
  - Execution Count: Number of times a query was run in the selected period
  - Query Hash: Unique fingerprint grouping identical SQL logic
  - Last Execution: Most recent run time/date for that query logic
  - Avg Duration: Mean execution time per run
  - Total Duration: Aggregate time spent executing that query
  - Query: SQL statement text as preview (expand for full)



- The Failed Queries view provides a powerful interface for data teams to detect, investigate, and resolve errors arising from SQL query failures in the warehouse. It surfaces key diagnostic details for each failed query, supporting rapid root cause analysis and operational improvement.
- The admin/privileged user will be able to view the failed queries by the following:
  - Max Duration - Use "Max Duration" to prioritize queries that consumed the most resources before failing
  - Query ID: Unique identifier for each failed query run.
  - Start Time: Timestamp marking when a query was submitted for execution.
  - Duration: Time taken before the query failed, helpful for spotting failed long-running requests
  - User Name: User or service account who submitted the query
  - Error Code: Numeric code categorizing the failure type (e.g., compilation error, missing object, permission issue)
  - Error Message: Textual detail describing why the query failed (e.g., syntax error, missing identifier, runtime exception)
  - Query: The SQL text attempted (typically truncated for preview), expands for complete inspection.



- Execution Count - Use "Execution Count" to spot recurring failures quickly.
  - Query Hash: Fingerprint grouping identical logic; aggregates failures for easier pattern detection.
  - Start Time: Timestamp of first failure (per hash or logic).
  - Execution Count: Number of failed executions for this query logic in the selected timeframe.
  - Error Code & Error Message: As above, but aggregated for matching query patterns.
  - Query: Preview or truncated query text

FAILED QUERIES

Max Duration | Execution Count

QUERY HASH	START TIME	EXECUTION COUNT ↓	ERROR CODE	ERROR MESSAGE	QUERY
3397d97d615cefa5ac2c206c281d3a8d	Sep 23 2025 11:39 AM	30	003030	SQL compilation error: Shared database is no longer available for use. It will need to be re-...	<redacted>
5cc670b75dc83ccd855a0799f4164293	Sep 22 2025 05:29 PM	16	002003		<redacted>
f5d69968d839ec48f54a8dbb488355e8	Sep 22 2025 05:29 PM	11	-20000	Uncaught exception of type 'ACCOUNT_ROOT_BUDGET_NOT_ACTIVATED' ...	CALL "SNOWFLAKE"."LOCAL"."ACCOUNT_ROOT_BUDGET"!get_spending_history('2025-09-01', '2025-09-30')
9543f67475dae1f6b370964lff697dad	Sep 22 2025 10:18 PM	3	002043	SQL compilation error: Object does not exist, or operation cannot be performed.	<redacted>
d9bcb6d5f0dd502225c0796709e92968	Sep 22 2025 02:13 PM	2	002003	SQL compilation error: Object 'DQLABS.STAGING.TEST_MEDICAL_FULL' does...	create or replace view DQLABS.ANALYTICS_staging.stg_medical as (-- Staging model for medical data -- This model cleans and standardizes the raw medical data from STAGING.TEST_MEDICAL_FULL with source as (select *...
ccdda0dd8f4aaff645c5c9f1e79d0c20	Sep 22 2025 02:13 PM	2	002003	SQL compilation error: Object 'DQLABS.STAGING.TEST_CUSTOMER_EXISTING...	create or replace view DQLABS.ANALYTICS_staging.stg_customers_existing as (-- Staging model for existing customer data -- This model cleans and standardizes the raw existing customer data from...
42157461891f4bbe72994fc3b5ee6e2f	Sep 22 2025 10:23 PM	1	003001	SQL access control error: Insufficient privileges to operate on user 'SVC_PRIZM_DEV'.	<redacted>

Total rows 8

50 Rows per page | 1 of 1

Insights

- Recurring queries with identical errors indicate systemic issues in data structures or access policies.
- High-duration failures suggest wasted compute resources and may require deeper flow or logic review before submission.
- Permissions and missing object errors can be proactively addressed with updated grants or improved ETL dependency documentation.
- Tracking error frequency and types guides user training, data pipeline robustness, and proactive monitoring rule updates

Compute and Storage

This section provides visibility into queuing and concurrency issues across compute warehouses, helping operations teams diagnose bottlenecks and optimize resource allocation.

- The "Queuing and Concurrency Issues" table shows detailed information about query queuing and concurrency problems occurring in various compute warehouses within your data environment
  - Warehouse: The name of the compute warehouse where queries were processed.
  - Date: The specific day when the queuing metrics were recorded.
  - Queued Queries: The count of queries that experienced queuing, i.e., those that waited for compute resources before execution.
  - Total Queries: The total number of queries executed on that day in the warehouse.
  - Avg Queue Time: The average amount of time queries spent waiting in the queue before execution.
  - Max Queue Time: The longest single query queue wait time recorded, showing the worst case of delay.

Key Insights

- High queue counts or long delays indicate insufficient resource sizing or poorly distributed workload.
- Warehouses with NA in wait times usually are well-sized or lightly loaded.
- Daily review and alerting on queue metrics prevent user complaints and missed reporting deadlines.
- Queuing metrics, when correlated with business events (e.g., batch ETL, end-of-month reporting), drive effective infrastructure planning

QUEUING AND CONCURRENCY ISSUES

WAREHOUSE	DATE	QUEUED QUERIES ↓	TOTAL QUERIES	AVG QUEUE TIME	MAX QUEUE TIME
COMPUTE_WH	Sep 22 2025	3429	10131	9 m 24 s	1 h 16 m 57 s
SVC_QA_WH	Sep 22 2025	5	61	750 ms	46 s
SVC_DEMO_WH	Sep 22 2025	4	32	NA	NA
SVC_REPORTING_DEV_WH	Sep 22 2025	2	76	NA	NA
FIVETRAN_WAREHOUSE	Sep 22 2025	1	369	NA	NA
USER_ADHOC	Sep 22 2025	0	135	NA	NA
COMPUTE_SERVICE_WH_USER_TASKS_POOL_XSMALL_0	Sep 22 2025	0	5	NA	NA

Total rows 9

50 Rows per page | 1 of 1

- The Data Spillage dashboard table shows metrics for queries that caused data to spill to disk during execution in each compute warehouse on a specific date
  - Warehouse: Name of the compute warehouse where data spillage was observed.
  - Date: The day the spillage was measured, enabling tracking and comparison over time.
  - Total Spill (GB): The total volume of data (in gigabytes) that spilled to disk across all queries for the warehouse on the selected day.
  - Local Spill (GB): Amount of spill that occurred locally within the warehouse, rather than remote nodes or storage.
  - Remote Spill (GB): Spill activity to remote storage or external disk (here, all shown as zero, meaning only the local disk was used).
  - Remove Spill Pct: Percentage of data spilled that was successfully removed or cleaned up post-computation (zero in this sample).
  - Spill Query Count: Number of queries that caused a spill event.
  - Avg Spill GB per Query: Average amount of data spilled per spilling query; highlights typical impact per query.
  - Max Spill GB per Query: Largest data spill associated with a single query, showing the worst-case outlier.

Insights

- High spill counts or volumes per query may indicate resource limits, inefficient joins, or large sort operations.
- Monitoring spillage helps teams proactively tune queries, resize resources, and prevent slowdowns or failures due to disk usage.
- Warehouses with consistently high spill metrics should be prioritized for optimization or configuration review.

DATA SPILLAGE

WAREHOUSE	DATE	TOTAL SPILL (GB) ↓	LOCAL SPILL (GB)	REMOTE SPILL (GB)	REMOVE SPILL PCT	SPILL QUERY COUNT	AVG SPILL GB PER QUERY	MAX SPILL GB PER QUERY
COMPUTE_WH	Sep 22 2025	18.172967768274	18.172967768274	0	0	13	1.39792059756	6.050121307373
SVC_QA_WH	Sep 22 2025	5.501033782959	5.501033782959	0	0	5	1.100206756592	5.3957862854
FIVETRAN_WAREHOUSE	Sep 22 2025	0.007823944092	0.007823944092	0	0	1	0.007823944092	0.007823944092
SVC_REPORTING_DEV_WH	Sep 22 2025	0.006183624268	0.006183624268	0	0	1	0.006183624268	0.006183624268

Total rows 4

50 Rows per page 1 of 1

- Idle Warehouse Detection shows which compute warehouses have idle compute time where resources were allocated but not actively processing queries. Helps identify underutilized warehouses wasting money, guiding teams to rightsize, pause, or optimize scheduling to reduce cloud costs.
    - Warehouse: Name of the compute cluster evaluated for idle time.
    - Idle Cost (USD): Dollar value attributed to credits consumed when the warehouse was running but not performing queries (idle).
    - Idle Percentage: Proportion of intervals found to be idle, indicating efficiency.
    - Idle Intervals / Idle Credits Used: Number of checked time intervals and credits wasted on idle compute.
    - Intervals Checked: Total periods assessed for warehouse activity.
    - Active Cost / Credits: Cost and credits spent for intervals with active processing.
    - Total Cost (USD) / Total Credits Used: Overall spend and usage across both active and idle intervals.

Insights

- All listed warehouses show zero idle cost and credits, meaning they were either properly scheduled, right-sized, or underutilized and candidates for scale-down or shutdown.
- Regular monitoring prevents unnecessary cloud expenses from running idle resources.

IDLE WAREHOUSE DETECTION

WAREHOUSE	IDLE COST (USD) ↓	IDLE PERCENTAGE	IDLE INTERVALS	IDLE CREDITS USED	INTERVALS CHECKED	ACTIVE COST (USD)	ACTIVE CREDITS USED	TOTAL COST (USD)	TOTAL CREDITS USED
USER_ADHOC	0	0	0	0	3	0	0	0	0
SVC_QA_WH	0	0	0	0	5	0.88	0.44	0.88	0.44
PRIZM_QA	0	0	0	0	1	0	0	0	0
SVC_REPORTING_DEV_WH	0	0	0	0	4	0.21	0.11	0.21	0.11
FIVETRAN_WAREHOUSE	0	0	0	0	1	0.06	0.03	0.06	0.03
PRIZM	0	0	0	0	2	0	0	0	0
COMPUTE_WH	0	0	0	0	288	224.6	112.3	224.6	112.3

Total rows 8

50 Rows per page 1 of 1

- Automatic Clustering Performance for Very Large Tables focuses on very large tables (100 million+ rows) and reports their clustering status, row count, size, and query performance count classifications. Indicates if automatic clustering is enabled for each table, helping optimize query speed for huge datasets.
    - Table Catalog/Schema/Name: Identifies large tables by catalog and schema.
    - Row Count / Table Size (GB): Indicates table scale and resource impact.
    - Clustering Status: Indicates clustering state (e.g., 'No clustering'), signaling if optimization has been applied.
    - Total Queries / Query Categories: Number of queries run, segmented by performance (e.g., very slow, slow, medium, fast).
    - Fast Queries: Count of queries deemed fast among total operations.

Insights

- None of the sampled very large tables use clustering, providing an immediate target for storage and query optimization efforts.
- Tables with high 'fast queries' counts, despite their size, offer learnings for schema or workload tuning

AUTOMATIC CLUSTERING PERFORMANCE - VERY LARGE TABLES (100M+ ROWS)

TABLE CATALOG	TABLE SCHEMA	TABLE NAME	ROW COUNT ↓	TABLE SIZE (GB)	CLUSTERING STATUS	TOTAL QUERIES	VERY SLOW QUERIES	SLOW QUERIES	MEDIUM QUERIES	FAST QUERIES	PI
DQLABS_PROSPECTS	BANKING_DEMO	CUSTOMER_ONE_BILLION	1440033112	77.24	No clustering	33	0	0	0	33	
DQLABS	CARMAX	COLGATE_TESTING_1B	1000000000	24.56	No clustering	7	0	0	0	7	
DQLABS	CARMAX	COLGATE_TESTING_500M	500000000	12.3	No clustering	199	0	0	0	199	

Total rows 3

50 Rows per page 1 of 1

Insights

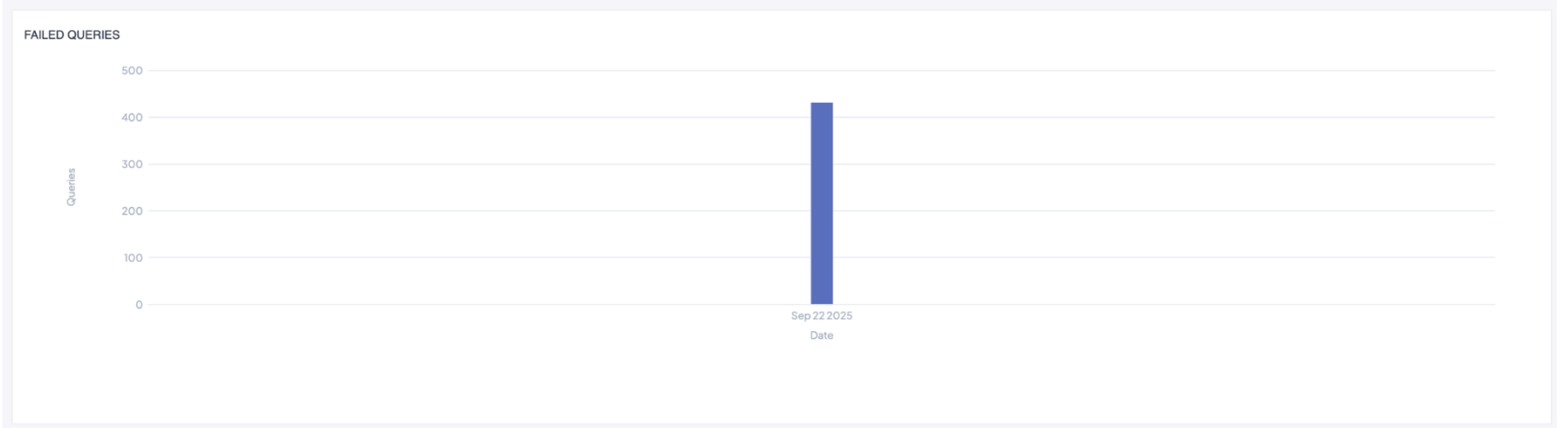
The insights tab provides details on the query volume, failed queries, and data spillage

- The "Query Volume Trend Per Day" chart in the Insights tab displays the total count of queries executed each day in your data warehouse environment.
- Visualizes daily query workload as a bar chart, with the x-axis representing dates and the y-axis indicating the number of executed queries.

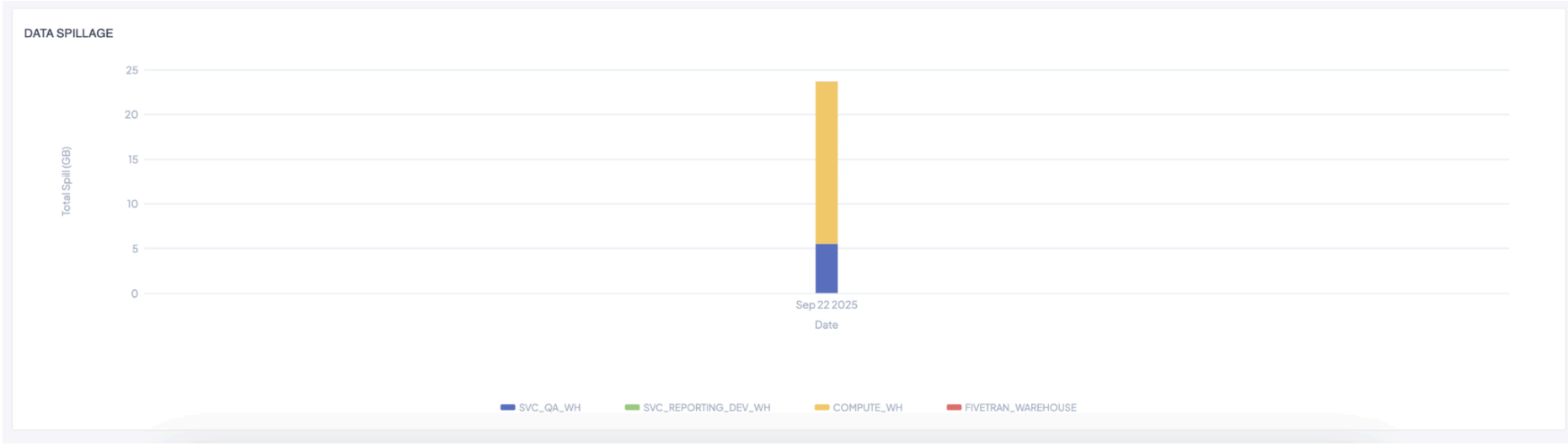
Key Insights

- Large spikes may signal batch ETL jobs, end-of-month reporting, or sudden user access increases.
- Sustained high volume can indicate increasing adoption, while sudden drops may highlight outages or misconfiguration.
- Regular tracking of query trends aids capacity planning and ensures infrastructure is aligned to real business needs

- Failed Queries displays the total number of queries that failed to execute on a given day.
- Visualized as a bar chart, this lets teams quickly spot error spikes, which may indicate infrastructure issues, broken pipelines, or permission problems.
- Tracking failed queries helps prioritize troubleshooting and ensures robust system health.



- Data Spillage shows the total volume (in GB) of data that spilled to disk during query execution for the day by warehouse
- Data spill typically occurs when datasets exceed memory limits, impacting query speed and resource usage.
- Monitoring spillage enables teams to target query tuning, optimize resource allocation, and prevent future performance bottlenecks.

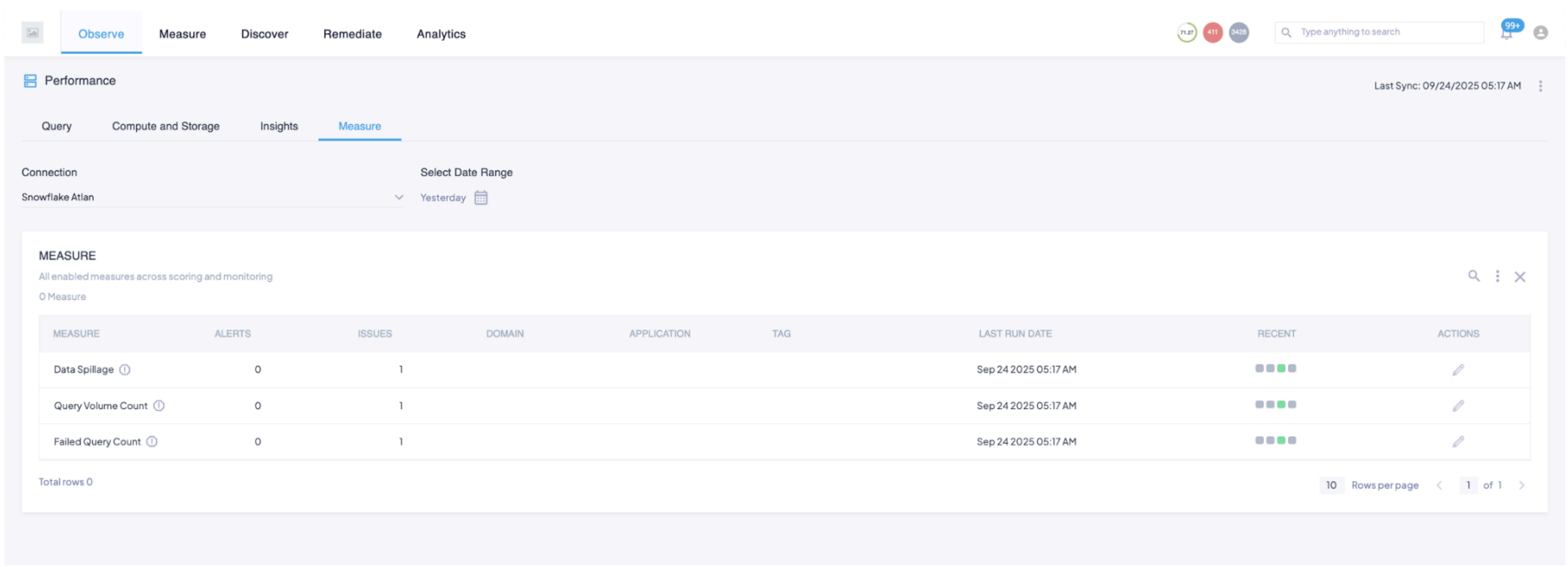


Measure

The Measure tab summarizes the status of enabled cost monitoring measures applied to the connection.The Measure Table shows a list of custom-defined cost measures enabled for monitoring and scoring within the data platform. It displays:

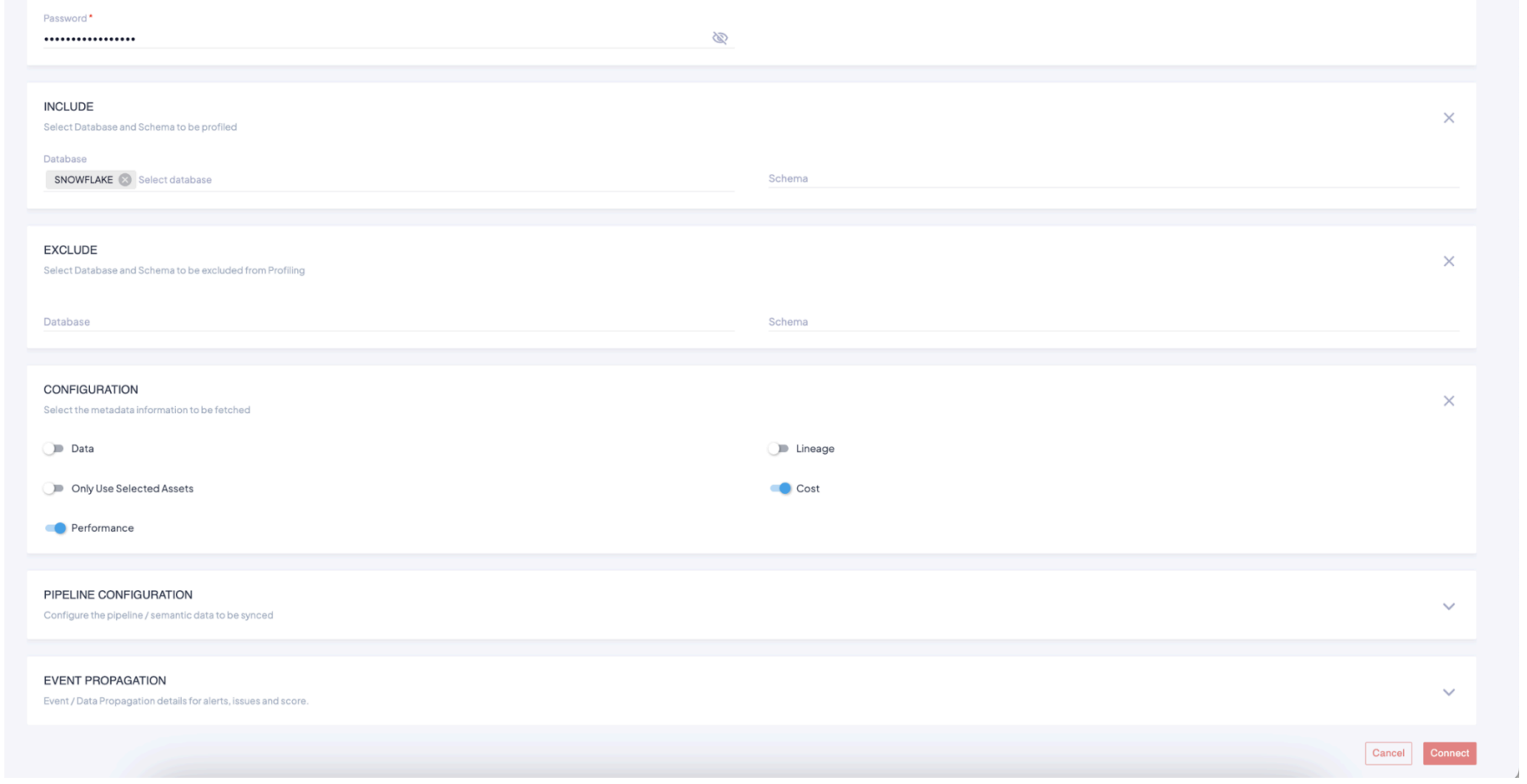
- The name of each measure.
- The number of alerts triggered for that measure.
- The count of associated issues.
- The timestamp of the last time the measure was run.
- A visual recent status indicator of the measure's health or activity.
- An action button to edit the measure settings.
- The admin/privileged user will able to able to select other applicable columns for the select columns list

This table helps users track key financial KPIs, see any anomalies or alerts related to cost spikes or trends, and maintain oversight over cost measure health across their data environment. It supports proactive optimization and governance of compute and storage expenses.



The admin/privileged user can enable the following measures from the connection configuration page by selecting the cost toggle.

- Data Spillage
- Query Volume Count
- Failed Query Count



16. Salesforce Data Cloud

Salesforce Data Cloud (previously known as Salesforce CDP – Customer Data Platform) is Salesforce's platform for unifying, managing, and activating customer data across multiple systems in real time. It's designed to help organizations create a single, 360-degree view of each customer to enable personalized marketing, sales, and service experiences. DQLabs allows users to connect to Salesforce data cloud and bring in data lake objects, data models, and transformations into DQLabs for observability

Pre-requisites

Whitelisting

If your organization uses a whitelist to manage Salesforce Data Cloud access, reach out to customersupport@dqlabs.ai to set up the whitelisting.

Account Access

To create a Client ID (Consumer Key) and Client Secret (Consumer Secret) for Salesforce Data Cloud, follow these steps:

- Log in to Salesforce: Access your Salesforce instance with an administrator account.
- Navigate to Setup: Click the gear icon in the top right corner and select "Setup."
- Go to App Manager: In the Quick Find box, type "App Manager" and select it under "Apps."
- Create a New Connected App: Click "New Connected App" in the App Manager.
- Configure Basic Information:
  - Provide a "Connected App Name," "API Name," and "Contact Email."
- Enable OAuth Settings:
  - Under the "API (Enable OAuth Settings)" section, check "Enable OAuth Settings."
  - Specify a "Callback URL." This URL is where the user's browser is redirected after successful authorization.

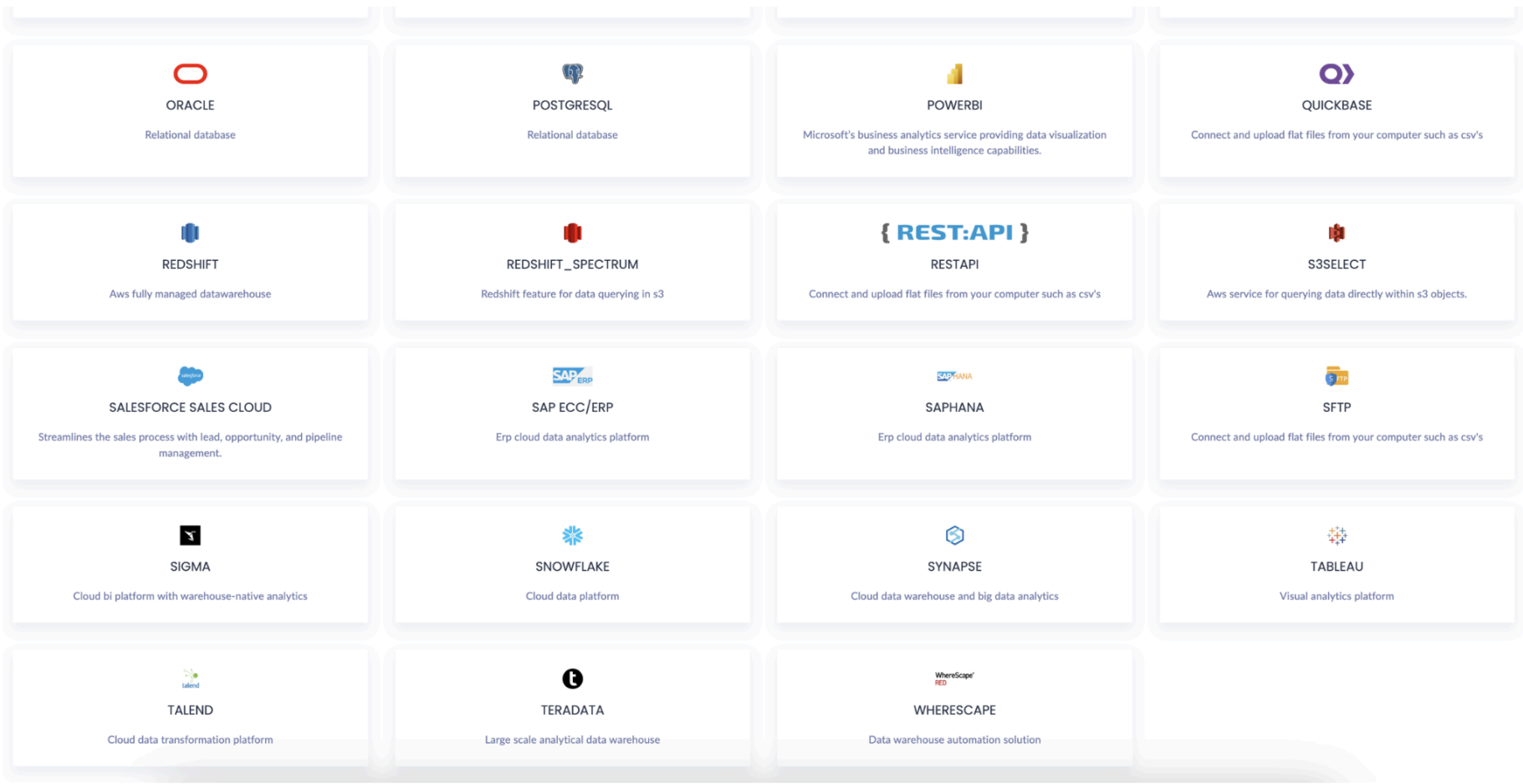
- Select the necessary "OAuth Scopes." For Data Cloud integration, you will likely need scopes such as "Access and manage your data (api)" and "Perform requests on your behalf at any time (refresh_token, offline_access)." You might also need Data Cloud-specific scopes, depending on your integration needs (e.g., cdp_query_api, cdp_profile_api).

- Save the Connected App: Click "Save."
- Obtain Consumer Key and Secret:
  - After saving, you will be redirected to the Connected App detail page.
  - Click "Manage Consumer Details" to reveal the "Consumer Key" (Client ID) and "Consumer Secret" (Client Secret).
- Important: Copy these values and store them securely, as the Consumer Secret will only be displayed once.
  - These credentials (Consumer Key and Consumer Secret) can then be used to authenticate with Salesforce and subsequently with Salesforce Data Cloud APIs using OAuth 2.0 flows.
  - Provide permission for the client credential flow in the settings.
  - For tables, every data lake object needs to run needs to be added to any one data space; if it is not mapped to a data space, queries will not run, so it must map to a data space

Connect to Salesforce Data Cloud

Follow the steps below to connect to Salesforce Data Cloud:

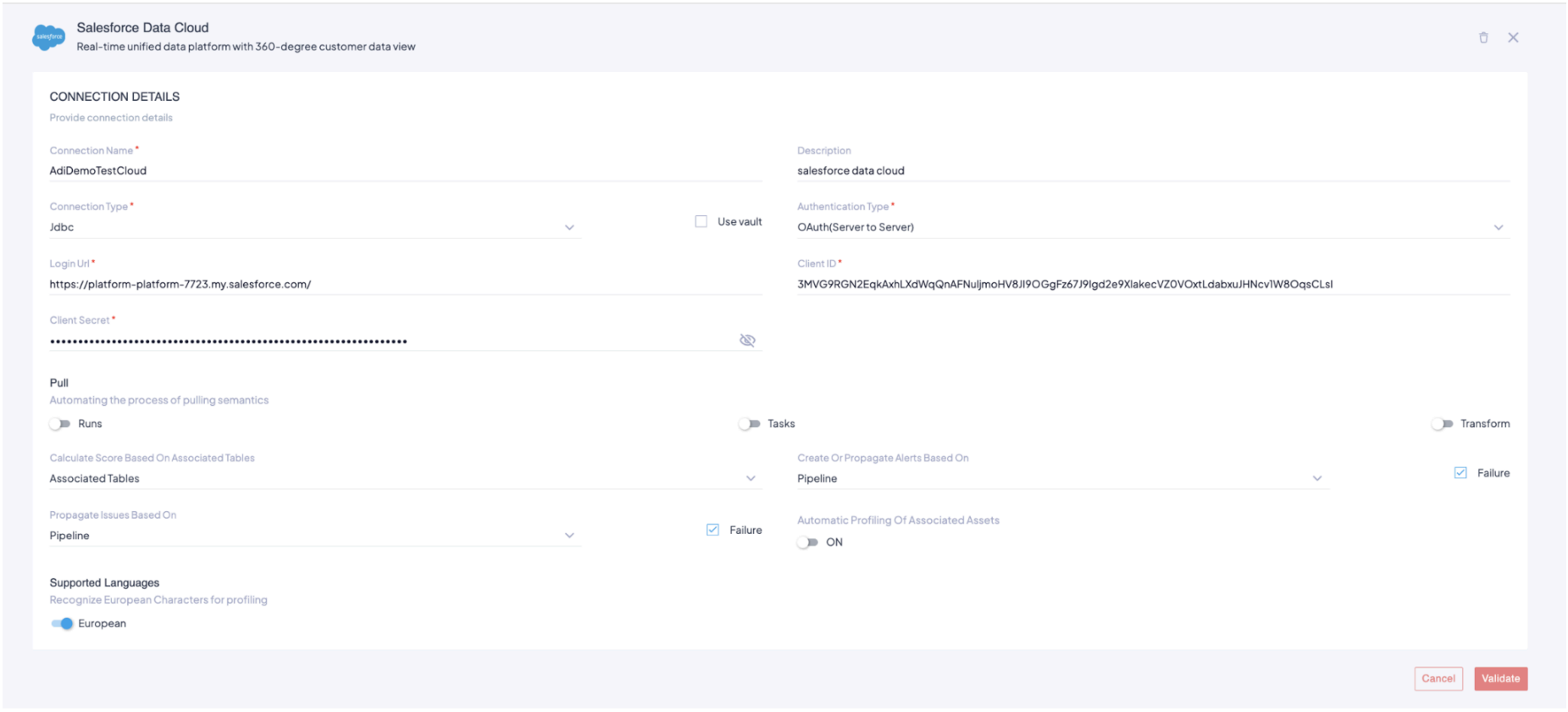
Step 1: Navigate to Settings → Connect → Source
Step 2: Click on the "+" icon



Step 3: Click on Salesforce data cloud and provide the following details:

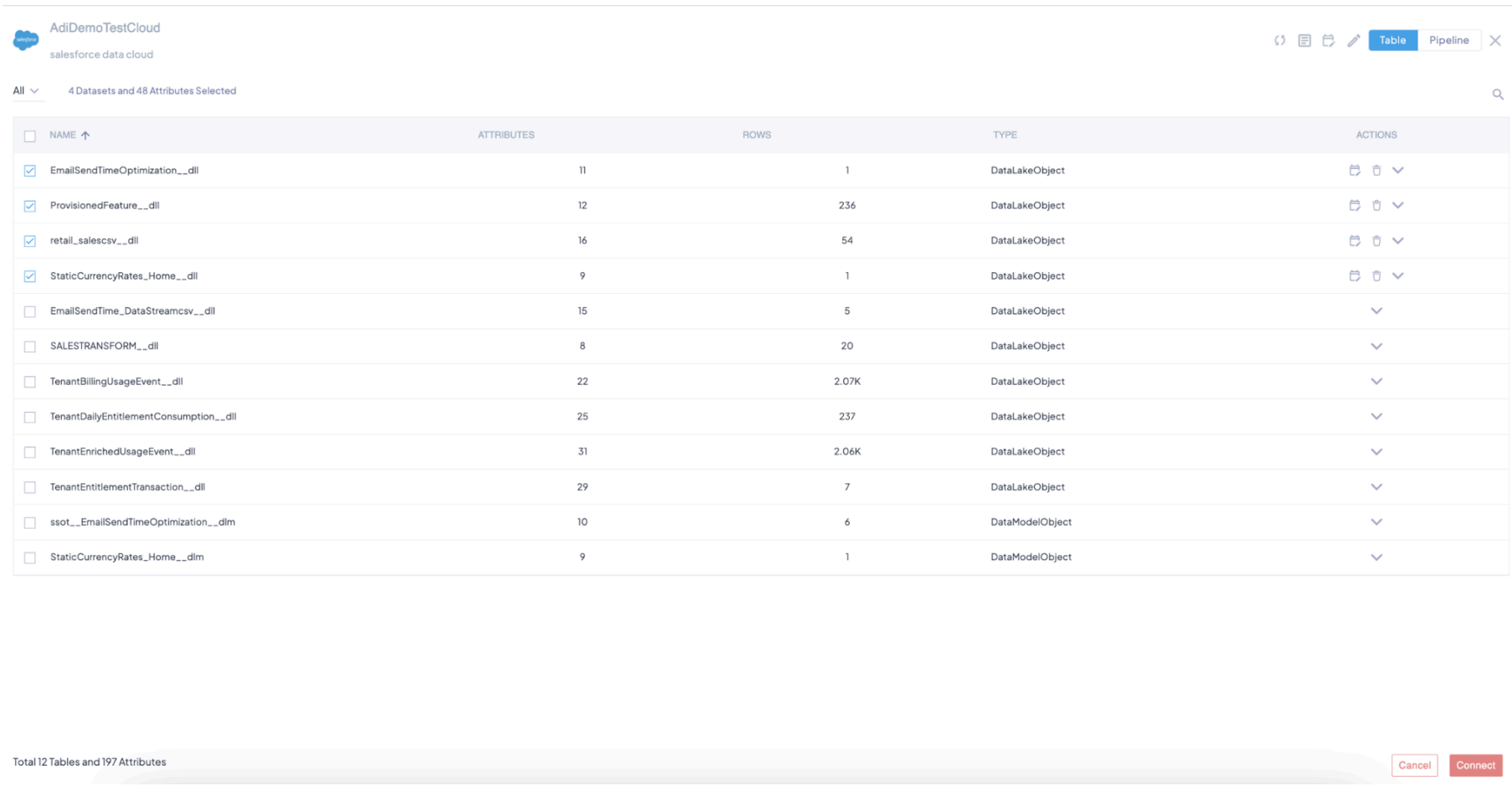
Field / Option	Description
Connection Name*	A required name for identifying your connection (e.g., AdiDemoTestCloud)
Connection Type*	Type of connection protocol (e.g., Jdbc)
Description	Optional text describing the connection purpose or context (e.g., Salesforce data cloud)
Login Url*	URL to authenticate against the Salesforce platform
Authentication Type*	Specifies the OAuth type used (e.g., OAuth(Server to Server))
Client ID*	OAuth client ID provided for authentication
Client Secret	Secret key/password used for authentication (hidden by default)
Use Vault	Option to store and retrieve sensitive credentials securely via a vault
Pull - Runs	Toggle to automate pulling pipeline runs
Pull - Tasks	Toggle to automate pulling pipeline tasks
Pull - Transform	Toggle to automate pulling pipeline transformations
Calculate Score Based On	Select what scoring calculation is based on

Propagate Issues Based On	Select on which criteria issues should propagate
Create or Propagate Alerts Based On	Select on which criteria alerts should be created
Failure Checkbox	When checked, failures will be tracked and propagated as issues
Automatic Profiling Of Associated Assets	Toggle to enable or disable automatic data profiling of linked assets
Supported Languages	Option to recognize specific character sets, e.g., European, for profiling



Step 4: Once connected, the user will be able to view the list of transformations in Salesforce Data Cloud

Step 5: Select the required transformations and click on connect.



Once connected, the admin/privileged user will be able to select Table and Pipelines. Tables include the following objects in Salesforce data cloud:

- Data lake objects
- Data Models

Pipelines include the following objects:

- Data Transformations

The user can select the objects and click on connect, and the user will be redirected to the asset detail page

Limitations

- Export failed rows for different connectors is only supported (MSSQL and Snowflake)
- Short pattern and long pattern measures are not supported

- Behavioural Measures and Look-up Measure are not support
- Need to configure data models and data lake objects for auto-mapping.

---

17. Salesforce Marketing Cloud

Salesforce Marketing Cloud is Salesforce's digital marketing automation platform. It helps organizations manage and personalize customer interactions across multiple channels — like email, SMS, social media, web, advertising, and apps — to drive engagement, retention, and growth. DQLabs allows the users to connect to salesforce markeing cloud and bring in both data assets and pipeline assets for observability

Pre-requisites

Whitelisting

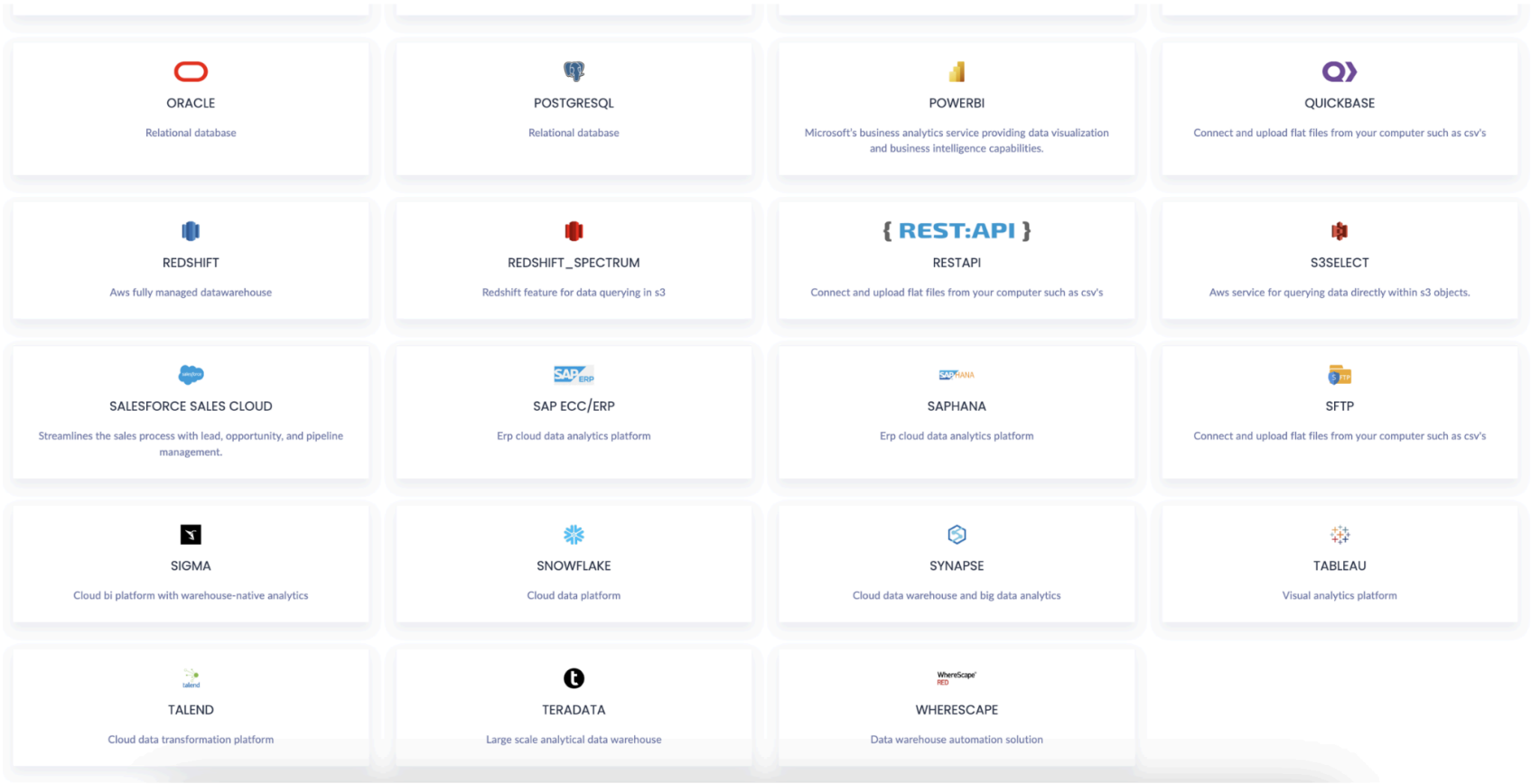
If your organization uses a whitelist to manage Salesforce Marketing Cloud access, reach out to customersupport@dqlabs.ai to set up the whitelisting.

Connect to Salesforce Marketing Pipelines

Follow the steps below to connect to Salesforce marketing pipelines:

Step 1: Navigate to Settings → Connect → Source

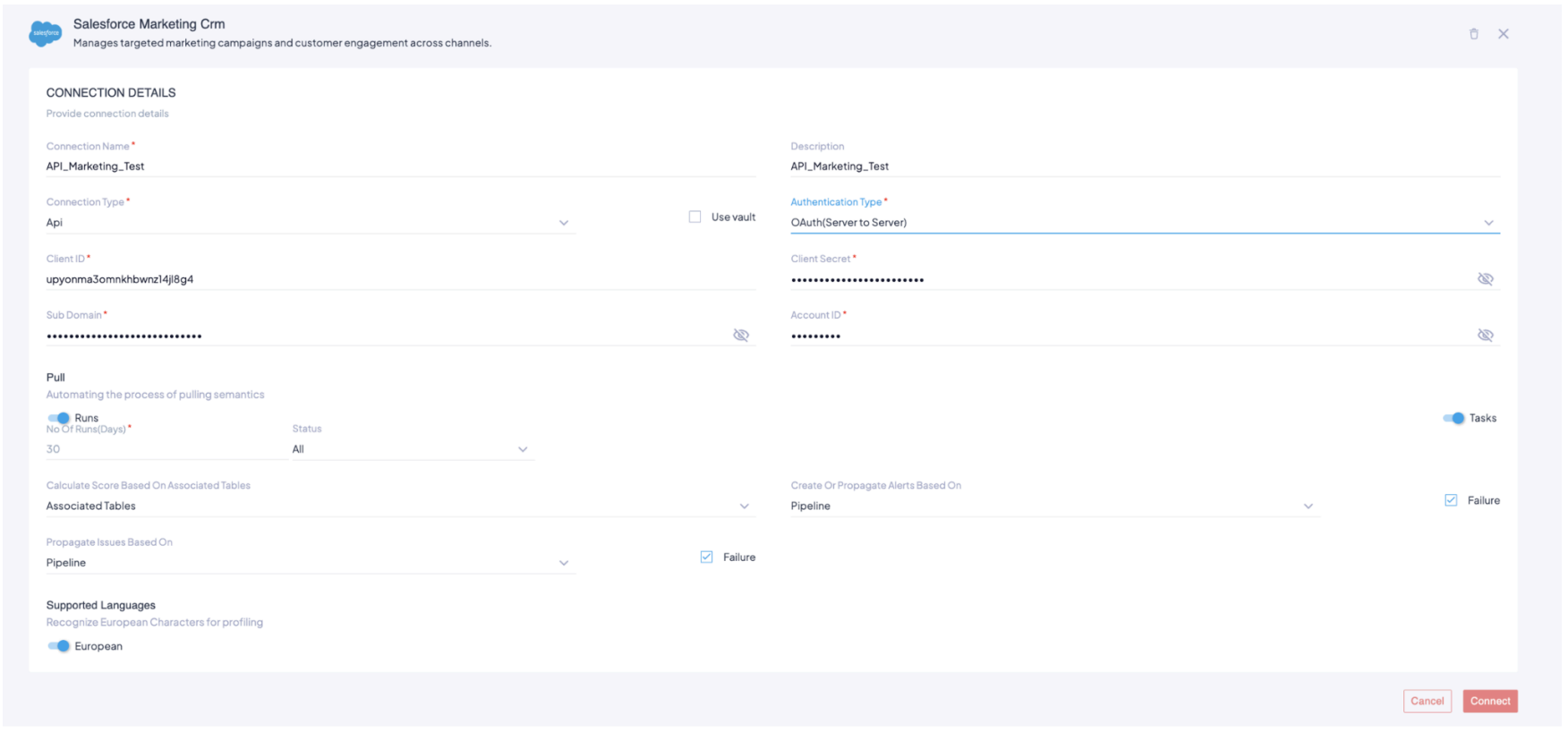
Step 2: Click on the "+" icon



Step 3: Select "Salesforce Marketing CRM" and provide the following details

Field / Option	Description
Connection Name*	A required name for identifying your connection (e.g., API_Marketing_Test)
Connection Type*	Specifies the protocol or service type , Select API for Pipeline Assets and JDBC for Data Assets
Description	Optional description about the connection (e.g., API_Marketing_Test)
Client ID*	Required OAuth client identifier for authenticating API calls
Sub Domain*	The Salesforce sub-domain value, typically a masked/secret input
Authentication Type*	Specifies the authentication protocol (e.g., OAuth(Server to Server))
Client Secret*	Secret key or password for authorization
Account ID*	Required account identifier for API operations (
Use Vault	Option to store sensitive credentials securely in a vault
Pull – Runs	Toggle ON to automate the collection of pipeline run data
Pull – Tasks	Toggle ON to automate the collection of pipeline task data

No Of Runs (Days)*	Specify how many days' worth of pipeline run history to pull (e.g., 30)
Status	Specify which run status to include
Calculate Score Based On	Select basis for scoring calculation
Propagate Issues Based On	Choose the source for propagating issues
Failure Checkbox	When enabled, failures are tracked and propagated as operational issues
Create or Propagate Alerts Based On	Determine how alerts are created or propagated (e.g., Pipeline)
Supported Languages	Recognize and process specific character sets, such as European, for profiling



Step 4: Once connected, the user will be able to view the list of pipelines in Salesforce Marketing

Step 5: Select the required pipelines and click on connect.

Once connected, the admin/privileged user will be redirected to the asset detail page. The following objects are mapped in Salesforce marketing pipelines

- Jobs - Automations in Salesforce marketing pipelines
- Task - Steps in Salesforce marketing pipelines
- Runs - Runs for the jobs

Limitations

- Issue and Alert propagation are not supported
- Automapping not supported

Improvements

This section contains the list of improvements that are identified as part of 3.1.3

1. Recommend AI Enhancements

The recommended AI feature has been enhanced to recommend/add descriptions to the following sections.

Asset Level

1. Recommend a description for an asset
2. Recommend a description for all the measures under an asset
3. Recommend a description for all the active measures under an asset
4. Recommend a description for all the inactive measures under an asset

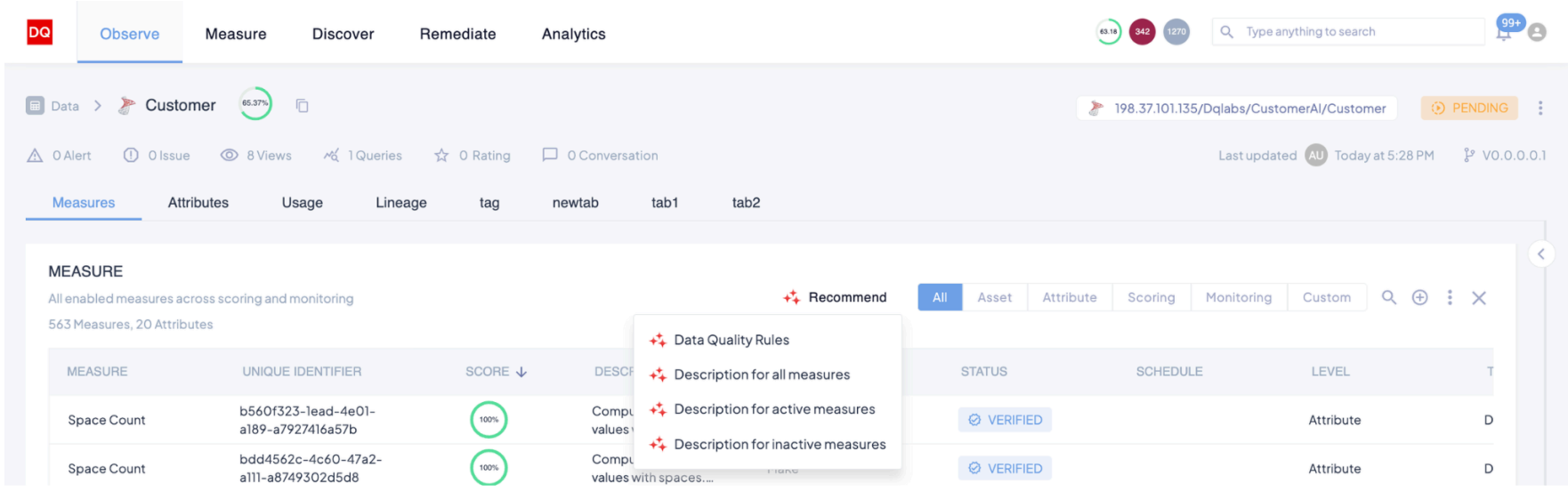
Attribute Level

1. Recommend a description for an attribute
2. Recommend a description for all the measures
3. Recommend a description for all the active measures
4. Recommend a description for all the inactive measures

Semantics

1. Recommend a description for a semantic term
2. Recommend a description for all the semantic terms
3. Recommend a description for all the active terms
4. Recommend a description for all the inactive terms

The admin/privileged user will be able to use the button under the recommend tab to create descriptions using AI for the related objects.



2. File Connector - On-prem Support

DQLabs now supports connecting to files stored in local storage, allowing users to configure them as assets within the platform. Once configured, out-of-the-box (OOB) measures can be applied, and profiling can be executed on these assets. Users can set up external storage by navigating to Settings → Configuration → External Storage. Local storage can now be designated as an external storage option, where files uploaded through the UI are saved and subsequently used for data processing.

Query building standardization

The new **Query Editor** replaces the traditional query measure build interface, providing a more intuitive and efficient way to write and validate SQL queries. This enhancement significantly reduces the effort required to identify and reference database elements like table names, column names, and schemas, while also offering real-time syntax validation.

Navigating to the Query Editor

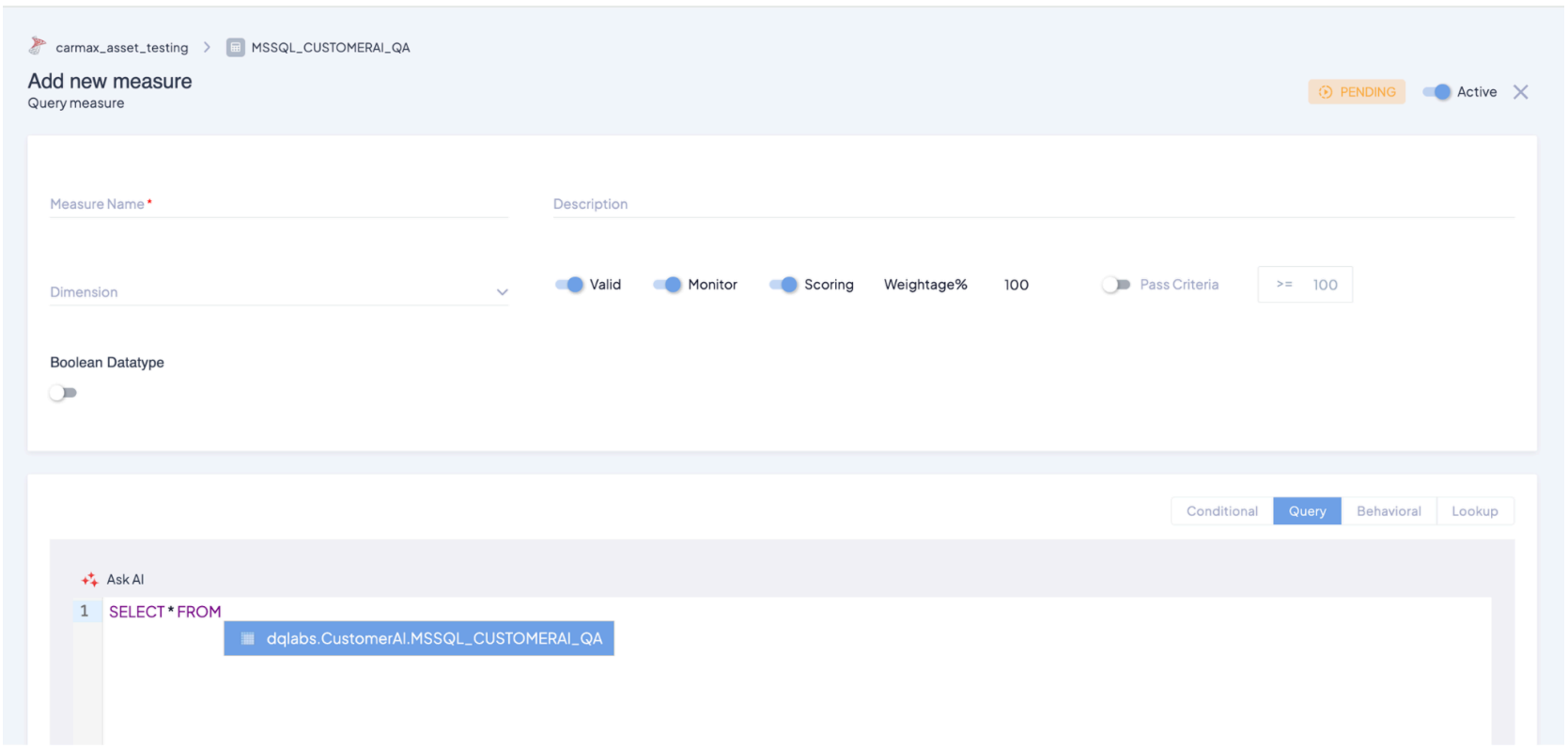
- Go to the **Measures** section in the DQLabs Portal.
- Select **"Create New Measure"** or open an existing **Query Measure**.
- The interface now opens a full **Query Editor** view instead of the basic query input box.

Writing Queries with Assistance

- Begin typing your SQL query.
- The editor will **auto-suggest**:
  - **Database names**
  - **Schema and table names**
  - **Column names**
- This helps you quickly reference the correct structures without needing to search manually.

Syntax Validation

- As you write, the editor automatically checks for:
  - **Syntax errors**
  - **Incomplete clauses**
  - **Proper SQL structure**
- Errors are **highlighted inline**, making them easier to locate and fix before running the query.



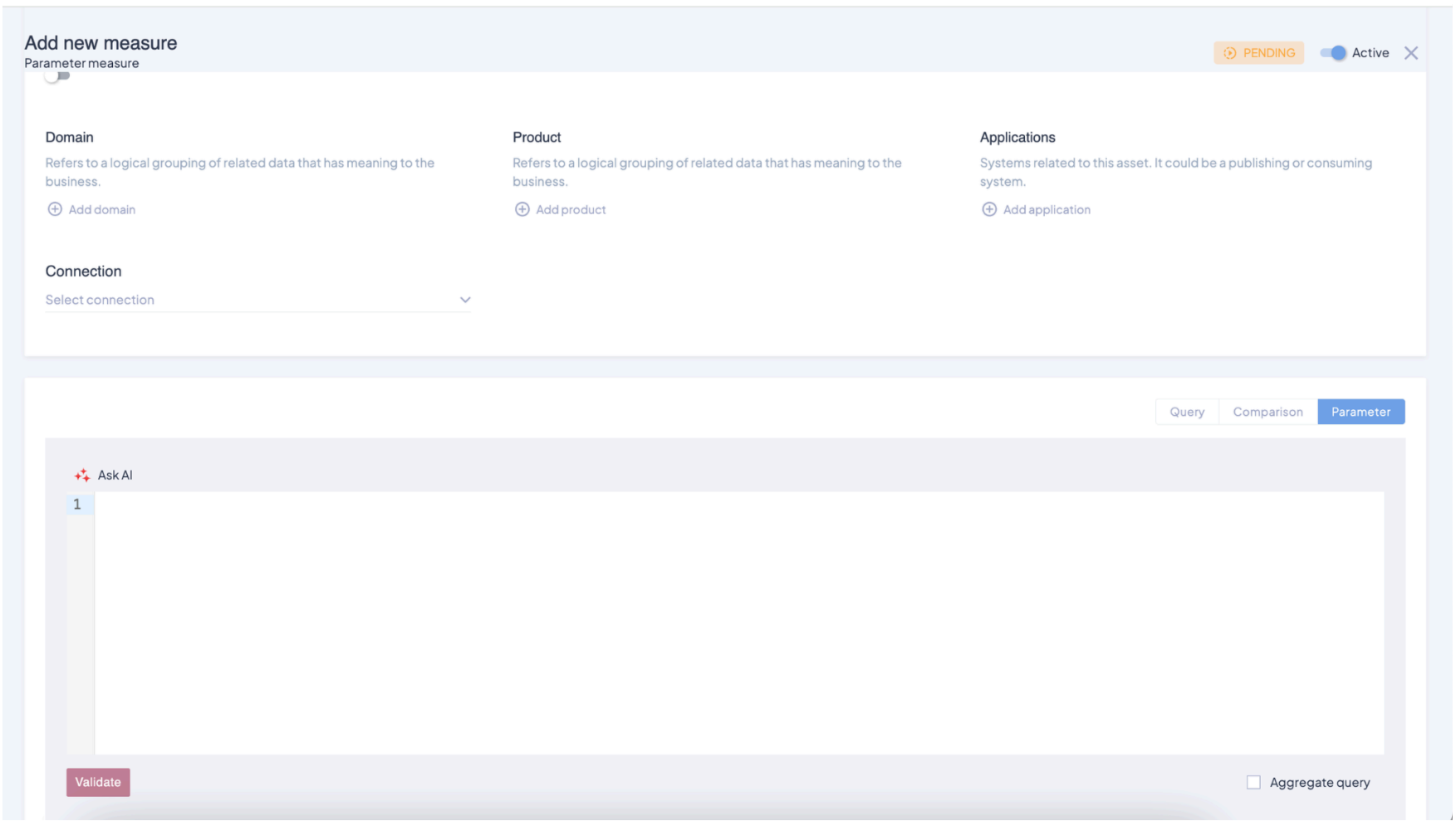
---

3. Pass Input Parameter for Query Measure Execution through API and CLI

This feature introduces the ability to run custom query-based measures with dynamic input parameters through both the API and the Command Line Interface (CLI). It enables users to reuse the same measure logic with different runtime inputs, enhancing flexibility and automation.

Follow the steps below to create a parameterized measure:

Step 1: Navigate to Measures → Add New Measure → Parameter Tab



Step 2: Define your SQL query with a parameter placeholder.

SELECT * FROM job_run_status WHERE PROCESS_ID = <PROCESS_ID>

Step 3: Save the Measure

Step 4: Click on the run button to provide the values and run from the UI

Step 5: To run via API, include the measure ID and parameter values in the request payload for the measure run API

```
{
 "measure_id": "MEASURE_12345",
 "parameters": {
  "PROCESS_ID": "XYZ001"
 }
}
```

- The execution result will include:
  - The **parameter values used**
  - The **status and output** of the query
- These will be displayed under the **"Results"** section.

---

4. Issue Page - Enhancements

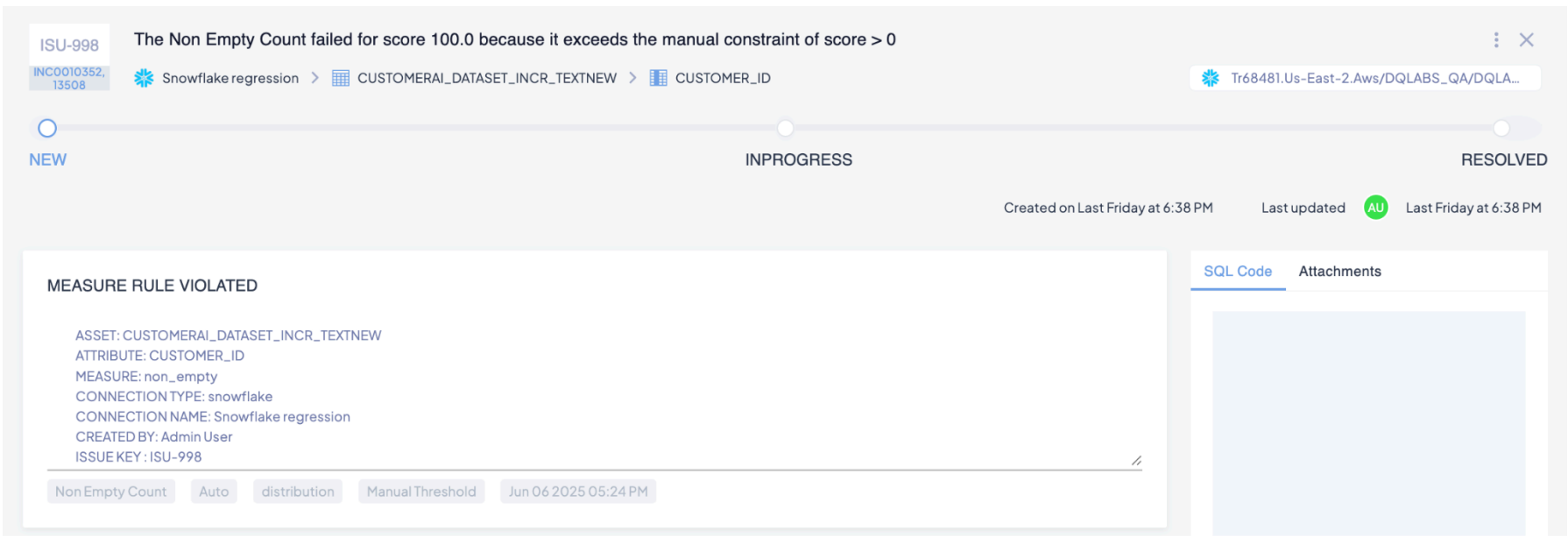
The Issue Page has been enhanced to provide deeper insights and better contextual information to help users quickly understand, analyze, and resolve issues related to assets.This update enriches the Issue Page with detailed information at the issue level, along with visibility into all impacted downstream assets. The goal is to enable users to:

- Grasp what failed at a glance
- Understand why it failed (root cause analysis)
- Identify the impacted downstream assets
- Take informed action to resolve the issue

The following are the details that have been newly added:

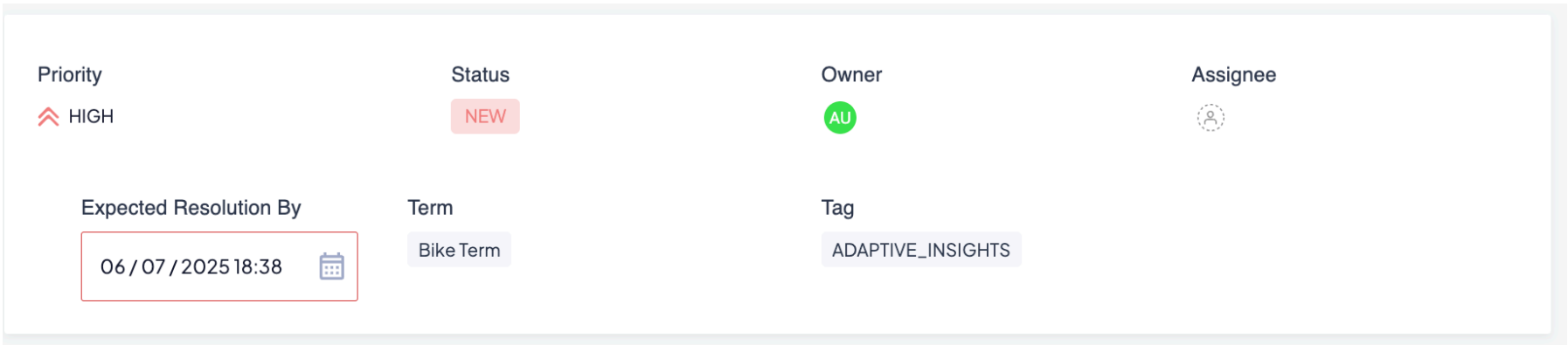
Measure Details

- The admin/privileged user will be able to view the following details in the measure details section
  - Measure Name
  - Measure Type
  - Measure Category
  - Threshold Type
  - Last Failure



Semantic Layer Definitions

- The admin/privileged user will now be able to view semantic layer definitions, such as terms and tags for an issue that has been added to the asset



---

5. Enhancement of /api/system_status/ API to include Platform Version

Previously, the /api/system_status/ API only provided information about the server status. It has now been enhanced to also include the platform version and the last updated date of the build for the specific environment. The following shows the response structure of the updated API:

```
{
  "message": "Server is up and running",
  "success": true,
  "version": {
    "release": "3.1.2",
    "last_updated": "2025-08-28 09:25:03.217809"
  }
}
```

---

6. Enhanced Connection Logs Download in DQLabs

When downloading connection logs from the application, users can now access detailed error information directly in the exported file. This enhancement ensures that error messages—previously visible only in the UI—are included in the download for easier analysis and troubleshooting.

Benefits:

- Provides a complete record of both successful and failed connections in a single extract.
- Enables offline analysis and troubleshooting without relying solely on the UI.
- Improves auditability and reporting by including error context in downloaded logs.

Error Message Column:

- A new column named “Error Message” has been added to the exported logs.
- Displays the same error messages shown in the UI for failed connections.
- For successful connections, the column is either left blank or shows “N/A”.
- Values in the downloaded file are identical to those in the UI, ensuring consistency.

---

7. Exclusion of Conversations in Retention Period

The Retention Period feature in DQLabs automatically manages the lifecycle of asset metadata by deleting old metadata once the configured number of days has passed.

Previously, this deletion applied to all metadata, including critical items like Conversations, which created challenges for customers—especially in Catalog—where some metadata needs to be preserved indefinitely.

With this enhancement, Conversations are now excluded from the retention period and will remain until users choose to delete them manually.

---

8. Enhancement: Export Metadata

The Push Down Metrics feature allows customers to export metadata into Snowflake for reporting and analysis.

Previously, using the Run Now button exported all metadata types automatically, including:

- ASSET_METADATA
- ATTRIBUTE_METADATA

- MEASURE_METADATA
- CONNECTION_METADATA
- USER_ACTIVITY
- USER_SESSION

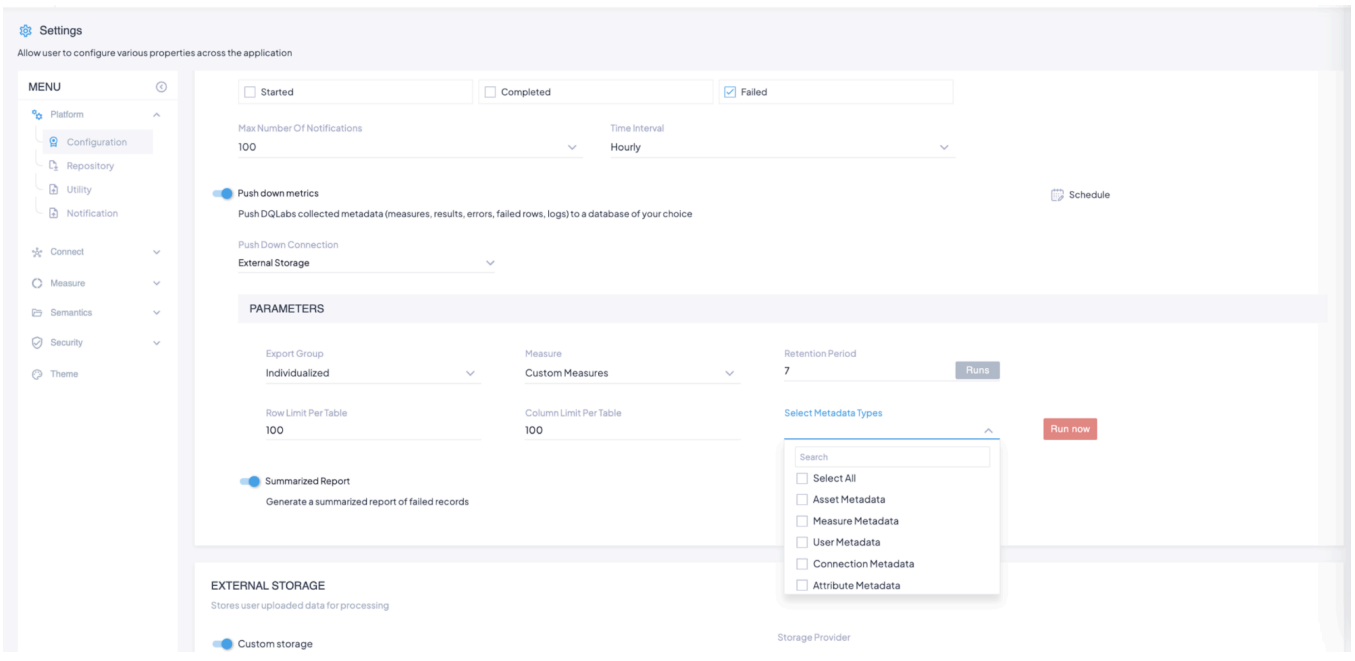
This often resulted in long runtimes and high compute costs, especially when only a subset of metadata was needed.

Enhancement: Customers can now select which metadata types to export, reducing execution time and cost.

Steps to configure export metadata:

1. Navigate to Settings → Configuration → Push Down Metrics.

2. In the Select Metadata Types dropdown, check the boxes for the metadata types you want to export (e.g., ASSET_METADATA, USER_ACTIVITY).
3. Leave unneeded metadata types unchecked.
4. Trigger the job—the export will now run only for the selected metadata types.



---

9. Selected/Unselected Assets as a Filter in Connection Asset List Page

Previously, the Selected/Unselected option on the Connection List page acted as a sort order:

- Selected → Displayed selected assets first, followed by unselected ones.
- Unselected → Displayed unselected assets first, followed by selected ones.

This caused issues when users applied additional sorting (e.g., by Asset Name, Schema Name, or Project Name for Tableau connections). The secondary sort ignored the Selected/Unselected order, sometimes showing selected items even when Show Unselected was chosen.

Enhancement:

The Selected/Unselected option now functions as a filter rather than a sort. Users will only see assets that match the filter (either Selected or Unselected) and can freely sort the filtered results by any column.

Steps to Use:

1. Navigate to the Connection List Page.
2. Open a connection to view associated assets.
3. Apply the Selected/Unselected Filter:
    a. Show Selected → Displays only selected assets.
    b. Show Unselected → Displays only unselected assets.
4. Sort the filtered results by:
    a. Asset Name
    b. Schema Name
    c. Project Name (for Tableau connections)

Best Practices:

- Use Show Selected to quickly review curated or approved assets.
- Use Show Unselected to identify potential assets for onboarding or validation.
- Apply sorting after filtering to refine searches and improve navigation across large asset lists.

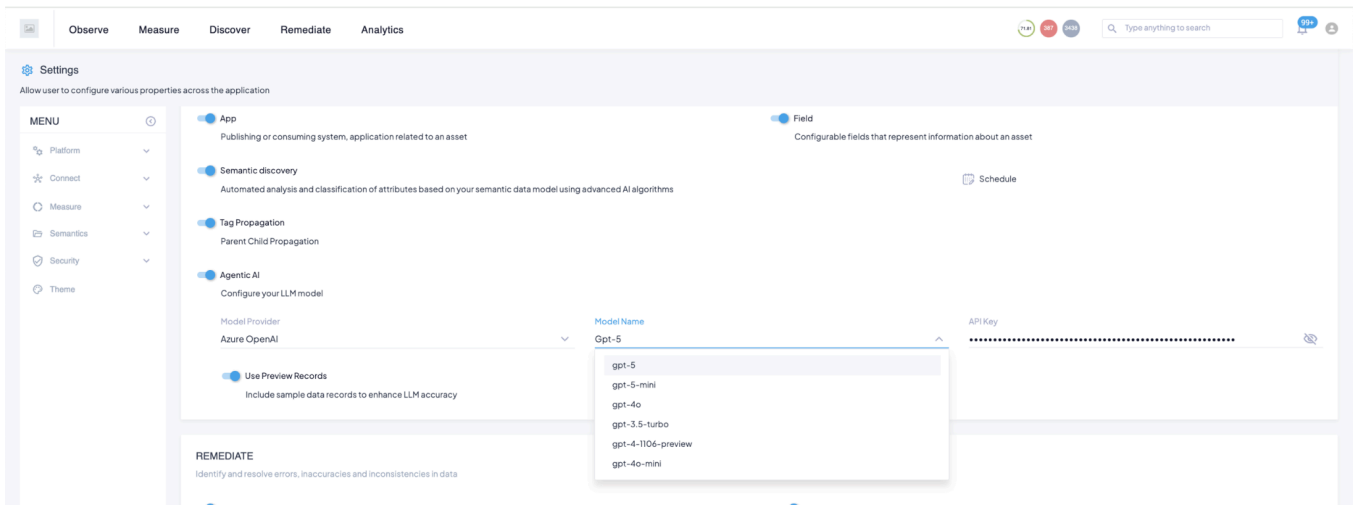
9. Converse Module Enhancements

The Converse module has been upgraded to deliver greater flexibility and smarter measure generation.

- Advanced LLM options – Administrators can now select from advanced LLM models to power Converse.
- Smarter measure generation – Asset and attribute descriptions are now leveraged to provide more accurate and context-aware recommendations.

1. Ability to Choose Advanced LLM Models

- Admin and privileged users can now select Azure AI LLM models to power Converse AI in DQLabs.
- From the Settings page, administrators can:
  - View all available LLM models supported in the platform.
  - Select the preferred model for Converse interactions.



- This flexibility enables organizations to choose the best-fit model based on accuracy, performance, or cost considerations.

2. Generate Measures Using Metadata Context

- When generating measures through Converse, the system now automatically incorporates:
  - Asset descriptions
  - Attribute/column descriptions
- This ensures recommendations are smarter, more accurate, and aligned with business meaning, reducing manual effort and improving relevance.