2024

Data Observability

COMPARATIVE STUDY REPORT

WALID BIROUK

Table of Contents

Abstract

The ever-increasing demand for data has significantly complicated the architecture and management of modern data pipelines. Enterprises now require robust data observability to ensure data quality, reliability, and trustworthiness across diverse and rapidly evolving data sources. This document presents a comparative study on data observability tools, detailing their evolution, significance, and the necessity for robust data governance in the modern data landscape.

The study begins by examining the historical context and the rising complexity of data pipelines over the past decade. It identifies the challenges faced by data teams, such as data quality issues, infrastructure monitoring, and budgetary constraints. A thorough internal requirement gathering process reveals specific needs and pain points, guiding the selection criteria for suitable data observability tools.

Key tools evaluated in this study include OpenMetadata, DataHub, Sifflet, Anomalo, Metaplane, Databand, Monte Carlo, Elementary, Soda, and Great Expectations. Each tool's features, strengths, and weaknesses are analysed, with a particular focus on OpenMetadata due to its comprehensive capabilities and cost-effectiveness.

A Proof of Concept (POC) was implemented to demonstrate the practical application of OpenMetadata within a sandbox environment. This setup incorporated Airflow, Airbyte, dbt, Spark, Hive Metastore, S3, and PostgreSQL to create a fully functional data pipeline. The POC aimed to achieve comprehensive data monitoring, real-time alerts, and improved data governance through a data catalog and glossary.

Results from the POC highlight the effectiveness of OpenMetadata in ensuring data quality, lineage, schema integrity, and timely alerts for data issues. The study concludes with recommendations for selecting data observability tools and outlines future work to enhance data observability and governance practices, such as deploying in Kubernetes, integrating with external systems, and implementing advanced security measures.

Overall, this document underscores the critical role of data observability in managing complex data pipelines and provides valuable insights for enterprises looking to enhance their data management strategies.

1 Introduction

The relentless increase in data supply and demand has significantly complicated the architecture and management of modern data pipelines. Enterprises are now more reliant than ever on vast amounts of data from a wide array of sources, often in real-time, to drive decision-making and optimize operations. However, this surge in data volume and diversity has presented substantial challenges for data teams tasked with ensuring that data systems can keep pace with these demands.

Despite advancements in tools and platforms and increased investments in engineering and operations, many enterprise data teams continue to grapple with daily operational issues. The integration of massive data volumes, the complexity of data pipelines, and the adoption of new technologies have further strained the capabilities of these teams, diminishing the overall business value of data systems.

Data observability has emerged as a vital strategy to address these challenges. Building on the foundations of application performance monitoring (APM), data observability offers a systematic approach to monitoring and correlating data events across application, data, and infrastructure layers. This holistic approach empowers business owners, DevOps engineers, data architects, data engineers, and site reliability engineers to detect, predict, prevent, and resolve issues—sometimes automatically—that could disrupt production analytics and artificial intelligence (AI) workflows.

The purpose of this report is to provide a comprehensive comparative study of data observability tools, emphasizing their importance in the contemporary data landscape. It explores the evolution of data pipelines, the increasing complexity of managing them, and the essential role of data observability in ensuring robust data governance. Through an internal requirement gathering process, the report identifies key challenges and evaluates various data observability tools, culminating in a detailed examination of OpenMetadata and its implementation in a proof of concept (POC) environment.

The POC demonstrates the practical application of OpenMetadata within a Dockerized modern data stack sandbox, incorporating tools such as Airflow, Airbyte, dbt, Spark, Hive Metastore, S3, and PostgreSQL. The objectives include comprehensive data monitoring, real-time alerts, and enhanced data governance through a data catalog and glossary. The report concludes with recommendations for selecting data observability tools and outlines future work to further enhance data observability and governance practices.

1.1 Purpose and Scope

The purpose of this document is to provide a comparative study of data observability, highlighting its significance in the modern data landscape, the evolution of related tools, and the necessity of data observability in achieving data governance. The scope includes an overview of available tools, their classification and selection criteria, and a detailed case study on OpenMetadata.

1.2 Evolution of Data Pipelines

For years, enterprise data pipelines served rigorous but relatively stable requirements for business analytics. Small teams of business intelligence (BI) analysts needed periodic historical measures of their sales pipeline, financial position, inventory levels, and other functional metrics. They relied on data engineers to build basic data pipelines using a handful of data extraction, transformation, and loading (ETL) and change data capture (CDC) tools to ingest structured data from databases and applications. These tools transformed the data and loaded it into data warehouses in batch jobs, usually overnight, allowing analysts to generate dashboards and reports using conventional BI software. [1] [2] [3]

1.3 Growing Complexity

About a decade ago, the supply and demand of data exploded, leading to a steady increase in the complexity of data pipelines. A growing population of enterprise data consumers, ranging from operational managers to data analysts and data scientists, began using new algorithms to generate new intelligence and analysis from diverse data sources, sometimes in real-time. This shift prompted data teams to adopt new tools and platforms, such as Snowflake, Databricks, and Synapse Analytics. These tools leverage cloud-native storage and compute infrastructure, offering unprecedented elasticity and scalability.

Architects and data engineers now build data pipelines with a plethora of tools, including extraction, loading, and transformation (ELT) tools, change data capture (CDC), application programming interfaces (APIs), and event streaming systems. They ingest structured, semi-structured, and unstructured data from sources like social media, IT logs, and Internet of Things (IoT) sensors, transforming and storing it in data warehouses, data lakes, NoSQL databases, and even streaming platforms. Cloud object stores and compute engines must integrate with legacy on-premises systems, increasing complexity. Amidst this hybrid and multi-cloud environment, companies must construct fragile data pipelines that feed data to a growing set of targets, including BI tools, artificial intelligence (AI), and embedded analytics workflows.

The complexity and rising tide of data can overwhelm enterprise teams managing the infrastructure, applications, and networks underpinning modern analytics pipelines. They struggle with slow, unwieldy Hadoop data lakes, which persist on-premises due to data gravity. Controlling the performance and cost of cloud data platforms while maintaining integrated views across hybrid, multi-cloud environments has become increasingly challenging. Delivering data and analytics pipelines with sufficient performance, availability, and reliability is difficult, and this complexity undermines the value these pipelines deliver to the business. [1] [3]

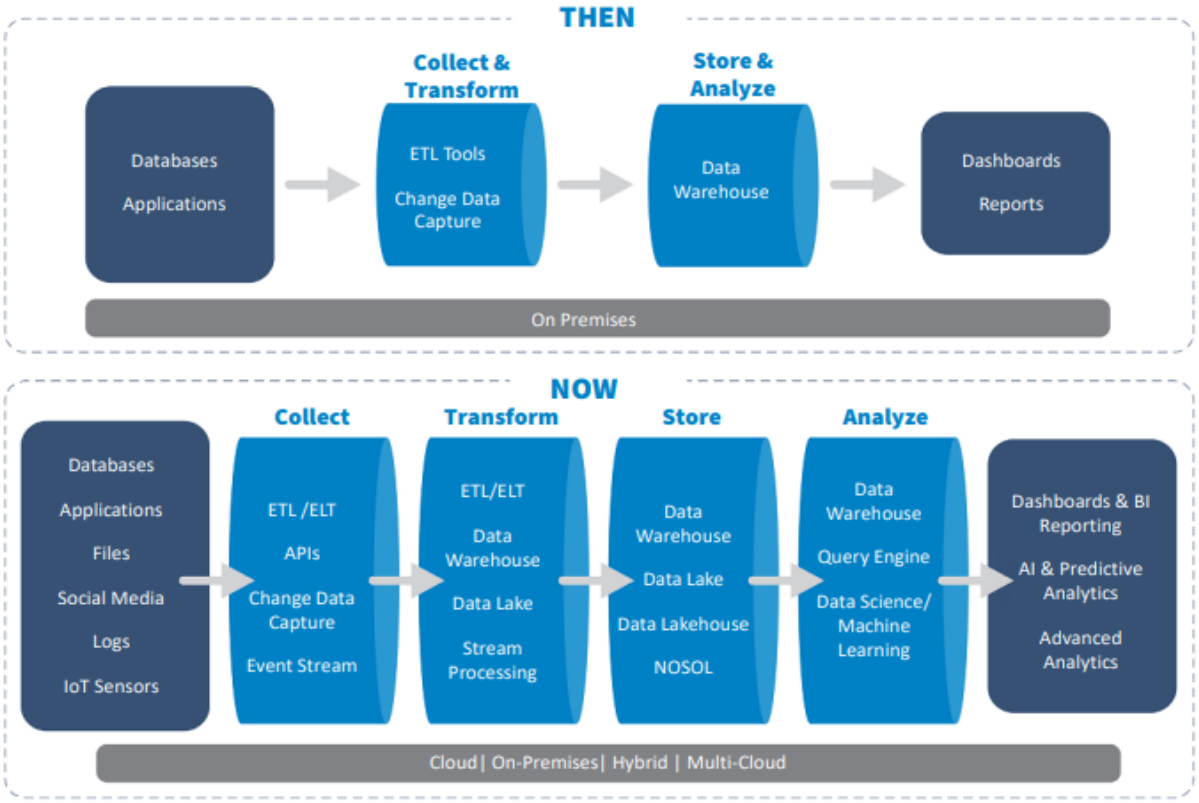


Figure 1: The Growing Complexity of Data Pipelines [3]

1.4 Challenges

The increasing complexity of data pipelines presents several significant challenges for enterprise data teams:

- **Data Quality Issues:** The proliferation of data sources leads to duplicate and contradictory datasets, compromising data accuracy.
- **Data Cleansing and Integration:** High volumes and varieties of data require significant manual effort for cleansing and integration, hindering scalability.
- **Automation and Scaling:** Manual processes are insufficient for handling growing data volumes and real-time processing, necessitating advanced automation.
- **Technology Proliferation:** Diverse tools and platforms create a fragmented landscape, complicating collaboration and problem-solving among data teams.
- **Performance and Cost Management:** Balancing performance with budget constraints is challenging in hybrid and multi-cloud environments, with limited visibility into cost drivers.
- **Infrastructure Management:** Managing complex infrastructures, especially slow, on-premises Hadoop data lakes, adds to operational burdens.
- **Timely Data Delivery:** Complexity causes delays and reduces reliability, impacting decision-making and operational efficiency.

- **Customer-Facing Applications:** Issues with data quality and performance in customer-facing applications can harm customer satisfaction and business reputation.

These challenges impede the delivery of high-quality, timely data necessary for competitive advantage. As a result, robust data observability practices and investments in automation and integration technologies are essential. [3] [4]

2 Data Observability

Data observability is a critical aspect of modern data management, providing comprehensive insights into the health and performance of data systems. It extends beyond traditional data monitoring to offer a holistic view of data pipelines, enabling proactive issue detection and resolution. This section explores the definition of data observability and its evolution from simple monitoring practices to advanced observability solutions.

2.1 Definition

Data observability refers to an organization's ability to fully understand the health of the data within its systems. It builds on the principles of application performance monitoring (APM) by providing comprehensive, end-to-end visibility into data pipelines, ensuring data quality, reliability, and trustworthiness. This approach leverages automated monitoring, alerting, and triaging to identify and resolve data quality and discoverability issues, leading to healthier data pipelines and more efficient data teams. [5] [6] [7] [8]

2.2 Evolution from Monitoring to Observability

Initially, data monitoring focused on tracking the behaviour of data pipelines and alerting data teams when anomalies occurred. This involved monitoring pipeline performance metrics such as data freshness, volume, and schema changes, often using tools that provided only a partial view of the data landscape. These tools could indicate when something went wrong, but they offered limited insights into the root cause or the broader impact of the issues. [5] [6] [7] [8]

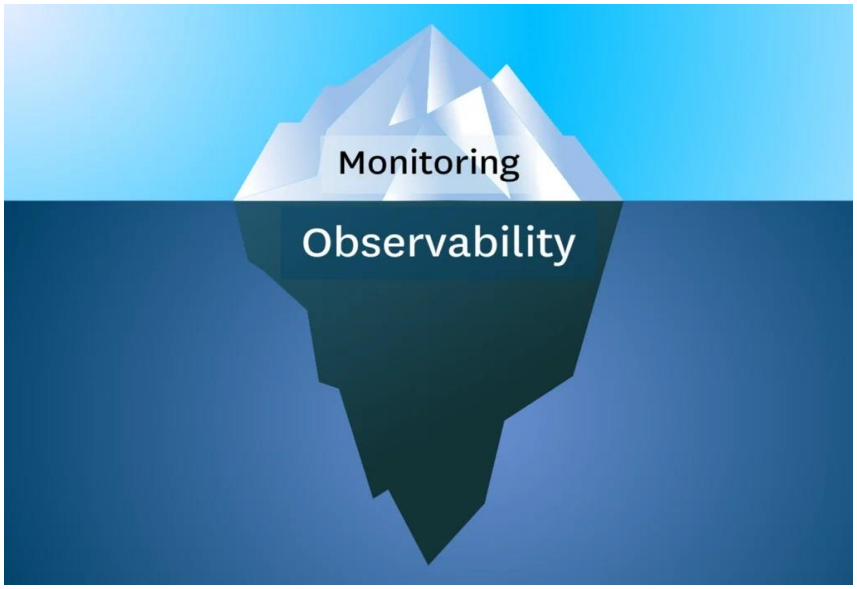


Figure 2 Monitoring Vs Data Observability [5]

Data observability, however, expands on traditional monitoring by incorporating several key elements:

1. **End-to-End Coverage**: Unlike basic monitoring tools that only observe isolated parts of the data pipeline, data observability provides visibility from data ingestion through transformation to final consumption. This holistic view helps in understanding how data flows across different systems and stages.
2. **Data Quality Monitoring**: Observability includes rigorous monitoring of data quality dimensions such as accuracy, completeness, consistency, timeliness, validity, and uniqueness. This ensures that the data meets business requirements at all times.
3. **Proactive Issue Resolution**: Data observability tools use machine learning to predict potential issues before they impact the business. Automated alerting and advanced analytics enable data teams to diagnose and resolve problems swiftly, reducing data downtime.
4. **Lineage and Impact Analysis**: One of the critical advancements in data observability is the ability to trace data lineage. This means understanding how data moves through various systems and identifying dependencies. When an issue arises, lineage tracking helps in pinpointing the exact location and impact, facilitating quicker resolution.
5. **Real-Time Insights**: By continuously monitoring data in real-time, data observability tools ensure that any anomalies or deviations are detected immediately, allowing for timely interventions and maintaining the integrity of analytics and AI output.

2.3 Current Trends in Data-Driven Enterprises

Data observability is crucial for modern data-driven organizations as it ensures the reliability and trustworthiness of data across various business functions. According to a comprehensive study conducted by CDO Magazine in collaboration with Kensu, data observability is recognized as a core strategy by 92% of data management leaders within the next 1-3 years. The study, which surveyed 185 data management decision-makers from companies with more than $50 million in annual revenue, highlights the pressing need for improved data quality and reliability. [9]

2.3.1 Importance of Data Observability

- **High Priority on Data Quality and Reliability**: Companies prioritize improving data quality (85%), building data products, and maintaining a data catalog (66% each). Cloud migration (44%) and visibility into data pipelines (45%) are also key focuses, with less emphasis on team performance (33%) and managing cloud costs (29%)

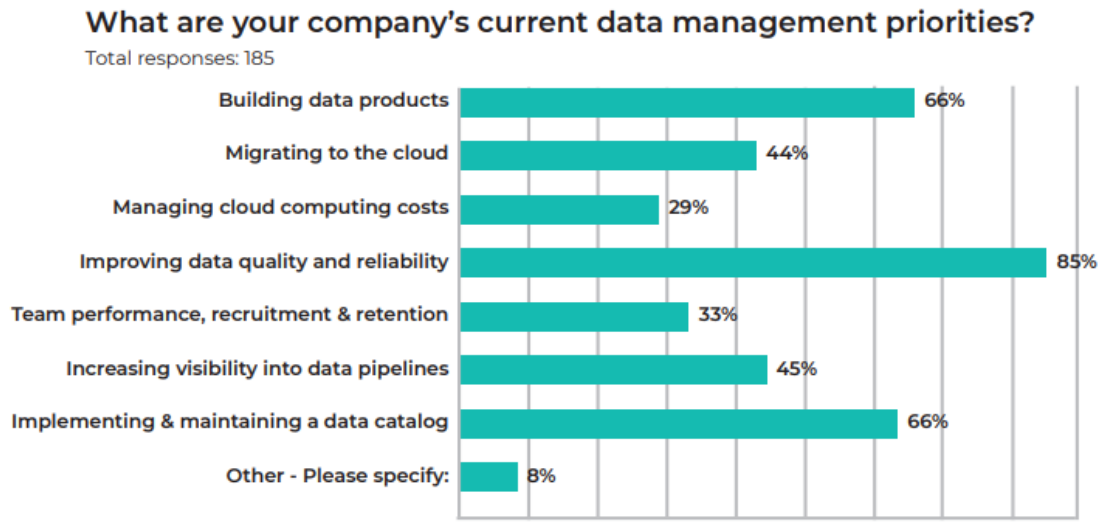


Figure 3 Survey Results: Top Data Management Priorities for Companies [9]

- **Significant Data Issues**: While most teams can sometimes identify data pipeline issues before they impact users (72%), the unresolved issues primarily cause conflicts with data consumers (58%), waste time (57%), and induce stress (55%), highlighting significant areas for process improvement.

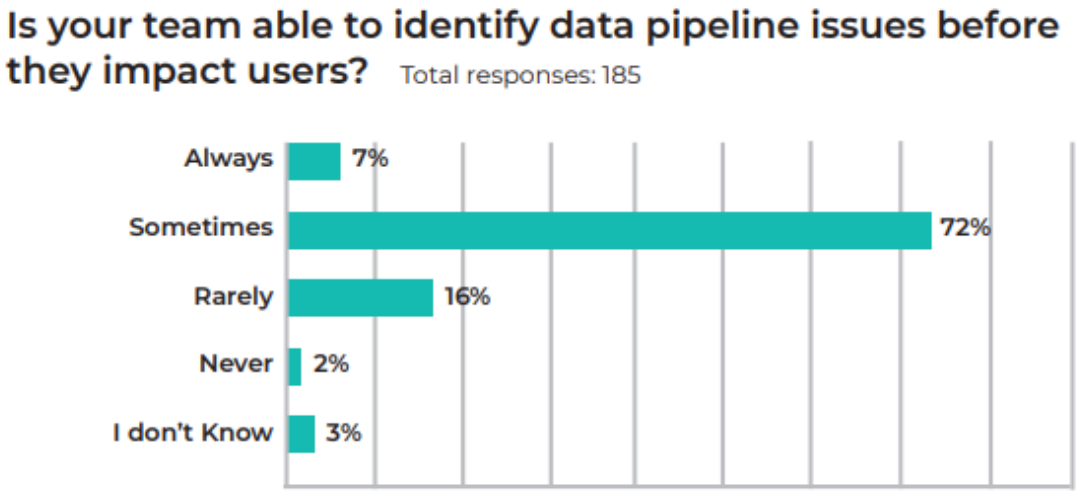


Figure 4 Team Effectiveness in Identifying Data Pipeline Issues Before User Impact [9]

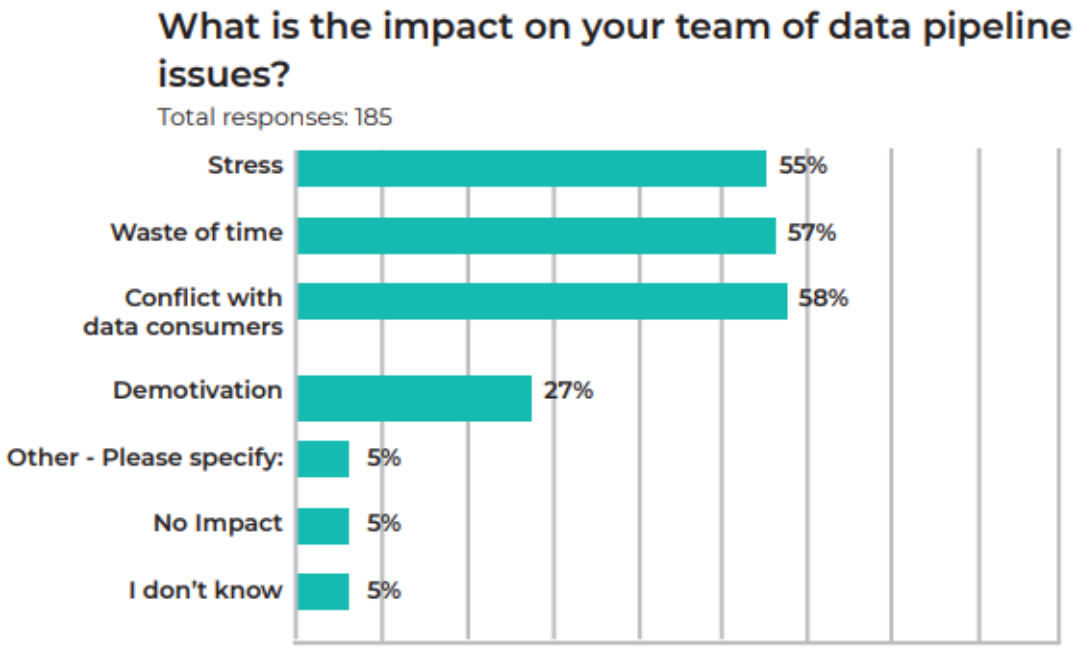


Figure 5 Impact of Data Pipeline Issues on Team Dynamics [9]

- **Time Spent on Fixing Data Pipelines**: Data teams spend a significant amount of time on pipeline issues, with 78% spending less than 40% of their month on this. Most issues are resolved within a few hours (51%) or days (28%).

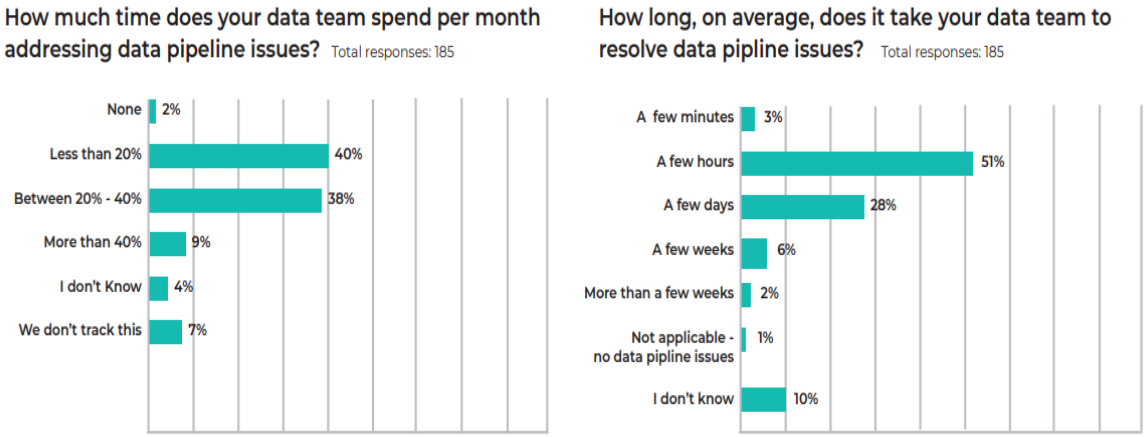


Figure 6 Time Spent and Resolution Duration for Data Pipeline Issues [9]

2.3.2 Benefits of Data Observability

- **Preventing Data Issues**: The primary benefits of implementing data observability are the prevention of data issues (79%), visibility in data pipelines (75%), and faster troubleshooting (74%). Availability of data lineage (63%) and populating a data catalog (36%) are also significant advantages.

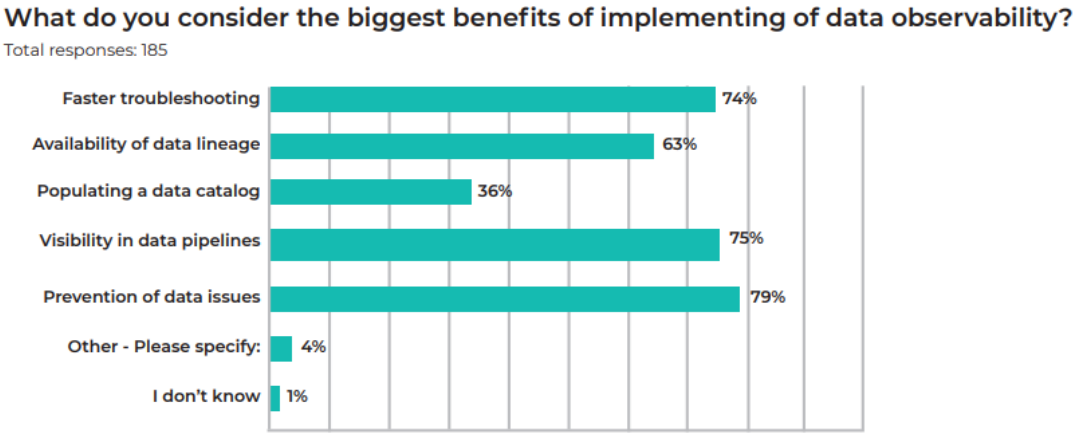


Figure 7 Top Benefits of Implementing Data Observability [9]

- **Strategic Implementation**: The data observability market is expected to mature by 2024, with 37% of organizations planning to implement a data observability tool within the next 12 months, and 35% within the next 2-5 years.

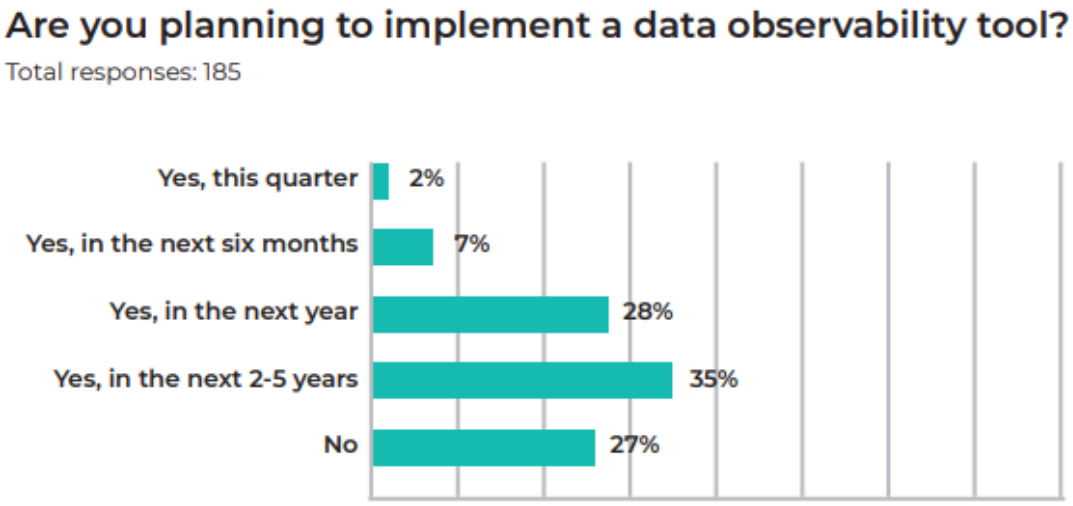


Figure 8 Plans for Implementing Data Observability Tools [9]

2.3.3 Conclusion

The transition from traditional monitoring to data observability marks a significant advancement in managing data pipelines. Data observability provides deeper insights and proactive issue resolution capabilities, ensuring that data remains a reliable asset that drives business value and operational efficiency. The Kensu and CDO Magazine study highlights the growing recognition of data observability as an essential component of modern data strategies, with maturity on the horizon.

Survey results show that while many companies prioritize improving data quality, building data products, and maintaining a data catalog, identifying, and resolving data pipeline issues remains a challenge. Most teams can sometimes detect these issues before they impact users, but unresolved problems often lead to conflicts, wasted time, and stress. Companies acknowledge the benefits of data observability, such as preventing issues and enhancing pipeline visibility, and many plan to implement observability tools within the next few years. However, a significant portion still lacks immediate plans for such tools, indicating an area for potential growth and improvement in data management practices. [9]

3 Internal Requirement Gathering for Data Observability Tools

This section outlines the findings from an internal investigation conducted with key members of our data consultancy team at Algorhythm. The investigation aimed to uncover the specific challenges and requirements for implementing data observability tools within our operations. It involved a comprehensive analysis of our current data infrastructure and common issues faced by our clients. The insights gathered have been categorized into key challenges and technical and non-technical considerations. The analysis concludes with strategic recommendations for addressing these pain points to enhance data quality, reliability, and overall data governance.

3.1 Key Challenges and Insights

1. **Data Quality Issues**

    - **Frequency:** Data quality issues account for 30% to 60% of managed services tickets. This high frequency underscores the need for robust data quality checks.
    - **Common Problems:** Duplicate data causing pipeline failures, difficulties in data discoverability and governance, and issues with data freshness.

2. **Data Infrastructure Monitoring**

    Infrastructure monitoring is critical for ensuring system availability and performance. Issues such as system downtime, slow performance, and load failures due to sudden changes in data volume or issues were common.

3. **Budgetary Constraints**

    Budget considerations vary significantly depending on the customer. Some clients may have substantial budgets for comprehensive solutions, while others may need more cost-effective options. Data observability platforms typically start from around $40K per year, which can be a significant investment for smaller clients.

4. **Environment and Monitoring Needs**
    - **On-Premise vs. Cloud**: Most of our work is client-dependent, with many clients operating on-premise environments. Monitoring solutions need to be adaptable to both on-premise and cloud infrastructures.
    - **Key Monitoring Requirements:**
        - **System Availability**: Monitoring whether the database and user front-end are accessible.

        - **Resource Utilization**: Monitoring CPU, disk usage, RAM, network, etc.
        - **Performance Metrics**: Detection of significant slowness, monitoring load timings at different levels, and providing statistics over time.
    - **Tools**: While data observability focuses on data pipelines, infrastructure monitoring often relies on Application Performance Monitoring (APM) tools like Datadog for example.

3.2 Technical and Non-Technical Considerations

- **Non-Technical Considerations**:
    - Scalability.
    - Integration (Cloud & On-Premises).
    - Documentation Availability.
    - Community and/or Vendor Support.
    - Cost.
    - Hosting.
- **Technical Pain Points**:
    - Data Quality.
    - Data Lineage.
    - Data Catalog.
    - Data Freshness.
    - Data Schema.
    - Data Glossary.
    - Data Security.
    - Access Control.
    - Master Data Management.
    - Infrastructure Monitoring.

3.3 Analysis and Conclusion

After analysing these technical pain points, we can categorize the issues as follows:

- **Data Observability Issues**
- **Data Governance Issues**
- **Infrastructure Monitoring Issues**: Although crucial, these issues are outside the scope of this comparative study and are typically addressed by APM tools.

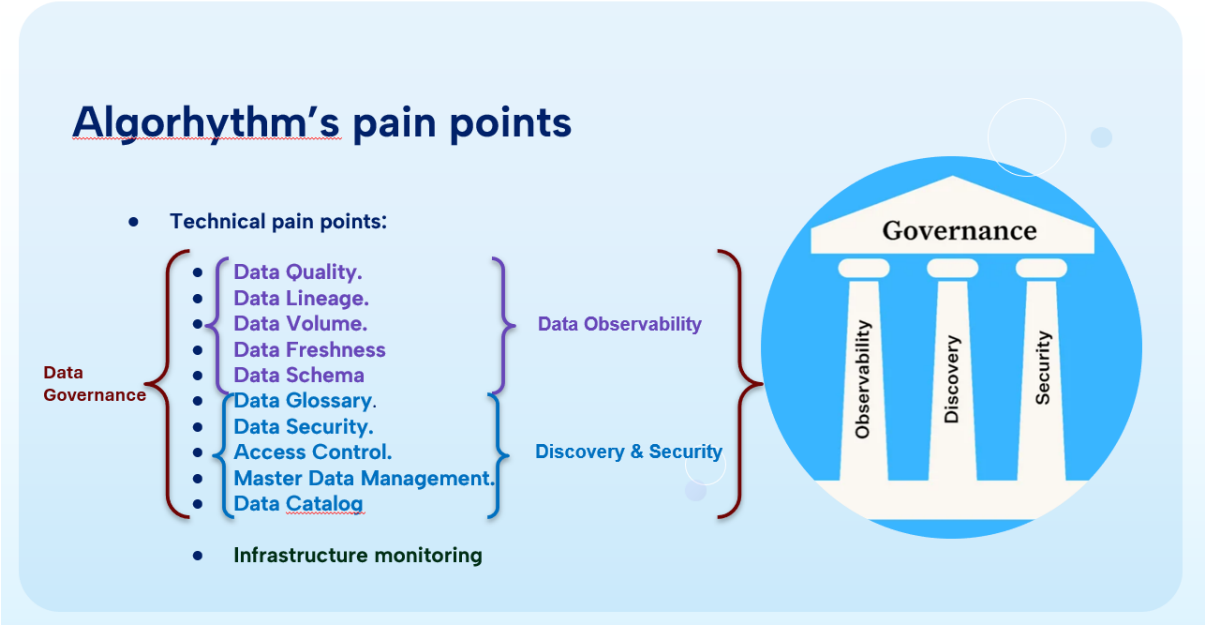


Figure 9 Algorhythm' s pain points [Author's contribution]

By addressing most of these pain points, we can implement a comprehensive data observability strategy that not only enhances data quality and reliability but also improves overall data governance. This approach will ensure that data teams are more productive, data pipelines are healthier, and clients can better leverage their data for business insights.

4 Tools Overview

In this section, we explore various tools available for data observability, examining their architecture, features, and capabilities. Understanding the core components of these tools helps in evaluating their effectiveness in managing and monitoring data systems. This overview aims to provide a comprehensive comparison of leading data observability platforms to assist in selecting the most suitable solution for specific organizational needs.

4.1 Observability Tool Architecture

A robust data observability platform integrates seamlessly with existing data tools and infrastructures, ensuring effective monitoring and management of data systems. Below are the essential components of such a platform: [10] [11] [12]

1. **Out-of-the-Box Integrations**: The platform should support seamless integration with popular data tools like Snowflake, Databricks, and BigQuery, facilitating easy connectivity and immediate functionality within existing data environments.
2. **Metadata Ingestion Service**: This service is crucial for fetching and managing metadata from various data tools, enabling enhanced visibility and control over the data landscape.
3. **Metadata Storage and Transformation Database**: Dedicated databases for metadata storage and transformation are vital for organizing and manipulating metadata to extract actionable insights.
4. **User Interface for Metadata Exploration**: A user-friendly and comprehensive interface allows for efficient interaction with and exploration of metadata, simplifying the understanding of data structures and dependencies.
5. **Alerting Service**: This component proactively notifies users about anomalies or issues detected within data pipelines, allowing for timely interventions.
6. **Testing Service**: Regularly scheduled queries test data quality and integrity within environments such as data warehouses, ensuring continuous data accuracy and reliability.

Additional Features for Debugging and Auditing:

7. **Data Lineage**: Visual representations show the dependencies between data assets and models, providing detailed insights at both macro and micro levels.
8. **Data Catalog**: A searchable engine for data assets facilitates efficient data management and accessibility.
9. **Data Security**: Features like access controls, data tagging for personally identifiable information (PII), and sensitive data tracking bolster data protection.

Emerging Open-Source Platforms

Platforms like OpenMetadata and DataHub are increasingly recognized for their capabilities in data observability. Although native deployment poses challenges, DataHub, with its documented real-world applications, demonstrates significant potential compared to OpenMetadata, which is yet to showcase similar success. [13] [14]

Economic Considerations

Many top-tier data observability tools operate independently of cloud data warehouses, leading to pricing models with higher central costs and reduced economies of scale, unlike cloud-integrated services.

Evaluation Criteria

Understanding these components and features allows for a thorough evaluation of data observability tools based on their efficacy in enhancing aspects of data quality, aiming to improve data pipeline performance and reliability.

4.2 Comparison Table

After establishing the architecture essential for a robust data observability platform, it is crucial to assess how well current tools adhere to these architectural requirements. This next section provides a comparison table that evaluates leading data observability tools against a set of critical features essential for comprehensive data management and observability.

[10] [11] [12]

Feature /Tool	Quality	Freshness	Schema	Volume	Lineage	Data Catalog	Data Glossary	Data Security
Acryl Datahub	✓	✓	✓	✓	✓	✓	✓	✓
Datahub			✓		✓	✓	✓	✓
OpenMetadata (Collate)	✓	✓	✓	✓	✓	✓	✓	✓
Sifflet	✓	✓	✓	✓	✓	✓	✓	✓
Monte Carlo	✓	✓	✓	✓	✓	✓		
Databand	✓	✓	✓	✓	✓			
Metaplane	✓	✓	✓	✓	✓			
Anomalo	✓	✓	✓	✓	✓			
Elementary (SaaS)	✓	✓	✓	✓	✓			
Soda	✓	✓	✓	✓				
Great Expectations	✓	✓	✓	✓				

- **Quality**: Ensuring data is accurate, consistent, and reliable.
- **Freshness**: Keeping data current and timely.
- **Schema**: Managing the structure and schema of data effectively.
- **Volume**: Handling large volumes of data efficiently.
- **Lineage**: Tracking the origin and transformation of data.
- **Data Catalog**: Providing a searchable system for data assets.
- **Data Glossary**: Defining data terms and relationships clearly.
- **Data Security**: Securing data against unauthorized access.

4.3 Detailed Tool Descriptions

In this section, we provide an in-depth analysis of each data observability tool listed in the comparison table. These detailed descriptions cover the core features, key capabilities, pricing models, pros, cons, and an overall conclusion for each tool. This comprehensive evaluation aims to help organizations understand the unique strengths and potential limitations of each tool, facilitating a more informed selection process based on specific needs and requirements.

4.3.1 DataHub

Overview: DataHub is an open-source data catalog platform with robust metadata management, basic governance functionalities, and extensive integration capabilities. [15]

	Quality	Freshness	Schema	Volume	Lineage	Data Catalog	Data Glossary	Data Security
Acryl Datahub	✓	✓	✓	✓	✓	✓	✓	✓
Datahub			✓		✓	✓	✓	✓

Meets Requirements:

- **Out-of-the-Box Integrations**: Yes
- **Metadata Ingestion Service**: Yes
- **Metadata Storage and Transformation Database**: Yes
- **User Interface for Metadata Exploration**: Yes
- **Alerting Service**: Yes

Key Features:

1. Metadata 360
2. Shift-Left Practices
3. Active Metadata
4. Open Source with community support
5. Forward-Looking Architecture
6. Extensive Integrations

Pricing: Managed Service available at $60,000 & OSS (requires dedicated platform team). [16]

Pros:

- Open-source
- User-friendly interface
- Extensive integrations
- Managed service from Acryl Data

Cons:

- Maintenance intensive
- Basic quality control
- No SLA

Conclusion: A cost-effective, flexible data catalog platform for organizations with substantial platform teams. Offers robust metadata management but requires significant maintenance.

4.3.2 OpenMetadata (Collate)

Overview: OpenMetadata is a comprehensive open-source platform for data discovery, observability, and governance, with flexible deployment options and strong community support. [17] [18]

	1. Quality	2. Freshness	3. Schema	4. Volume	5. Lineage	6. Data Catalog	7. Data Glossary	8. Data Security
OpenMetadata (Collate)	✓	✓	✓	✓	✓	✓	✓	✓

Meets Requirements:

- **Out-of-the-Box Integrations**: Yes
- **Metadata Ingestion Service**: Yes
- **Metadata Storage and Transformation Database**: Yes
- **User Interface for Metadata Exploration**: Yes
- **Alerting Service**: Yes

Key Features:

1. Data Discovery
2. Governance Enforcement
3. Data Insights
4. Security
5. Detailed Data Lineage
6. Extensive Ingestion Framework
7. APIs for Integration
8. Team Collaboration Tools

Deployment Options: OSS, SaaS, on-premises, BYOC

Pricing: [18]

- Starter Plan: From $15,000 per year
- Premium+: From $25,000 per year

Pros:

- Comprehensive features
- Open-source flexibility
- Scalable plans
- Strong community support

Cons:

- Higher cost for advanced features
- Significant initial setup

Conclusion: Provides a robust solution for data discovery, governance, and observability with a comprehensive feature set and flexible deployment options.

4.3.3 Sifflet

Overview: Sifflet enhances data observability through organization, accessibility, and problem-solving features, ensuring efficient data management and anomaly detection. [19]

	1. Quality	2. Freshness	3. Schema	4. Volume	5. Lineage	6. Data Catalog	7. Data Glossary	8. Data Security
Sifflet	✓	✓	✓	✓	✓	✓	✓	✓

Meets Requirements:

- **Out-of-the-Box Integrations**: Yes
- **Metadata Ingestion Service**: Yes
- **Metadata Storage and Transformation Database**: Yes
- **User Interface for Metadata Exploration**: Yes
- **Alerting Service**: Yes

Key Features:

1. Unified Platform for team collaboration
2. Centralized documentation and data lineage
3. User-friendly interface and data catalog
4. Anomaly detection and quality monitoring

Pricing: $48,000 per year [20]

Pros:

- Comprehensive features for data management
- Robust integration capabilities
- Effective anomaly detection

Cons:

- Higher cost

Conclusion: Ideal for organizations seeking a comprehensive, integrated data observability platform with strong anomaly detection and user-friendly features.

4.3.4 Monte Carlo

Overview: Monte Carlo's platform monitors and alerts on data issues across your entire data stack, leveraging machine learning for proactive identification and resolution. [21]

	1. Quality	2. Freshness	3. Schema	4. Volume	5. Lineage	6. Data Catalog	7. Data Glossary	8. Data Security
Monte Carlo	✓	✓	✓	✓	✓	✓		

Meets Requirements:

- **Out-of-the-Box Integrations**: Yes
- **Metadata Ingestion Service**: Yes
- **Metadata Storage and Transformation Database**: Yes
- **User Interface for Metadata Exploration**: Yes
- **Alerting Service**: Yes

Key Features:

1. Proactive Monitoring
2. Automated Root Cause Analysis
3. Data Lineage and Cataloging
4. Comprehensive Alerting
5. CI/CD Integration

Pricing:

- $50,000 per year (Starter). [22]
- $100,000+ per year (Enterprise). (Appendix A.1)

Pros:

- End-to-end observability
- Machine learning-driven monitoring
- Centralized data catalog

Cons:

- Higher cost for smaller organizations

Conclusion: Provides a robust solution for data observability, with machine learning-driven monitoring and extensive features for large-scale data management.

4.3.5 Databand

Overview: Databand provides a comprehensive platform for data incident management, pipeline monitoring, quality assurance, anomaly detection, and lineage analysis. [23]

	1. Quality	2. Freshness	3. Schema	4. Volume	5. Lineage	6. Data Catalog	7. Data Glossary	8. Data Security
Databand	✓	✓	✓	✓	✓			

Meets Requirements:

- **Out-of-the-Box Integrations**: Yes
- **Metadata Ingestion Service**: Yes
- **Metadata Storage and Transformation Database**: Yes
- **User Interface for Metadata Exploration**: Yes
- **Alerting Service**: Yes

Key Features:

1. Data Incident Management
2. Data Pipeline Monitoring
3. Data Quality Monitoring
4. Automated Anomaly Detection
5. Data Lineage and Impact Analysis

Pricing: EUR 250 per pipeline per year (volume discounts available). (Appendix A.2)

Pros:

- Extensive feature set
- Real-time alerts
- Suitable for complex environments

Cons:

- Higher pricing for smaller organizations

Conclusion: Offers a robust solution for managing and resolving data incidents, with extensive features and flexible pricing for larger organizations.

4.3.6 Metaplane

Overview: Metaplane enhances continuous integration and machine learning-based monitoring, with capabilities for anomaly detection, data lineage, and quality maintenance. [24]

	1. Quality	2. Freshness	3. Schema	4. Volume	5. Lineage	6. Data Catalog	7. Data Glossary	8. Data Security
Metaplane	✓	✓	✓	✓	✓			

Meets Requirements:

- **Out-of-the-Box Integrations**: Yes
- **Metadata Ingestion Service**: Yes
- **Metadata Storage and Transformation Database**: Yes
- **User Interface for Metadata Exploration**: Yes
- **Alerting Service**: Yes

Key Features:

1. ML-Based Anomaly Detection
2. Automated Data Lineage
3. Data Usage Tracking
4. Real-Time Job Monitoring
5. Schema Change Alerts

Pricing: Starting at $15,000+ per year [12]

Pros:

- Comprehensive monitoring and lineage tracking
- Real-time alerts and preventative measures

Cons:

- Higher cost

Conclusion: Suitable for organizations needing advanced monitoring and integration capabilities, ensuring high data quality and operational efficiency.

4.3.7 Anomalo

Overview: Anomalo focuses on anomaly detection using unsupervised machine learning, making it suitable for complex datasets with frequent anomalies. [25]

Anomalo	1. Quality	2. Freshness	3. Schema	4. Volume	5. Lineage	6. Data Catalog	7. Data Glossary	8. Data Security
Anomalo	✓	✓	✓	✓	✓			

Meets Requirements:

- **Out-of-the-Box Integrations**: Yes
- **Metadata Ingestion Service**: Yes
- **Metadata Storage and Transformation Database**: Yes
- **User Interface for Metadata Exploration**: Yes
- **Alerting Service**: Yes

Key Features:

1. Unsupervised Machine Learning
2. Automatic detection of anomalies
3. User-friendly configuration (API and no-code)
4. Deep monitoring and alerting

Pricing: Starting at $20,000+ per year

Pros:

- Effective anomaly detection
- User-friendly
- Comprehensive monitoring features

Cons:

- None noted for primary anomaly detection use

Conclusion: Excels in anomaly detection for complex datasets, offering a user-friendly, comprehensive solution.

4.3.8 Elementary

Overview: Elementary is a dbt-native data observability tool that integrates testing and alerting within data pipelines. It combines dbt tests with a user-friendly interface for metadata exploration. [26]

	1. Quality	2. Freshness	3. Schema	4. Volume	5. Lineage	6. Data Catalog	7. Data Glossary	8. Data Security
Elementary (SaaS)	✓	✓	✓	✓	✓			

Meets Requirements:

- **Out-of-the-Box Integrations**: Partial (primarily focused on dbt)
- **Metadata Ingestion Service**: No
- **Metadata Storage and Transformation Database**: No
- **User Interface for Metadata Exploration**: Yes
- **Alerting Service**: Yes

Key Features:

1. User Interface for Metadata Exploration
2. Testing Service leveraging dbt tests
3. Alerting Service for proactive issue detection

Pricing: $7,200 per year [26]

Pros:

- Open-source
- Preventative testing and alerting
- Easy to use for dbt users

Cons:

- Limited to dbt tests

Conclusion: Ideal for organizations using dbt, offering proactive data quality enhancements through testing and alerting.

4.3.9 Soda

Overview: Soda provides a data observability framework embedded within data pipelines. It offers both an open-source variant (Soda Core) and a more feature-rich library with cloud services. [27]

	1. Quality	2. Freshness	3. Schema	4. Volume	5. Lineage	6. Data Catalog	7. Data Glossary	8. Data Security
Soda	✓	✓	✓	✓				

Meets Requirements:

- **Out-of-the-Box Integrations**: Yes
- **Metadata Ingestion Service**: No
- **Metadata Storage and Transformation Database**: Yes (in the full library)
- **User Interface for Metadata Exploration**: Yes
- **Alerting Service**: Yes

Key Features:

1. Embedded Testing Framework
2. Soda Core (OSS) with custom SQL tests
3. Soda Library with additional features
4. Soda Checks Language (SodaCL)

Pricing: Free Trial Available (Estimated Price: $15,000+)

Pros:

- Open-source
- Preventative testing
- Easy to use

Cons:

- Cloud dependency for full features
- Potential support issues

Conclusion: Suitable for teams seeking an open-source, customizable testing framework. The cloud dependency may be a drawback for some organizations.

4.3.10 Great Expectations

Overview: Great Expectations is a flexible data validation framework with open-source and cloud-hosted versions, enabling data quality checks integrated into data workflows. [28]

	1. Quality	2. Freshness	3. Schema	4. Volume	5. Lineage	6. Data Catalog	7. Data Glossary	8. Data Security
Great Expectations	✓	✓	✓	✓				

Meets architectural requirements:

- **Out-of-the-Box Integrations**: Yes
- **Metadata Ingestion Service**: No
- **Metadata Storage and Transformation Database**: Yes (in the cloud version)
- **User Interface for Metadata Exploration**: Yes
- **Alerting Service**: Yes

Key Features:

1. Testing Framework for data workflows
2. Great Expectations Cloud
3. Wide adoption and strong community support
4. Customizable data validation

Pricing: Free (OSS), Cloud starts at $1,200 per year [28]

Pros:

- Open-source
- Preventative checks
- Strong community support

Cons:

- Justifying Cloud investment can be challenging
- Requires a cultural shift towards test-driven development
- Insufficient documentation

Conclusion: A powerful solution for teams focusing on test-driven development and needing flexible, automated data validation.

4.4 Tool Selection

In the initial phase of evaluating tools for our data management framework, we narrowed down our choices to DataHub and OpenMetadata. DataHub, having been longer in the field, offers robustness and has been battle-tested [29], which made it an appealing option. It serves primarily as an open-source data catalog, well-suited for robust metadata management. However, it lacks specific data quality features essential for comprehensive data observability, which are critical for our needs. [15]

Initially, we considered coupling DataHub with Great Expectations [28], a tool known for its data validation capabilities, to create a full-blown data management solution. Unfortunately, the open-source version of Great Expectations presented several challenges. It proved cumbersome with inadequate documentation and complex integration into continuous integration/continuous deployment (CI/CD) pipelines.

Upon further evaluation, OpenMetadata emerged as the more fitting choice for our proof of concept (POC). Unlike DataHub, OpenMetadata integrates both data cataloging and observability functionalities into a single platform. This integration is particularly advantageous as it simplifies the architecture by reducing the need for multiple tools. More importantly, OpenMetadata offers these features for free in its open-source version, which is pivotal for conducting a cost-effective POC. [17]

The OpenMetadata platform not only meets our immediate needs for cataloging and basic observability but also provides a path for scaling our capabilities through its SaaS offering, Collate. Collate offers advanced features such as AI-driven anomaly detection and Service Level Agreements (SLAs), which could be beneficial as our data operations grow. [18]

4.5 Conclusion

Choosing OpenMetadata allows us to leverage a comprehensive, integrated approach to data management and observability. This tool provides all necessary functionalities out-of-the-box, enhancing our ability to implement a proof of concept effectively without the complexities and additional costs associated with integrating multiple tools. OpenMetadata's capability to expand its features through its SaaS model ensures that we can scale our operations seamlessly in the future, making it the most strategic choice for our current and anticipated needs.

5 Data Stack Sandbox for POC

Our proof of concept (POC) for the data observability project involves setting up a comprehensive sandbox environment that supports a fully functional data pipeline. This pipeline connects to multiple data sources, executes essential ETL processes, and leverages OpenMetadata for enhanced data management features such as cataloging, tagging, and insights. Below is a detailed breakdown of this sandbox environment's structure and function:

5.1 Use Case Overview

The POC leverages three diverse datasets to demonstrate the data pipeline's versatility and effectiveness:

1. **Transactional Data**: This dataset comprises loan transactions and customer interactions, offering a robust source of relational data. (Appendix B.1)
2. **API Data for Soccer Players**: Extracted from a sports API, this dataset provides detailed information on soccer players, ideal for showcasing dynamic data ingestion capabilities. (Appendix B.2)
3. **Open Data from Inside Airbnb**: Features publicly available data from Inside Airbnb to analyse property listing trends and user interactions. (Appendix B.3)

5.2 Sandbox Environment

To create a robust and reliable data observability framework, the sandbox environment is meticulously designed to incorporate various essential components. Each component plays a vital role in ensuring seamless data integration, transformation, and management. The architecture integrates modern data processing tools and platforms to handle diverse data sources, maintain data quality, and provide comprehensive metadata management.

The sandbox environment setup includes workflow orchestration with Airflow, data integration via Airbyte, scalable storage using S3, data transformations managed by dbt Core, and efficient data processing with a Spark cluster. Additionally, the environment supports interactive data analysis through Jupyter notebooks, metadata management with Hive Metastore, and centralized structured data storage using a Postgres Data Warehouse. OpenMetadata is integrated across all components to facilitate enhanced data governance, visibility, and control.

5.3 Architecture

The architecture diagram bellow labelled as (Figure 10), illustrates the setup of our data observability sandbox. This environment includes several key components designed to work in harmony, ensuring comprehensive management and observability of the data pipeline:

- **Airflow:** Manages workflow orchestration, controlling the sequence and execution of tasks**.**
- **Airbyte:** Facilitates data integration from various sources into the system**.**
- **S3 Storage:** Stores raw, processed, and transformed data, dbt metadata files, and Hive SQL warehouse files, enabling scalable and durable data storage.

- **dbt Core:** Handles data transformation ensuring consistency, metadata management through tags and descriptions and quality across data layers.
- **Spark Cluster:** Processes large datasets efficiently, working in conjunction with the Jupyter server for data analysis tasks.
- **Jupyter Server:** Provides an interactive interface for data processing, analysis and visualization.
- **Hive Metastore:** Manages metadata for Spark and the Airbnb lakehouse, ensuring efficient data retrieval and management.
- **Postgres Data Warehouse (DWH):** Acts as the central repository for structured data storage, supporting dbt transformations and PostgreSQL operations.
- **OpenMetadata:** Integrates with all components to manage metadata and enhance data governance through visibility and control.

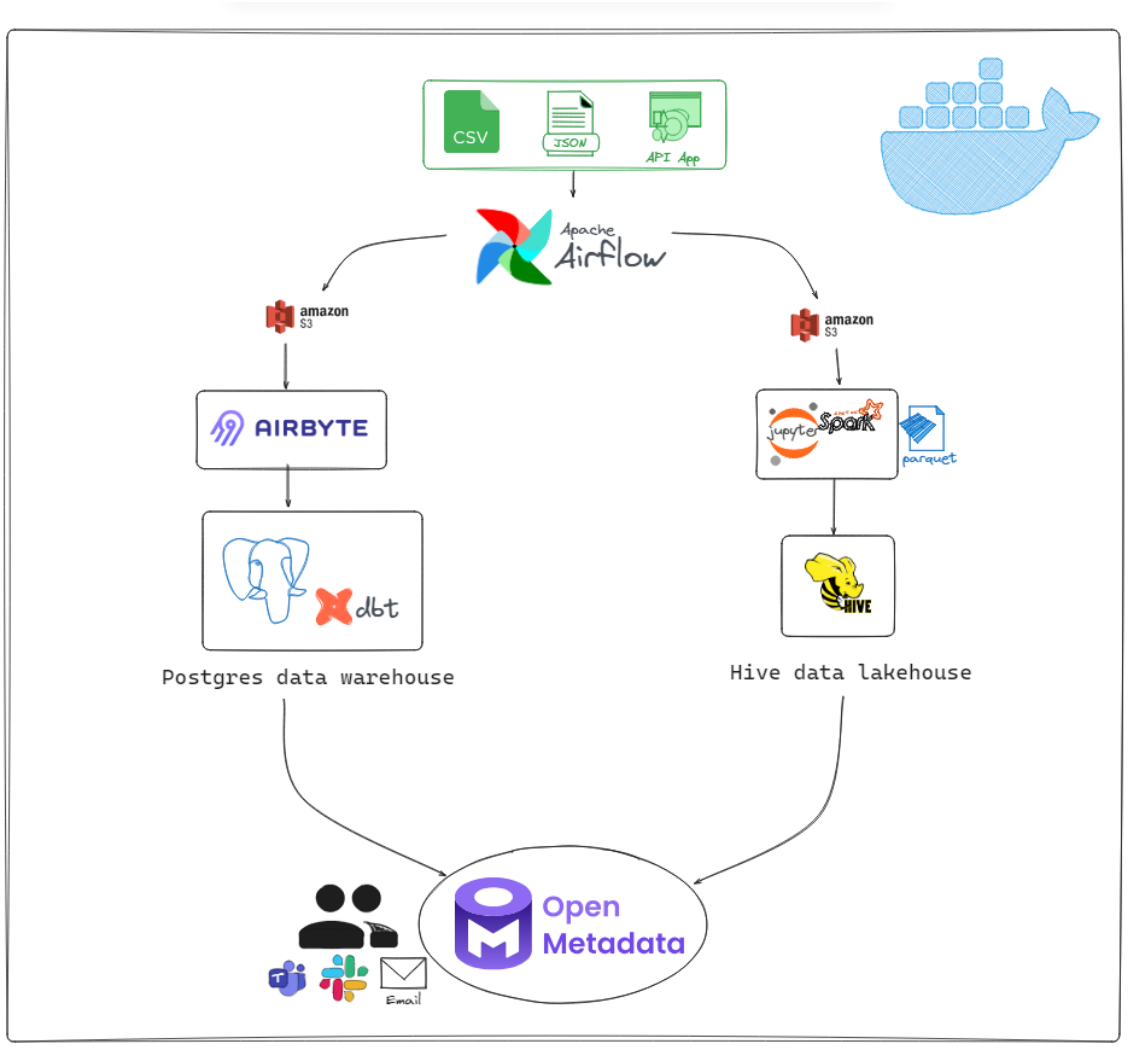


Figure 10 High-Level Architecture of the Data Observability Sandbox Environment [Author's contribution]

5.4 Data Ingestion and ETL Processes

The data ingestion and ETL processes are pivotal to maintaining a smooth and efficient data pipeline. They involve several components and tools working together to ingest data from various sources, transform it as necessary, and store it in the appropriate data warehouses or lakes. Below is a detailed breakdown of the processes involved:

5.4.1 Airflow:

Includes 4 DAGs:

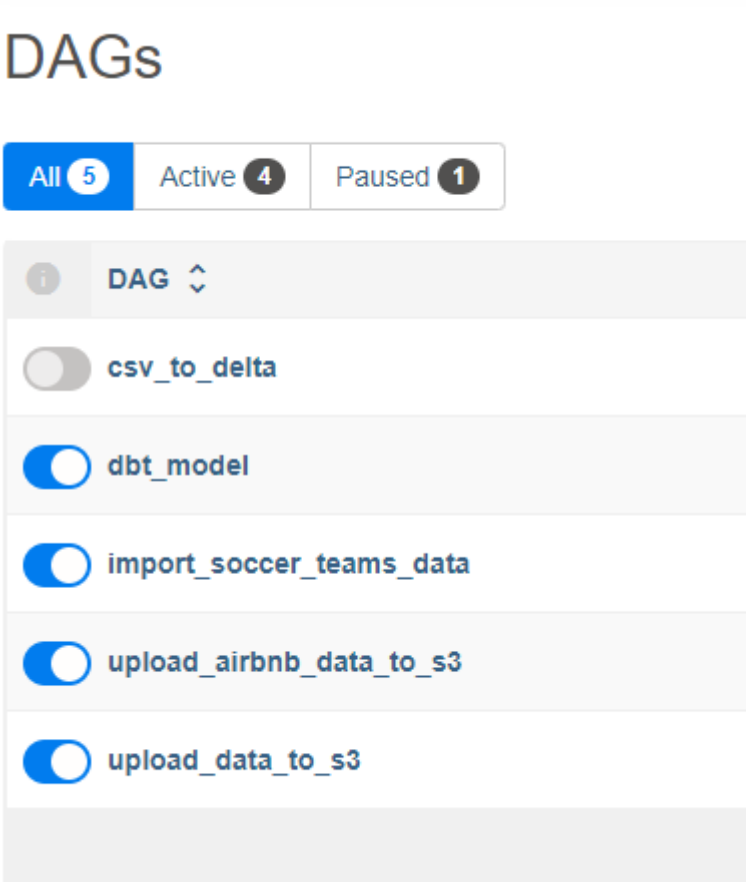
- upload_data_to_s3.py and upload_airbnb_data_to_s3.py: Automate the transfer of CSV files from local storage to AWS S3, setting the stage for further processing.
- import_soccer_teams_data.py: Connects to the sports API, uses the Airflow PostgreSQL operator to create an "ods" schema, loads player data into tables, and categorizes them by position (goalkeepers, defenders, midfielders, attackers).
- run_dbt_models.py: Executes the dbt core project, connecting to the Airbyte schema to create data layers (bronze, silver, gold), runs dbt tests, adds tags and descriptions, and saves metadata manifest in JSON files.
- 

Figure 11 Airflow DAGs Overview in the Data Stack Sandbox Environment for POC

5.4.2 Airbyte

Airbyte is configured to handle three tasks that connect to the S3 source and replicate data into a Postgres data warehouse (DWH) within a schema named 'airbyte':

- S3_daily_loan_transactions -> Postgres
- S3_daily_customer_drivers -> Postgres
- S3_customer_information_cdc -> Postgres

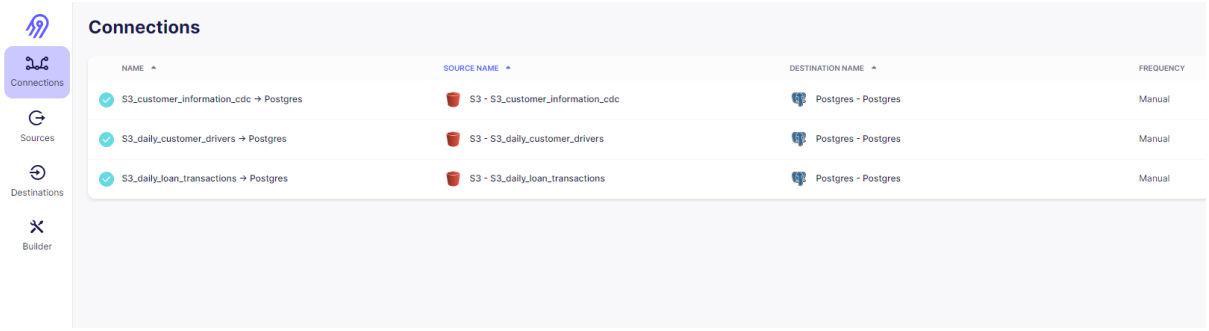


Figure 12 Airbyte Connections for Data Replication to Postgres DWH

5.4.3 Spark + Hive Metastore & Jupyter Notebook:

Utilizes a Spark cluster (1 master, 2 workers) and a Jupyter server connected to the Spark master to process CSV and JSON files from Inside Airbnb. Data is processed for three cities (Antwerp, Brussels, Ghent), and output is stored as Parquet files and ".db" files in a data lakehouse based on S3, registered in a Hive metastore.

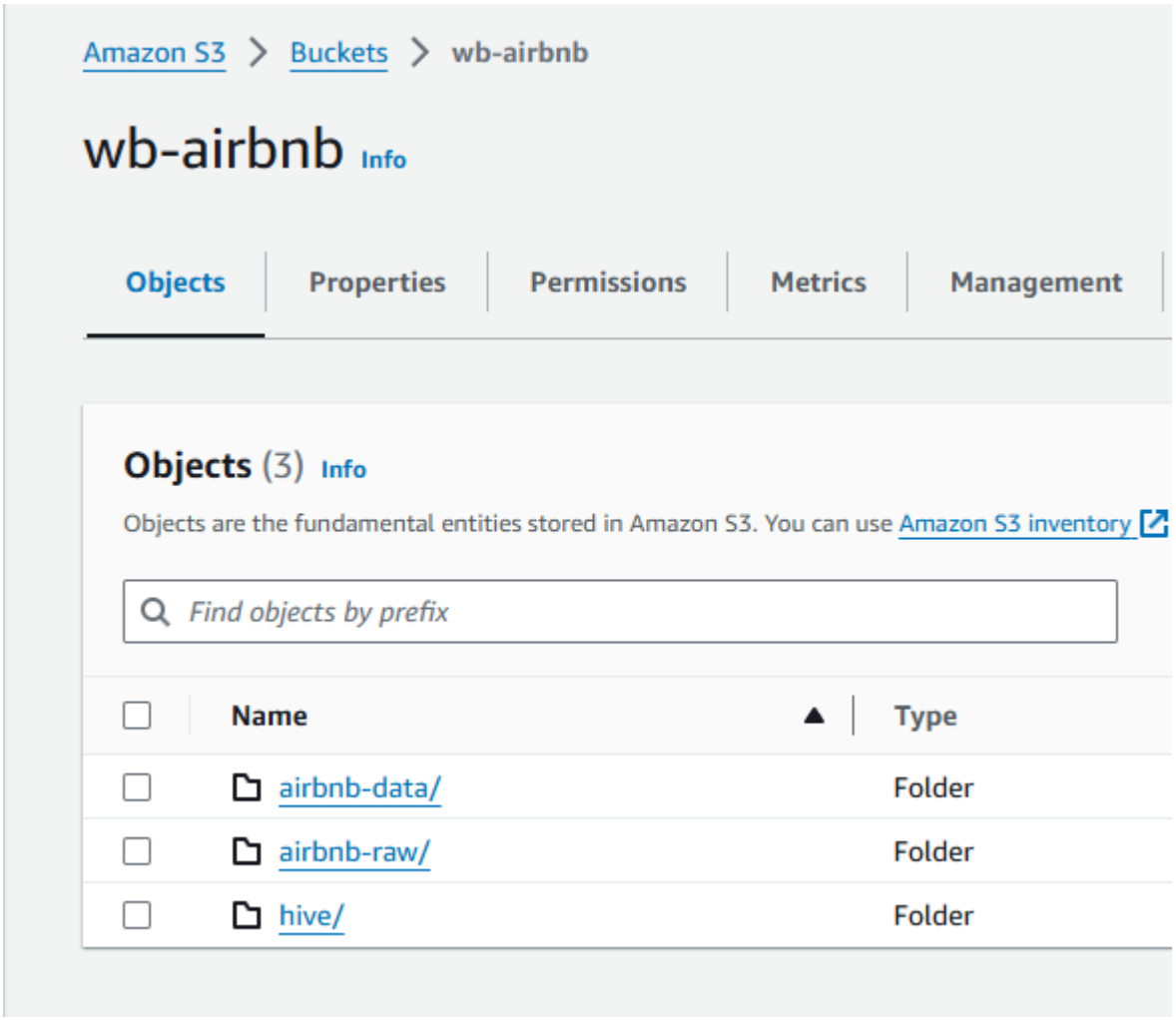
- 

Figure 13 Amazon S3 Bucket Structure for Airbnb Data Processing

5.4.4 DBT (Data Build Tool)

Manages data transformation through stages—bronze, silver, and gold. It applies stringent DBT tests to ensure data integrity and consistency, manages schema definitions for the silver and gold layers, and enriches datasets with metadata and DBT tags.

-

Figure 14 Airflow DAG for DBT Data Transformation and Quality Assurance

5.4.5 PostgreSQL

The sports API data is fetched, structured into an "ods" schema in PostgreSQL, and segmented by player positions.

-

Figure 15 Airflow DAG for Fetching and Structuring Sports API Data in PostgreSQL

6 POC Implementation Details

This section details the implementation process of the Proof of Concept (POC) aimed at ensuring comprehensive data monitoring, real-time alerts and data governance features within our data pipeline. The implementation is broken down into clear objectives, setting up connections, metadata ingestions, and addressing challenges encountered.

6.1 Objectives and success criteria

The primary objectives of this Proof of Concept (POC) are to ensure comprehensive data monitoring, implement real-time alerts for data pipeline issues and establish a robust data governance. Success will be measured by:

1. **Comprehensive Data Monitoring**: Ensuring freshness, quality, schema integrity, lineage, and volume of data.
2. **Real-Time Alerts**: Implementing immediate notifications for upstream or source issues that could disrupt the pipeline, such as duplicates or schema changes.
3. **Data Glossary and Catalog**: Establishing a clear and accessible data glossary and catalog to improve data governance and discovery.

6.2 Setting up connections and Metadata Ingestions

To achieve these objectives, various connections and metadata ingestion jobs were set up using different tools within the sandbox environment. Each tool was configured to ensure smooth data ingestion, transformation, and metadata management.

After setting up the sandbox environment and executing the pipelines, Airflow and Airbyte pipelines as shown in (Figures 16 and 17), we end up with five schemas in the Postgres Data Warehouse as shown in (Figure 20). (Figure 30) shows three schemas in the Hive Metastore, while managing Parquet files, DBT JSONs, or Hive DB files in two S3 buckets as shown in (Figure 33). The setup of the ingestion jobs is straightforward; for the most part, you just go to the connectors page of OpenMetadata [30], look for your tool, click on it, and configure the metadata to be ingested as detailed in the sections below.

6.2.1 Airbyte

- **Service Connection**: Configured a pipeline service named airbyte_con.
- **Metadata Ingestion:** Set up one metadata ingestion job, extracting metadata from three Airbyte pipelines.

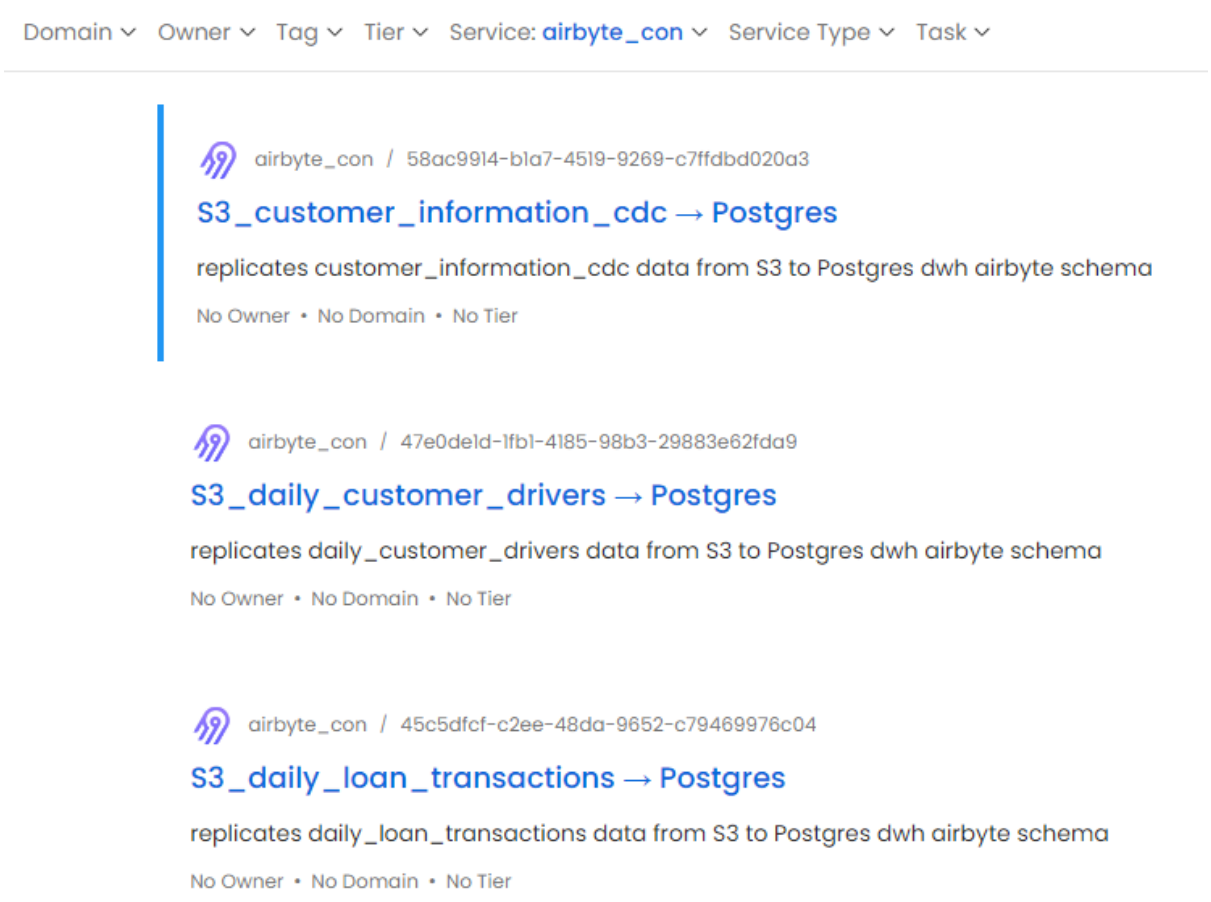


Figure 16 Represents three Airbyte pipelines with descriptions, used to replicate data located in the wb-om bucket to the Airbyte schema in the Postgres data warehouse.

6.2.2 Airflow

- Service **Connection**: Established a pipeline service named airflow_con.
- Metadata **Ingestion**: No specific configuration needed due to the integration of Airflow lineage backend with OpenMetadata, which uses OpenLineage to push Airflow metadata. This setup pushes DAG graphs, execution times, and lineage.

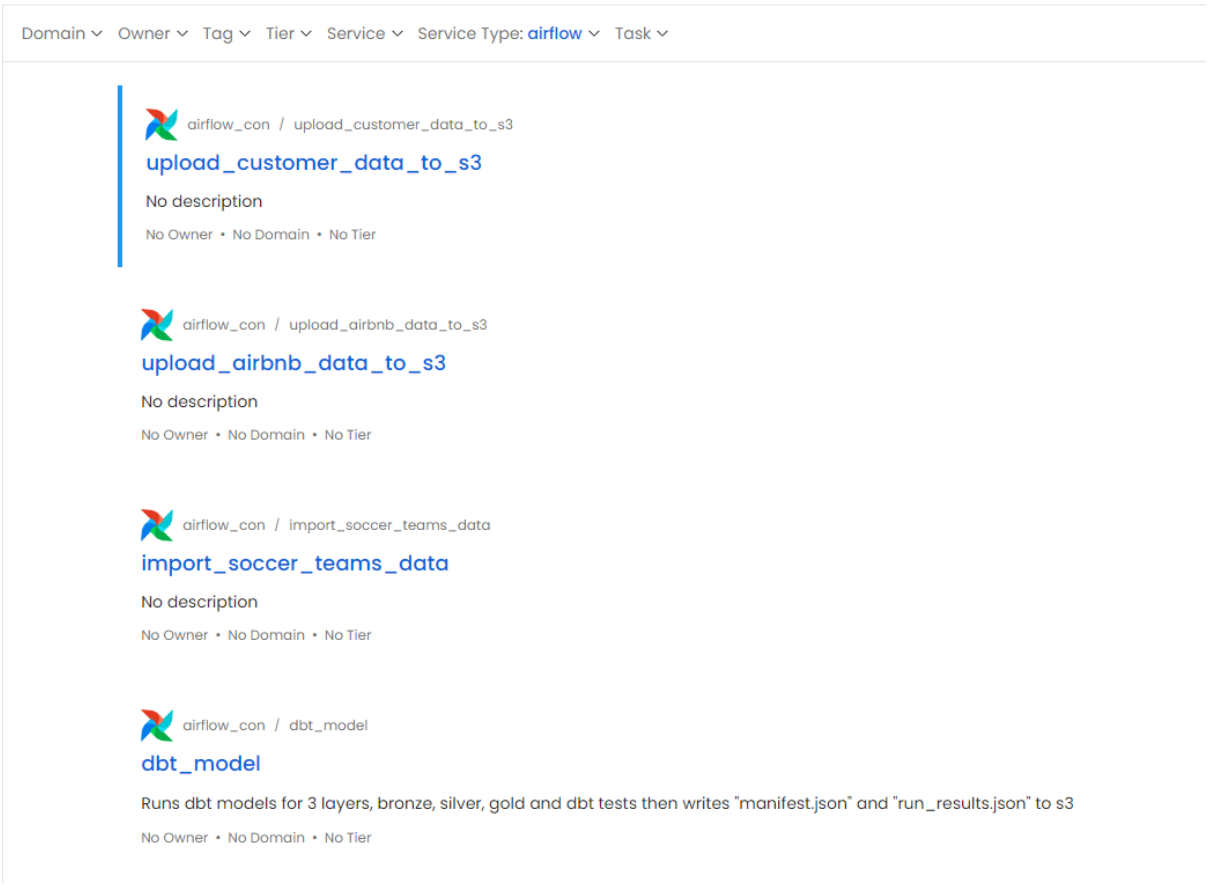


Figure 17 shows the Airflow pipelines, highlighting the four pipelines used for Airflow DAGs or pipelines.

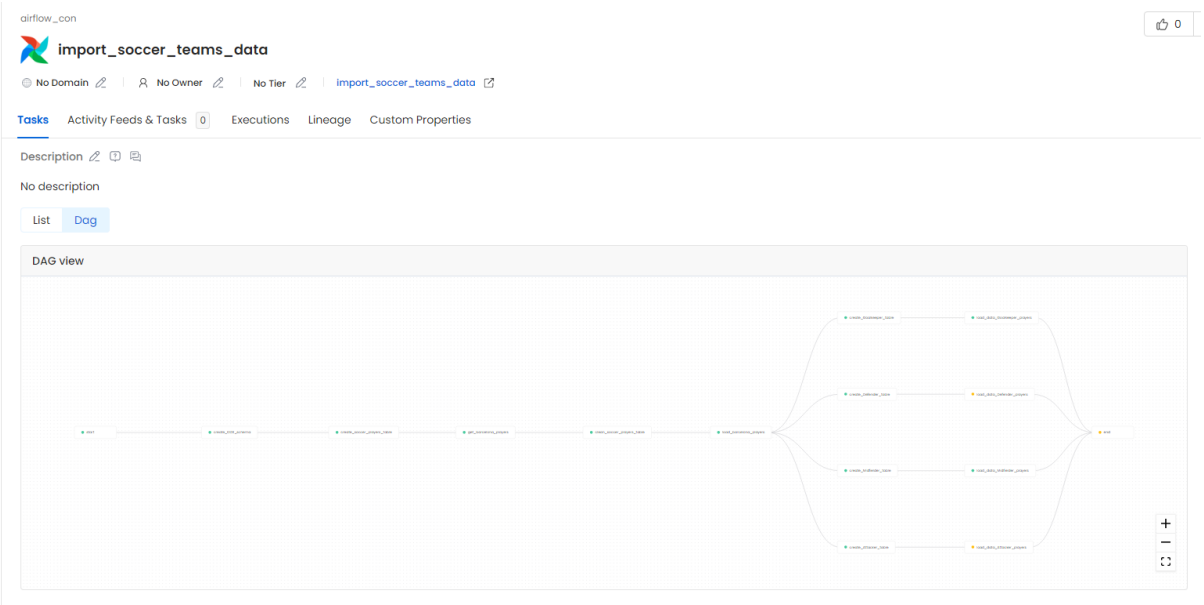


Figure 18 illustrates the Airflow DAG representation in OpenMetadata, showing the graph of the DAG used to create the ODS schema and insert data for the players' dataset from an API as shown previously in Figure 13.

6.2.3 Postgres Data Warehouse (DWH)

- **Service Connection:** Set up a database service named postgres_con for the Postgres DWH.
- **Metadata Ingestion Jobs**: Configured five key metadata ingestion jobs.

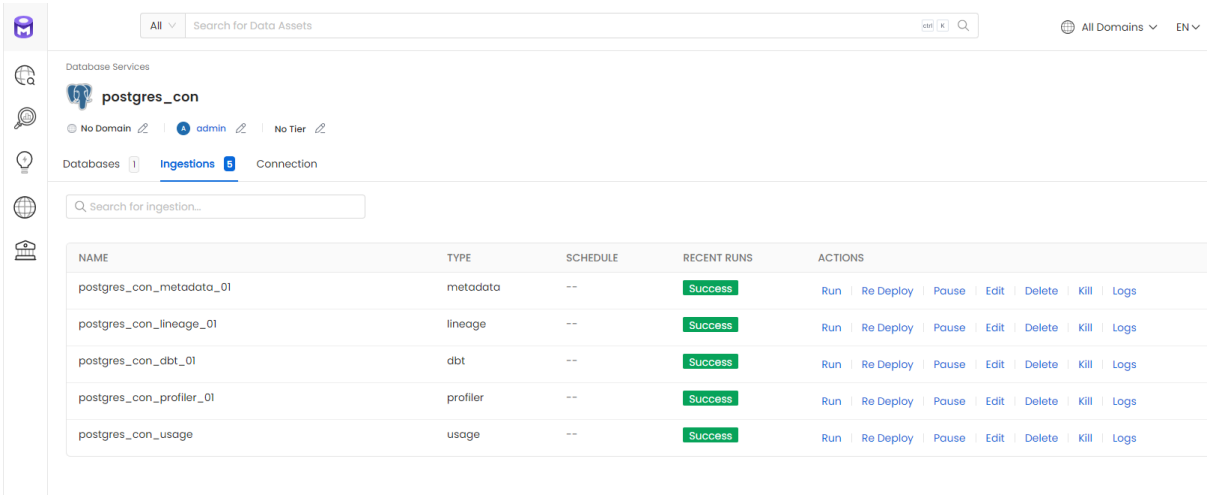


Figure 19 postgres_con ingestions

- **Metadata Ingestion:** Extracts metadata from the Airbyte, Bronze, Silver, Gold, and ODS schemas, including table names and schemas.

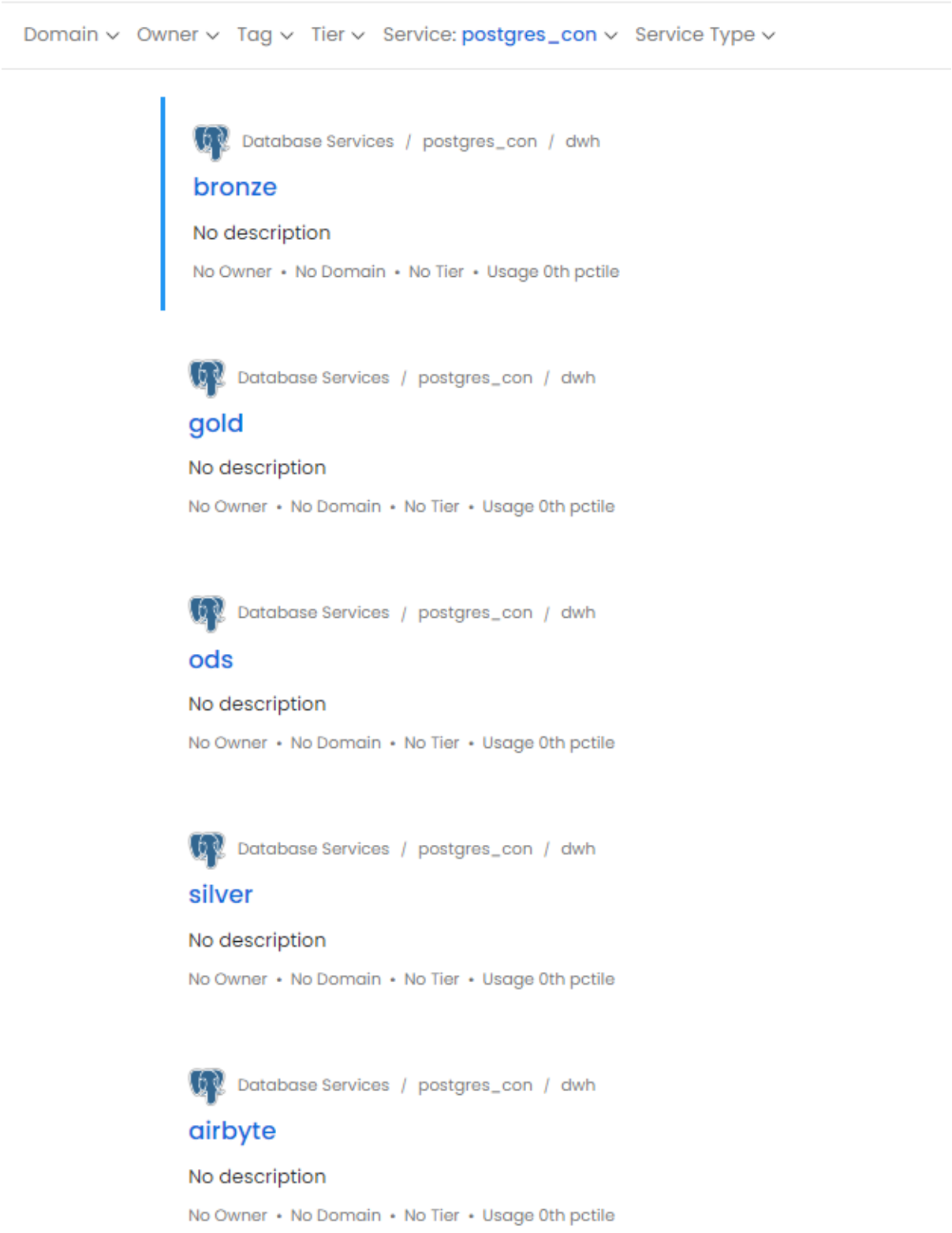


Figure 20 showcases the five schemas within the Postgres DWH.

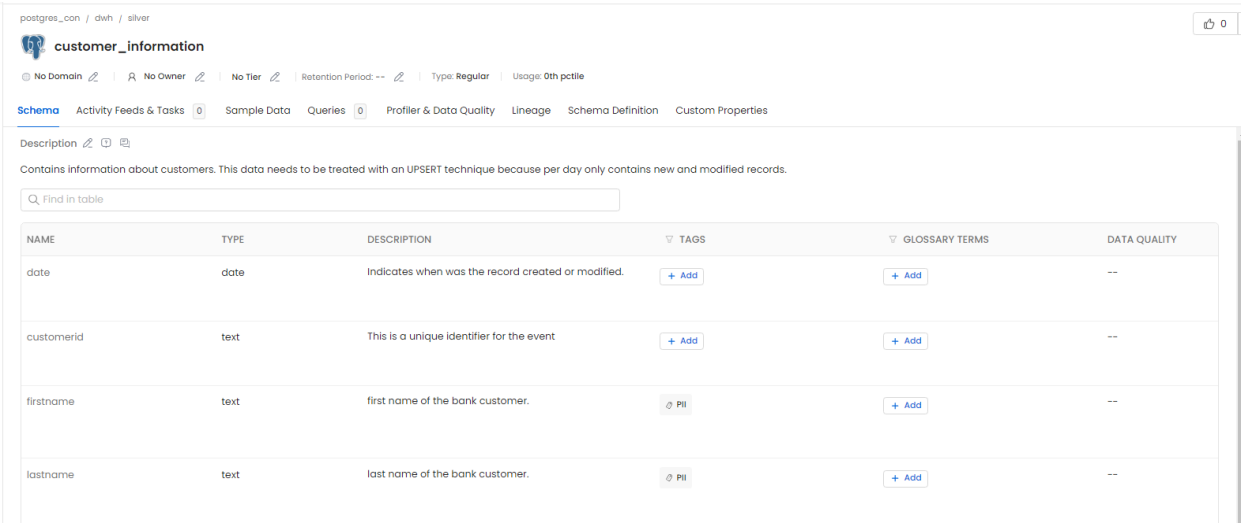


Figure 21 provides an example of a table name and schema, showing the table schema as well as owner description, etc.

- **Profiler Job**: Gathers comprehensive metrics from all ingested tables, including sample data, row and column counts, update frequency, changes in data volume, and unique and null percentage values at the column level.

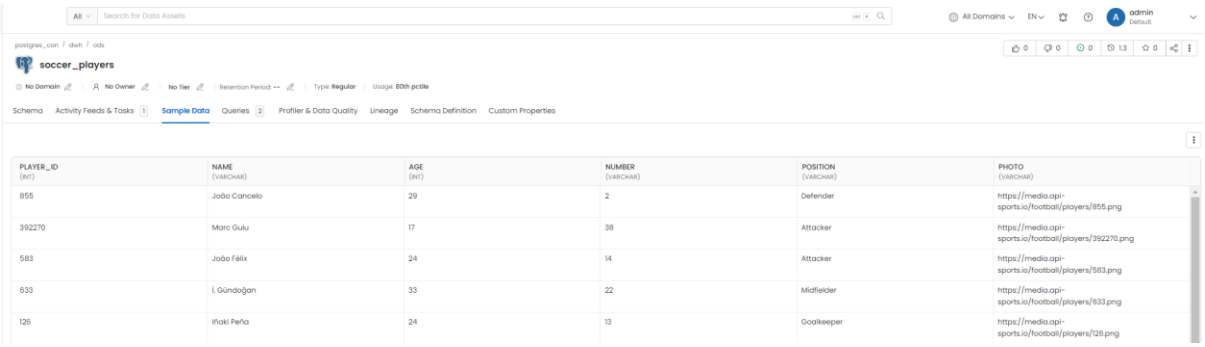


Figure 22 shows a data sample after profiling.

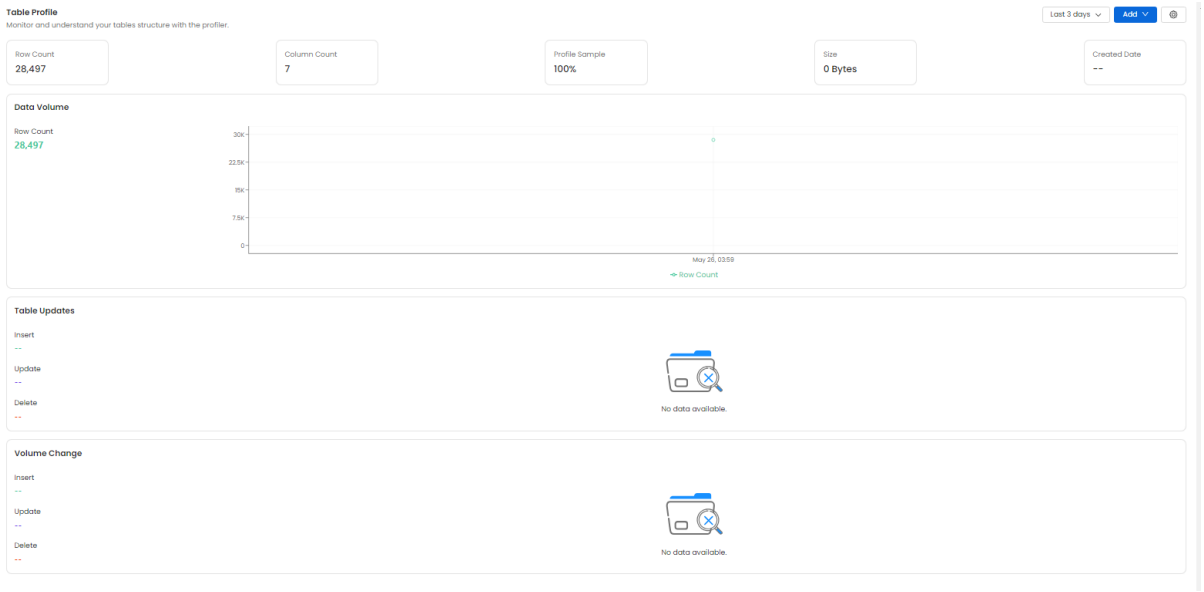


Figure 23 represents a table-level profile, including row count, data volume over time, and column count. Note that table updates (freshness) and volume change over time are not supported by Postgres but can be displayed for BigQuery or Snowflake.

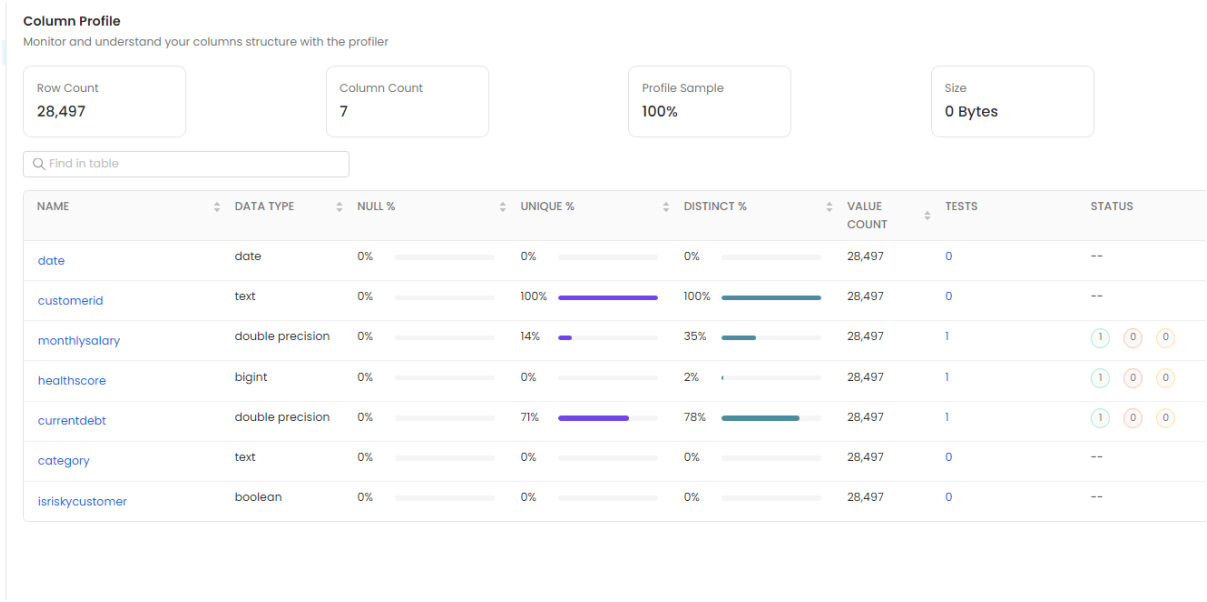


Figure 24 showcases a column-level profile, including column datatype, null percentage, unique percentage, value counts, test on that specific column, and test status.

- **DBT Ingestion Job**: Retrieves “run_results.json” and “manifest.json” from a DBT project stored in S3 after processing, including data quality checks, lineage data, and detailed metadata.

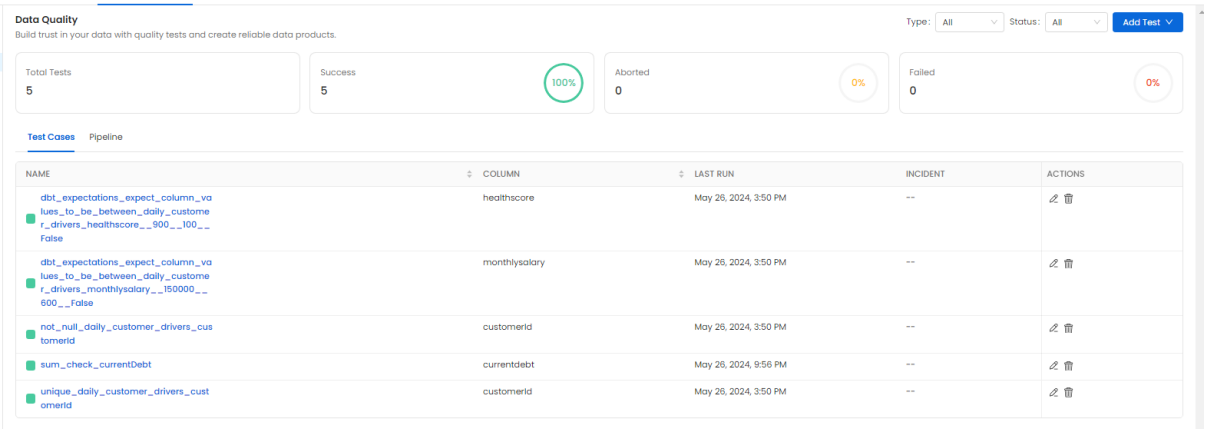


Figure 25 DBT displays DBT test ingestion results, showing a dashboard of total tests, including successful, aborted, and failed test cases.

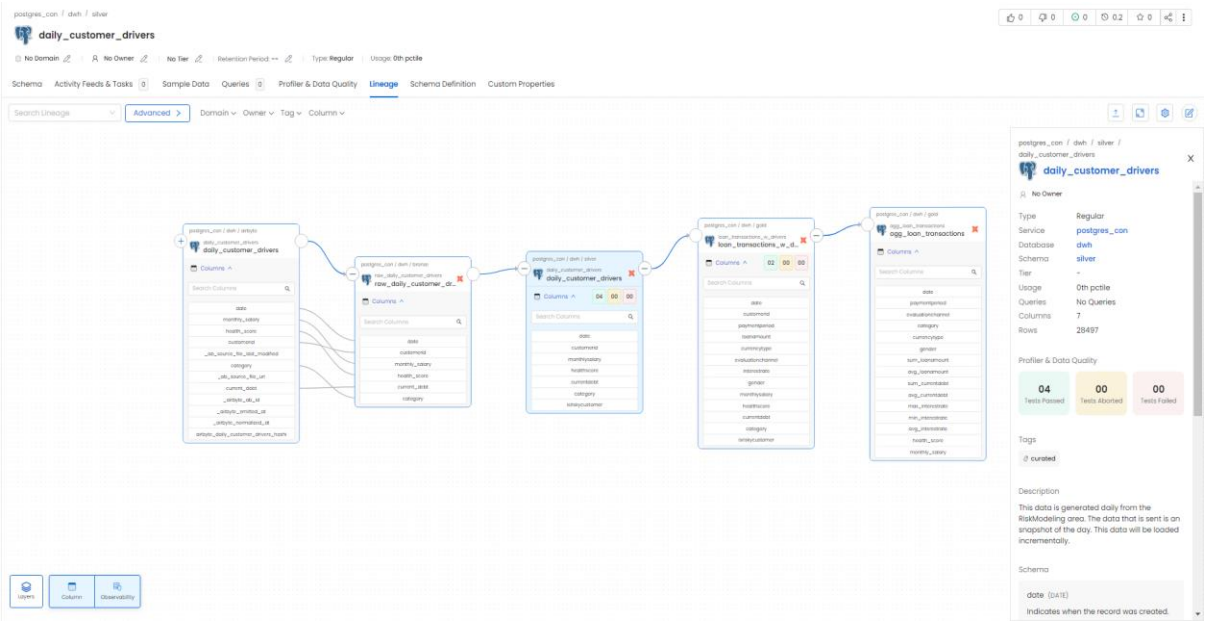


Figure 26 illustrates the lineage graph tab, showing DBT expectations (test results), lineage, tags, descriptions.

- **Lineage Job**: Automatically captures and organizes lineage information for specified schemas, particularly focusing on the ODS schema.

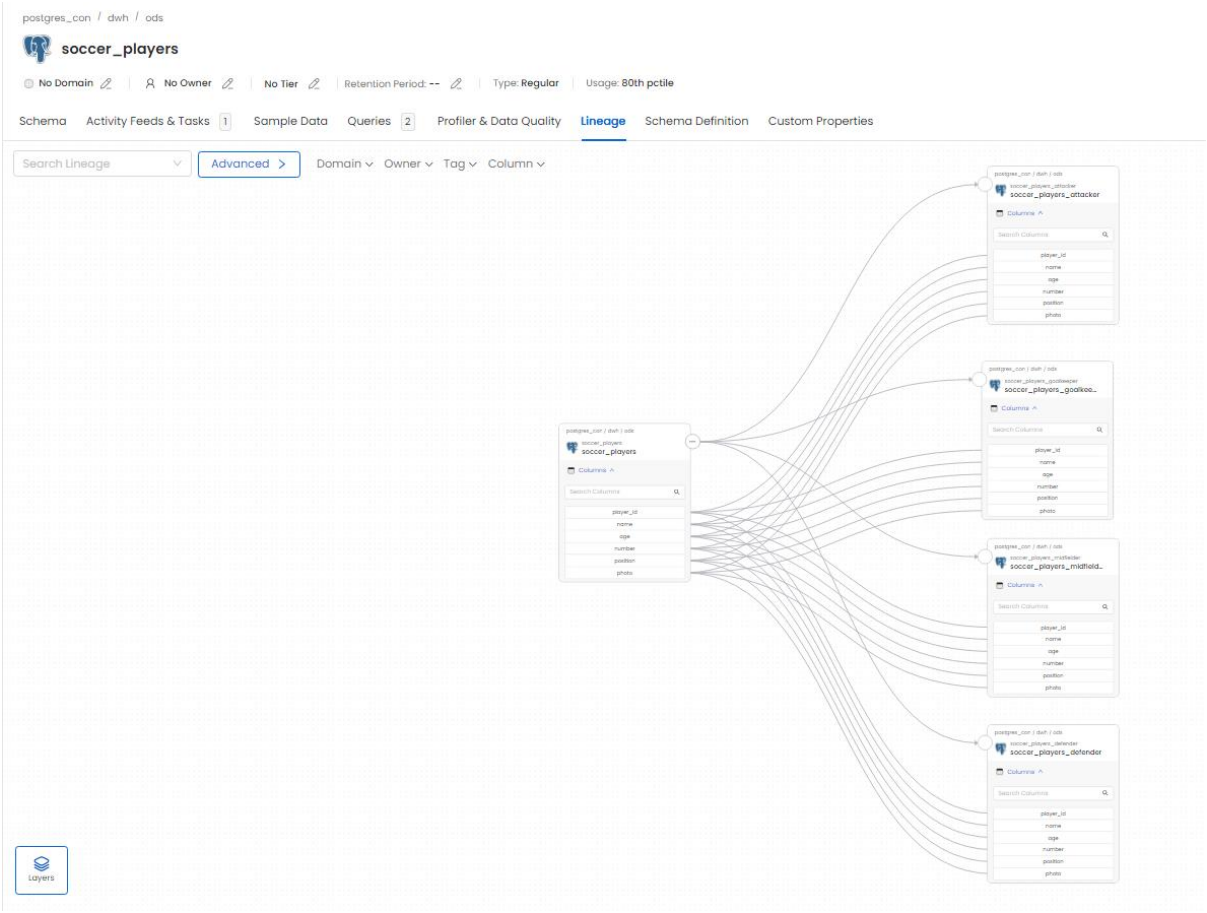


Figure 27 shows an image of the "soccer_players" table from the Postgres ODS schema, showcasing automatic column-level lineage extraction from Postgres (no DBT, ETL was done in an Airflow DAG).

- **Usage Job**: Collects and analyses historical query data for performance audits and system optimizations.

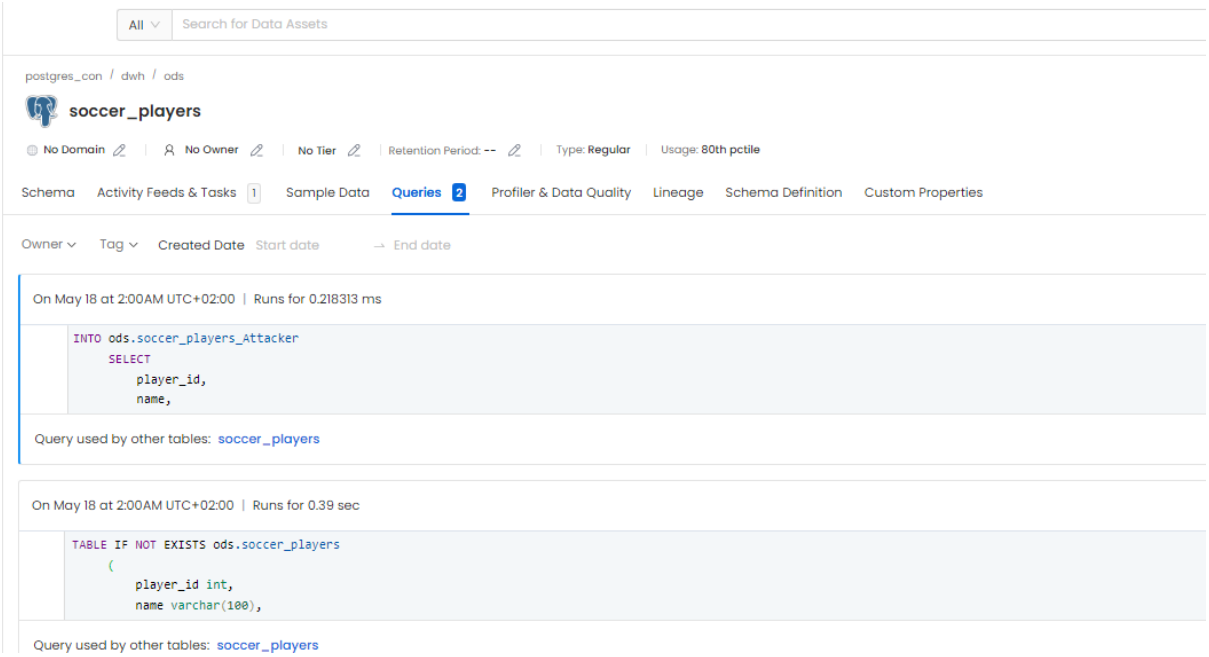


Figure 28 illustrates query usage history, execution timestamp, and execution time (duration).

6.2.4 Hive Metastore

- **Service Connection**: Configured Hive database service “hive_con” with two ingestion jobs.

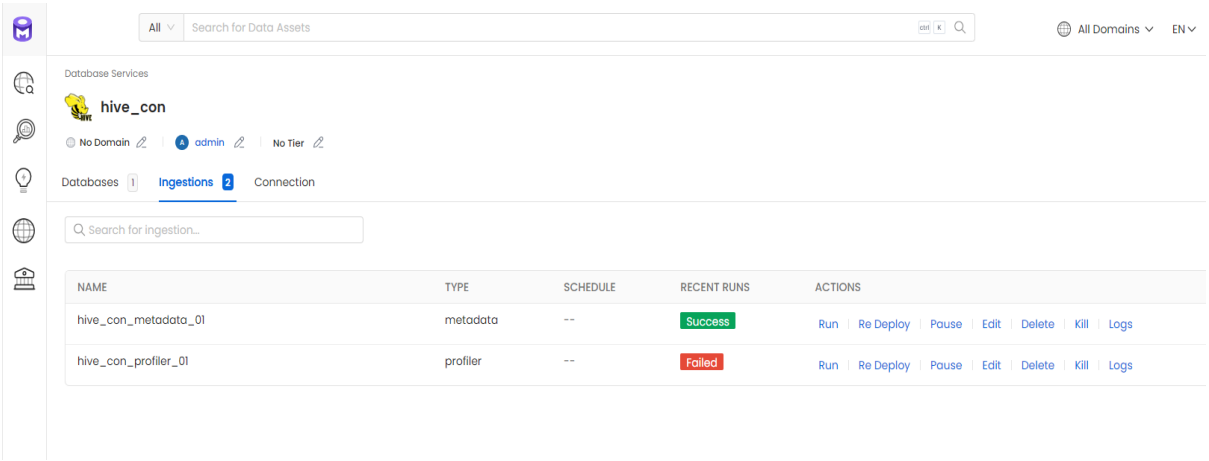


Figure 29 shows “hive_con” ingestions.

  - **Metadata Ingestion**: Ingests metadata from the bronze, silver, and gold schemas within the Hive metastore, along with their table structures.

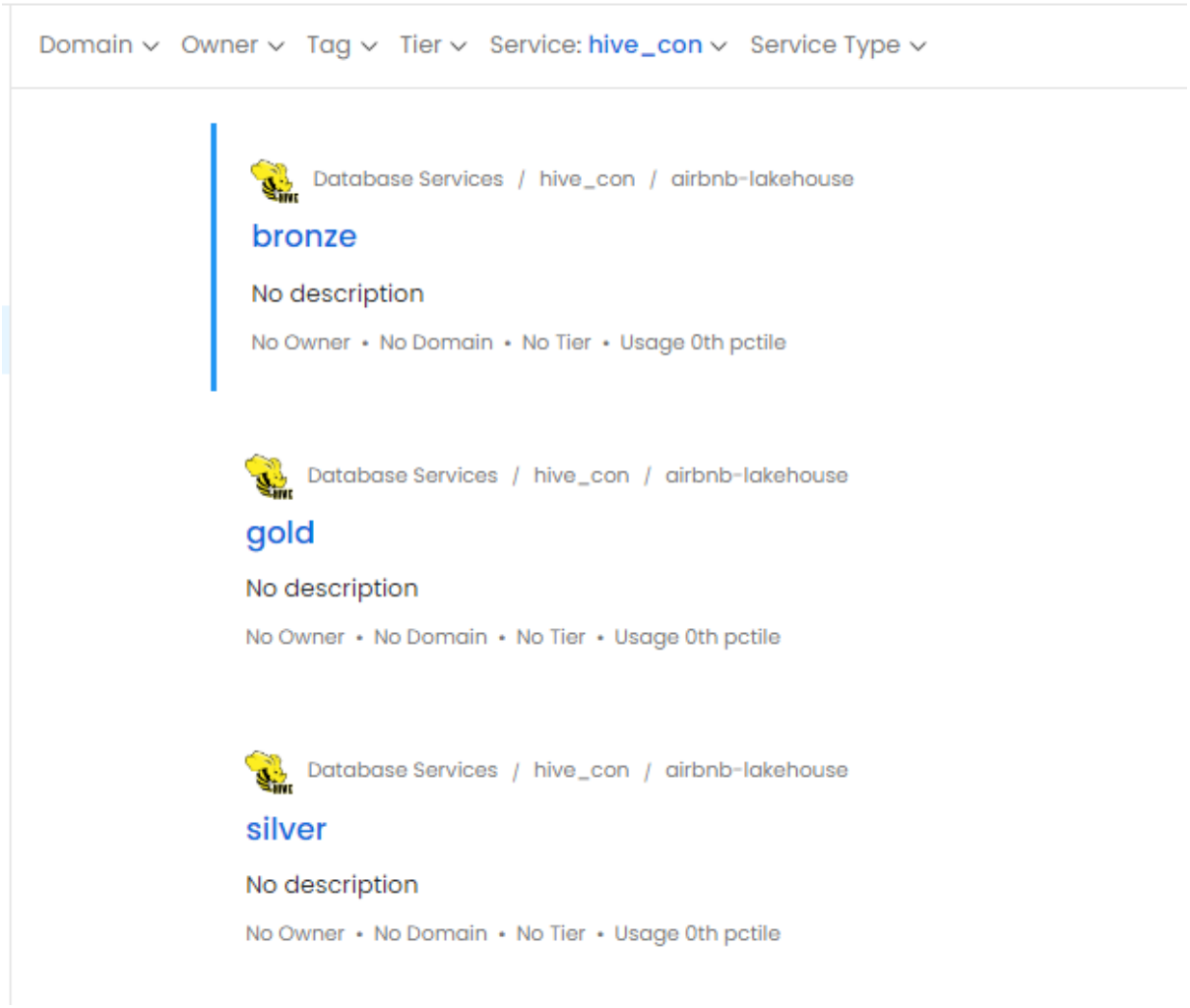


Figure 30 shows all three Hive schemas (Bronze, Silver, Gold).

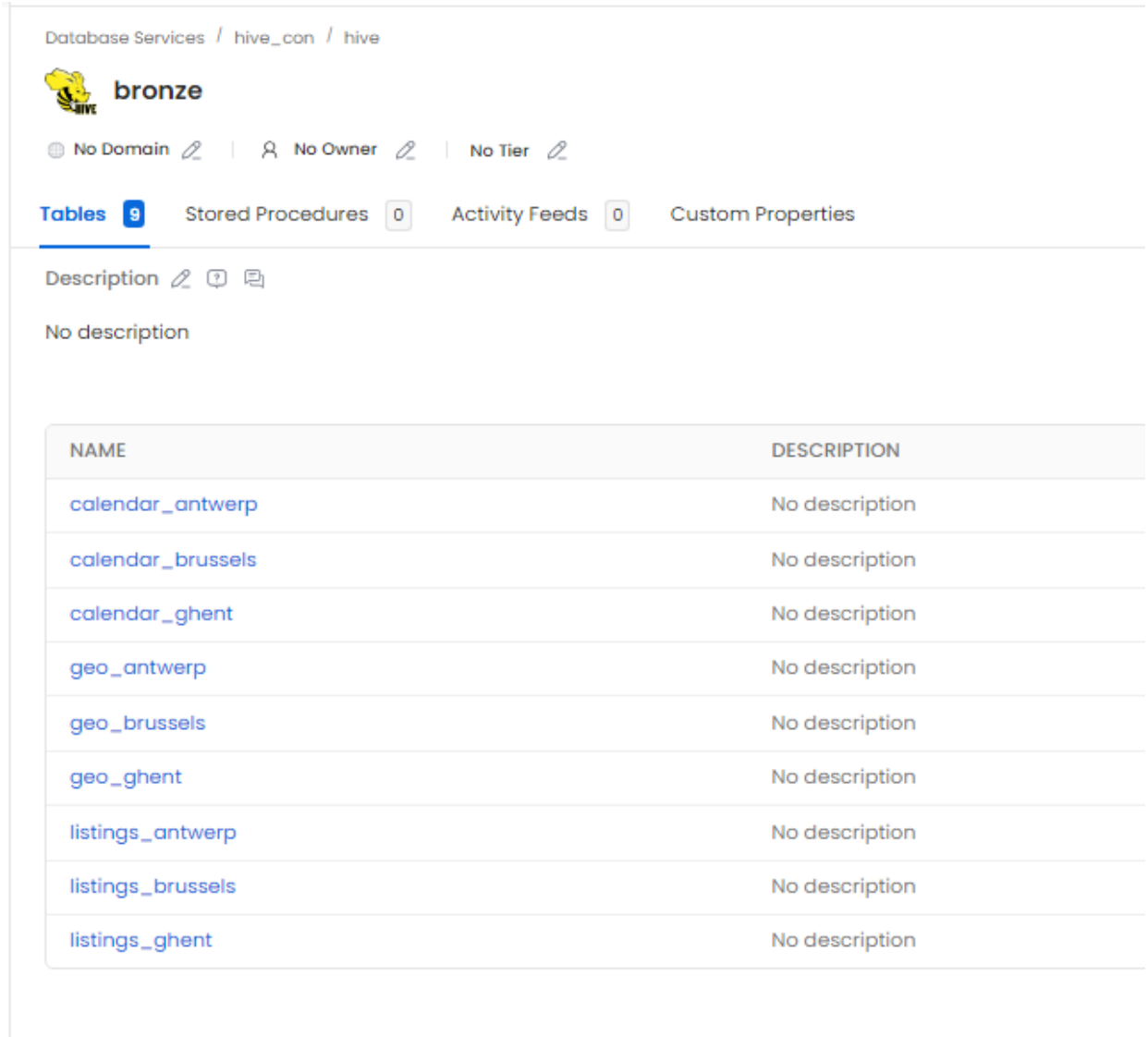


Figure 31 provides an example of tables inside the bronze schema.

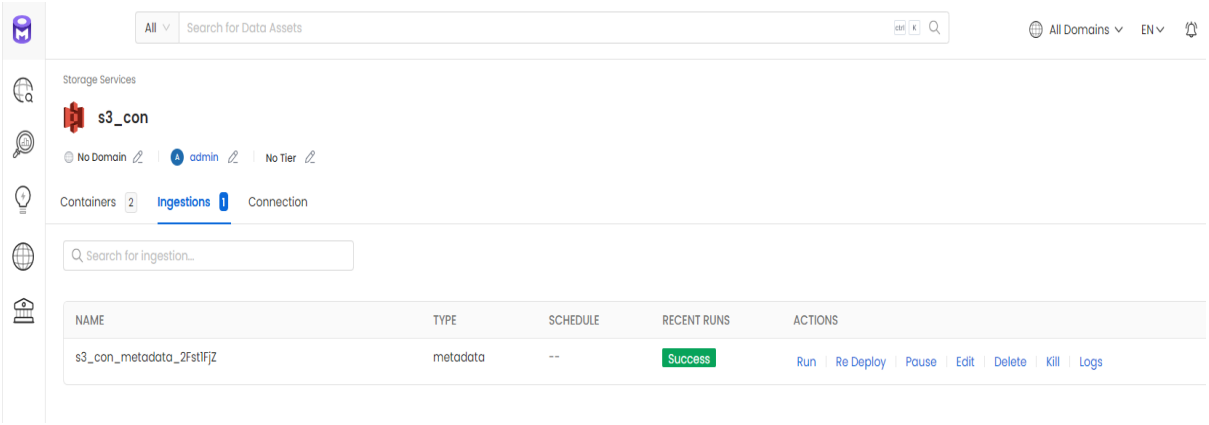
- **Profiler Ingestion Job**: Faces limitations as the functionality is for Hive data warehouse and not Hive metastore, leading to job failures.

```
[2024-05-24T00:56:14.419+0000] {server_mixin.py:75} INFO - OpenMetadata client running with Server version [1.4.0] and Client version [1.4.0.0]
[2024-05-24T00:56:14.618+0000] {metadata.py:107} INFO - Starting profiler for service hive_con:hive
[2024-05-24T00:56:18.842+0000] {test_connections.py:215} INFO - Test connection results:
[2024-05-24T00:56:18.843+0000] {test_connections.py:216} INFO - failed=[] success=["'CheckAccess': Pass", "'GetSchemas': Pass", "'GetTables': Pass", "'GetViews': Pass"] warning=[]
[2024-05-24T00:56:20.369+0000] {profiler_interface.py:60} WARNING - Error trying to compute profile for calendar_antwerp.adjusted_price: TSocket read 0 bytes
[2024-05-24T00:56:20.370+0000] {profiler_interface.py:458} ERROR - calendar_antwerp.adjusted_price metric_type.value: TSocket read 0 bytes
[2024-05-24T00:56:20.371+0000] {status.py:76} WARNING - calendar_antwerp.adjusted_price metric_type.value: TSocket read 0 bytes
[2024-05-24T00:56:20.374+0000] {profiler_interface.py:202} WARNING - Error trying to compute profile for calendar_antwerp: TSocket read 0 bytes
[2024-05-24T00:56:20.375+0000] {profiler_interface.py:458} ERROR - calendar_antwerp metric_type.value: TSocket read 0 bytes
```

Figure 32 shows the logs of a profiler job failure with a TSocket 0 bytes error.

6.2.5 AWS S3 Containers

- **Service Connection**: Configured S3 storage service s3_con with one ingestion job.



- **Metadata Ingestion Job**: Connects to S3 to ingest metadata, specifically targeting the "wb-om" & "wb-airbnb" buckets.

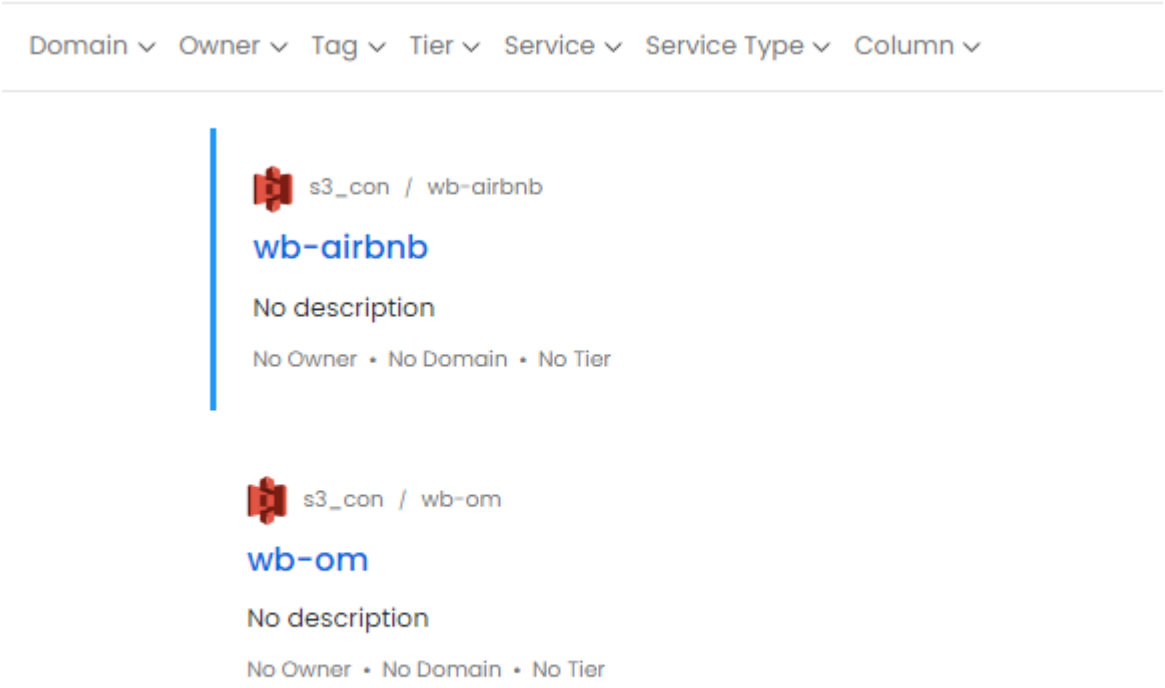


Figure 33 lists the two ingested buckets based on a preset filter.

- **Bucket children listing task**: Encounters permission issues when trying to list contents within the bucket, requiring a review and adjustment of CloudWatch and IAM settings to resolve.

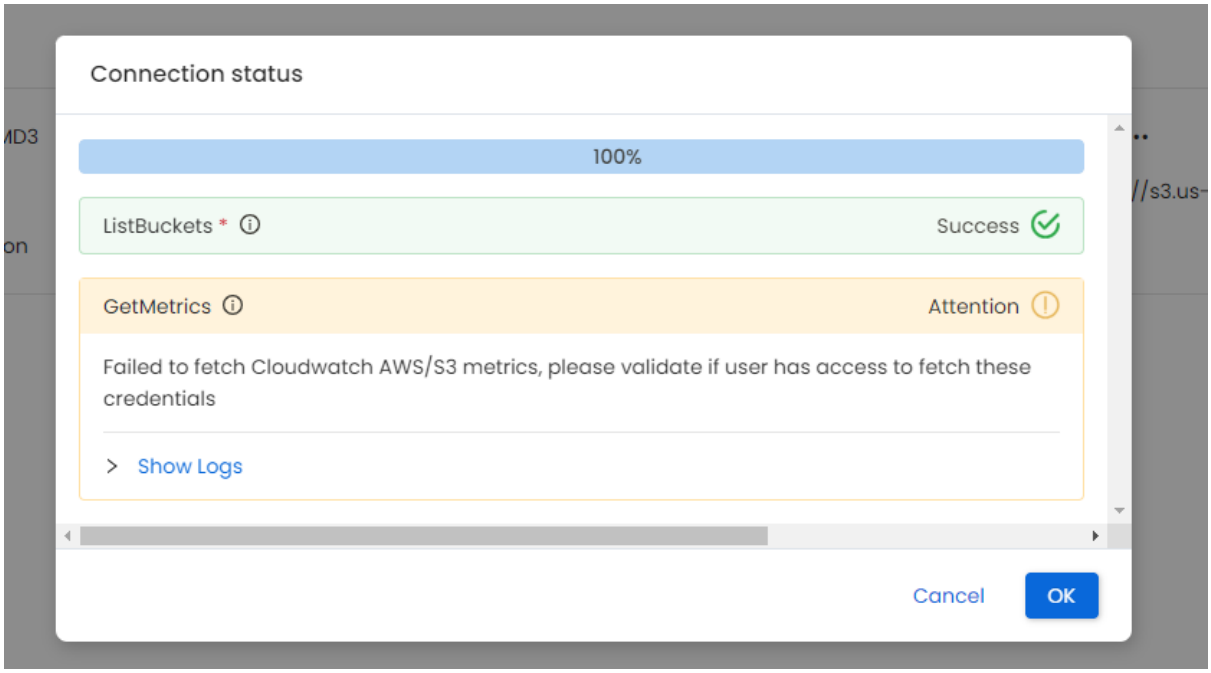


Figure 34 shows an error with the GetMetrics function, caused by insufficient permissions (unable to list bucket children), only listing buckets at a high level.

6.3 Addressing Challenges:

- **Hive metastore Profiler Job**: Consider upgrading to Hive data warehouse for compatibility.
- **S3 Bucket Listing**: Correcting IAM roles and CloudWatch permissions is essential for resolving access and listing issues, ensuring successful metadata management.

7 POC Results:

In this section, we will evaluate whether we have achieved our objectives:

- **Comprehensive Data Monitoring:** Ensuring the five pillars of data observability: freshness, quality, schema, lineage, and volume of data.
- **Real-Time Alerts**
- **Data Glossary and Catalog**

7.1 Comprehensive Data Monitoring:

Using OpenMetadata and dbt tests, we have implemented a comprehensive set of tests to ensure data in our data warehouse is consistently monitored. This ensures data integrity and reliability across all our data connectors. Our data profiler aids in understanding data configurations and usage, which is crucial for maintaining data integrity over time.

7.1.1 Data Quality

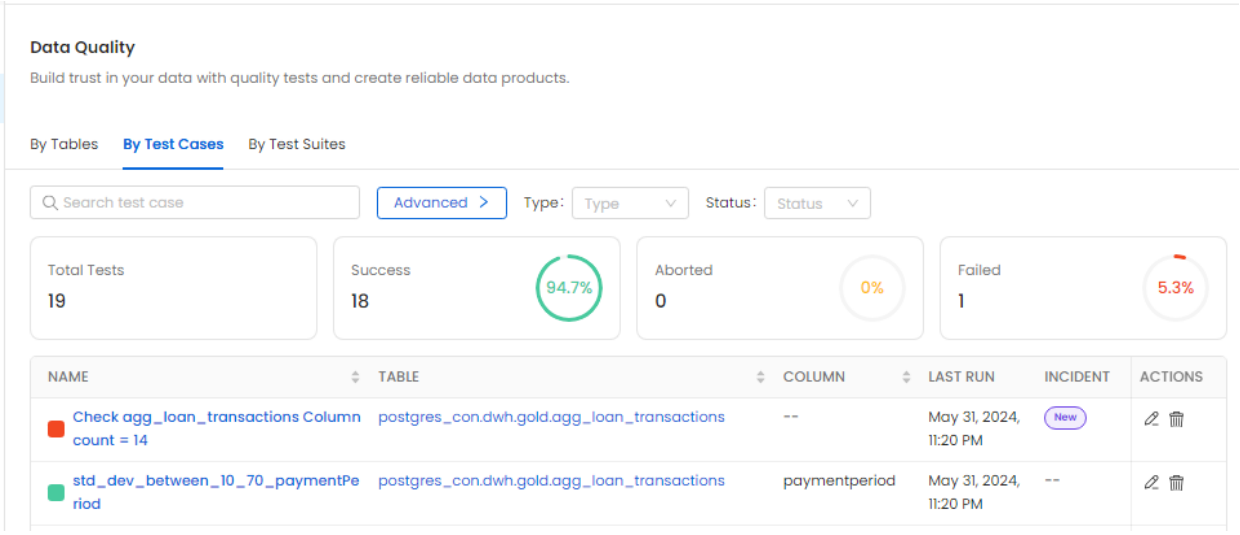


Figure 35 Data Quality Dashboard 1

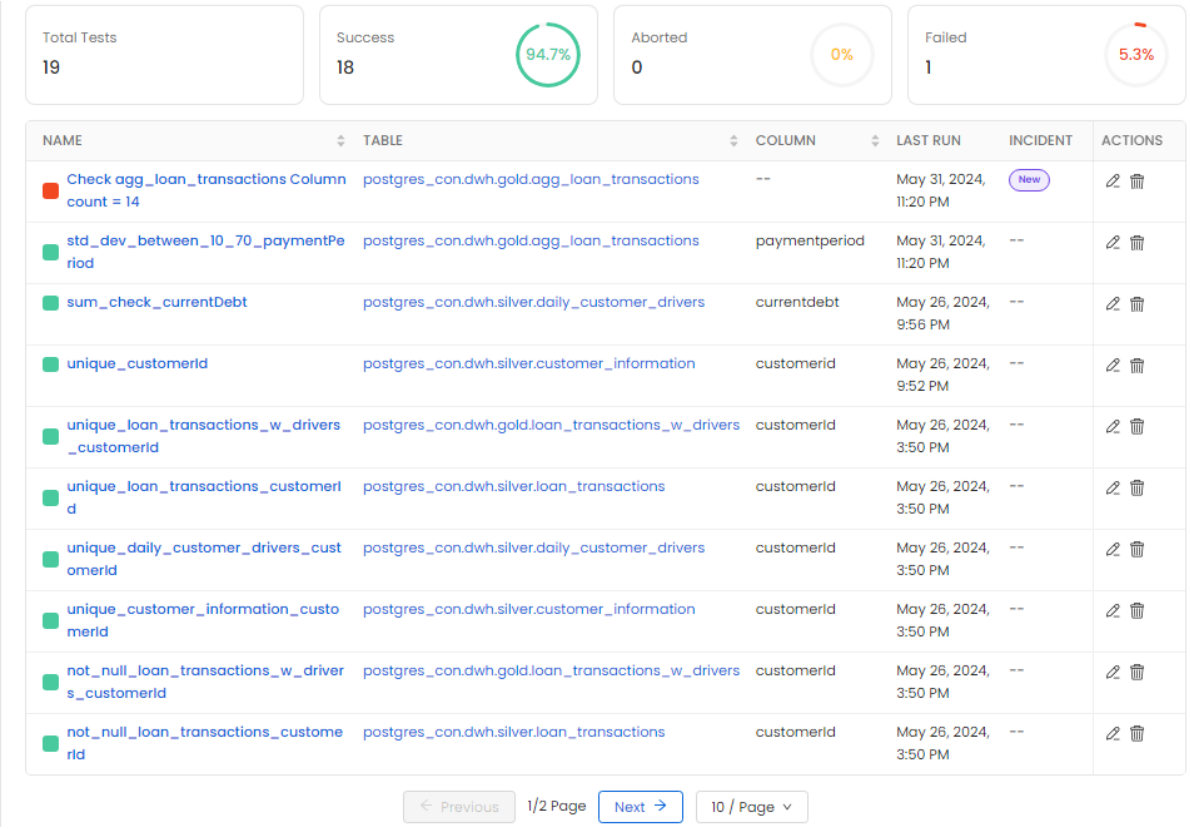


Figure 36 Data Quality Dashboard 2

- **Figure 35 & 36: Data Quality Dashboard** - These dashboards showcase a comprehensive view of data quality tests, highlighting success rates and detailed test results for quick assessments. The filter panel allows for easy navigation and pinpointing specific data quality issues.

- The data quality dashboards provide real-time insights into various aspects of data quality, including accuracy, consistency, and reliability. By identifying and addressing issues such as duplicate records or missing data promptly, we have significantly improved the overall quality of our data. This proactive approach ensures that our analytics and decision-making processes are based on trustworthy data.

7.1.2 Schema

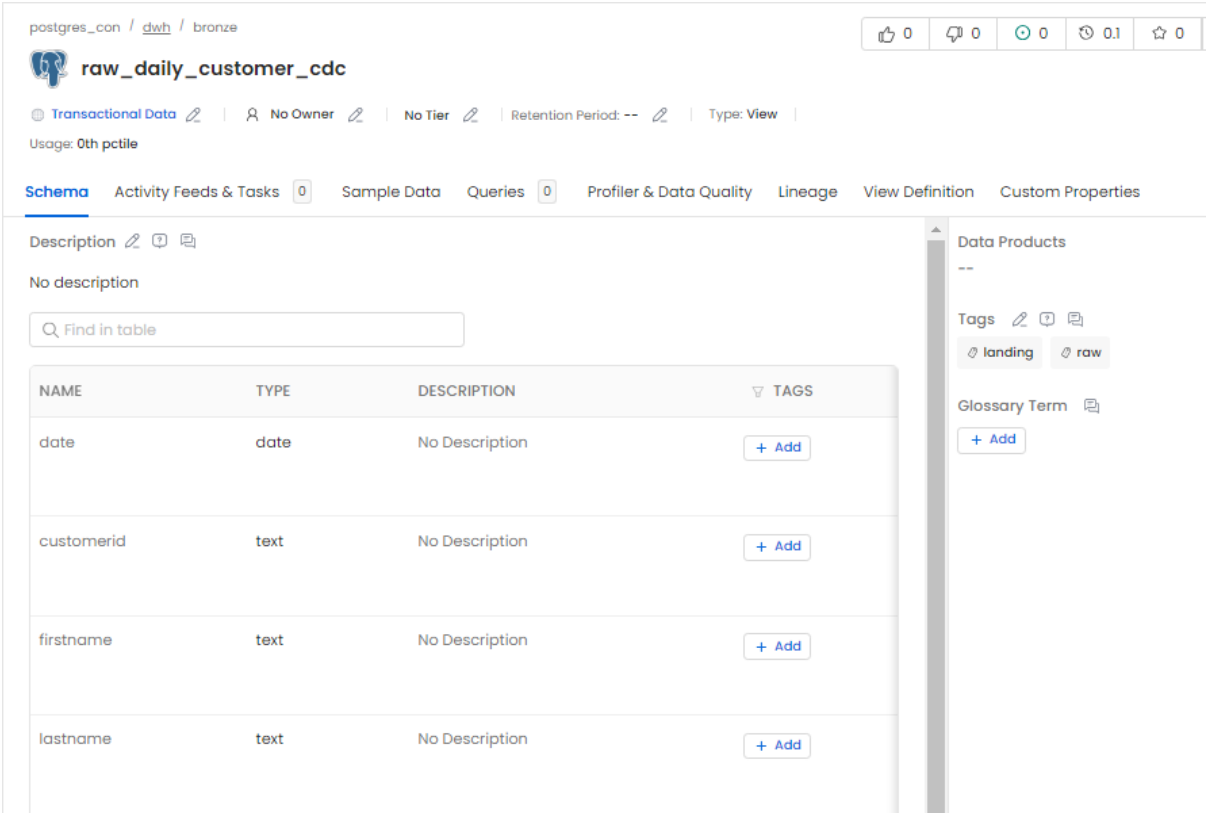


Figure 37 Schema example 1 (Daily Customer View at bronze layer)

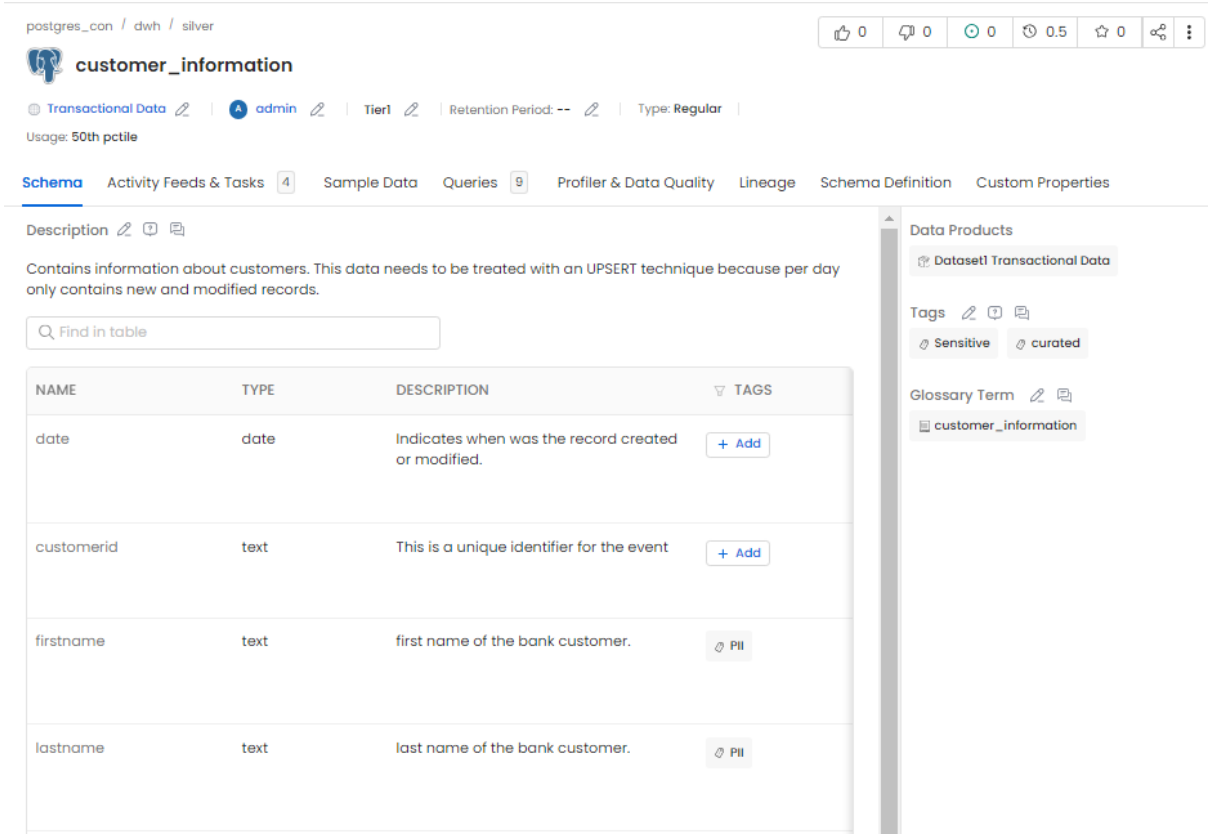


Figure 38 Schema example 2 (Daily Customer Table at silver layer)

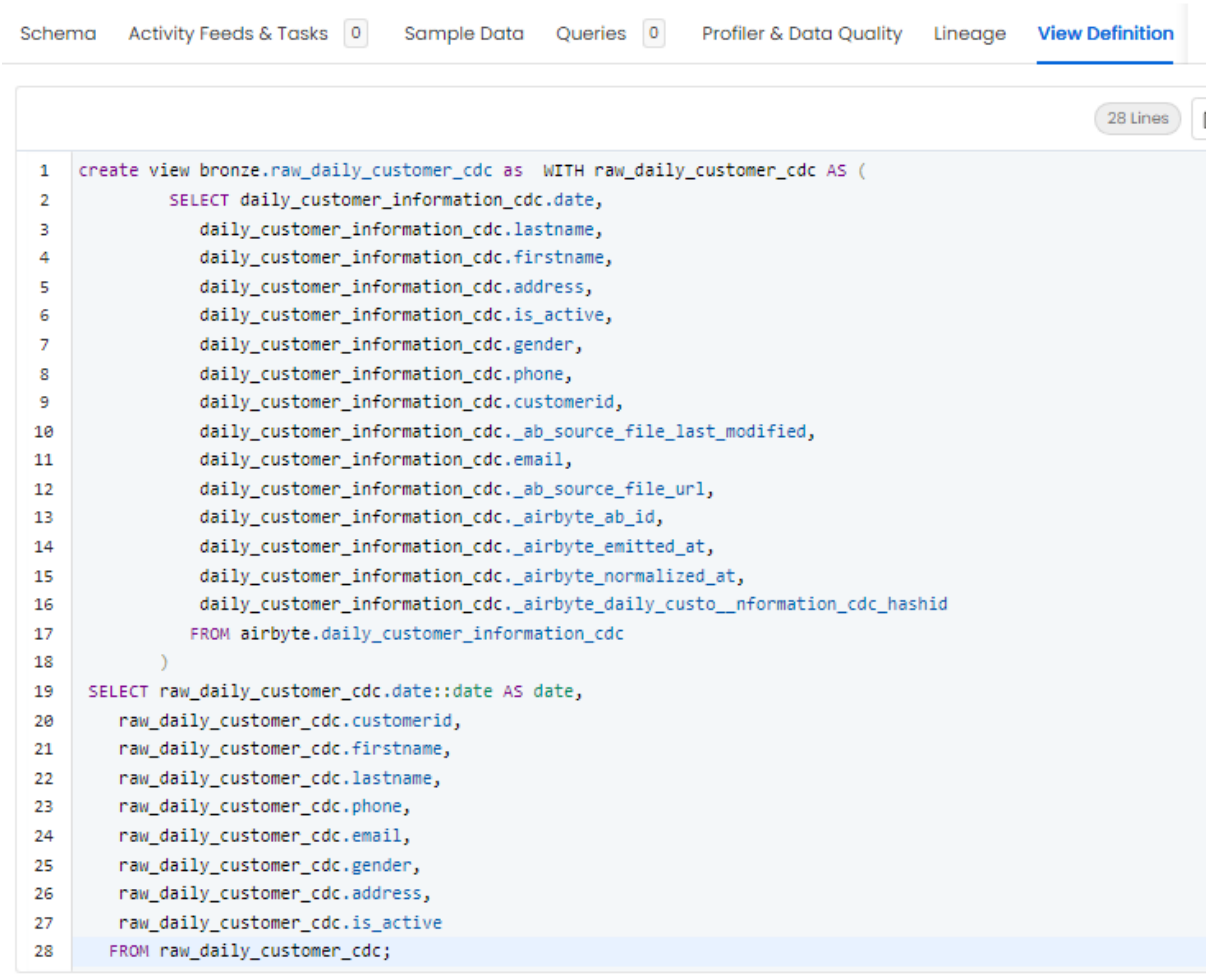


Figure 39 Schema definition example (Daily Customer View definition at bronze layer)

- **Figure 37 & 38: Schema Examples** - Demonstrates our capabilities in maintaining consistent dataset schemas across databases (Bronze - Silver).
- **Figure 39**: Schema Definition Example.

Maintaining consistent schema integrity across different layers (Bronze to Silver) is essential for data reliability. Figures 37 and 38 demonstrate how our schema definitions remain consistent through the transformation stages, ensuring that data transformations do not introduce schema inconsistencies. Figure 39 provides detailed schema definitions, enhancing documentation and aiding in data governance. This consistency helps prevent data-related errors and ensures smooth operation of dependent systems.

7.1.3 Lineage

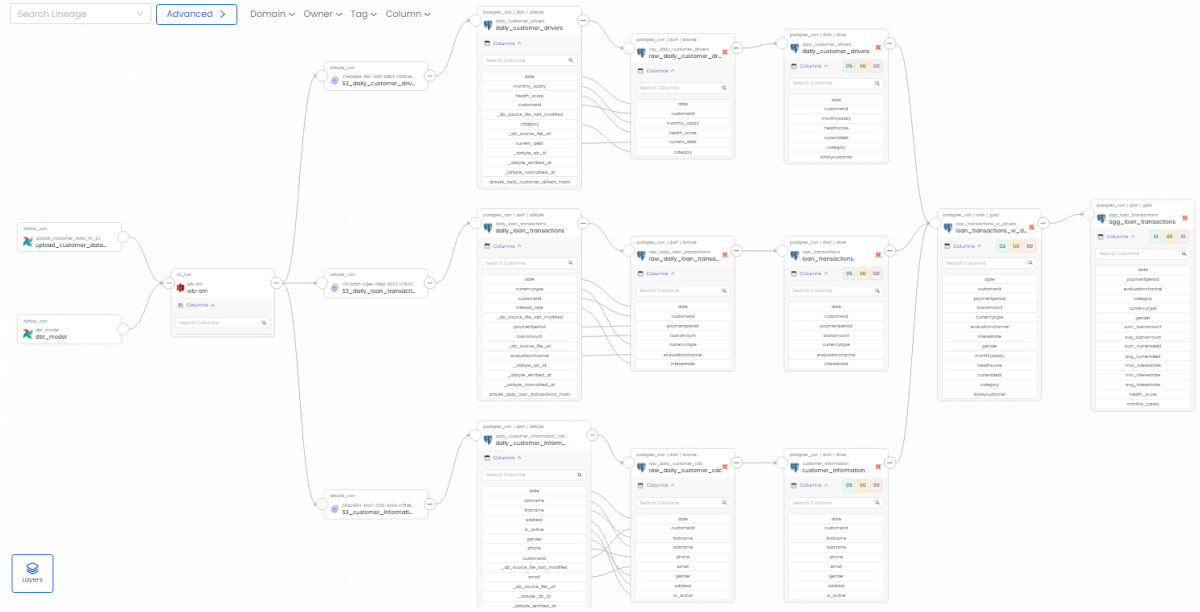


Figure 40 End to End Lineage example 1

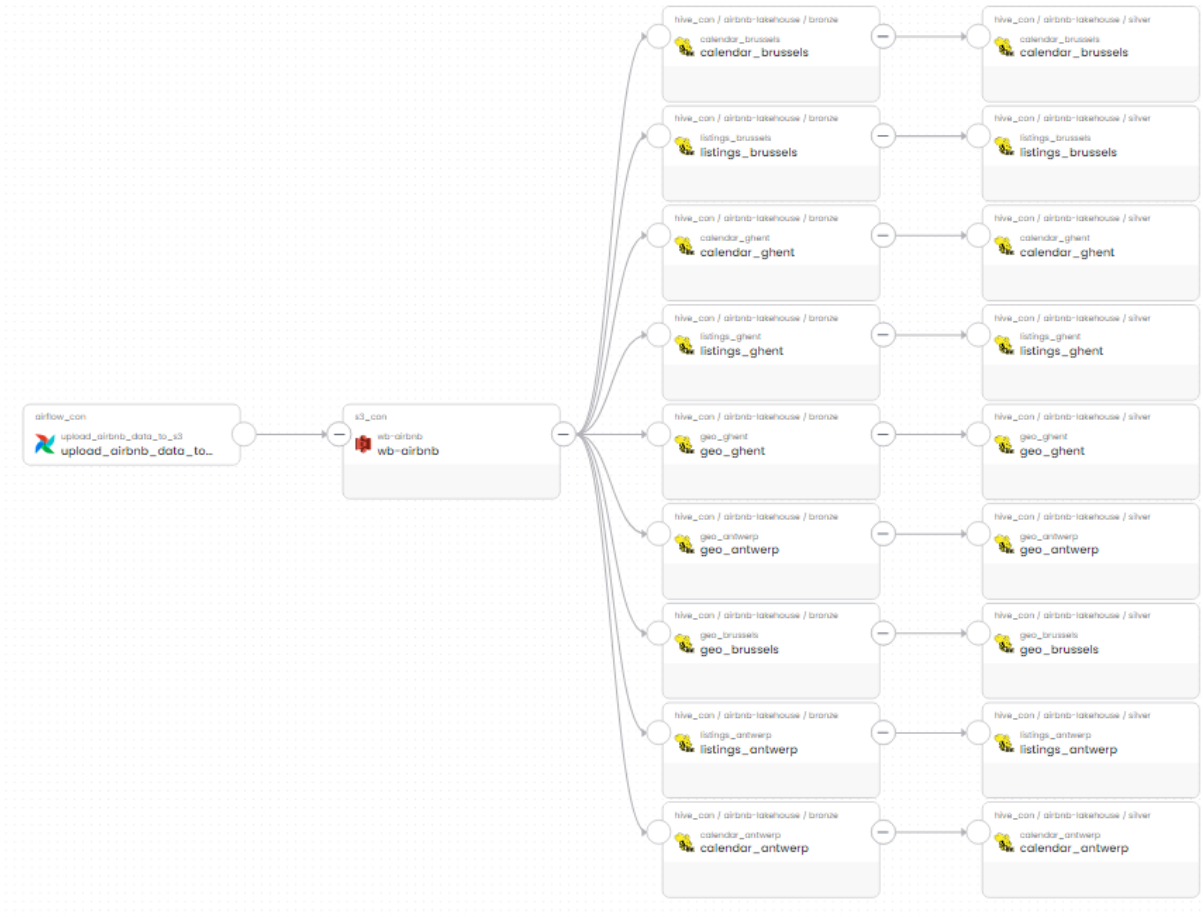


Figure 41 End to End Lineage example 2

- **Figure 40 & 41: Lineage Tracking** - Illustrates the complete path data travels through, from source to destination, ensuring transparency and trust.

- Lineage tracking is crucial for understanding data flow and dependencies. It provides a clear visual representation of the data journey from source to destination, making it easier to trace issues back to their origins. This has been instrumental in debugging, auditing, and ensuring compliance with data governance policies. By knowing the exact path data takes, we can quickly identify and resolve any discrepancies, ensuring data accuracy and reliability.

7.1.4 Volume

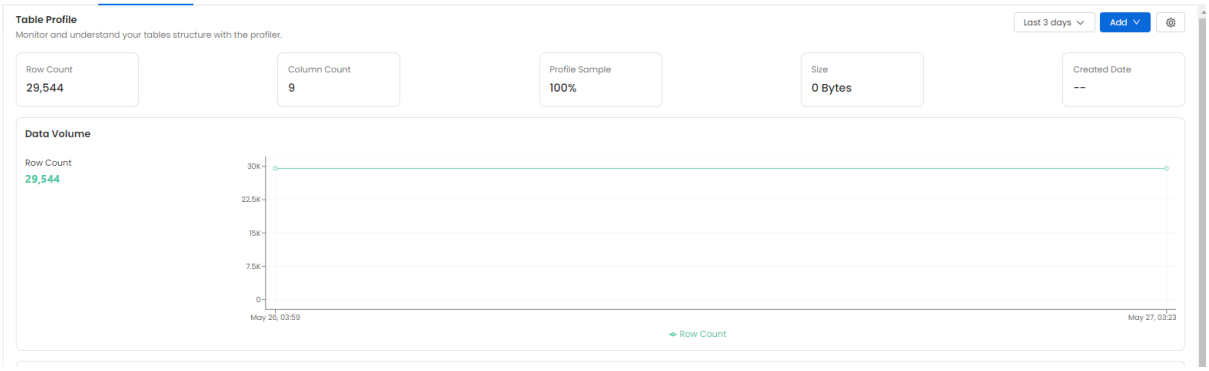


Figure 42 Data Volume example

- **Figure 41: Data Volume Monitoring** - Tracks data volume to manage capacity planning effectively.
- Monitoring data volume helps us manage storage resources efficiently and plan for future capacity needs. By keeping track of data growth and usage patterns, we can anticipate and prevent potential issues related to data storage and processing capabilities. This proactive management ensures that our data infrastructure remains scalable and capable of handling increasing data loads without compromising performance.

7.1.5 Freshness

Figure 43 Data freshness and volume change over time, picture from Collate sandbox

- **Figure 43: Data Freshness Visualization** - Details how recent and relevant our data remains, crucial for dynamic and timely decision-making.
- Ensuring data freshness means our analytics are based on the most current data, enhancing the accuracy and relevance of our insights. This is particularly important for time-sensitive applications where outdated data can lead to incorrect conclusions. It's important to note that the data freshness feature is not available for the Postgres data warehouse we are currently using. The picture shown is taken from the Collate SaaS sandbox environment to illustrate the functionality for display purposes. By continuously monitoring data freshness in environments where it's supported, we can provide stakeholders with up-to-date information, supporting effective decision-making and operational efficiency.

7.2 Real-Time Alerts:

Immediate Notifications for Data Issues

OpenMetadata simplifies monitoring, allowing quick setup of alerts for changes in metadata, data quality failures, pipeline status, and more. Custom notifications ensure that teams are immediately aware of issues, enhancing proactive management.

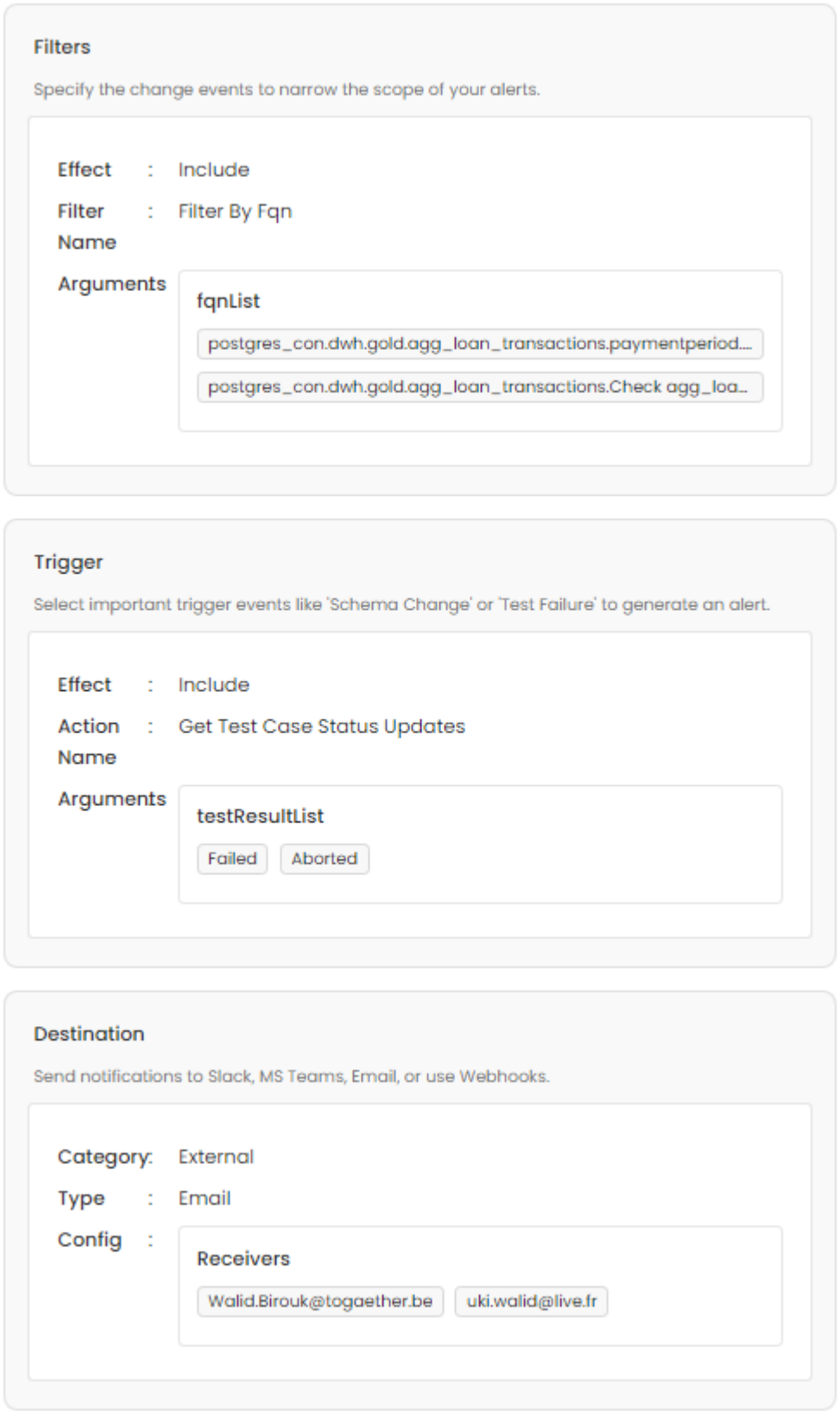


Figure 44 Alert Setup for Select Tests

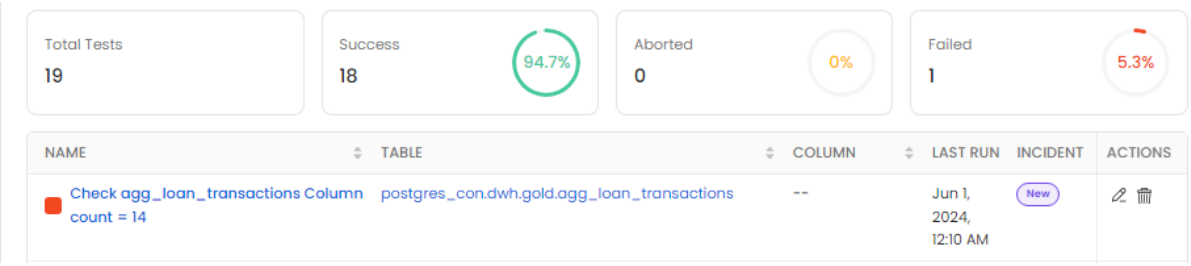


Figure 45 Check column count test failed, triggering an email notification.

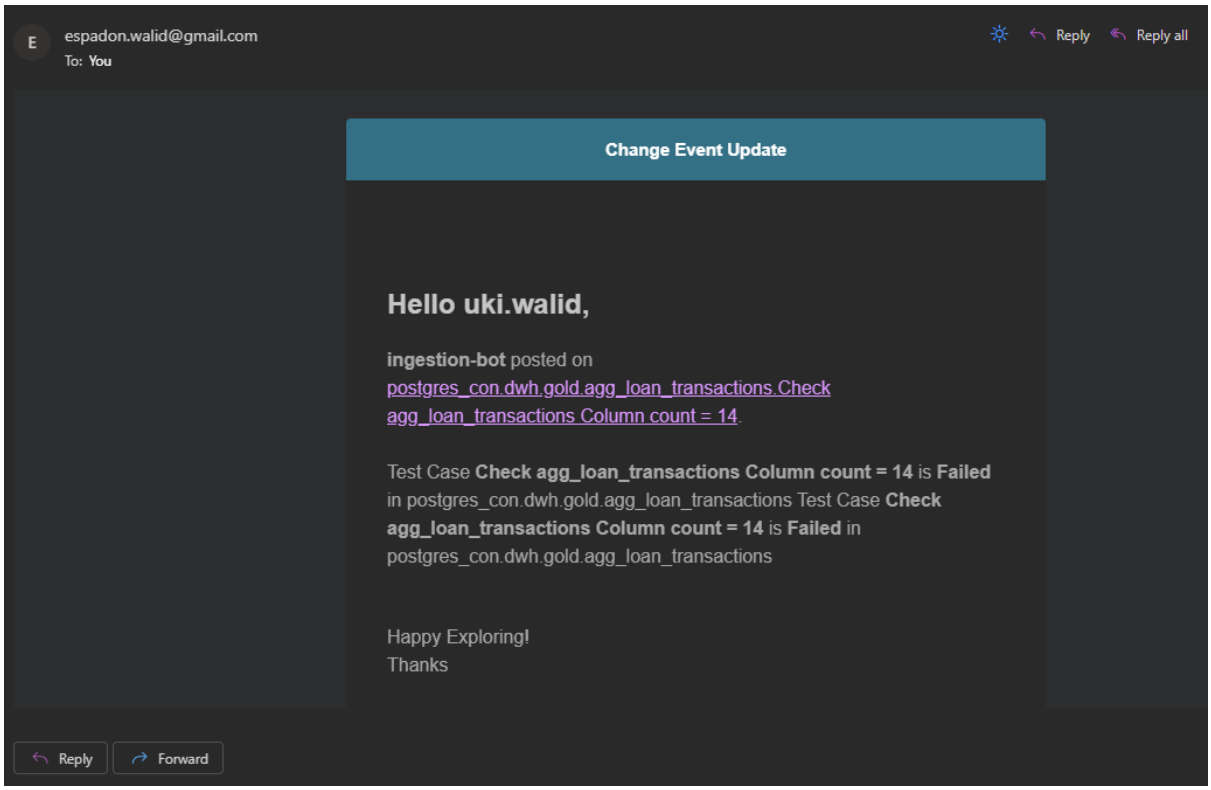


Figure 46 email alert of the failed test case

- **Figure 44 & 45:** Notification Examples - Shows real-time alerts triggered by data discrepancies, ensuring immediate attention and resolution.
- Real-time alerts are crucial for maintaining the integrity of our data pipeline. Figures 45 and 46 show how alerts are set up and triggered by specific tests, such as changes in column counts. These alerts ensure that any issues are promptly addressed, minimizing downtime and preventing potential data loss or corruption. For instance, an alert for a schema change prevented a potential data pipeline failure by allowing timely intervention.

7.3 Data Glossary and Catalog:

Enhancing Data Governance and Usability

With our data catalog, tags, and glossary powered by OpenMetadata, we provide sophisticated search capabilities and detailed metadata management, which is essential for effective data governance and usage.

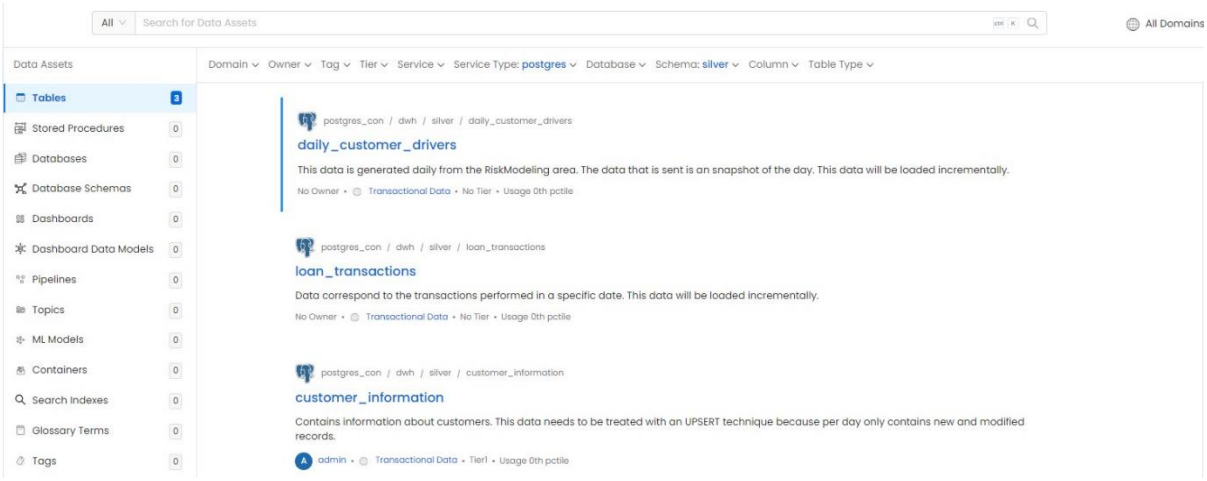


Figure 47 Example filtering for postgres service and targeting silver schema specifically

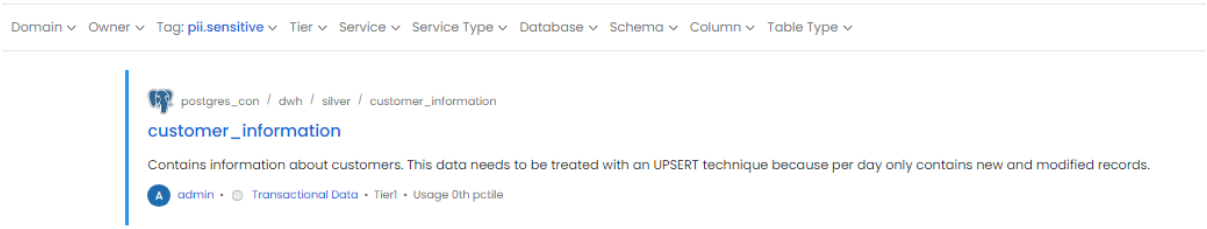


Figure 48 Filtering for PII Sensitive information

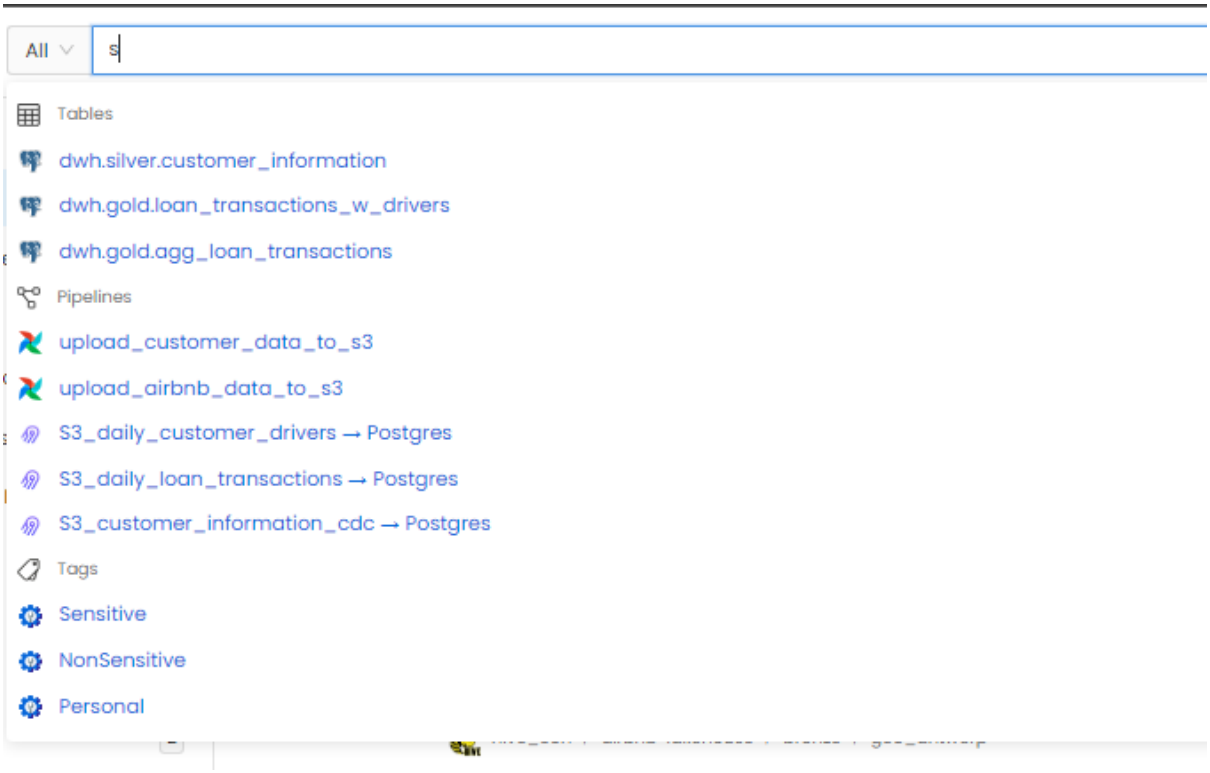


Figure 49 ElasticSearch Engine for data discovery

- **Figure 45: Specific Schema Targeting** - Example of how users can filter data views to specific needs, such as Postgres services targeting the silver schema.
- **Figure 46: Sensitive Data Filtering** - Showcases our capabilities to isolate and manage access to PII-sensitive information.
- **Figure 47: Searchable Data Assets** - Highlights the extensive search functionalities that enhance data discoverability across the organization.

OpenMetadata enables data teams to access and share organizational metadata. It supports fine-grained access control roles and policies to ensure data security. Effective data governance is further enhanced through a common vocabulary established by a Business Glossary, which defines terminology within the organization. Additionally, data classification and tagging are used to enforce various policies, such as privacy, data management, and data retention policies. This allows for managed access to sensitive PII data in OpenMetadata.

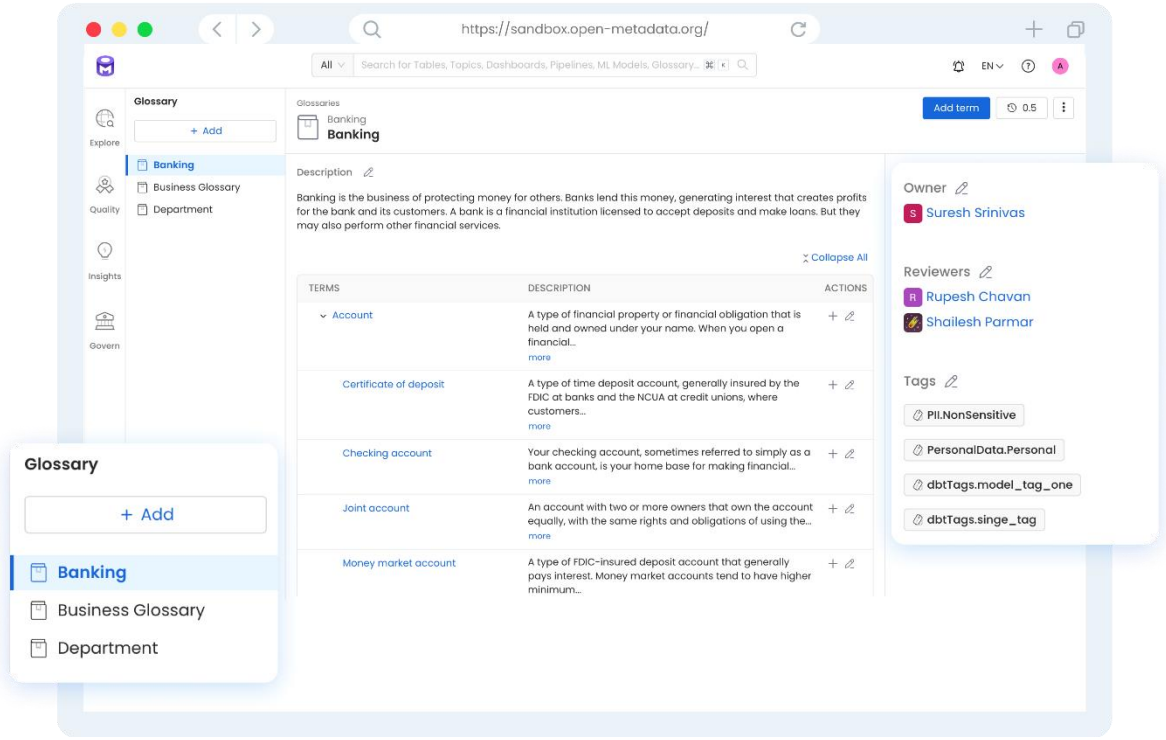


Figure 50 OpenMetadata' s Data Glossary example

(Figure 50) taken from OpenMetadata's [12] website and serves as an example of how glossary terms can be organized and utilized within the platform. Although we have not set up our data glossary terms in the POC, this figure demonstrates the following features:

- **Nested Glossary Terms:** Glossary terms can be nested within broader categories, providing a hierarchical structure that is easy to navigate.
- **Owners and Reviewers:** Each glossary term can have designated owners and reviewers, ensuring that the definitions are maintained by responsible parties.
- **Tags:** Tags can be applied to glossary terms to categorize and manage data effectively.
- **Description and Actions:** Each term has a detailed description and action options for managing the glossary entries.

OpenMetadata enables data teams to access and share organizational metadata. It supports fine-grained access control roles and policies to ensure data security. Effective data governance is further enhanced through a common vocabulary established by a Business Glossary, which defines terminology within the organization. Additionally, data classification and tagging enforce various policies, such as privacy and data retention, allowing for managed access to sensitive PII data.

7.4 Data Insights

Driving Platform Adoption and Data Culture

The Data Insights Dashboard serves as the nucleus for monitoring and driving the adoption of OpenMetadata within the organization. It provides essential metrics that guide strategic decisions and operational improvements.

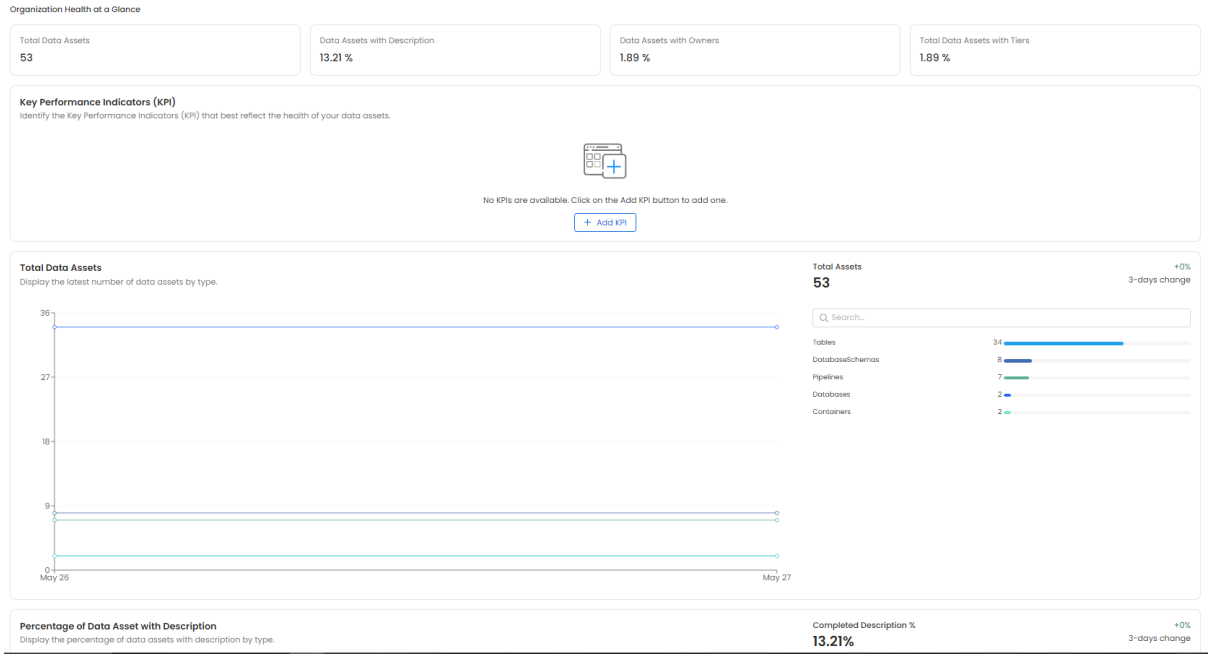


Figure 51 Data Asset Overview

- **Figure 51: Data Asset Overview** - Offers insights into the total number of assets managed, showcasing the scale and scope of our data management efforts.

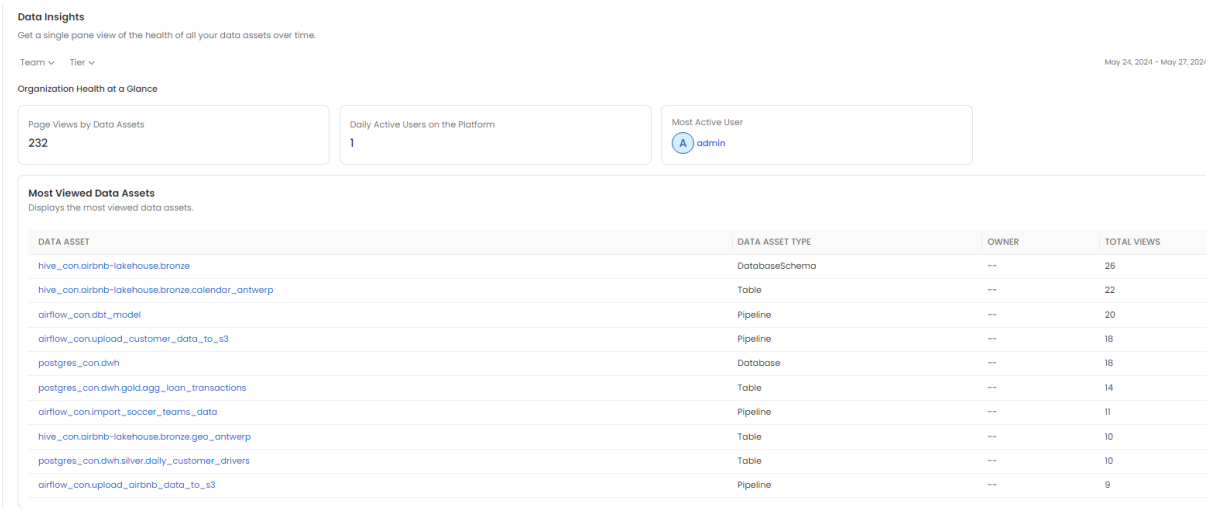


Figure 52 Insights on most used, viewed data asset, most active user

- **Figure 51: User Engagement and Asset Utilization** - Details the most frequently accessed and utilized data assets, helping identify key areas of focus for user engagement and resource allocation.

7.5 POC Conclusion

Reflecting on our established success criteria, we can confidently state that our objectives have been met, thanks to robust data governance, comprehensive monitoring, and efficient

real-time alerts. These achievements are further supported by our advanced data insights and collaboration tools, which have significantly enhanced our data governance strategy.

7.5.1 Key Achievements:

- **Data Quality and Monitoring:** Achieved high data quality and integrity through comprehensive monitoring.
- **Real-Time Alerts:** Implemented effective real-time alerts, reducing response times to data issues.
- **Data Glossary and Catalog:** Improved data governance and usability with a detailed data catalog and glossary example.
- **Data Insights:** Provided actionable insights that guided strategic decisions and enhanced data culture.

7.5.2 Future Steps:

- **Scalability:** Deploy in Kubernetes for better scalability and robustness.
- **Integration:** Integrate with external systems and managed services such as Redshift and BigQuery.
- **Security:** Implement better security authentication and switch to service accounts with limited permissions.
- **Access Control:** Implement access control based on data asset or product.

The POC involved setting up a Dockerized modern data stack sandbox environment, incorporating tools like Airflow, Airbyte, dbt, Spark, Hive Metastore, S3, and Postgres DWH. OpenMetadata was deployed to manage and govern the data pipelines, ensuring data quality, monitoring, real-time alerts, and providing a data catalog and glossary. These outcomes demonstrate that OpenMetadata can significantly enhance data governance and operational efficiency, meeting our objectives and success criteria.

7.5.3 Future Work:

Future work should focus on the following areas to further enhance data observability and governance capabilities:

1. **Deploy in Kubernetes:** For better scalability and robustness.
2. **Integrate with External Systems:** Connect with managed services such as Redshift and BigQuery.
3. **Implement Better Security Authentication:** Enhance security with improved authentication mechanisms.
4. **Switch to Service Accounts with Limited Permissions:** Improve security by using service accounts with limited access.
5. **Implement Access Control Based on Data Asset or Product:** Establish more granular access control policies to secure data assets effectively.

This enhanced explanation provides a comprehensive view of the POC results, clearly demonstrating how the objectives were achieved and setting the stage for future improvements.

8 Conclusion

Data observability offers a holistic view of data pipelines, covering end-to-end visibility, data quality monitoring, proactive issue resolution, lineage and impact analysis, and real-time insights. This approach addresses the multifaceted challenges faced by enterprise data teams, such as data quality issues, infrastructure monitoring, budgetary constraints, and the need for scalable and integrated solutions.

Through internal requirement gathering, we identified key pain points and evaluated various data observability tools, including Sifflet, Anomalo, Metaplane, Databand, Monte Carlo, DataHub, and OpenMetadata. Each tool's strengths and weaknesses were meticulously analysed, leading to the selection of OpenMetadata for our POC due to its comprehensive features, integration capabilities, and cost-effectiveness.

The POC involved setting up a Dockerized modern data stack sandbox environment, incorporating tools like Airflow, Airbyte, dbt, Spark, Hive Metastore, S3, and Postgres DWH. OpenMetadata was deployed to manage and govern the data pipelines, ensuring data quality, monitoring, real-time alerts, and providing a data catalog and glossary.

Key achievements of the POC include:

1. Comprehensive Data Monitoring: Implemented robust tests and data profiling to ensure data completeness, freshness, and accuracy.
2. Real-Time Alerts: Established immediate notifications for data issues, enhancing proactive management.
3. Data Glossary and Catalog: Enhanced data governance and usability through sophisticated search capabilities and detailed metadata management.
4. Data Insights: The Data Insights Dashboard facilitated monitoring and driving the adoption of OpenMetadata, providing essential metrics for strategic decisions.

These outcomes demonstrate that OpenMetadata can significantly enhance data governance and operational efficiency, meeting our objectives and success criteria.

Based on our findings, we recommend selecting data observability tools that offer comprehensive features, seamless integration with existing data environments, and scalability. Both DataHub and OpenMetadata stand out as robust choices, addressing data cataloging, search, discovery, governance, and quality. Although these tools differ in their release history and maturity, there is significant overlap in their features. OpenMetadata offers built-in data profiling with the possibility of using external tools, an in-house Data Quality Framework, and the flexibility to scale through its SaaS offerings (Collate). DataHub relies on third-party integrations for data profiling and quality, such as dbt and Great Expectations, but also supports native data contract enforcement and integration capabilities, as well as the flexibility to scale through its SaaS offerings (Acryl Data). These tools are particularly suitable for organizations seeking open-source solutions that can scale with their evolving needs.

The study and POC implementation have highlighted the transformative potential of data observability in modern data management. By providing deep visibility into data pipelines, enabling proactive issue resolution, and enhancing data governance, data observability and

governance tools like OpenMetadata and DataHub can empower organizations to leverage their data assets more effectively, driving business insights and operational excellence. This report serves as a foundational reference for organizations looking to adopt data observability practices, offering insights into tool selection, implementation strategies, and the tangible benefits of enhanced data governance.

9 References

1. [1] Sankaran, S. (2022). *Data Observability: A Tech Primer for Modern Data Teams*. USEReady Blog. Retrieved from https://www.useready.com/blog/data-observability-a-tech-primer
2. [2] DZone. (2023). *The Evolution of Data Pipelines*. Retrieved from https://dzone.com/articles/the-evolution-of-data-pipelines
3. [3] Acceldata. (2023). *The Definitive Guide to Data Observability for Analytics & AI.* Retrieved from https://www.acceldata.io/ebzook/the-definitive-guide-to-data-observability-for-analytics-ai
4. [4] SplashBI. (2024). *Data Pipeline for Seamless Enterprise Reporting*. SplashBI. Retrieved from https://splashbi.com/data-pipeline-for-seamless-enterprise-reporting/
5. [5] Monte Carlo. (2023). *What is Data Observability?* Retrieved from https://www.montecarlodata.com/blog-what-is-data-observability/
6. [6] IBM. (2023). *Data Observability*. Retrieved https://www.ibm.com/topics/data-observability.
7. [7] Airbyte. (2024). *What is Data Observability: 5 Pillars, Benefits, & Challenges*. Airbyte. Retrieved from https://airbyte.com/data-engineering-resources/data-observability
8. [8] DevOps.com. (2023). *Data Observability and Its Importance: Everything You Need to Know.* Retrieved from https://devops.com/data-observability-and-its-importance-everything-you-need-to-know/
9. [9] Kensu. (2024). *The State of Data Observability 2023 | Research Report*. Kensu. Retrieved from https://www.kensu.io/researches/state-of-data-observability
10. [10] strongDM. (2023). *Data Observability*. Retrieved from https://www.strongdm.com/blog/data-observability
11. [11] Monte Carlo. (2023). *Data Observability Architecture and Optimizing Your Coverage*. Retrieved from https://www.montecarlodata.com/data-observability-architecture-and-optimizing-your-coverage/
12. [12] GetOrchestra. (2024). *Best Data Observability Tools 2024 Ranked*. Retrieved from https://www.getorchestra.io/blog/best-data-observability-tools-2024-ranked
13. [13] Castor. (2024). *OpenMetadata vs. DataHub: Compare Architecture, Capabilities, Integrations & More*. Castor. Retrieved from https://www.castordoc.com/data-strategy/openmetadata-vs-datahub
14. [14] Atlan. (2024). *OpenMetadata vs. DataHub: Architecture, Features, Integrations*. Atlan. Retrieved from https://atlan.com/openmetadata-vs-datahub/
15. [15] DataHub. (2023). *Open-Source Data Catalog Platform*. Retrieved from https://www.datahubproject.io
16. [16] AWS Marketplace. (2024). *Acryl Data: Metadata Management Platform*. AWS. Retrieved from https://aws.amazon.com/marketplace/pp/prodview-sum2rwvczvwvi?sr=0-1&ref_=beagle&applicationId=AWSMPContessa
17. [17] OpenMetadata. (2023). *Comprehensive Data Management and Observability Platform*. Retrieved from https://www.open-metadata.org
18. [18] Collate. (2024). *Collate: SaaS for OpenMetadata (Pricing).* Collate. Retrieved from https://www.getcollate.io/pricing
19. [19] Sifflet. (2023). *Data Observability for Comprehensive Data Management*. Retrieved from https://www.siffletdata.com
20. [20] AWS Marketplace. (2024). *Sifflet: Data Observability Platform*. AWS. Retrieved from https://aws.amazon.com/marketplace/pp/prodview-lz6cw37v4bkna

21. [21] Monte Carlo. (2023). *Data Observability for Modern Data Teams*. Retrieved from https://www.montecarlodata.com
22. [22] AWS Marketplace. (2024). *Monte Carlo Data Observability Platform*. AWS. Retrieved from https://aws.amazon.com/marketplace/pp/prodview-hikicsfohm3gg?sr=0-1&ref_=beagle&applicationId=AWSMPContessa
23. [23] Databand. (2023). *Comprehensive Platform for Data Incident Management*. Retrieved from https://www.databand.ai
24. [24] Metaplane. (2023). *Continuous Integration and Machine Learning-Based Monitoring*. Retrieved from https://www.metaplane.dev
25. [25] Anomalo. (2023). *Anomaly Detection with Unsupervised Machine Learning*. Retrieved from https://www.anomalo.com
26. [26] Elementary. (2023). *DBT-Native Data Observability Tool*. Retrieved from https://www.elementary-data.com/
27. [27] Soda. (2023). *Embedded Testing Framework for Data Observability*. Retrieved from https://www.soda.io
28. [28] Great Expectations. (2023). *Flexible Data Validation Framework*. Retrieved from https://www.greatexpectations.io
29. [29] YouTube. (2023). *DataHub Adoption Stories* [Video playlist]. Retrieved from https://www.youtube.com/playlist?list=PLdCtLs64vZvGCKMQC2dJEZ6cUqWsREbFi
30. [30] OpenMetadata. *Connectors Documentation*. Retrieved from https://docs.open-metadata.org/v1.4.x/connectors
31. Ventana Research. (2023). *Buyer's Guide to Data Observability*. Retrieved from https://www.ventanaresearch.com/buyers_guide/data/data_observability_2023/thank_you?submissionGuid=60ed1c4b-3f9f-4f76-815c-a0adfa53d92c
32. TrustRadius. (2023). *Data Observability*. Retrieved from https://www.trustradius.com/data-observability
33. AIMultiple. (2023). *Data Observability Tools: Vendor Selection Criteria*. Retrieved from https://research.aimultiple.com/data-observability-tools/#vendor-selection-criteria
34. Gartner. (2023). *Cool Vendors in Monitoring and Observability*. Retrieved from https://www.gartner.com/doc/reprints?id=1-2EJNEPFD&ct=230725&st=sb
35. IBM. Grid® Report for Observability Solution Suites | Winter 2024. Retrieved from https://www.ibm.com/downloads/cas/KXA1GLDP
36. OpenDataDiscovery. (2023). *Awesome Data Catalogs*. Retrieved from https://github.com/opendatadiscovery/awesome-data-catalogs

9.1 Appendix

Appendix A: Pricing Information References

- **A.1 Monte Carlo Pricing Information** The pricing information for Monte Carlo was obtained through an online meeting held on May 10th. During the meeting, the following details were provided:
  - Starter Plan: $40,000+ per year
  - Enterprise Plan: $100,000+ per year
- **A.2 Databand Pricing Information** The pricing information for Databand was obtained through an email conversation after a series of phone calls. Below is the full email transcript for reference:

Subject: Databand Pricing Information

Dear Walid,

As discussed, coming back on the pricing. There are two models, SaaS or subscription. Personally, I would advise to go for SaaS, because all IT, all version updates etc. are taken care of.

The SaaS pricing is calculated per pipeline and per year. List price is EUR 250/pipeline/year. Depending on the number of pipelines that you need to monitor and manage, I can adapt the price. If you need to monitor 1000 pipelines/year, then I could offer EUR 220/pipeline/year. If there were more pipelines than 1000, then the price would clearly be lower.

The subscription model cost is also depending on the amount of pipelines. For 1000 pipelines in the subscription model, we would end at EUR 240/pipeline/year.

However, we can scale the volume and price down/up to your needs. It would be great if you could let me know what your expectations are in terms of pipelines and needs? If you have mostly 30-50 pipelines clients, then it might maybe make sense to focus on those first? And if the budget indication is approx. within your estimations?

In essence, I am of course open to discuss everything, and I hope this indication makes sense.

Thank you for your feedback.

Ruven

Appendix B: Data Source References

- **B.1 Transactional Data Source** This dataset comprises loan transactions and customer interactions, offering a robust source of relational data. Retrieved from Google Drive: Transactional Data Source

- **B.2 API Data for Soccer Players** This dataset provides detailed information on soccer players, including squad details and player statistics, extracted from the sports API. Retrieved from: API Data for Soccer Players
- **B.3 Open Data from Inside Airbnb** This dataset features publicly available data from Inside Airbnb for Belgium (Brussels, Ghent, Antwerp). Retrieved from Google Drive: Inside Airbnb Data